ORACLE NetSuite OpenAir

NetSuite Integration

2024.1

July 24, 2024

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

If this document is in public or private pre-General Availability status:

This documentation is in pre-General Availability status and is intended for demonstration and preliminary use only. It may not be specific to the hardware on which you are using the software. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to this documentation and will not be responsible for any loss, costs, or damages incurred due to the use of this documentation.

If this document is in private pre-General Availability status:

The information contained in this document is for informational sharing purposes only and should be considered in your capacity as a customer advisory board member or pursuant to your pre-General Availability trial agreement only. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described in this document may change and remains at the sole discretion of Oracle.

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle Master Agreement, Oracle License and Services Agreement, Oracle PartnerNetwork Agreement, Oracle distribution agreement, or other license agreement which has been executed by you and Oracle and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced, or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trsif you are hearing impaired.

Sample Code

Oracle may provide sample code in SuiteAnswers, the Help Center, User Guides, or elsewhere through help links. All such sample code is provided "as is" and “as available”, for use only with an authorized NetSuite Service account, and is made available as a SuiteCloud Technology subject to the SuiteCloud Terms of Service at www.netsuite.com/tos, where the term “Service” shall mean the OpenAir Service.

Oracle may modify or remove sample code at any time without notice.

No Excessive Use of the Service

As the Service is a multi-tenant service offering on shared databases, Customer may not use the Service in excess of limits or thresholds that Oracle considers commercially reasonable for the Service. If Oracle reasonably concludes that a Customer’s use is excessive and/or will cause immediate or ongoing performance issues for one or more of Oracle’s other customers, Oracle may slow down or throttle Customer’s excess use until such time that Customer’s use stays within reasonable limits. If Customer’s particular usage pattern requires a higher limit or threshold, then the Customer should procure a subscription to the Service that accommodates a higher limit and/or threshold that more effectively aligns with the Customer’s actual usage pattern.

Table of Contents

NetSuite Integration Overview

The OpenAir NetSuite integration provides a seamless flow of information between the OpenAir Professional Services Automation (PSA) and NetSuite Financials (ERP). It lets you leverage the strength of both system and create an integrated solutions for greater collaboration, automation and efficiency gains across your enterprise.

You can use the NetSuite integration to integrate your data on demand, on a defined schedule, or in real time.

You can specify the data you want to exchange between NetSuite and OpenAir to suit your business processes and requirements. Common integrated data elements include customers, employees, items you sell and the projects and invoices you create. With the integration set up, you enter information one time and it is shared automatically between the two applications, leading to better data quality and accuracy.

With the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature, you can view all your project information from within NetSuite without having to sign in to OpenAir separately. Company executives can view all company project metrics alongside sales, financials and other key performance indicators from within NetSuite. Users gain visibility into service delivery from a single dashboard, with an overview of project execution, staffing and financials.

For an overview of the main NetSuite integration features, see NetSuite Integration Main Features.

For an overview of NetSuite features requiring special consideration to work with the NetSuite <> OpenAir integration, see Feature Compatibility Check List.

For an overview of how to get started with the NetSuite <> OpenAir integration, see Getting Started With the NetSuite integration.

Example of Typical Usage

In a typical scenario, you would use NetSuite as your primary application for employees, leads, prospects, customers, contacts, vendors, expense categories, projects, and service items and import these into OpenAir. The integration creates and updates OpenAir records so your OpenAir and NetSuite data is synchronized.

You then build a project and connect it to a sales order in NetSuite when estimating the scope, identifying the work to be done for the customer, and specify how each service delivered through the project will be billed (for example by time, by fixed date, …). The sales order in NetSuite may show other revenue streams in the company, such as hardware sales and subscriptions that are billed separately from your project delivery. You may also create a work breakdown structure in OpenAir to generate and send item estimates back to an opportunity in NetSuite.

You can use OpenAir project templates to prepopulate your projects with commonly used tasks, billing rules, loaded costs, revenue recognition rules, auto-bill settings, and project approvers. OpenAir project templates provide an efficient way to capture and reuse information specific to a particular customer or type of project.

After the sales order is closed and you start delivering work against the project, you can export revenue recognition transactions and invoices generated in OpenAir to NetSuite, using the approval workflow in OpenAir to review the invoice before sending it to NetSuite. You receive payments in OpenAir and automatically update OpenAir so your service delivery staff are kept informed even if they do not have access to your NetSuite financials. Aside from your billing operations, you can manage payments for consultant time and expenses submitted and approved in OpenAir using NetSuite.

With the OpenAir Within NetSuite and One View features providing access to your OpenAir dashboard and report data in NetSuite, you can view your project and financials metrics side-by-side and get a unified view of KPIs such as profitability and revenue forecast to inform both operational and strategical decisions.

▶ NetSuite SuiteSuccess for Services: OpenAir Integration — The video showcases how the OpenAir <> NetSuite integration lets you manage project financials from sales order to billing, invoice payments, expenses tracking and reimbursement from the NetSuite UI.

▶ NetSuite OpenAir: OneView — The video showcases how you can combine project data and reporting from OpenAir into NetSuite to have all important metrics and KPIs in one place and support data-driven decisions.

About this Guide

This guide is provided for account administrators and employees responsible for managing the NetSuite <> OpenAir integration. It covers the information you need to know to manage the integration, to troubleshoot simple problems, to specify the data you want to exchange as your business processes evolve and to take advantage of new features as they are released.

The initial setup of the integration is described for reference purposes only. OpenAir Professional Services configures your OpenAir and NetSuite accounts initially following the leading practice so you can use the integration.

The guide focuses on the latest recommended features available for your OpenAir account. For more information about latest recommendations and to review the features enabled in your current integration setup, see OpenAir NetSuite Connector Health Check.

The guide is organized around the following conceptual topics for ease of reference:

- NetSuite Integration Main Features — High level overview of main integration features.
- How the NetSuite <> OpenAir Integration Works — High level overview of the information flow between NetSuite and OpenAir.
- Configuring NetSuite for the Integration — Reference material for configuring NetSuite to use both standard and optional integration features.
- Configuring OpenAir for the Integration — Reference material for configuring OpenAir to use both standard and optional integration features. Includes reference material for using the OpenAir NetSuite Connector UI to control what information is exchanged, scheduling the integration to run automatically, and configuring the real-time export of records from OpenAir to NetSuite.
- Running the NetSuite <> OpenAir Integration — Reference material for running the integration on demand for a single record or in bulk either manually or using OpenAir user scripting.
- Importing Data From NetSuite Into OpenAir — Reference material for all available integration workflows importing data from NetSuite into OpenAir.
- Exporting Data From OpenAir to NetSuite — Reference material for all available integration workflows exporting data from OpenAir to NetSuite, including custom export workflows.

Note: This guide is undergoing a full review and restructure with a view to making it available in the OpenAir Help Center and in OpenAir SuiteAnswers.

Some of the content may have been relocated. Please use the search functionality on your PDF reader or the bookmarks to find the relevant information, if necessary.

The following content is still awaiting a full review and is largely the same as in previous versions of this guide

- Working with Integrated Records in NetSuite and OpenAir
- Configuring Optional Features (Some sections in this chapter have been relocated to Configuring NetSuite for the Integration or Configuring OpenAir for the Integration.
- Using the Integration
- Data Mapping in the Integration
- Support and Troubleshooting

The following new chapters are placeholders in will hold consolidated reference material for all available integration workflows:

- Importing Data From NetSuite Into OpenAir

Exporting Data From OpenAir to NetSuite

NetSuite Integration Main Features

The following features are available for all NetSuite <> OpenAir integration deployments:

- Bulk Import and Export On Demand
- Scheduled Bulk Import and Export
- Single Record Export On Demand
- Monitored Fields

The following integration features are optional and require additional configuration:

- Real-Time Record Export
- Real-Time Record Import
- OpenAir Within NetSuite
- OneView — Published Report Export to NetSuite
- Multithreading

Bulk Import and Export On Demand

You can trigger the integration manually at any time to import and export records in bulk.

- You can select the integration workflows to be included in the run. Integration workflows are defined for each record type you can export from OpenAir to NetSuite or import from NetSuite into OpenAir.
- You can use an OpenAir user scripting function to trigger an integration run to import and export records in bulk from your scheduled scripts. The run will include all integration workflows that are active at the time the run is triggered.
- When importing records in bulk from NetSuite into OpenAir, new and modified NetSuite records are imported into OpenAir. Import workflows create records in OpenAir and update your OpenAir data based on your NetSuite data. For each record in NetSuite, you can choose whether to import the record into OpenAir.
- When exporting records in bulk from OpenAir to NetSuite, new OpenAir records are exported to NetSuite. Export workflows only create records in NetSuite for most record types. You can configure the NetSuite integration to update projects and tasks data based on your OpenAir data.
- Only the integrated records modified since the last integration run are included. You can run a full synchronization to synchronize all integrated records for selected workflows.

This feature is available by default for all NetSuite <> OpenAir integration deployments.

See also:

- Adding and Managing Integration Workflows
- Running the Bulk Import and Export Integration on Demand
- Running a Full Synchronization on Demand
- Running the NetSuite <> OpenAir Integration From OpenAir User Scripts

Scheduled Bulk Import and Export

You can schedule the integration to import and export records in bulk automatically.

- You can select the integration workflows to be included in each scheduled run. Integration workflows are defined for each record type you can export from OpenAir to NetSuite or import from NetSuite into OpenAir.
- You can select the day and time you want each scheduled run to start.
- When importing records in bulk from NetSuite into OpenAir, new and modified NetSuite records are imported into OpenAir. Import workflows create records in OpenAir and update your OpenAir data based on your NetSuite data. For each record in NetSuite, you can choose whether to import the record into OpenAir.
- When exporting records in bulk from OpenAir to NetSuite, new OpenAir records are exported to NetSuite. Export workflows only create records in NetSuite for most record types. You can configure the NetSuite integration to update projects and tasks data based on your OpenAir data.
- Only the integrated records modified since the last integration run are included. You can run a full synchronization to synchronize all integrated records for selected workflows.

This feature is available by default for all NetSuite <> OpenAir integration deployments. For more information, see Scheduling the Integration to Run Automatically.

Single Record Export On Demand

You can export a single record from OpenAir to NetSuite on demand.

- You can run the integration manually to export a single OpenAir record to NetSuite. Export workflows only create records in NetSuite for most record types. You can configure the NetSuite integration to update projects and tasks data based on your OpenAir data.
- You can use a user scripting function to export a single OpenAir record from your form scripts.

- Single record export on demand follows the field mapping and settings configured for the relevant integration workflow. See Adding and Managing Integration Workflows.

Single record export on demand can be used with the following OpenAir record types:

- Expense Reports — See Expense Reports (Expense Reports) Export Workflow.
- Invoices — See Invoices (Invoices) Export Workflow.
- Revenue Recognition Transactions — See Recognition Transactions (Revenue Recognition Transactions) Export Workflow.
- Customers — See Customers (Customers) Export Workflow.
- Timesheets — See Time Entries (Time) Export Workflow.
- Purchase Requests — See Purchase Requests (Purchase Requisitions) Export Workflow.
- Projects — See Projects (Projects) Export Workflow and Update NetSuite Projects with OpenAir Project Data.
- Project Tasks — See Update NetSuite Tasks with OpenAir Task Data.
- Work Breakdown Structure — See WBS (Opportunity) Export Workflow.

This feature is available by default for all NetSuite <> OpenAir integration deployments.

See also:

- Exporting a Single Record from OpenAir to NetSuite on Demand.
- Running the NetSuite <> OpenAir Integration From OpenAir User Scripts

Monitored Fields

You can speed up certain import workflows by ensuring that records are only included in the integration run if one or more integrated field values have changed.

With this feature enabled, the fields designated for integration are monitored for change. NetSuite records will be imported only if one of the monitored field values have changed.

This feature is available by default for all NetSuite <> OpenAir integration deployments with the latest integration bundles installed in NetSuite. For more information, see Enabling Monitored Fields for Import Workflows.

For more information about designating fields for integration, see Mapping OpenAir Fields and NetSuite Fields.

Note: The following workflows support the Monitored Fields feature:

- Customers (Customers) Import Workflow
- Employees (Employees) Import Workflow
- Projects (Projects) Import Workflow
- Tasks (Project Tasks) Import Workflow

Real-Time Record Export

You can configure the NetSuite integration to export an OpenAir record automatically to NetSuite when the record is approved or saved by any user in OpenAir.

Real-time export from OpenAir to NetSuite runs in the background. When saving, approving or running an action on records, either individually or in bulk, users no longer have to wait until the export finishes or for a confirmation notification to continue working. An email is sent to the administrator in case of an integration error.

The following OpenAir records can be exported to NetSuite upon approval:

- Invoices
- Expense Reports
- Timesheets
- Purchase requests

The following OpenAir records can be exported to NetSuite upon saving:

- Customers
- Projects
- Tasks

For all OpenAir record types except Project and Task records, the real-time export only creates a record in NetSuite when the record is approved or saved for the first time. For Project and Task records, the real-time integration can create and update the record in NetSuite.

Real-time record exports follow the field mapping and settings configured for the relevant integration workflow. See Adding and Managing Integration Workflows.

For more information, see Configuring Real-Time Record Export from OpenAir to NetSuite.

Note: This is an optional NetSuite <> OpenAir integration feature. It requires an active NetSuite SuiteCloud Plus license and the Multithreading feature. See Multithreading.

Real-Time Record Import

You can configure the NetSuite <> OpenAir integration to import a NetSuite record automatically into OpenAir when the record is saved by any user in NetSuite.

If a user has the relevant permissions to create or modify a record in NetSuite and saves the record, the record is imported in to OpenAir. The NetSuite user making the change does not need to be an OpenAir user.

The following NetSuite records can be imported in to OpenAir upon saving. The real-time import creates a new record in OpenAir if it is imported for the first time, and updates the OpenAir record if it was created during a previous import.

- Employees
- Customers
- Projects
- Cases (imported to OpenAir as project tasks)
- Project Tasks
- Items
- Expense categories
- Sales orders
- Credit memos

- Invoice payments
- Vendor bills
- Vendor credits
- Purchase orders
- Expense report reimbursements
- OpenAir: Cost Center (custom record)
- OpenAir: Project Rate Card (custom record)

Real-time record imports follow the field mapping and settings configured for the relevant integration workflow. See Adding and Managing Integration Workflows

For more information, see Configuring Real-Time Record Import from NetSuite into OpenAir.

Important Notes and Guidelines About Real-Time Record Import

Real-Time Record Import is an optional NetSuite <> OpenAir integration feature and requires additional configuration.

Until the OpenAir 2023.2 release, Real-Time Record Import relied on the NetSuite Outbound Single Sign-On (SuiteSignOn) feature as outbound single sign-on method. NetSuite plans to end support for SuiteSignOn with the NetSuite 2024.2 for production accounts. The functionality relying on SuiteSignOn may stop working at any time after its end-of-support date.

An upgraded version of the Real-Time Record Import feature is available and uses certificate-based authentication instead of SuiteSignOn to validate real-time record import requests.

Warning: If you are using the previous feature version relying on NetSuite Outbound Single Sign-On (SuiteSignOn), upgrade before the NetSuite 2024.2 release to ensure service continuity. For upgrade instructions, see Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn.

OpenAir changes supporting the upgraded Real-Time Record Import feature were delivered as part of the OpenAir 2023.2 release on October 7, 2023.

NetSuite bundles supporting the upgraded Real-Time Record Import feature were released on February 7, 2024. The upgrade requires the following bundles and release versions:

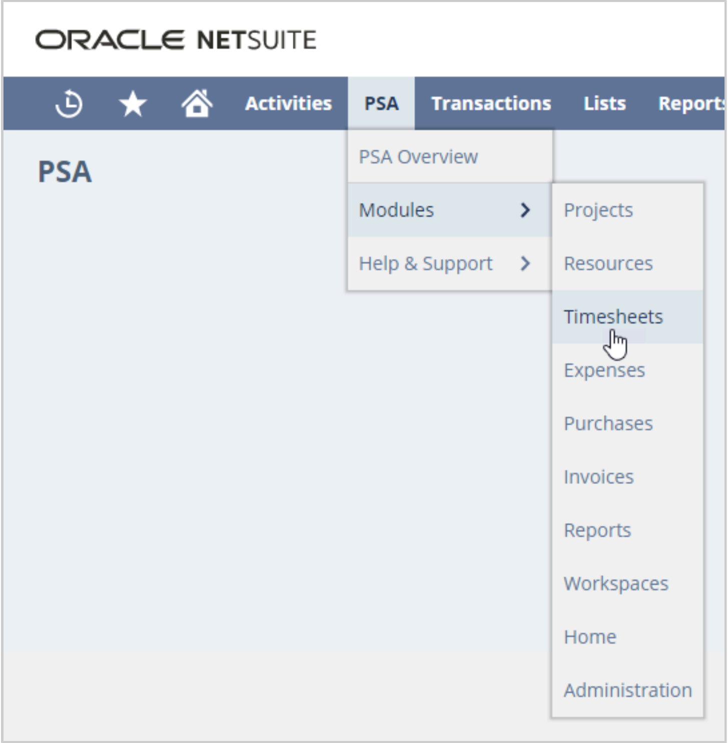
- NetSuite OpenAir SRP Integration (Bundle ID: 369637) 4.0
- NetSuite OpenAir SRP Real-time Add-on (Bundle ID: 369630) 3.0

OpenAir Within NetSuite

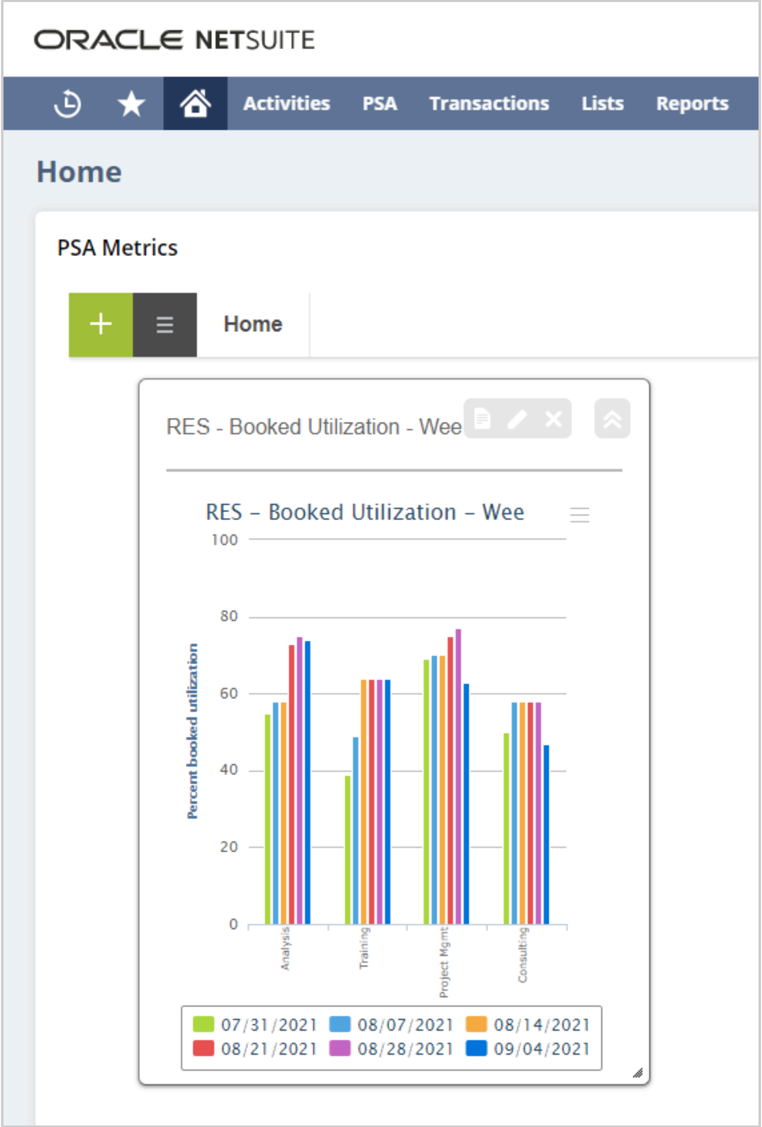
You can give users of both NetSuite and OpenAir access to OpenAir directly from within the NetSuite user interface.

With the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature, your users only need to sign in to your company's NetSuite account. Then, they also have access to the following functionality without having to sign in to OpenAir:

- A PSA center tab in the NetSuite navigation bar, with any of the OpenAir modules, Support page and OpenAir Help Center as submenu options from the PSA center tab.



- The OpenAir dashboard as a portlet on the NetSuite dashboard.



- An OpenAir subtab showing the OpenAir record on record pages for supported NetSuite record types. The following table lists the supported NetSuite record types and the corresponding types of OpenAir record shown in the OpenAir subtab

NetSuite record type	OpenAir record type
Contact	Contact
Customer	Customer
Employee	Employee
Expense Report	Expense Report
Invoice	Invoice
Journal Entry	Revenue Recognition Transaction
OpenAir: Project Rate Card (custom record type)	Rate Card
Project	Project

NetSuite record type	OpenAir record type
Service Item	Category

For more information, see Enabling Access to OpenAir From Within NetSuite.

Important Notes and Guidelines About OpenAir Within NetSuite

OpenAir Within NetSuite (OpenAir NetSuite Single Sign-On Integration) is an optional NetSuite <> OpenAir integration feature and requires additional configuration.

Until the OpenAir 2023.2 release, OpenAir Within NetSuite relied on the NetSuite Outbound Single Sign-On (SuiteSignOn) feature as outbound single sign-on method. NetSuite plans to end support for SuiteSignOn with the NetSuite 2024.2 for production accounts. The functionality relying on SuiteSignOn may stop working at any time after its end-of-support date.

An upgraded version of the OpenAir Within NetSuite feature is available and uses NetSuite as OIDC Provider as outbound single sign-on method instead of SuiteSignOn.

Warning: If you are using the previous feature version relying on NetSuite Outbound Single Sign-On (SuiteSignOn), upgrade before the NetSuite 2024.2 release to ensure service continuity. For upgrade instructions, see Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn.

For more information about NetSuite as OIDC Provider, see NetSuite as OIDC Provider (External link to NetSuite SuiteAnswer | Answer ID: 98241).

OpenAir changes supporting the upgraded OpenAir Within NetSuite feature were delivered as part of the OpenAir 2023.2 release on October 7, 2023.

NetSuite bundles supporting the upgraded OpenAir Within feature were released on February 7, 2024. The upgrade requires the following bundles and release versions:

- NetSuite OpenAir SRP Integration (Bundle ID: 369637) 4.0
- NetSuite OpenAir SRP Single Sign-On (Bundle ID: 369619) 7.0

Note: Some of the configuration-based customization available when using the SuiteSignOn feature is not currently available when using the NetSuite as OIDC Provider feature. You would need to deploy custom SuiteScripts to achieve the following functionality:

- Ability to show the Timesheets module on Support Case records as an OpenAir: Timesheets subtab in NetSuite when using the Tasks (Support Cases) Import Workflow.
- Ability to set up custom OpenAir subtabs on NetSuite records and displaying OpenAir content other than those listed in the table above.

OneView — Published Report Export to NetSuite

If the Business Intelligence Connector feature is enabled for your OpenAir account, you can publish OpenAir reports to the OpenAir OData service with the OpenAir NetSuite Connector scope so you can then export your OpenAir report data to NetSuite. You can configure OpenAir to refresh your published report data according to a defined schedule.

After you publish a report to your OpenAir OData feed for access by OpenAir NetSuite Connector, you can create a custom export workflow to export data from your OpenAir published report to a custom record type in NetSuite. You can add this custom export workflow to workflow groups and include them in your scheduled integration runs, or when running the integration manually.

You can then combine your exported OpenAir published report data and your NetSuite data to get a unified view of key metrics in NetSuite such as profitability and revenue forecast.

NetSuite OpenAir: OneView — The video showcases how you can combine project data and reporting from OpenAir into NetSuite to have all important metrics and KPIs in one place and support data-driven decisions.

For more information, see Exporting OpenAir Published Report Data To NetSuite Custom Record Types.

> **Note:** Custom export workflows to export OpenAir published report data to NetSuite require the Business Intelligence Connector feature to be enabled for your OpenAir account. To enable this features, contact your OpenAir account manager.

Multithreading

If you have an active NetSuite SuiteCloud Plus license, OpenAir NetSuite Connector can use multiple threads concurrently to export records from OpenAir to NetSuite. NetSuite SuiteCloud Plus allows up to 10 threads to be allocated concurrently for communication with NetSuite web services.

Multithreading support is available for:

- **Single record export on-demand and real-time export from OpenAir to NetSuite** — The NetSuite integration can use any available threads concurrently for real-time export and single record export on-demand. With this feature, multiple users can trigger the export of a record at the same time. If all threads are in use, Open Air shows the message "All available threads are in use". If this message shows, try to export the record again later. By default, the NetSuite integration can use all 10 threads concurrently (if available) for real-time export and export on-demand.
- **On-demand and scheduled bulk export of records from OpenAir to NetSuite** — Specify the number of threads (up to 10) the NetSuite integration can use to export records from OpenAir to NetSuite as part of scheduled or on-demand bulk integration runs. This allows your integration export workflows to run up to 10 times faster.

> **Note:** To enable the Multithreading Support for On-Demand and Real-Time Export, or Multithreading Support for the Main Integration features, contact OpenAir Customer Support. You can specify the maximum number of threads that the NetSuite integration can use at any one time to a lesser value. You may want to ensure that threads are available for other processes, for example.
>
> The Multithreading Support features require an active NetSuite SuiteCloud Plus license. If you do not have an active NetSuite SuiteCloud Plus, the OpenAir NetSuite integration is limited to a single thread communication with NetSuite web services at all time.
>
> Multithreading is supported only for export from OpenAir to NetSuite and not for import from NetSuite to OpenAir.

Feature Compatibility Check List

The following NetSuite features require special consideration to work with the NetSuite <> OpenAir integration:

- Multiple Currencies
- Classic Revenue Recognition or Advanced Revenue Management
- Legacy Tax or SuiteTax

Multiple Currencies

If you use more than one currency for business transactions, you may already have your OpenAir account, NetSuite account or both accounts configured to support multiple currencies. If you use multiple currencies in either OpenAir or NetSuite, enable and configure multiple currencies in both applications. See Enabling the Multiple Currencies Feature (external link to NetSuite SuiteAnswers — Answer ID: 7714) for more information about the Multiple Currencies feature in NetSuite, and Optional Features for more information about the Multicurrency feature in OpenAir.

After you have enabled and configured multiple currencies in both NetSuite and OpenAir, you can use the exchange rate integration to:

- Import exchange rates from NetSuite in to OpenAir daily so today’s exchange rates in OpenAir are the same as in NetSuite.
- Support multiple base currencies for cross rates, if you have a NetSuite OneWorld account.

For more information, see Exchange rate integration.

Classic Revenue Recognition or Advanced Revenue Management

When you amortize revenue and set up rules for recognizing revenue on a project-by-project basis, you can exchange this information between OpenAir and NetSuite. OpenAir can export your OpenAir revenue recognition rules and revenue recognition transactions to NetSuite for use either with the classic revenue recognition features, available in older NetSuite implementations, or with the Advanced Revenue Management feature available in new NetSuite implementations.

The NetSuite integration supports both classic revenue recognition and the Advance Revenue Management feature using the following workflows:

- Workflows used with NetSuite classic revenue recognition features:
  - Recognition Transactions (Revenue Recognition Transactions) Export Workflow
- Workflows used with the NetSuite Advance Revenue Management feature:
  - Recognition Rules (Custom Record: OpenAir: Revenue Rules for ARM) Export Workflow
  - Recognition Transactions (Custom Record: OpenAir: Revenue Transactions for ARM) Export Workflow

  Advanced Revenue Management currently supports the following OpenAir revenue recognition rules: As Billed, Fixed Amount on Date or Milestone, Fixed Amount % Complete, % Complete, Incurred vs. Forecast, and Time Project.

Note: Classic revenue recognition features and Advanced Revenue Management are mutually exclusive in NetSuite. The two set of workflows listed above are also mutually exclusive.

For more information about the NetSuite features, see the following help topics:

- Advanced Revenue Management (external link to NetSuite SuiteAnswers | Answer ID: 49167)
- Using Revenue Recognition (external link to NetSuite SuiteAnswers | Answer ID: 7931)

For more information about revenue recognition in OpenAir, see Administrator Guide.

Legacy Tax or SuiteTax

The NetSuite integration is compatible with the legacy Tax features.

The following OpenAir <> NetSuite integration workflows are compatible with the NetSuite SuiteTax feature:

- Expense Reports (Expense Reports) Export Workflow, when expense reports are exported from OpenAir to NetSuite vendor bills.
- Invoices (Invoices) Export Workflow
- Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Export Workflow

Important: Review the following guidelines:

- Update NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) to version 2.11 or later. With earlier versions, the tax amount calculated in NetSuite cannot be imported back into OpenAir if the SuiteTax feature is enabled for your NetSuite account.
- When exporting OpenAir expense reports to NetSuite as expense reports using the Expense Reports (Expense Reports) export workflow, the tax details need to be entered manually in NetSuite when using the SuiteTax engine. See Entering Expense Reports with Tax Impact in SuiteTax (external link to NetSuite SuiteAnswers | Answer ID: 65664).
- You should test the NetSuite <> OpenAir integration with the SuiteTax feature on sandbox accounts before you enable the feature on your NetSuite production account.
- For more information about the NetSuite features, see the following help topics:
  - SuiteTax (external link to NetSuite SuiteAnswers | Answer ID: 64700)
  - Legacy Tax (external link to NetSuite SuiteAnswers | Answer ID: 50295)

Getting Started With the NetSuite integration

You should use the following steps to set up and get familiar with the NetSuite integration before using it to exchange information between NetSuite and OpenAir:

- Step 1 — Prerequisites and Setup
- Step 2 — Read the Relevant Documentation
- Step 3 — Configure the Integration in NetSuite
- Step 4 — Configure the Integration in OpenAir
- Step 5 — Run the Integration

⚠ **Important:** You should test the initial configuration and any subsequent changes using NetSuite and OpenAir sandbox accounts, before implementing the integration on a production environment.

Step 1 — Prerequisites and Setup

NetSuite Professional services configure your NetSuite and OpenAir initially so you can use the integration.

The following table lists the main requirements to be able to implement the NetSuite <> OpenAir integration with references to the relevant help topics.

Requirements	How to enable
NetSuite Features	See NetSuite Features Required for the Integration
NetSuite Bundles	See NetSuite Bundles for the Integration
OpenAir Features: - OpenAir NetSuite Connector. - New NetSuite Connector UI. See New NetSuite Connector User Interface.	Contact OpenAir Customer Support

Step 2 — Read the Relevant Documentation

This guide provides a reference for using the NetSuite integration to exchange information between NetSuite and OpenAir. The guide is organized around the following conceptual topics for ease of reference:

- NetSuite Integration Main Features — High level overview of main integration features.
- How the NetSuite <> OpenAir Integration Works — High level overview of the information flow between NetSuite and OpenAir.
- Configuring NetSuite for the Integration — Reference material for configuring NetSuite to use both standard and optional integration features.
- Configuring OpenAir for the Integration — Reference material for configuring OpenAir to use both standard and optional integration features. Includes reference material for using the OpenAir NetSuite Connector UI to control what information is exchanged, scheduling the integration to run automatically, and configuring the real-time export of records from OpenAir to NetSuite.
- Running the NetSuite <> OpenAir Integration — Reference material for running the integration on demand for a single record or in bulk either manually or using OpenAir user scripting.
- Importing Data From NetSuite Into OpenAir — Reference material for all available integration workflows importing data from NetSuite into OpenAir.

- Exporting Data From OpenAir to NetSuite — Reference material for all available integration workflows exporting data from OpenAir to NetSuite, including custom export workflows.

Step 3 — Configure the Integration in NetSuite

See Configuring NetSuite for the Integration.

Data that can be imported from NetSuite into OpenAir include users, customers, contacts, projects, items, expense categories, vendors, purchase items, sales orders, expense report reimbursements, invoice payments, purchase orders, and customer deposits and credit memos applied to invoices. See Supported Record Types and Integration Operations.

Step 4 — Configure the Integration in OpenAir

See Configuring OpenAir for the Integration.

Data that can be exported from OpenAir to NetSuite include customers, projects approved expense reports, approved invoices, approved timesheets, and revenue recognition transactions. See Supported Record Types and Integration Operations.

Step 5 — Run the Integration

After NetSuite and OpenAir are configured for the integration, you can:

- Run the integration on demand or use OpenAir form and schedule scripts to trigger the integration. See Running the NetSuite <> OpenAir Integration.
- Schedule the integration to run automatically. See Scheduling the Integration to Run Automatically.
- Configure NetSuite to import records from NetSuite into OpenAir in real time. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- Configure OpenAir to export records from OpenAir to NetSuite in real time. See Configuring Real-Time Record Export from OpenAir to NetSuite.

How the NetSuite <> OpenAir Integration Works

This section describes the fundamentals of the NetSuite <> OpenAir integration. It is important that you read the following help topics before you start using OpenAir NetSuite Connector to exchange data between OpenAir and NetSuite.

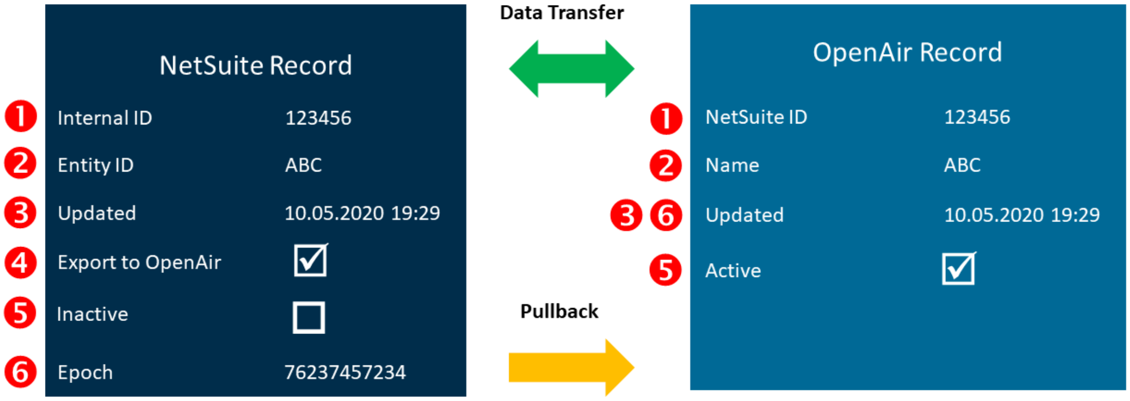
- OpenAir NetSuite Connector matches NetSuite records with OpenAir records.
    - **When importing a record from NetSuite into OpenAir**, the integration:
        - Creates a record in OpenAir, if a matching record does not already exist or cannot be found, and matches this new record with the NetSuite record.
        - Updates the matching record in OpenAir if a matching record exists and can be found.
    - **When exporting a record from OpenAir to NetSuite**, the integration:
        - Creates a record in NetSuite, if a matching record does not already exist or cannot be found, and matches this new record with the OpenAir record.
    - **When exporting a Project or Task record from OpenAir to NetSuite**, the integration:
        - Updates the matching record in NetSuite if a matching record exists and can be found.

        > ⚠ **Important:** Updating information for existing projects and tasks in NetSuite requires the following integration workflows to be enabled and active:
        >
        > - Update NetSuite Projects with OpenAir Project Data Export Workflow
        > - Update NetSuite Tasks with OpenAir Task Data

- OpenAir NetSuite Connector uses record information to match an OpenAir record with a NetSuite record, and to determine whether the matching record needs to be created or updated. See Integrated Records and Records Marked for Integration.
- Each OpenAir NetSuite Connector integration workflow let you exchange information between NetSuite records of a specific type and OpenAir record of a specific type. Each integration workflow either import NetSuite record information into OpenAir, or export OpenAir record information to NetSuite. For a list of supported NetSuite record types and OpenAir record types and supported integration operations as standard, see Supported Record Types and Integration Operations.
- Records of a specific type in both NetSuite and OpenAir may have dependencies on records of other types. For the NetSuite <> OpenAir integration to run smoothly, required associated records must be integrated first before a record with these dependencies can be integrated. To respect these dependencies, OpenAir NetSuite Connector runs integration workflows in a specific order and checks for dependencies when you enable integration workflows or run the integration. However, use caution when marking NetSuite records for import into OpenAir or setting filters for each integration workflows to make sure that all required records are integrated.
- Field mapping definitions for each OpenAir NetSuite Connector integration workflows specify what information is exchanged, for which records and how it is exchanged. See Field Mapping.

Integrated Records and Records Marked for Integration

OpenAir NetSuite Connector uses the following record information to match an OpenAir record with a NetSuite record, and to determine whether the matching record needs to be created or updated.



#	NetSuite Field	OpenAir Field	Description
1	**Internal ID**	**NetSuite ID** custom field	OpenAir NetSuite Connector creates the **NetSuite ID** custom field in OpenAir and associates it with the OpenAir record type when you enable the relevant integration workflow for the first time. The **NetSuite ID** custom fields on the OpenAir record stores the **Internal ID** of the NetSuite record. It is used to map an OpenAir record with a NetSuite record in a 1:1 relationship. For some records, the custom field label includes the NetSuite record type. For example, **NetSuite Customer ID** for integrated customer records. ■ For records imported from NetSuite in to OpenAir: □ The NetSuite **Internal ID** is imported with the other record data when the integration creates a new record in OpenAir. □ The **NetSuite ID** custom field is used to find the matching OpenAir record when the integration updates an existing record in OpenAir. ■ For records exported from OpenAir to NetSuite: □ The **Internal ID** of the new NetSuite record is pulled back when the integration creates a new record in NetSuite. □ The **NetSuite ID** custom field is used to find the matching NetSuite record when the integration updates an existing project or project task record in NetSuite. You can enter the **NetSuite ID** in the OpenAir record. You may need to do so for maintenance purposes. For example, if you run the integration to match existing records in NetSuite and OpenAir based on the `entityId` in NetSuite and `name` or `nickname` in OpenAir and there are discrepancies between the `entityId` in NetSuite and `name` or `nickname`, the integration creates a new record for the same entity (for example, a user or customer). In this case, you can delete the new record created in OpenAir by the integration and enter the **NetSuite ID** on the original OpenAir record to match it with the corresponding NetSuite ID.

#	NetSuite Field	OpenAir Field	Description
			Tip: Add the **NetSuite ID** as a column to your list views in OpenAir. You can use the **NetSuite ID** to identify integrated records or to locate a specific integrated record on your list views.
2	**Entity ID**	**Name**	When the integration updates an existing record in OpenAir (import from NetSuite into OpenAir), OpenAir NetSuite Connector uses the **NetSuite ID** custom field to find the matching OpenAir record. If there are no matching results, it uses the `entityId` on the NetSuite record and the `name` or `nickname` on the OpenAir record to find a matching record. For example, if you want the integration to match employee records in NetSuite, with existing user records in OpenAir, make sure the **Employee ID** (`Employee.entityId`) on the NetSuite record and the **User ID** (`User.nickname`) on the OpenAir record are the same. If you want the integration to match customer records in NetSuite, with existing customer records in OpenAir, make sure the **Customer ID** (`Customer.entityId`) on the NetSuite record and the **Nickname** (`Customer.name`) on the OpenAir record are the same. **Note:** The `entityId` and `name` or `nickname` is case insensitive. If the case of the `entityId` is changed in NetSuite, the case of the `name` or `nickname` will be changed accordingly in OpenAir.
3	**Updated**	**Updated**	Only the integrated records modified since the last integration run are included. OpenAir NetSuite Connector compares the **Updated** timestamp on the NetSuite record and the OpenAir record to determine which records to include in the integration. **Note:** You can run a full synchronization to synchronize all integrated records for all or selected workflows. See Running a Full Synchronization on Demand.
4	**Export to OpenAir**		Check the **Export to OpenAir** box on the NetSuite record to mark the NetSuite records you want to import into OpenAir. **Note:** You can configure NetSuite to check the **Export to OpenAir** box by default for new records using the **Export <Record Type> by Default** NetSuite OpenAir SRP Integration custom preference. See Configuring Integration Preferences and Terminology Preferences in NetSuite. You can configure OpenAir NetSuite Connector to import NetSuite records that are not marked for export to OpenAir as well as those marked for export to OpenAir using the **Disable Export to OpenAir filtering for <Record Type> in NetSuite integration** workflow setting. See Workflow Settings. **Tip:** Add the **Export to OpenAir** field as a column to your list views in NetSuite and use inline-editing to check or clear the **Export to OpenAir** directly from your list views.
5	**Inactive**	**Active**	By default, NetSuite records marked as **Inactive** are not imported into OpenAir. You can use the **Enable import of inactive <Record Type>** workflow setting to import NetSuite records marked as **Inactive** as well as active NetSuite records.

#	NetSuite Field	OpenAir Field	Description
			Note: If you check the **Inactive** box on a NetSuite record, make sure you edit the corresponding OpenAir record and clear the **Active** box, if required. If you delete a NetSuite record, make sure you delete the corresponding OpenAir record or clear the **Active** box on the OpenAir record, according to your company policies.
6	**Epoch**		If the latest version of NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) is installed on your NetSuite account, a hidden custom field for Customer records and Project Task records is used to store a timestamp (**epoch**) whenever any of the monitored field values are changed. This timestamp is used to import customer and project task records into OpenAir only if one of the monitored field values have changed. See Monitored Fields and Enabling Monitored Fields for Import Workflows.
6	*Any*	*Any*	You can set filters for each integration workflow to import selected NetSuite records into OpenAir, or export selected OpenAir records to NetSuite. You can use any of the information on the source record in your filter criteria, including standard fields and custom fields native to the source record type, or information on related other record types. See Using Filters.

Supported Record Types and Integration Operations

The following table lists the NetSuite record types and OpenAir record types available for integration as standard, together with information about how NetSuite record types and OpenAir record types can be paired and the integration operations supported for each pair.

NetSuite Record	OpenAir Record	Import from NetSuite Create & Update	Export OpenAir > NetSuite Create	Export OpenAir > NetSuite Update
Customer	Customer	Customers (Customers) Import Workflow	Customers (Customers) Export Workflow	
Credit Memo	Invoice		Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Export Workflow	
Contact	Contact	Contacts (Contacts) Import Workflow		
Custom Record	Revenue Recognition Rule		OpenAir Revenue Recognition with NetSuite Advanced Revenue Management	
Custom Record	Revenue Recognition Transaction		OpenAir Revenue Recognition with NetSuite Advanced Revenue Management	

NetSuite Record	OpenAir Record	Import from NetSuite Create & Update	Export OpenAir > NetSuite Create	Export OpenAir > NetSuite Update
Employee	Employee	Employees (Employees) Import Workflow		
Expense Category	Item	Expense Items (Expense Categories) Import Workflow		
Expense Report	Expense Report		Expense Reports (Expense Reports) Export Workflow	
Job	Project	Projects (Projects) Import Workflow	Projects (Projects) Export Workflow	Update NetSuite Projects with OpenAir Project Data Export Workflow
Invoice	Invoice		Invoices (Invoices) Export Workflow	
Item	Category	Services (Services Items) Import Workflow		
Item	Product	Products (Purchase Items) Import Workflow		
Journal Entry	Revenue Recognition Transaction		Configuring Revenue Recognition	
Project Task	Project Task	Tasks (Project Tasks) Import Workflow		Update NetSuite Tasks with OpenAir Task Data
Opportunity Item	Project Task		WBS (Opportunity) Export Workflow	
Purchase Requisition	Purchase Request		Purchase Requests (Purchase Requisitions) Export Workflow	
Support Case	Project Task	Tasks (Support Cases) Import Workflow		
TimeBill	Task		Time Entries (Time) Export Workflow	
Transaction (Credit Memo)	Invoice	Credits (Applied Credit Memos) Import Workflow		
Transaction (Customer Deposit)	Retainer	Retainers (Applied Customer Deposits) Import Workflow		
Transaction (Customer Payment)	Payment	Payments (Invoice Payments) Import Workflow		

NetSuite Record	OpenAir Record	Import from NetSuite Create & Update	Export OpenAir > NetSuite Create	Export OpenAir > NetSuite Update
Transaction (Purchase Order)	Purchase Order	POs (Purchase Orders) Import Workflow		
Transaction (Reimbursement)	Reimbursement	Reimbursements (Expense Report Reimbursements) Import Workflow		
Transaction (Sales Order)	Customer PO	Customer POs (Sales Orders) Import Workflow		
Transaction (Vendor Bill)	Purchase Order	POs (Vendor Bills) Import Workflow		
Vendor	User	Employees (Vendors) Import Workflow		
Vendor	Vendor	Vendors (Vendors) Import Workflow		

Field Mapping

The information exchanged for each record is determined by the field mapping definitions for the relevant integration workflow.

- The NetSuite record **Internal ID** is always mapped to a custom field created on the OpenAir record. This mapping is used to match a specific NetSuite record to a specific OpenAir record. See Integrated Records and Records Marked for Integration
- Default field mapping definitions are provided for available for each integration workflow. Default field mapping definitions can be changed or disabled to meet your unique business requirements. The default mapping definitions are available as part of the reference information for each integration workflow under Importing Data From NetSuite Into OpenAir and Exporting Data From OpenAir to NetSuite.
- Any NetSuite fields exposed by the NetSuite communication endpoint or any custom fields created on the NetSuite record can be mapped to any OpenAir field listed in the OpenAir data dictionary or custom field associated with the OpenAir record type. See Mapping OpenAir Fields and NetSuite Fields.
- Advanced mappings can be created to meet complex custom requirements. See Using the Advanced Tag Notation.

Configuring NetSuite for the Integration

OpenAir Professional Services does the initial configuration so you can use the NetSuite <> OpenAir integration.

The following help topics describe how to configure NetSuite for the integration, and are provided for reference.

- NetSuite Features Required for the Integration
- NetSuite Bundles for the Integration
- Enabling the OpenAir Integration in NetSuite
- Enabling Access to OpenAir From Within NetSuite
- Configuring Real-Time Record Import from NetSuite into OpenAir
- Enabling Monitored Fields for Import Workflows
- Configuring Integration Preferences and Terminology Preferences in NetSuite

⚠ **Important:** Review the following guidelines:

- You should check the list of Installed Bundles regularly and update the integration bundles installed on your NetSuite account to the latest available version, to take advantage of the latest NetSuite integration features.
- You should test any configuration changes on sandbox accounts before implementing them on your production accounts.
- You can contact OpenAir Professional Services if you need to make changes to your integration configuration.

NetSuite Features Required for the Integration

The following table lists the NetSuite features that are required for the NetSuite <> OpenAir integration and for optional integration features.

To enable any of these features, go to Setup > Company > Setup Tasks > Enable Features in NetSuite and click the relevant subtab. You must be signed in as an account administrator to access these settings.

⚠ **Important:** See also Feature Compatibility Check List for other NetSuite features requiring special considerations to work with the NetSuite <> OpenAir integration.

Feature	Subtab	Subtab section	Required / Optional	Additional information
Projects	Company	Projects	Required	—
Project Management	Company	Projects	Optional	Strongly recommended. With this feature: ■ You can create and manage Customers and Projects in separate lists of records. Without the Project Management feature,

Feature	Subtab	Subtab section	Required / Optional	Additional information
				you must create new projects from the customer record. - Customer and Project show as separate fields on Sales transaction records. - Saved searches can be created for Projects instead of subcustomer records. - The Project Profitability report is available to report on transactions exported from OpenAir to NetSuite. - Utilization reports are available for time exported from OpenAir to NetSuite. - You can use advanced integration settings that are not otherwise available, including Project Revenue Rules or Project Task Integration, for example. See Projects (external link to NetSuite SuiteAnswers | Answer ID: 61148) for a summary of functionality supported by the Projects and Project Management features.
Custom Records	SuiteCloud	SuiteBuilder	Required	—
Client SuiteScript	SuiteCloud	SuiteScript	Required	—
Server SuiteScript	SuiteCloud	SuiteScript	Required	—
SOAP Web Services	SuiteCloud	SuiteTalk (Web Services)	Required	—
Token-Based Authentication	SuiteCloud	Manage Authentication	Required	As of the October 10, 2020 OpenAir release, token-based authentication is the only supported authentication method for the integration.
OAuth2.0	SuiteCloud	Manage Authentication	Optional	*Required to enable and use the NetSuite as OIDC Provider feature. See table entry below.*
NetSuite as OIDC Provider	SuiteCloud	Manage Authentication	Optional	Required to give users access to OpenAir from within the NetSuite user interface. See Enabling Access to OpenAir From Within NetSuite.

Feature	Subtab	Subtab section	Required / Optional	Additional information
				⚠ **Important:** Until the OpenAir 2023.2 release, OpenAir Within NetSuite relied on the NetSuite Outbound Single Sign-On (SuiteSignOn) feature as outbound single sign-on method. NetSuite plans to end support for SuiteSignOn with the NetSuite 2024.2 for production accounts. Support for SuiteSignOn has already ended on non-production accounts. See Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn.

NetSuite Bundles for the Integration

The following table lists the NetSuite bundles that are required for the NetSuite <> OpenAir integration and for optional integration features.

To install any of these bundles, go to Customization > SuiteBundler > Search & Install Bundles in NetSuite and search for “OpenAir” to show the available bundles. You must be signed in as an account administrator to access these settings. See Installing a Bundle (external link to NetSuite SuiteAnswers | Answer ID: 10223).

You should check the list of Installed Bundles regularly and update the integration bundles installed on your NetSuite account to the latest available version, to take advantage of the latest NetSuite integration features. A version history for each bundles summarizes the changes included in each release version – See NetSuite Bundle Version History.

> ⚠ **Important:** Review the latest bundle upgrade guidelines before you update NetSuite bundles for the integration. See Important Notes and Guidelines About Bundle Updates.

Bundle name	Bundle ID	Legacy bundle ID	Required / Optional	Key components	Additional Information
NetSuite OpenAir SRP Integration	369637	2851	Required	■ Custom fields and custom scripts for the OpenAir integration. ■ Custom role “NSOA Integration”. ■ Custom Preferences for the OpenAir integration.	—
NetSuite OpenAir SRP Single Sign-on	369619	13512	Optional	■ PSA center tab. ■ 12 Suitelet SuiteScripts, one for each OpenAir module, one for the Support page, and one for the OpenAir Help center.	Required to give users access to OpenAir from within the NetSuite user interface. Enabling Access to OpenAir From Within NetSuite.

Bundle name	Bundle ID	Legacy bundle ID	Required / Optional	Key components	Additional Information
				■ One portlet SuiteScript for the OpenAir Home module. ■ One subtab to show the OpenAir Timesheet module as a subtab on Support Case records in NetSuite, when the Tasks (Support Cases) Import Workflow is enabled. ■ Custom preferences for the OpenAir NetSuite Single Sign-On integration.	
NetSuite OpenAir SRP Real-time Add-on	369630	12610	Optional	■ Two user event SuiteScripts to send NetSuite records to OpenAir (SuiteScript 1.0 and SuiteScript 2.0 versions) ■ Custom preferences for the NetSuite > OpenAir real-time integration.	Required for the real-time import of records from NetSuite in to OpenAir. See Configuring Real-Time Record Import from NetSuite into OpenAir.
NetSuite OpenAir SRP ARM	370366	217172	Optional	■ Custom fields and records ■ OpenAir: ARM Plug-in Implementation". ■ Three user event SuiteScripts. ■ Subtab for OpenAir transactions	Required to export OpenAir revenue recognition rules and transactions to NetSuite for use with the Advanced Revenue Management feature in NetSuite.

Important Notes and Guidelines About Bundle Updates

Review the following guidelines before you update your NetSuite bundles for the integration:

- After you update an integration bundles in NetSuite, do the following:
  - Go to Administration > NetSuite Connector in OpenAir, click the Tips button then **Synchronize NetSuite custom fields**.
  - Verify integration script deployments settings and user event script running order. See Bundle SuiteScript Deployment Guidelines.
- Update NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) to Version 4.0 or later before you update NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619) to version 7.0 or later, or NetSuite OpenAir SRP Real-time Add-On (Bundle ID 369630) to version 3.0 or later. NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) 4.0 updates the `OpenAir: SRP Integration Settings` custom record type with settings required for using the upgraded version of.
- Update the NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) to version 3.0 or later before you update NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619) to version 6.0 or later, or

NetSuite OpenAir SRP Real-time Add-On (Bundle ID 369630) to version 2.03 or later. NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) 3.0 creates a custom record type and a custom record that are required when updating the other bundles.

- The following bundle versions include SuiteScript 2 versions of all scripts. After you update the bundles, you can switch to the SuiteScript 2 versions of scripts in each bundle, or continue to use the updated SuiteScript 1.0 versions. See Switching to SuiteScript 2 Versions of Integration Bundle Scripts.
    - NetSuite OpenAir SRP Integration (Bundle ID 369637) Version 3.0 or later. Includes both SuiteScript 2.1 and SuiteScript 1.0 versions of scripts.
    - NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619) Version 6.0 or later. Includes both SuiteScript 2.0 (`OpenAir: Home module portlet` script) or SuiteScript 2.1 (all other scripts) and SuiteScript 1.0 versions of scripts.
    - NetSuite OpenAir SRP Real-time Add-On (Bundle ID 369630) Version 2.03 or later. Includes both SuiteScript 2.0 and SuiteScript 1.0 versions of scripts
    - NetSuite OpenAir SRP ARM (Bundle ID 370366) Version 0.96 or later. Includes both SuiteScript 2.0 and SuiteScript 1.0 versions of scripts.
- Only the bundles listed in the table above are supported.

  The bundle search results may also show the OpenAir: WIP Accounting – Bundle ID 369609 (previously 17953) bundle. This bundle lets you transfer revenue and billing between subsidiaries while work is in progress, using a scheduled script to generate intercompany journal entries that show incremental revenue accrual on balance sheets. This functionality requires a NetSuite OneWorld account. Access to this bundle is restricted. The bundle was created for a specific use case, it is not actively maintained and does not comply with the ASC 606 revenue recognition standard. It should not be installed on a production account. For more information about this bundle, contact OpenAir Professional Services.

Switching to SuiteScript 2 Versions of Integration Bundle Scripts

After you update the bundles, you can switch to the SuiteScript 2 versions of scripts in each bundle, or continue to use the updated SuiteScript 1.0 versions. All SuiteScript 1.0 versions that were deployed before the update are deployed by default after the update. When updating the NetSuite OpenAir SRP Real-time Add-On from version 2.0, 2.01, or 2.02, SuiteScript 1.0 or SuiteScript 2.0 versions that were deployed before the update are deployed by default after the update.

> **Note:** When updating NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619) to version 7.0 or later, the switch to SuiteScript 2 versions of scripts in this bundle is done automatically.

To switch to SuiteScript 2 versions of integration bundle scripts:

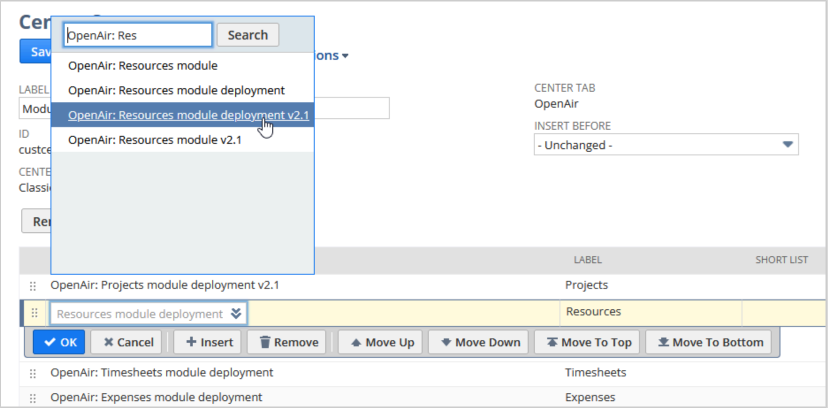
1. In NetSuite, go to Customization > Scripting > Scripts.
2. For each deployed SuiteScript 1.0 version script in any of the NetSuite bundles for the OpenAir integration:
    a. Locate the script in the list and click **Edit**.
    b. On the Deployments subtab, make a note of the script and all deployment list entries with a YES under **Deployed**. If the script applies to different record types or events, and is deployed only for specific record types or events, you will need to refer to this list to deploy the SuiteScript 2 equivalent later.
    c. For each list entry, clear the **Deployed** box then click **OK**.

   d. Click **Save**.

3. For each SuiteScript 2 version script in any of the NetSuite bundles for the OpenAir integration:
   a. Locate the script in the list and click **Edit**.
   b. On the Deployments subtab, for each deployment list entry equivalent to the SuiteScript 1.0 versions you undeployed earlier, check the **Deployed** box then click **OK**.
   c. Verify that the script is deployed for all roles and, in the case of portlet, Suitelet, or user event scripts, executed as the `NSOA Connector` or the custom role you created for the integration. See Verifying Integration Script Deployment Settings.
   d. Verify that all SuiteScript 2 equivalent to the SuiteScript 1.0 versions you undeployed earlier are deployed to ensure that you retain the same integration functionality.
   e. Click **Save**.

> **Important:** Review the Switch to SuiteScript 2 Versions Additional Guidelines.

4. Skip this step when updating NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619) to version 7.0 or later. Otherwise, after you switch to SuiteScript 2 versions of the NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619), update the category links for the OpenAir (PSA) center tab. To do so:
   a. Go to Customization > Centers and Tabs > Center Tabs.
   b. Locate the row for the OpenAir (PSA) center tab and click the center tab name.
   c. On the Content > Categories subtab, click **Edit** on the `Modules` row.
   d. For each line on the category link sublist, update the link to point to the SuiteScript 2.1 Suitelet script deployment.
      i. Click the name of SuiteScript 1.0 Suitelet script deployment under **Link** – `OpenAir: <Module> module deployment`, where `<Module>` is the name of the OpenAir module.
      ii. Search and select the name of the SuiteScript 2.1 Suitelet script deployment – `OpenAir: <Module> module deployment v2.1`.



OpenAir: Res

OpenAir: Resources module
OpenAir: Resources module deployment
OpenAir: Resources module deployment v2.1
OpenAir: Resources module v2.1

LABEL
Modu

ID
custce

CENTE
Classi

CENTER TAB
OpenAir

INSERT BEFORE
- Unchanged -

	LABEL	SHORT LIST
OpenAir: Projects module deployment v2.1	Projects	
Resources module deployment	Resources	
OpenAir: Timesheets module deployment	Timesheets	
OpenAir: Expenses module deployment	Expenses	

      iii. Click **OK**.
   e. Click **Save**.
   f. On the Content > Categories subtab, click **Edit** on the `Help & Support` row.
   g. For each line on the category link sublist, update the link to point to the SuiteScript 2.1 Suitelet script deployment (Use the same steps as for the Modules center category).
   h. Click **Save**.

5. Skip this step when updating NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619) to version 7.0 or later. Otherwise, after you switch to SuiteScript 2 versions of the NetSuite OpenAir SRP

Single Sign-On (Bundle ID 369619) or NetSuite OpenAir SRP Real-time Add-On (Bundle ID 369630) scripts, edit existing Portlet, Suitelet and User Event connection points in the SuiteSignOn record for OpenAir. To do so:

a. Go to Setup > Integration > SuiteSignOn.

> **Important:** Until the OpenAir 2023.2 release, the OpenAir Within NetSuite and Real-Time Record Import features relied on the NetSuite Outbound Single Sign-On (SuiteSignOn) feature as outbound single sign-on method. NetSuite plans to end support for SuiteSignOn with the NetSuite 2024.2 for production accounts. Support for SuiteSignOn has already ended on non-production accounts. If you are using older version of these features relying on SuiteSignOn, upgrade your integration configuration before the NetSuite 2024.2 release to ensure service continuity.
>
> See Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn.

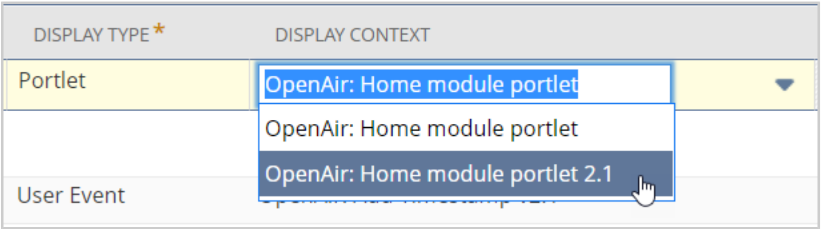
b. Click the name of the SuiteSignOn record for OpenAir.

c. Click **Edit**.

d. On the Connection Points subtab, for each Portlet, Suitelet and User Event connection points, do the following:

   i. Click the SuiteScript 1.0 script name under **Display Context**.

   ii. Select the SuiteScript 2 equivalent from the dropdown list.



DISPLAY TYPE *	DISPLAY CONTEXT
Portlet	OpenAir: Home module portlet
	OpenAir: Home module portlet OpenAir: Home module portlet 2.1
User Event	

   iii. Click **OK**.

e. Click **Save**.

6. If you are using real-time record imports from NetSuite into OpenAir, verify that the user event script sending the record to OpenAir runs after all other user event scripts for that record type. See Verifying User Event Scripts Running Order.

7. Verify the integration configuration settings for SuiteScript 2 versions of the bundle scripts.

> **Important:** When using SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637), NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619), or NetSuite OpenAir SRP Real-time Add-On (Bundle ID 369630) bundle scripts, integration configuration settings are sourced from the `OpenAir: SRP Integration Settings` custom record.
>
> When using SuiteScript 1 versions NetSuite OpenAir SRP Integration (Bundle ID 369637), NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619), or NetSuite OpenAir SRP Real-time Add-On (Bundle ID 369630) bundle scripts, integration configuration settings are sourced from Setup > Company > General Preferences (Custom Preferences tab).

To do so:

a. Go to Customization > Lists, Records and Fields > Record Types.

b. Locate `OpenAir: SRP Integration Settings` and click **list**.

   The list for the `OpenAir: SRP Integration Settings` record type appears and should include one record only (`OpenAir SRP Integration Settings Record`).

c. Click **View**.

d. Verify the settings and edit the record to change any settings, if required.

Switch to SuiteScript 2 Versions Additional Guidelines

In most cases, upgraded versions of the script have the same name as their SuiteScript 1.0 equivalent followed by the version number (`v2.1` or `2.1` for example).

Note the following exceptions:

- There are no SuiteScript 1.0 equivalent for `OpenAir: Add Timestamp v2.1`. Only the SuiteScript 2.0 version was present in previous versions of the bundle. The script has been upgraded to SuiteScript 2.1 and are deployed by default. There are no actions required for this script.
- `OpenAir: 'Export to OpenAir' Flags v2.1` is equivalent to the SuiteScript 1.0 `OpenAir: Set 'Export to OpenAir' Flags`
- `OpenAir: Set 'Limit Time...' v2.1` is equivalent to the SuiteScript 1.0 `OpenAir: Set 'Limit Time and Expenses'`
- `OpenAir: 'Limit Time...' Client v2.1` is equivalent to the SuiteScript 1.0 `OpenAir: Set 'Limit Time and Exp' Client`
- `OpenAir: Rev. Rec. % Complete v2.1` is equivalent to the SuiteScript 1.0 `OpenAir: Rev Rec PCO`
- The SuiteScript 1.0 `OpenAir: SFA Integration` has no SuiteScript 2 equivalent. Support for this feature ended with the April 2021 OpenAir release.

Most of your integration settings and preferences are copied to the `OpenAir: SRP Integration Settings` custom record so they can be maintained after you update the bundles and switch to SuiteScript 2 versions.

Note the following exceptions:

- **NetSuite OpenAir SRP Integration**
  - After you deploy the `OpenAir: PO/VB/VC Validate v2.1` script, you need to change preferences at the user level to let employees view projects and items and check the **Billable** and **Export to OpenAir** boxes for purchase order (or vendor bill and vendor credit) line items and purchase order (or vendor bill and vendor credit) records. To do so, go to Home > Set Preferences and check the **Validate OpenAir PO/Vendor Bill 2** box on the Custom Preferences tab.
  - After you deploy the `OpenAir: Create JE from Time v2.1` script, you need to change the parameters for this script deployment manually to match the settings for the previous SuiteScript 1.0 deployment. To do so, go to Customization > Scripting > Script Deployments, locate the deployment for the `OpenAir: Create JE from Time v2.1` script and click **Edit**, go to the Parameters subtab and edit the parameters.

Bundle SuiteScript Deployment Guidelines

When installing a bundle, or deploying a script included in one of the NetSuite bundle for the integration, you should ensure that:

- The script is deployed for all roles and, in the case of portlet, Suitelet, or user event scripts, executed as the `NSOA Connector` or the custom role you created for the integration. See Verifying Integration Script Deployment Settings.
- If you are using real-time record imports from NetSuite into OpenAir, the user event script sending the record to OpenAir runs after all other user event scripts for that record type. See Verifying User Event Scripts Running Order.

Verifying Integration Script Deployment Settings

Deployed integration scripts must be deployed for all roles and, in the case of portlet, Suitelet, or user event scripts, executed as the `NSOA Connector` or the custom role you created for the integration.

You should use the following steps when deploying a script included in one of the NetSuite bundles for the integration, or when troubleshooting NetSuite records not imported into OpenAir, for example.

To verify integration script deployment settings:

1. Go to the script deployment record for a script and record type. To do so:
   a. Go to Customization > Scripting > Script Deployments.
   b. Use the filters to show only the relevant script deployments for a **Script**, a **Record type**, or both.

   ✓ **Tip:** If you are looking for script deployments for a script in a specific bundle, go to Customization > Scripting > Scripts, use filters to show only the scripts in the bundle (**From bundle**), and view the script. The script deployments for the script are listed in the Deployments subtab.

2. Click **Edit**.
3. Change the following settings:
   - **Deployed** – Check the box
   - **Execute as Role** – (Portlet, Suitelet and User event scripts) Select `NSOA Connector` or the custom role you created for the integration. See Creating a Custom Role for the Integration in NetSuite.
   - **Roles** (Audience subtab) – Check the **Select All** box.
4. Click **Save**.

Verifying User Event Scripts Running Order

If you are using real-time record imports from NetSuite into OpenAir, the user event script sending the record to OpenAir (`OpenAir: Send Record to OpenAir` or `OpenAir: Send Record to OpenAir 2.0`) should run after all other user event scripts for that record type.

> ⚠ **Important:** The running order of user event scripts may change when you update a bundle in NetSuite. You should verify and, if necessary, change the script running order for all relevant scripted records in NetSuite. The `afterSubmit` user event script sending the record to OpenAir (`OpenAir: Send Record to OpenAir` or `OpenAir: Send Record to OpenAir 2.0`) should run after all other user event scripts for the record type.
>
> If the user event script for the real-time integration is not the last to run, the real-time integration may create duplicate records in some cases. The `OpenAir: Send Record to OpenAir` or `OpenAir: Send Record to OpenAir 2.0` triggers the real-time integration process. If an `afterSubmit` script runs afterward and modifies the record in NetSuite, this causes the real-time integration process to be triggered again before the previous process finishes, resulting in the creation of duplicate records.
>
> Effective with the OpenAir 2021.2 release, additional validation and a temporary record lock prevent the creation of duplicate records in OpenAir when importing a record from NetSuite to OpenAir using the real-time integration. OpenAir NetSuite Connector now checks for any processes already importing the record and locks the record until the import process completes so the same record cannot be imported by more than one process at the same time. This removes the script running order requirement and prevents the creation of duplicate records in OpenAir. However, it is best practice to verify that the OpenAir user event script for real-time integration is the last in the script running order.
>
> For more information about changing the running order of user event scripts, see The Scripted Records Page (external link to NetSuite SuiteAnswers | Answer ID: 10555.

To verify the running order of user event scripts for a record type:

1. Go to Customization > Scripting > Scripted Records > [*Select a scripted record type*].
2. Click `Edit`.
3. In the User Event Scripts subtab, move `OpenAir: Send Record to OpenAir 2.0` to the bottom of the sublist.
4. Click **Save**.

Enabling the OpenAir Integration in NetSuite

Use the following steps to enable the NetSuite <> OpenAir integration in NetSuite. These steps describe the minimum set up requirements to enable the integration. For information about configuring optional integration features, see the following help topics:

- Enabling Access to OpenAir From Within NetSuite
- Configuring Real-Time Record Import from NetSuite into OpenAir

To enable the OpenAir integration in NetSuite:

1. Enable the NetSuite features required for the integration. To do so:
   a. Sign in to NetSuite as an account administrator.
   b. Go to Setup > Company > Setup Tasks > Enable Features, and enable all the features required for the integration that are not already enabled, and any of the features required

for the optional integration features. For a list of NetSuite features required for the integration, see NetSuite Features Required for the Integration.

   c. Go to Setup > Integration > SOAP Web Services Preferences, check the **Run server SuiteScript and trigger workflows** box, and click **Save**.

> ⚠ **Important:** The **Run server SuiteScript and trigger workflows** preference must be enabled for OpenAir NetSuite Connector to fetch integration preferences configured in the NetSuite account during an integration run. This operation uses a SuiteScript triggered by a `beforeLoad` user event. This server SuiteScript is disabled when the **Run server SuiteScript and trigger workflows** SOAP web services preference is disabled and NetSuite returns empty or false values instead of the integration preferences. Empty or false integration preferences in turn may lead to integration errors. For more information about this and other company-wide SOAP web services preferences, see Company-Wide Preferences (External link to NetSuite SuiteAnswers | Answer ID: 10805).

2. Install the latest version of the NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle. To do so:
   a. Sign in to NetSuite as an account administrator.
   b. Go to Customization > SuiteBundler > Search & Install Bundles.
   c. Search for “NetSuite OpenAir SRP Integration”.
   d. Click **NetSuite OpenAir SRP Integration** from the search results.
   e. Click **Install**.

   See also Installing a Bundle (external link to NetSuite SuiteAnswers | Answer ID: 10223)

> ⓘ **Note:** NetSuite OpenAir SRP Integration (Bundle ID 369637) Version 3.0 or later includes SuiteScript 2.1 and SuiteScript 1.0 versions of all scripts. After you install or update the bundle from a previous version, SuiteScript 1.0 version of scripts are deployed by default. You can switch to the SuiteScript 2 versions, or continue to use the updated SuiteScript 1.0 versions. See Switching to SuiteScript 2 Versions of Integration Bundle Scripts.

3. Create an integration record for the OpenAir integration in NetSuite. See Creating an Integration Record for the OpenAir Integration.

> ⓘ **Note:** You do not need to create an integration record for the OpenAir integration if you are using the Direct Token-Based Authentication feature. When this feature is enabled, the NetSuite <> OpenAir integration uses an integration record that is added automatically by the latest version of the NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle and has the name “`Internal NS Application OPENAIR`”. See Direct Token-Based Authentication.

4. (Optional) Create a custom role for the OpenAir integration in NetSuite. See Creating a Custom Role for the Integration in NetSuite.
5. Verify that scripts in the bundle are deployed for all roles and executed as the `NSOA Connector` or the custom role you created for the integration. See Verifying Integration Script Deployment Settings.

> ✓ **Tip:** Go to Customization > Scripting > Scripts and use filters to show only scripts **From bundle** 369637 in the list.

6. Create and assign a Token-Based Authentication access token for the OpenAir integration in NetSuite. OpenAir will use this access token to establish a connection with NetSuite for the integration and send SOAP web services requests to your NetSuite account. See Creating and Assigning an Access Token for the Integration.

Creating an Integration Record for the OpenAir Integration

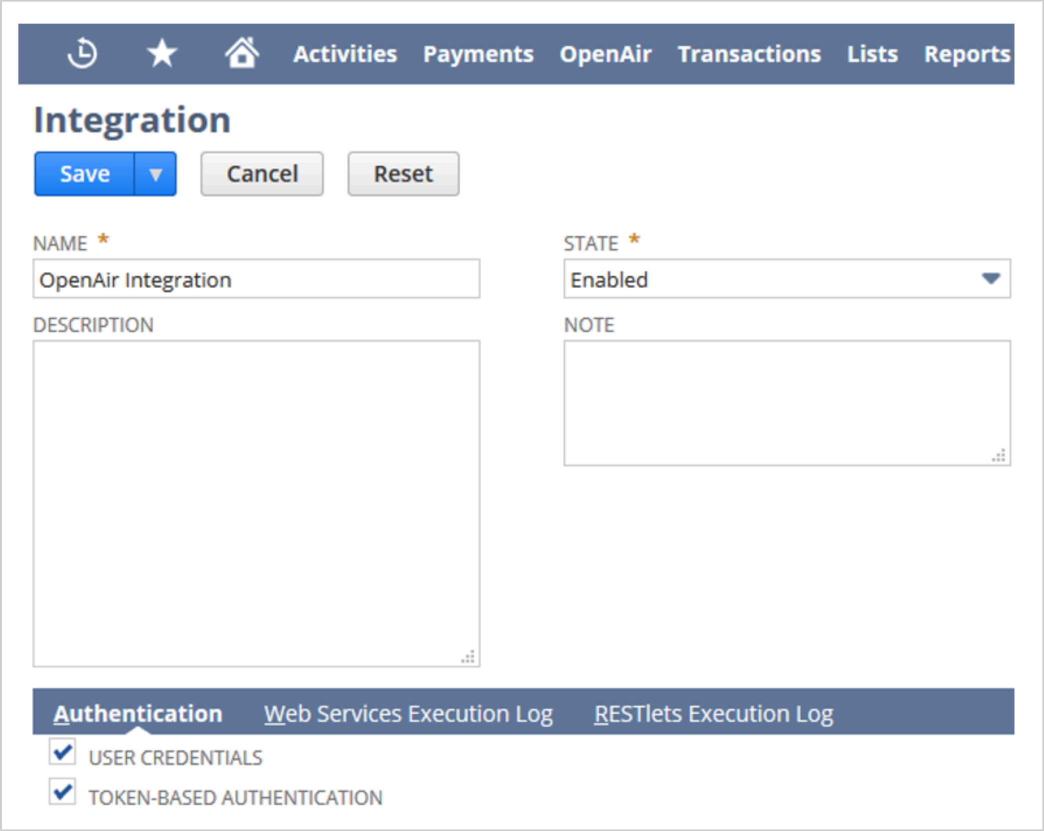
The following steps describe how to create an integration record for OpenAir. All external applications that use token-based authentication to send SOAP web services requests to your NetSuite account are represented by one or more integration records. You can use integration records to manage applications, view activity logs, and more.

> **Note:** You do not need to create an integration record for the OpenAir integration if you are using the Direct Token-Based Authentication feature. When this feature is enabled, the NetSuite <> OpenAir integration uses an integration record that is added automatically by the latest version of the NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle and has the name “`Internal NS Application OPENAIR`”. See Direct Token-Based Authentication.

To create an integration record for the OpenAir integration in NetSuite:

1. Sign in to NetSuite as an account administrator or a user with the Integration Application permission.
2. Go to Setup > Integration > Manage Integrations > New.

   The Integration form appears.



Integration

NAME * OpenAir Integration

STATE * Enabled

DESCRIPTION

NOTE

☑ USER CREDENTIALS

☑ TOKEN-BASED AUTHENTICATION

3. Enter a **Name** for the integration — for example, “OpenAir Integration”.
4. (Optional) Enter a **Description**.
5. Set **State** to “Enabled”.
6. (Optional) Enter a **Note**.

7. On the Authentication subtab, check the **User credentials** and **Token-based authentication** boxes.
8. Click **Save**.

   NetSuite generates the following information and displays it on the Integration page: Application ID, Consumer key and Consumer secret.
9. Copy the Application ID, Consumer key and Consumer secret and store the information in a safe place. You will need to enter this information when configuring OpenAir for the integration.

> ⚠ **Important:** The Consumer Key and Consumer Secret are displayed only one time. You will not be able to retrieve this information after you close, refresh or leave the page. You can generate a new Consumer key and Consumer secret at any time. To generate a new Consumer Key and Consumer Secret, go to the integration record for the OpenAir integration, click **Edit**, then click **Reset Credentials** and **OK** to confirm.

Creating a Custom Role for the Integration in NetSuite

The latest version of the NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle creates the “NSOA Connector” custom role for the integration. The NSOA Connector role includes all the permissions required for the integration. You can edit the NSOA Connector role or create your own custom role for the integration, if required.

To create a custom role for the Integration in NetSuite:

1. Sign in to NetSuite as an account administrator.
2. Go to Setup > Manager Users & Roles > Manager Roles and create a new role.
3. Do not add any restrictions to the header portions.
4. On the Permissions tab, set the appropriate permissions for the integration role. For a list of the required permissions and minimum permission levels, see NetSuite Permissions Required for the Integration. For a non-exhaustive list of optional permissions that can be set for the integration role, see Optional NetSuite Permissions for the Integration.

For more information about creating or modifying custom roles in NetSuite, see Customizing or Creating NetSuite Roles (External link to NetSuite SuiteAnswers | Answer ID: 9896).

NetSuite Permissions Required for the Integration

The following table lists the NetSuite permissions that are required for the integration with the level for each permission indicated in parentheses. The “NSOA Connector” role created by the NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle includes the permissions listed below. When creating a custom role for the integration, set all the following permissions and permission level at a minimum.

Transactions	Lists	Setup	Custom Record
Sales Order (Full)	Clients (Full)	Accounting Lists (full)	Features and Preferences (Full)
Credit Memo (Full)	Contacts (Full)	Certificate management (Full)	OpenAir: Exchange Rate Setting (Full)
Currency Revaluation (Full)	Currency (Full)	Enable Features (Full)	OpenAir: Exchange Rates (Full)
Expense Report (Full)	Custom Record Entries (Full)	Log in using OAuth 2.0 Access Tokens	OpenAir: List User Vendor (Full)

Transactions	Lists	Setup	Custom Record
Finance Charge (Full)	Documents and Files (Full)	OAuth 2.0 Authorized Applications Management	OpenAir: Project Templates (Full)
Find Transaction (Full)	Employees (Full)	OIDC Provider Setup	OpenAir: User Tax Nexus Type (Full)
Invoice (Full)	Expense Categories (Full)	SuiteBundler (Full)	—
Journal Approval (Full)	Items (Full)	SuiteScript (Full)	—
Make Journal Entry (Full)	Projects (Full)	SuiteSignOn (Full)	—
Posting Period on Transactions (View)	Vendors (Full)	User Access Tokens (Full)	—
Purchase Order (full)	—	Web Services (Full)	—

Optional NetSuite Permissions for the Integration

The following table lists some of the NetSuite permissions that can be used with the integration with the level for each permission indicated in parentheses. The list is provided for illustration purposes only, and is not exhaustive.

⚠ **Important:** Some role permissions may have an undesirable effect on the integration. For example, OpenAir transactions may be posted to a closed accounting period in NetSuite if the “Override Period Restrictions” role permission is enabled for the authenticated user.

See also Frequently Asked Questions (FAQs).

Transactions	Lists	Setup	Custom Record
Bills (Full)	Business Areas (View)	Accounting Preferences (Full)	Functional Area (Full)
Cash Sale (View)	Customers (Full)	Allow JS / HTML Uploads (Full)	LCCI (Full)
Customer Deposit (View)	Employee Record (Full)	Custom Body Fields (View)	OpenAir: Cost Center (Full)
Customer Payment (View)	Engagement Tasks (Full)	Custom Column Fields (View)	OpenAir: Prog/Proj Stages (Full)
Invoice Approval (Full)	Engagements (Full)	Custom Entity Fields (View)	OpenAir: Project Rate Card (Full)
Pay Bills (View)	MRUs (View)	Custom Fields (View)	OpenAir: Project Stages (Full)
Revenue Arrangement (Full)	Revenue Recognition Schedules (Full)	Custom Item Fields (View)	Profit Center (Full)
Track Time (Full)	Subsidiaries (View)	Custom Lists (View)	—
—	—	Other Lists (Full)	—
—	—	Set Up Company (Full)	—

Creating and Assigning an Access Token for the Integration

OpenAir NetSuite Connector uses token-based authentication (TBA) to establish a connection with NetSuite and send SOAP web services requests to your NetSuite account for the integration. After you create an integration record for OpenAir, you need to create an access token associated with the integration record for OpenAir, a designated user and the custom role created for the OpenAir integration. NetSuite will generate the **User Token ID** and **User Token Secret** required for OpenAir NetSuite Connector to access NetSuite.

For more information about the Token-Based Authentication (TBA) feature in NetSuite and about managing TBA tokens, see Token-based Authentication (TBA) (external link to NetSuite SuiteAnswers | Answer ID: 41827) and Manage TBA Tokens in the NetSuite UI (external link to NetSuite SuiteAnswers | Answer ID: 41902).

To create and assign an access token for the integration in NetSuite:

1. Sign in to NetSuite as an account administrator or as a user with the Access Token Management permission.
2. Choose the user to whom the access token will be assigned, or TBA user. You may use an existing employee record or create an employee record for the integration only. All authentication requests from OpenAir NetSuite Connector will show under this user in the logs.
3. Go to the Lists > Employees > Employees, locate the employee record for the TBA user and click **Edit**.
4. On the Access > Roles subtab, add the NSOA Connector role created by the NetSuite OpenAir SRP Integration bundle, or the custom role you created for the integration. See Creating a Custom Role for the Integration in NetSuite.
5. Go to Setup > Users/Roles > Access Tokens > New.
6. Select the **Application Name**.
   - If the **Enable Direct TBA** box is checked on the OpenAir NetSuite Connector administration form, select `Internal NS Application OPENAIR`. See Direct Token-Based Authentication.
   - Otherwise, select the name of the integration record you created for the OpenAir integration. See Creating an Integration Record for the OpenAir Integration.

   > ✓ **Tip:** Double the speed of your integration with an optimized link between OpenAir and NetSuite using Direct Token-Based Authentication. To create an access token for direct TBA select the following **Application Name**: `Internal NS Application OPENAIR`. You will also need to enable the feature on the OpenAir NetSuite Connector administration form in OpenAir. See Direct Token-Based Authentication.

7. Select the **User** — This is the **Name** on the employee record for the TBA user you chose.
8. Select the **Role** — You can use the NSOA Connector role created by the NetSuite OpenAir SRP Integration bundle, or the custom role you created for the integration. See Creating a Custom Role for the Integration in NetSuite.

   > ⚠ **Important:** You should not associate the access token with the Administrator role. Some role permissions may have an undesirable effect on the integration. For example, OpenAir transactions may be posted to a closed accounting period in NetSuite if the “Override Period Restrictions” role permission is enabled for the TBA user.
   >
   > See also Frequently Asked Questions (FAQs).

9. NetSuite generates the **Token Name** automatically. You can change the token name or keep the default value.

10. Click **Save**.

    NetSuite generates the **User Token ID** and **User Token Secret** and displays the information on the Access Token page.

11. Copy the **User Token ID** and **User Token Secret** and store the information in a safe place. You will need to enter this information when configuring OpenAir for the integration.

    > **Important:** The User Token ID and User Token Secret are displayed only one time. You will not be able to retrieve this information after you close, refresh or leave the page. You can generate a new User Token ID and User Token Secret at any time. To generate a new User Token ID and User Token Secret, revoke the existing token and create a new one.

Enabling Access to OpenAir From Within NetSuite

The OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature can be used to integrate NetSuite with OpenAir seamlessly through outbound single sign-on. You can give users of both NetSuite and OpenAir access to OpenAir directly from the NetSuite user interface without them having to sign in to OpenAir.

For an overview of the feature, see OpenAir Within NetSuite.

> **Warning:** The steps in this help topic describe how to enable and set up the upgraded version of the OpenAir Within NetSuite feature, which uses NetSuite as OIDC Provider as outbound single sign-on method instead of SuiteSignOn. For more information about NetSuite as OIDC Provider, see NetSuite as OIDC Provider (External link to NetSuite SuiteAnswer | Answer ID: 98241).
>
> Until the OpenAir 2023.2 release, OpenAir Within NetSuite relied on the NetSuite Outbound Single Sign-On (SuiteSignOn) feature as outbound single sign-on method. NetSuite plans to end support for SuiteSignOn with the NetSuite 2024.2 for production accounts. The functionality relying on SuiteSignOn may stop working at any time after its end-of-support date.
>
> Upgrade before the NetSuite 2024.2 release to ensure service continuity. For upgrade instructions, see Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn.
>
> You should test all integration configuration changes on sandbox NetSuite and OpenAir accounts before implementing them in your production accounts.

To enable access to OpenAir from within NetSuite:

1. In NetSuite, enable the OAuth 2.0 and NetSuite as OIDC feature. See Enable the NetSuite as OIDC Provider Feature (External link to NetSuite SuiteAnswers | Answer ID: 98270).
2. Install or update the following bundles. See Installing a Bundle (external link to NetSuite SuiteAnswers | Answer ID: 10223).
   - NetSuite OpenAir SRP Integration (Bundle ID: 369637) to version 4.0 or later.

     This bundle version:

     - Adds fields to the `OpenAir: SRP Integration Settings` custom record type. Some of the fields are required to enable OpenAir NetSuite Connector to use NetSuite as OIDC Provider for outbound single sign-on.
     - Adds the following permission to the `NSOA Connector` role:

        - OIDC Provider Setup (Full)
        - OAuth 2.0 Authorized Applications Management (Full)
        - Log in using OAuth 2.0 Access Tokens (Full)
        - Certificate management (Full)

    - NetSuite OpenAir SRP Single Sign-On (Bundle ID: 369619) to version 7.0 or later.

      This bundle version:

        - Updates SuiteScript 2 versions of Suitelet scripts to enable access to OpenAir modules in the PSA center tab using NetSuite as OIDC Provider.
        - Updates the `OpenAir: Home module portlet v2.0` Portlet script to enable access to the OpenAir dashboard as a custom portlet on the NetSuite dashboard using NetSuite as OIDC Provider.
        - Adds the script `OpenAir: Load subtab` to enable access to the OpenAir record as a subtab on the matching record in NetSuite for selected NetSuite record types using NetSuite as OIDC Provider.
        - Switches to using SuiteScript 2 versions of all scripts in the bundle automatically.

      > **Important:** Update NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) to Version 4.0 or later before you update NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619) to this version. NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) 4.0 updates the `OpenAir: SRP Integration Settings` custom record type with settings required for using the NetSuite as OIDC Provider for the OpenAir within NetSuite feature.
      >
      > Both bundles include SuiteScript 2 and SuiteScript 1.0 versions of all scripts.
      >
      > - After you install or update the NetSuite OpenAir SRP Integration (Bundle ID: 369637) bundle from a previous version, SuiteScript 1.0 version of scripts are deployed by default, unless you switched to SuiteScript 2 versions before the update. You can switch to the SuiteScript 2 versions, or continue to use the updated SuiteScript 1.0 versions. See Switching to SuiteScript 2 Versions of Integration Bundle Scripts.
      > - SuiteScript 1.0 versions of scripts in the NetSuite OpenAir SRP Single Sign-On (Bundle ID: 369619) bundle support only the SuiteSignOn method. OpenAir will end support for SuiteScript 1.0 versions of scripts in the NetSuite OpenAir SRP Single Sign-On (Bundle ID: 369619) bundle some time after the NetSuite 2024.2 release, when support for the SuiteSignOn feature ends (date to be confirmed).

3. If you are using your own custom role for the OpenAir integration instead of the “NSOA Connector” role created by the NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle, make sure that this custom role includes the following permissions and permission levels in the **Setup** secondary subtab. For more information, see Creating a Custom Role for the Integration in NetSuite.

    - OIDC Provider Setup (Full)
    - OAuth 2.0 Authorized Applications Management (Full)
    - Log in using OAuth 2.0 Access Tokens (Full)

4. Create a custom integration record for the OpenAir Within NetSuite feature and set it up for OpenAir NetSuite Connector to use NetSuite as OIDC Provider for outbound single sign-on. See Create Integration Records for Applications that Use NetSuite as OIDC Provider for Outbound Single Sign-on (External link to NetSuite SuiteAnswers | Answer ID: 98323).

a. Use the following settings under the OAuth 2.0 section of the integration record:

- **Authorization code grant** — Check the box. This is required for the NetSuite as OIDC Provider integration to work.
- **Redirect URI** — Enter one of the following URIs, depending on the type of OpenAir account type you are setting up for the integration:
  - □ Production account — `https://auth.openair.com/oidc/callback/internal_netsuite_<netsuite_account_id>`
  - □ Sandbox account — `https://auth.sandbox.openair.com/oidc/callback/internal_netsuite_<netsuite_account_id>`

  where <`netsuite_account_id`> is your NetSuite Account ID

  > **Important:** All alphabetical characters in <`netsuite_account_id`> must be lowercase. If your NetSuite Account ID includes uppercase alphabetical characters, replace them with the lowercase equivalent. When setting up the integration between OpenAir and NetSuite sandbox accounts, the NetSuite sandbox account ID typically includes the characters `SB` and should be replaced with the lowercase equivalent `sb`. For example, if your NetSuite Account ID is `345678_SB2`, the Redirect URI is `https://auth.sandbox.openair.com/oidc/callback/internal_netsuite_345678_sb2`.

- **OAuth2.0 consent policy** — You should select `Never Ask` so that the integration is autoapproved by an administrator. With other settings, users need to authorize the integration before being able to view OpenAir within NetSuite UI.

b. Copy the **Consumer key / Client ID** and **Consumer secret / Client secret** and store the information in a safe place. You will need to enter this information on the OpenAir NetSuite Connector credentials form in OpenAir for the NetSuite as OIDC Provider integration to work. See Enabling the OpenAir Within NetSuite Feature in OpenAir – Step 7.

5. Give users access to the NetSuite as OIDC Provider integration for the OpenAir Within NetSuite feature.

   a. Go to Setup > Integration > NetSuite as OIDC Provider Setup.

   b. Click the name of the custom integration record you created for the OpenAir Within NetSuite feature (Step 4).

      The Application Settings: <*Integration Name*> window appears.

   c. On the Entities tab and Employees subtab, check the **Select All** box to give all users access to the NetSuite as OIDC Provider integration for the OpenAir Within NetSuite feature.

   d. On the Roles tab, check the **Select All** box to give all roles access to the NetSuite as OIDC Provider integration for the OpenAir Within NetSuite feature.

   e. Click **Save**.

6. Add the **Custom Record Entries** permission with **View** permission level or higher to the relevant roles in NetSuite. Without this permission, users cannot access OpenAir from the NetSuite UI.

   For each role, do the following:

   a. Go to Setup > Users/Roles > User Management > Manage Roles > [*Select a role*].

   b. Click **Edit**.

   c. In the Permissions subtab, click the **Lists** secondary subtab.

   d. Add a row and select `Custom Record Entries` under **Permission**, and `View` under **Level**.

   e. Click **Add**.

   f. Click **Save**.

> **Note:** If you are using standard roles, you will need to customize standard roles, and assign these new custom roles to users to let these users access OpenAir from the NetSuite UI. For more information about managing roles in NetSuite, see Customizing or Creating NetSuite Roles (External link to NetSuite SuiteAnswers | Answer ID: 9896).

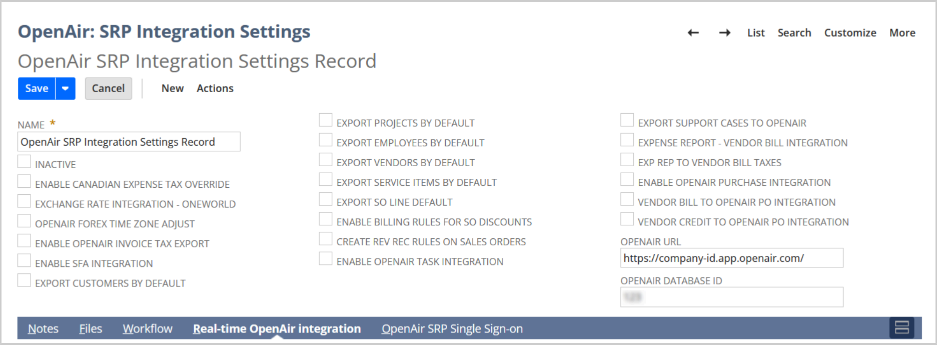
7. Update the preferences for the NetSuite <> OpenAir integration.

   a. Go to Customization > Lists, Records and Fields > Record Types > [**List** `OpenAir: SRP Integration Settings`] > [**Edit** *OpenAir SRP Integration Settings Record*].

   b. Enter the following information:

      - **OpenAir URL** — The URL for your OpenAir account. The URL must include the account-specific domain name for your OpenAir account. For more information about your account-specific domain name, see the help topic Your OpenAir Account URLs.

      - **OpenAir Database ID** — Your unique OpenAir account number. See the help topic Finding the Database ID for Your OpenAir Account.


← → List Search Customize More
Save
Cancel
New Actions
Notes Files Workflow Real-time OpenAir integration OpenAir SRP Single Sign-on

OpenAir: SRP Integration Settings

OpenAir SRP Integration Settings Record

NAME *
OpenAir SRP Integration Settings Record

☐ INACTIVE
☐ ENABLE CANADIAN EXPENSE TAX OVERRIDE
☐ EXCHANGE RATE INTEGRATION - ONEWORLD
☐ OPENAIR FOREX TIME ZONE ADJUST
☐ ENABLE OPENAIR INVOICE TAX EXPORT
☐ ENABLE SFA INTEGRATION
☐ EXPORT CUSTOMERS BY DEFAULT

☐ EXPORT PROJECTS BY DEFAULT
☐ EXPORT EMPLOYEES BY DEFAULT
☐ EXPORT VENDORS BY DEFAULT
☐ EXPORT SERVICE ITEMS BY DEFAULT
☐ EXPORT SO LINE DEFAULT
☐ ENABLE BILLING RULES FOR SO DISCOUNTS
☐ CREATE REV REC RULES ON SALES ORDERS
☐ ENABLE OPENAIR TASK INTEGRATION

☐ EXPORT SUPPORT CASES TO OPENAIR
☐ EXPENSE REPORT - VENDOR BILL INTEGRATION
☐ EXP REP TO VENDOR BILL TAXES
☐ ENABLE OPENAIR PURCHASE INTEGRATION
☐ VENDOR BILL TO OPENAIR PO INTEGRATION
☐ VENDOR CREDIT TO OPENAIR PO INTEGRATION

OPENAIR URL
https://company-id.app.openair.com/

OPENAIR DATABASE ID
[redacted]

   c. In the NetSuite OpenAir SRP Single Sign-On subtab.

      - (Optional) Change the default **Suitelet content height (px)** and **Suitelet content width (px)**. These values determine the size of the iFrame showing the OpenAir content in NetSuite for all users.

        > **Tip:** To utilize the whole of the available page space, set **Suitelet content height (px)** and **Suitelet content width (px)** in function of the browser window height and width. For example, set **Suitelet content height (px)** to `calc(100vh – 184px)` and **Suitelet content width (px)** to `calc(100vw – 39px)`.

      - (Optional) Enter any **terminology overrides** for OpenAir module names to match any custom terminology defined in OpenAir.

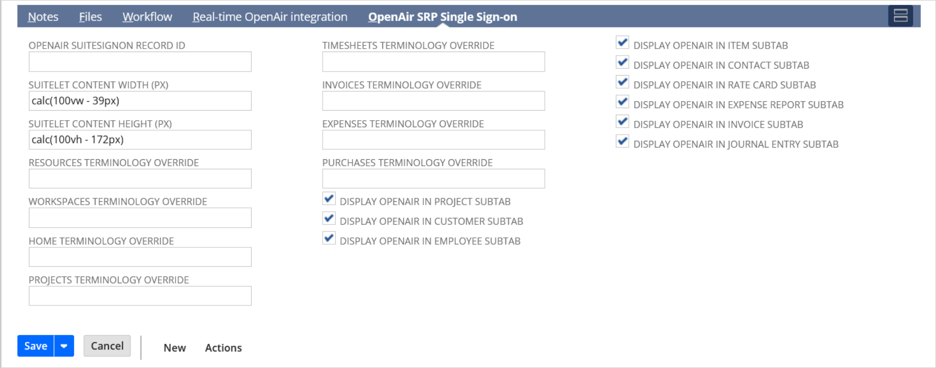
      - Check any of the boxes listed in the following table to show the OpenAir record as a subtab on the matching record in NetSuite. The table lists the integrations settings for each of the supported NetSuite record types and the corresponding types of OpenAir record shown in the OpenAir subtab.

Setting	NetSuite record type	OpenAir record type
Display OpenAir in Contact subtab	Contact	Contact
Display OpenAir in Customer subtab	Customer	Customer
Display OpenAir in Employee subtab	Employee	Employee

Setting	NetSuite record type	OpenAir record type
Display OpenAir in Expense report subtab	Expense Report	Expense Report
Display OpenAir in Invoice subtab	Invoice	Invoice
Display OpenAir in Journal entry subtab	Journal Entry	Revenue Recognition Transaction
Display OpenAir in Rate card subtab	`OpenAir: Project Rate Card` (custom record type)	Rate Card
Display OpenAir in Project subtab	Project	Project

Setting	NetSuite record type	OpenAir record type
Display OpenAir in Item subtab	Service Item	Category

> **Note:** Some of the configuration-based customization available when using the SuiteSignOn feature is not currently available when using the NetSuite as OIDC Provider feature. You would need to deploy custom SuiteScripts to achieve the following functionality.
>
> - Ability to show the Timesheets module on Support Case records as an OpenAir: Timesheets subtab in NetSuite when using the Tasks (Support Cases) Import Workflow.
> - Ability to set up custom OpenAir subtabs on NetSuite records and displaying OpenAir content other than those listed in the table above. With the SuiteSignOn feature, you could set up custom subtabs and use connection points to associate each custom subtab with a NetSuite record type and use integration variables to specify the OpenAir content shown in this subtab.
>
> If you want to deploy custom SuiteScripts to add support for custom subtabs and portlets, you can do the following:
>
> - Use the SuiteScripts included in the bundle as examples. Go to Customization > Scripting > Scripts, filter the list by bundle ID 369619 and review the script records.
> - Use the list of available integration parameters as reference. See Custom Subtabs and Portlets Integration Parameters.



OPENAIR SUITESIGNON RECORD ID

SUITELET CONTENT WIDTH (PX): calc(100vw - 39px)

SUITELET CONTENT HEIGHT (PX): calc(100vh - 172px)

RESOURCES TERMINOLOGY OVERRIDE

WORKSPACES TERMINOLOGY OVERRIDE

HOME TERMINOLOGY OVERRIDE

PROJECTS TERMINOLOGY OVERRIDE

TIMESHEETS TERMINOLOGY OVERRIDE

INVOICES TERMINOLOGY OVERRIDE

EXPENSES TERMINOLOGY OVERRIDE

PURCHASES TERMINOLOGY OVERRIDE

☑ DISPLAY OPENAIR IN PROJECT SUBTAB
☑ DISPLAY OPENAIR IN CUSTOMER SUBTAB
☑ DISPLAY OPENAIR IN EMPLOYEE SUBTAB
☑ DISPLAY OPENAIR IN ITEM SUBTAB
☑ DISPLAY OPENAIR IN CONTACT SUBTAB
☑ DISPLAY OPENAIR IN RATE CARD SUBTAB
☑ DISPLAY OPENAIR IN EXPENSE REPORT SUBTAB
☑ DISPLAY OPENAIR IN INVOICE SUBTAB
☑ DISPLAY OPENAIR IN JOURNAL ENTRY SUBTAB

d. Click **Save**.

8. Control which OpenAir modules are available from the PSA tab center submenu. See Managing OpenAir Modules Available from the PSA Center Tab.

9. Control who can access OpenAir from within NetSuite. See Controlling Access to the PSA Center Tab.

10. Enable the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature in OpenAir and allow NetSuite outbound single sign-on access for users. See Enabling the OpenAir Within NetSuite Feature in OpenAir.

Important: You must import user records from NetSuite into OpenAir in addition to explicitly allowing them to use the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature on the user demographic form in OpenAir.

Custom Subtabs and Portlets Integration Parameters

If you want to deploy custom SuiteScripts to add support for custom subtabs and portlets, you can do the following:

- Use the SuiteScripts included in the bundle as examples. Go to Customization > Scripting > Scripts, filter the list by bundle ID 369619 and review the script records.
- Use the following list of available integration parameters as reference.

Integration Parameters	Value	Required by
`app`	To show the default view for an OpenAir module in portlets or subtabs, set this variable to the two-letter code for that module. ▪ `km` — Workspaces ▪ `ma` — Account, Home, or Dashboard ▪ `pm` — Projects ▪ `po` — Purchases ▪ `rm` — Resources ▪ `ta` — Timesheets ▪ `tb` — Charges or Invoices ▪ `te` — Expenses To show the item or transaction record on a subtab for a NetSuite record, set this variable to the two-letter code for the relevant module. ▪ `ma` — Users, Customers, Contacts, Categories, Activities, or Services ▪ `pm` — Projects ▪ `tb` — Charges or Invoices ▪ `te` — Expenses	Custom subtabs and portlets
Additional integration parametes for specific record types:	Integration parameter for specific record type. Set the value for the record ID in NetSuite: `{id}`	Custom subtabs

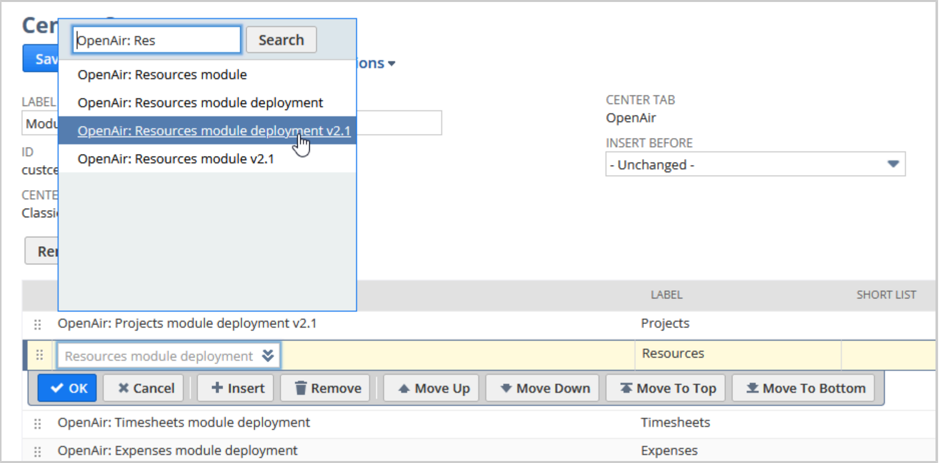
Integration Parameters	Value	Required by
- netsuite_employee_id - netsuite_customer_id - netsuite_project_id - netsuite_category_id - netsuite_company - netsuite_contact_id - netsuite_envelope_id - netsuite_invoice - netsuite_revenue_recognition_transaction	**Note:** A contact can be associated with more than one company in NetSuite. To refer to a specific contact, set both the company and contact IDs using the integration variables — `netsuite_company={company}, netsuite_contact_id={id}`	

Managing OpenAir Modules Available from the PSA Center Tab

You can remove OpenAir modules available from the PSA center tab, add these modules again, or change the module order in the submenu at any time.

To manage OpenAir modules available from the PSA center tab:

1. Go to Customization > Centers and Tabs > Center Tabs.
2. Locate the row for the PSA center tab and click the center tab name.
3. On the Content > Categories subtab, click **Edit** on the `Modules` row to edit the list of modules, or on the `Help & Support` row to edit the OpenAir help resources.
4. Do any of the following:
   - To remove a module, click the module, then **Remove**.
   - To change the module order, click a module and drag the row up or down the list.
   - To add a module again, select the **Link** for the relevant Suitelet script deployment, enter a **Label**, and click **Add**.



CENTER TAB
OpenAir

INSERT BEFORE
- Unchanged -

	LABEL	SHORT LIST
OpenAir: Projects module deployment v2.1	Projects	
Resources module deployment	Resources	
OpenAir: Timesheets module deployment	Timesheets	
OpenAir: Expenses module deployment	Expenses	

Cer
Sav
LABEL
Modu
ID
custc
CENTE
Classi
Re
OpenAir: Res
Search
ons ▾
OpenAir: Resources module
OpenAir: Resources module deployment
OpenAir: Resources module deployment v2.1
OpenAir: Resources module v2.1
✓ OK
✖ Cancel
+ Insert
Remove
Move Up
Move Down
Move To Top
Move To Bottom

> ⚠ **Important:** NetSuite OpenAir SRP Single Sign-on (Bundle ID 369619) Version 6.0 or later includes SuiteScript 2 and SuiteScript 1.0 versions of all scripts – Either of the script versions can be deployed but not both. The script deployment you select must correspond to the script deployed in your NetSuite account.
>
> - `OpenAir: <Module> module deployment` – SuiteScript 1.0 version.
> - `OpenAir: <Module> module deployment v2.1` – SuiteScript 2 version.
>
> NetSuite OpenAir SRP Single Sign-on (Bundle ID 369619) Version 7.0 or later is required and SuiteScript 2 versions of the bundle scripts must be deployed to enable access to OpenAir from within NetSuite using NetSuite as OIDC Provider. When you install NetSuite OpenAir SRP Single Sign-on (Bundle ID 369619) Version 7.0 or later, SuiteScript 1.0 versions of all scripts in the bundle are undeployed automatically and SuiteScript 2 versions are deployed automatically.
>
> SuiteScript 1.0 versions of scripts in the NetSuite OpenAir SRP Single Sign-On (Bundle ID: 369619) bundle support only the SuiteSignOn method. OpenAir will end support for SuiteScript 1.0 versions of scripts in the NetSuite OpenAir SRP Single Sign-On (Bundle ID: 369619) bundle some time after the NetSuite 2024.2 release, when support for the SuiteSignOn feature ends (date to be confirmed).

5. Click **Save**.

Controlling Access to the PSA Center Tab

By default, the PSA center tab is available to all roles. You can restrict access to the PSA center tab to specific roles, employees, subsidiaries, departments, groups, You can also grant access to specific or all customers, vendors and partners.

To control access to OpenAir from the PSA center tab in NetSuite:

1. Go to Customization > Centers and Tabs > Center Tabs.
2. Locate the row for the PSA center tab and click **Edit**.
3. On the Audience subtab, select specific **Roles**, **Employees**, **Subsidiaries**, **Departments**, **Groups**, **Customers**, **Vendors** and **Partners**, or check the **Select All** box to grant access to all users. The user must be either selected or part of a selected group to have access to the PSA center tab.

> ⚠ **Important:** Users must also have access to the NetSuite as OIDC Provider integration for the OpenAir Within NetSuite feature and the **Custom Record Entries** role permission with **View** permission level or higher to view OpenAir content from the NetSuite UI. See Enabling Access to OpenAir From Within NetSuite.

4. Click **Save**.

Configuring Real-Time Record Import from NetSuite into OpenAir

You can configure NetSuite so that NetSuite records are imported automatically in to OpenAir when the record is saved by any user in NetSuite. If a user has the relevant permissions to create or modify a record in NetSuite and saves the record, the record is imported in to OpenAir. The NetSuite user making the change does not need to be an OpenAir user.

The following NetSuite records can be imported in to OpenAir upon saving. The real-time import creates a new record in OpenAir if it is imported for the first time, and updates the OpenAir record if it was created during a previous import.

- Employees
- Customers
- Projects
- Cases (imported to OpenAir as project tasks)
- Project Tasks
- Items
- Expense categories
- Sales orders
- Credit memos
- Invoice payments
- Vendor bills
- Vendor credits
- Purchase orders
- Expense report reimbursements
- OpenAir: Cost Center (custom record)
- OpenAir: Project Rate Card (custom record)

Before using the following steps to configure real-time record import from NetSuite into OpenAir, review the Real-Time Import Functionality and Controls.

Warning: The steps in this help topic describe how to enable and set up the upgraded version of the Real-Time Record Import feature, which uses certificate-based authentication instead of SuiteSignOn to validate real-time record import requests.

Until the OpenAir 2023.2 release, Real-Time Record Import relied on the NetSuite Outbound Single Sign-On (SuiteSignOn) feature.

NetSuite plans to end support for SuiteSignOn with the NetSuite 2024.2 for production accounts. The functionality relying on SuiteSignOn may stop working at any time after its end-of-support date. Upgrade before the NetSuite 2024.2 release to ensure service continuity. For upgrade instructions, see Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn.

You should test all integration configuration changes on sandbox NetSuite and OpenAir accounts before implementing them in your production accounts.

To configure real-time record import from NetSuite into OpenAir:

1. In NetSuite, install or update the following bundles. See Installing a Bundle (external link to NetSuite SuiteAnswers | Answer ID: 10223).
   - NetSuite OpenAir SRP Integration (Bundle ID: 369637) – release version 4.0 or later.

     This bundle version:
     - Adds fields to the `OpenAir: SRP Integration Settings` custom record type. Some of the fields are required to enable OpenAir NetSuite Connector to use NetSuite as OIDC Provider for outbound single sign-on.
     - Adds the following permission to the `NSOA Connector` role:
       - OIDC Provider Setup (Full)
       - OAuth 2.0 Authorized Applications Management (Full)
       - Log in using OAuth 2.0 Access Tokens (Full)
       - Certificate management (Full)
   - NetSuite OpenAir SRP Real-time Add-on (Bundle ID 369630) – release version 3.0 or later.

     This bundle version:
     - Updates the `OpenAir: Send Record to OpenAir v2.0` user event script to import records from NetSuite into OpenAir in real-time using certificate-based authentication.
     - Adds the `custcertificate_oa_realtime` certificate record. The default certificate `openair-certificate.pem` associated with the `custcertificate_oa_realtime` certificate record is not valid. It is a required placeholder that will be replaced later (Steps 6 and 7).

⚠ **Important:** Update NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) to Version 4.0 or later before you update NetSuite OpenAir SRP Real-time Add-on (Bundle ID 369630) to version 3.0. NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) 4.0 updates the `OpenAir: SRP Integration Settings` custom record type with settings required for using certificate-based authentication for the Real-Time Record Import feature.

Both bundles include SuiteScript 2 and SuiteScript 1.0 versions of all scripts.

- After you install or update the NetSuite OpenAir SRP Integration (Bundle ID: 369637) bundle from a previous version, SuiteScript 1.0 version of scripts are deployed by default, unless you switched to SuiteScript 2 versions before the update. You can switch to the SuiteScript 2 versions, or continue to use the updated SuiteScript 1.0 versions. See Switching to SuiteScript 2 Versions of Integration Bundle Scripts.
- NetSuite OpenAir SRP Real-time Add-on (Bundle ID 369630) Version 3.0 or later is required and `OpenAir: Send Record to OpenAir v2.0` must be deployed to enable real-time record import from NetSuite into OpenAir using certificate-based authentication.

  `OpenAir: Send Record to OpenAir` (SuiteScript 1.0 version) supports only the SuiteSignOn method. `OpenAir: Send Record to OpenAir v2.0` (SuiteScript 2.0 version) supports both the SuiteSignOn and certificate-based authentication methods. OpenAir will end support for this SuiteScript 1.0 version some time after the NetSuite 2024.2 release, when support for the SuiteSignOn feature ends (date to be confirmed).

2. If you are using your own custom role for the OpenAir integration instead of the “NSOA Connector” role created by the NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle, make sure that this custom role includes the **Certificate management** permission (**Full** permission level) in the **Setup** secondary subtab. For more information, see Creating a Custom Role for the Integration in NetSuite.
3. Deploy `OpenAir: Send Record to OpenAir v2.0` for each record type you want to import into OpenAir using the real-time integration.

> **Note:** If you are updating NetSuite OpenAir SRP Real-time Add-on (Bundle ID 369630) bundle from version 1.x, undeploy the SuiteScripts 1.0 version of the script for all the record types before you deploy the SuiteScripts 2.0 version. To do so:
>
> 1. Go to Customization > Scripting > Scripts.
> 2. Locate the `OpenAir: Send Record to OpenAir` script and click **Edit**.
> 3. On the Deployments subtab, clear the **Deployed** boxes and click **OK** to confirm for all record types.
> 4. Click **Save**.
>
> See also Switching to SuiteScript 2 Versions of Integration Bundle Scripts.

a. Go to Customization > Scripting > Scripts.

b. Locate the `OpenAir: Send Record to OpenAir 2.0` script and click **View**.

> ✓ **Tip:** Use filters to show only the relevant scripts in the list. Set the **From bundle** filter to 369630.

c. Do the following for each record type you want to import into OpenAir using the real-time integration:

   i. In the Deployments subtab, click the record type name.

   The script deployment record appears.

   ii. Click **Edit**.

   iii. Change the following settings:

   - **Deployed** – Check the box
   - **Execute as Role** – Select `NSOA Connector` or the custom role you created for the integration. See Creating a Custom Role for the Integration in NetSuite.
   - **Roles** (Audience subtab) – Check the **Select All** box.

   iv. Click **Save**.

   v. Click the **Script** link to go back to the `OpenAir: Send Record to OpenAir 2.0` script record.

4. Verify that `OpenAir: Send Record to OpenAir 2.0`) runs after all other user event scripts for each record type you want to import into OpenAir using the real-time integration. See Verifying User Event Scripts Running Order.

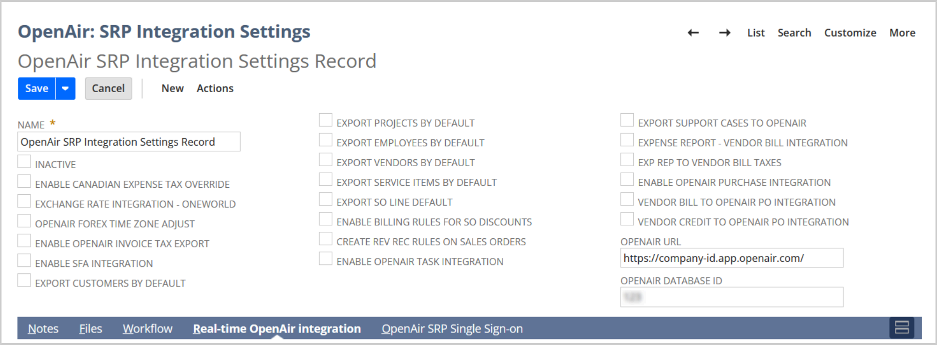
5. Update the preferences for the NetSuite <> OpenAir integration.

   a. Go to Customization > Lists, Records and Fields > Record Types > [**List** `OpenAir: SRP Integration Settings`] > [**Edit** *OpenAir SRP Integration Settings Record*].

   b. Enter the following information:

   - **OpenAir URL** — The URL for your OpenAir account. The URL must include the account-specific domain name for your OpenAir account. For more information about your account-specific domain name, see the help topic Your OpenAir Account URLs.

- **OpenAir Database ID** — Your unique OpenAir account number. See the help topic Finding the Database ID for Your OpenAir Account.


← → List Search Customize More
Save ▾ Cancel New Actions
Notes Files Workflow Real-time OpenAir integration OpenAir SRP Single Sign-on

OpenAir: SRP Integration Settings

OpenAir SRP Integration Settings Record

NAME *
OpenAir SRP Integration Settings Record

□ INACTIVE
□ ENABLE CANADIAN EXPENSE TAX OVERRIDE
□ EXCHANGE RATE INTEGRATION - ONEWORLD
□ OPENAIR FOREX TIME ZONE ADJUST
□ ENABLE OPENAIR INVOICE TAX EXPORT
□ ENABLE SFA INTEGRATION
□ EXPORT CUSTOMERS BY DEFAULT

□ EXPORT PROJECTS BY DEFAULT
□ EXPORT EMPLOYEES BY DEFAULT
□ EXPORT VENDORS BY DEFAULT
□ EXPORT SERVICE ITEMS BY DEFAULT
□ EXPORT SO LINE DEFAULT
□ ENABLE BILLING RULES FOR SO DISCOUNTS
□ CREATE REV REC RULES ON SALES ORDERS
□ ENABLE OPENAIR TASK INTEGRATION

□ EXPORT SUPPORT CASES TO OPENAIR
□ EXPENSE REPORT - VENDOR BILL INTEGRATION
□ EXP REP TO VENDOR BILL TAXES
□ ENABLE OPENAIR PURCHASE INTEGRATION
□ VENDOR BILL TO OPENAIR PO INTEGRATION
□ VENDOR CREDIT TO OPENAIR PO INTEGRATION

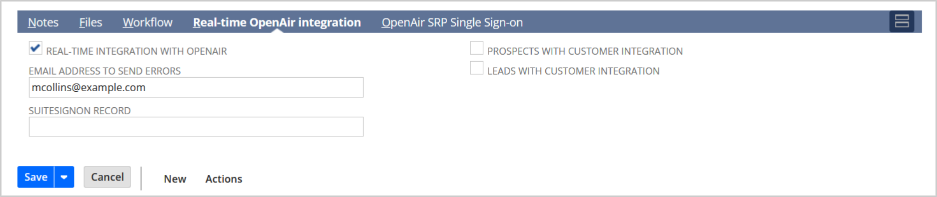
OPENAIR URL
https://company-id.app.openair.com/

OPENAIR DATABASE ID
[redacted]

c. In the NetSuite OpenAir SRP Real-Time Add-On subtab

- Check the **Real-time integration with OpenAir** box.
- Enter the email address to be used for error notifications (**Email address to send errors**).
- (Optional) Check the **Leads with customer integration** and **Prospects with customer integration** boxes to enable additional customer integration features for use with leads and prospects.

> **Note:** Make sure the following settings are enabled on the OpenAir NetSuite Connector administration form, before you enable these features in NetSuite:
>
> - Include NetSuite leads marked 'Export to OpenAir' in the customers import
> - Include NetSuite prospects marked 'Export to OpenAir' in the customers import
>
> Make sure that the real-time integration script is deployed for Leads (`customdeploy_oa_send_lead`) and Prospects (`customdeploy_oa_send_prospect`).


Notes Files Workflow Real-time OpenAir integration OpenAir SRP Single Sign-on
Save ▾ Cancel New Actions

☑ REAL-TIME INTEGRATION WITH OPENAIR

EMAIL ADDRESS TO SEND ERRORS
mcollins@example.com

SUITESIGNON RECORD

□ PROSPECTS WITH CUSTOMER INTEGRATION
□ LEADS WITH CUSTOMER INTEGRATION

d. Click **Save**.

6. Enable the Real-Time Record Import feature in OpenAir, and verify that a valid certificate was generated and saved to NetSuite and OpenAir. See Enabling the Real-Time Record Import Feature in OpenAir.

Note: OpenAir uses certificates to validate real-time record import requests.

Certificates are valid for a finite period. After the initial set up, OpenAir NetSuite Connector generates a new certificate and saves it to NetSuite and OpenAir automatically one to two weeks before the previous certificate expires. You can verify if the real-time record import certificate is valid, and generate a new certificate at any time in OpenAir. See Real-Time Record Import Certificate and Certificate Rotation.

Real-Time Import Functionality and Controls

Review the following functionality, controls and limitations:

- The OpenAir NetSuite Connector configuration determines real-time record imports from NetSuite into OpenAir:
  - The corresponding workflow must be enabled in the OpenAir NetSuite Connector administration form. See Imports.
  - The corresponding workflow must be activated on the OpenAir NetSuite Connector Mapping tab. See Adding and Managing Integration Workflows.
  - Real-time imports use the same field mapping definition as the corresponding workflow in manual and scheduled integration runs. See Mapping OpenAir Fields and NetSuite Fields.
- The real-time import also imports all records modified since the last synchronization. All updates are listed in the integration run log.
- Real-time imports are queued. Schedule integration runs take priority over real-time imports. Real-time imports will run after scheduled integration runs have completed. A message may appear indicating that the request was added to the queue and will run as soon as possible.
- Real-time synchronization is triggered as an `afterSubmit` user event when the user creates, or edits a record, either from the record form or from a list view using inline-editing.
- The running order of user event scripts may change when you update a bundle in NetSuite. You should verify and, if necessary, change the script running order for all relevant scripted records in NetSuite. The `afterSubmit` user event script sending the record to OpenAir (`OpenAir: Send Record to OpenAir` or `OpenAir: Send Record to OpenAir 2.0`) should run after all other user event scripts for the record type. See Verifying User Event Scripts Running Order.
- The real-time import only runs when a user clicks **Save** on the record form. It does not run if the record creation or update is automated. For example, if the data is imported into NetSuite from a CSV file, or added or updated using SuiteScript or SuiteFlow, the data is not transferred automatically to OpenAir using real-time import. Instead you need to schedule an integration run or trigger an integration run manually to import this data into OpenAir.
- If you are using the customer integration to capture changes to leads and prospects, the real-time import only runs if the lead or prospect is modified. The changes will be imported in to OpenAir if a lead or prospect changes to a customer or if a lead changes to a prospect. Other processes that change status automatically, such as using the lead conversion CRM feature, are not supported.
- If you use real-time integration for Project Tasks or Customers, you can improve the performance of real-time integration by enabling monitored fields for project tasks or customers integration. See Enabling Monitored Fields for Import Workflows.

Enabling Monitored Fields for Import Workflows

Note: The following workflows support the Monitored Fields feature:

- Customers (Customers) Import Workflow
- Employees (Employees) Import Workflow
- Projects (Projects) Import Workflow
- Tasks (Project Tasks) Import Workflow

You can speed up certain import workflows by ensuring that records are only included in the integration run if one or more integrated field values have changed.

With this feature enabled, the fields designated for integration are monitored for change. NetSuite records will be included in the integration run only if one of the monitored field values have changed.

For more information about designating fields for integration, see Configuring OpenAir for the Integration.

To enable monitored fields for import workflows:

1. Update the NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle to the latest version.
   - Monitored fields for Project tasks were introduced in Version 2.04. Version 2.04 of the bundle:
     - Creates a new custom record for Monitored Fields, which holds the list of fields to be updated to OpenAir.
     - Creates a new hidden custom field for project task records, which is used to store a timestamp whenever any of the monitored field values are changed.
     - Deploys a script to monitor listed fields and update the timestamp whenever any of the monitored field values are changed.
   - Monitored fields for Customers were introduced in Version 2.08. Version 2.08 of the bundle:
     - Creates a new hidden custom field for Customer records, which is used to store a timestamp whenever any of the monitored field values are changed.
   - Monitored fields for Employees and Projects were introduced in Version 3.0. Version 3.0 of the bundle:
     - Creates a new hidden custom field for Employee and Project records, which is used to store a timestamp whenever any of the monitored field values are changed.
2. If the real-time integration is installed and enabled on your account, update NetSuite OpenAir SRP Real-time Add-on (Bundle ID 369630) bundle to version 2.0 or later and deploy `OpenAir: Send Record to OpenAir v2.0` script (SuiteScript 2 version).

   See Installing NetSuite OpenAir SRP Real-time Add-on (bundle 12610) version 2.0 or higher.
3. Run the main integration to update the monitored fields used for the real-time integration. You must include the workflows supporting the Monitored Fields feature in the integration run. See Running the Bulk Import and Export Integration on Demand.

   Important: After you make changes to the field mapping for a workflow supporting the Monitored Fields feature, run the main integration and include that workflow to update the monitored fields used for the real-time integration.

Configuring Integration Preferences and Terminology Preferences in NetSuite

This help topic gives reference information for integration preferences and additional features and settings relevant to the NetSuite <> OpenAir integration.

You can control some of the NetSuite <> OpenAir integration settings from your NetSuite account.

To review or change the integration preferences, sign in to NetSuite as an account administrator and do one of the following:

- Go to Setup > Company > General Preferences (Custom Preferences tab), if you are using the SuiteScript 1 versions of bundle scripts in NetSuite OpenAir SRP Integration (Bundle ID 369637), NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619), or NetSuite OpenAir SRP Real-time Add-On (Bundle ID 369630) version 2.03 or later, or SuiteScript 2 versions of bundle scripts in NetSuite OpenAir SRP Real-time Add-On (Bundle ID 369630) version 2.02 or earlier. A new section becomes available on the Custom Preferences subtab after you install each OpenAir integration bundle.
- Go to Customization > Lists, Records and Fields > Record Types > [**List** `OpenAir: SRP Integration Settings`] > [**Edit** *OpenAir SRP Integration Settings Record*], if you are using the SuiteScript 2 versions of bundle scripts in NetSuite OpenAir SRP Integration (Bundle ID 369637), NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619), or NetSuite OpenAir SRP Real-time Add-On (Bundle ID 369630) version 2.03 or later. A new subtab becomes available on the record after you install each OpenAir integration bundle.

Depending on the bundles installed on your NetSuite account, the following sections or subtabs are available:

- NetSuite OpenAir SRP Integration
- NetSuite OpenAir SRP Single Sign-On
- NetSuite OpenAir SRP Real-Time Add-On

You can also make sure that both your NetSuite and OpenAir account use the same terminology. See Terminology

NetSuite OpenAir SRP Integration

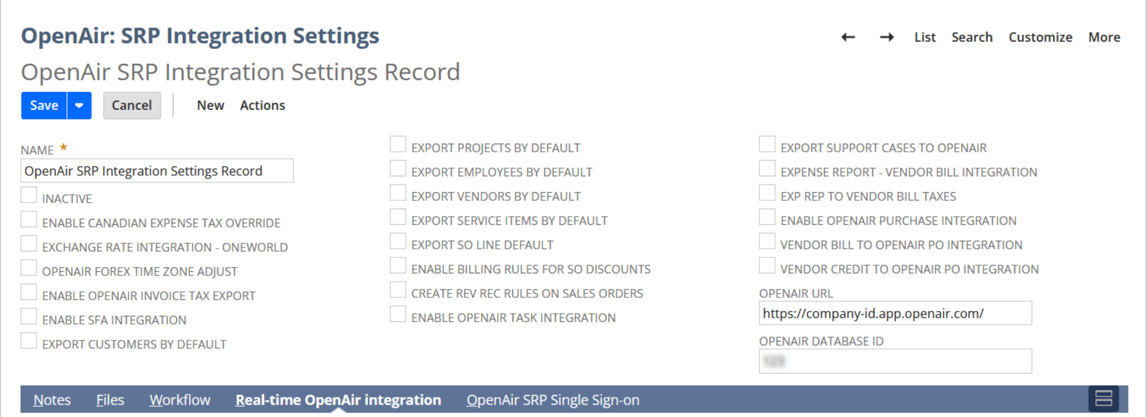
You can use the preferences in this section to control some of the NetSuite <> OpenAir integration functionality.

For example, the NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle adds a custom field (check box) to the record form for NetSuite record types that can be exported to OpenAir. You can configure NetSuite to check or clear the **Export to OpenAir** box by default when users add a new record.

Preference	Setting
Enable Canadian Expense Tax Override	Check this box to export Canadian tax amounts entered manually on OpenAir receipts to NetSuite when exporting expense reports and expense report tax information from OpenAir to NetSuite. See Allow manual override of Canadian taxes on Expense report export.
Exchange Rate Integration — OneWorld	Check this box to enable support for multiple base currencies for cross rates when using NetSuite OneWorld.
OpenAir Forex Time Zone Adjust	—

Preference	Setting
Enable OpenAir Invoice Tax Export	Check this box if you want to import the tax calculated in NetSuite back into OpenAir when exporting invoices from OpenAir to NetSuite. Additional setup is required. See Import taxes applied to exported invoices.
Enable SFA Integration	**Deprecated** — Check this box to enable the 3–way Salesforce, OpenAir and NetSuite integration for customers and projects. Additional setup is required. ⚠ **Important:** The 3-way Salesforce, OpenAir and NetSuite integration for customers and projects is deprecated and will be removed at a future date to be announced. You should use the Projects (Projects) and Customers (Customers) export workflows instead. See Replacing the Deprecated 3-Way Salesforce, OpenAir and NetSuite Integration With the Customers (Customers) or Projects (Projects) Export Workflows.
Export <Record Type> by Default	Check this box to mark new records for export to OpenAir by default. After you enable this preference, the **Export to OpenAir** box is checked by default when a user creates a new record. Users with the relevant permissions can clear the box on individual records if required. Record types include Customers, Projects, Employees, Vendors, and Service items.
Export SO Line Default	Check this box if you want to import sales order from NetSuite into OpenAir as Customer POs. Additional setup is required. See Configuring the Sales Order Integration.
Enable Billing Rules for SO Discounts	Check this box to create fixed fee billing rules automatically on a per project basis for discount and markup line items on the NetSuite sales order. See Configuring the Sales Order Integration.
Create Rev Rec Rules on Sales Orders	Check this box to create revenue recognition rules automatically associated with the project billing rule selected on the NetSuite sales order item line. See Configuring the Sales Order Integration.
Enable OpenAir Task Integration	Check this box if you want to import project tasks from NetSuite into OpenAir. Additional setup is required. See Tasks (Project Tasks) Import Workflow.
Export Support Cases to OpenAir	Check this box if you want to import support cases from NetSuite into OpenAir as project tasks. Additional setup is required. See Tasks (Support Cases) Import Workflow
Expense Report — Vendor Bill Integration	Check this box if you want to export expense reports from OpenAir to NetSuite as vendor bills. Additional setup is required. See Expense Reports (Expense Reports) Export Workflow.
Exp Rep to Vendor Bill Taxes	Check this box to include taxes breakout when exporting of expense report from OpenAir to NetSuite as vendor bills. Additional setup is required. See Enable tax export on expense reports.
Enable OpenAir Purchase Integration	Check this box if you want to import purchase orders from NetSuite into OpenAir. Additional setup is required. See POs (Purchase Orders) Import Workflow.
Vendor Bill to OpenAir PO Integration	Check this box if you want to import vendor bills from NetSuite into OpenAir as purchase orders. Additional setup is required. See POs (Vendor Bills) Import Workflow.
Vendor Credit to OpenAir PO Integration	Check this box if you want to import vendor credits from NetSuite into OpenAir as purchase orders. Additional setup is required. See POs (Vendor Credits) Import Workflow.

Preference	Setting
OpenAir URL	Enter the URL for your OpenAir account. The URL must include the account-specific domain name for your OpenAir account. For more information about your account-specific domain name, see the help topic Your OpenAir Account URLs. **Important:** This setting is available only if NetSuite OpenAir SRP Integration (Bundle ID: 369637) 4.0 or later version is installed in your NetSuite account. The OpenAir Within NetSuite (NetSuite Single Sign-On Integration) and Real-Time Record Import use NetSuite as OIDC Provider and certificate-based authentication instead of SuiteSignOn if an **OpenAir URL** and an **OpenAir Database ID** are specified on the `OpenAir: SRP Integration Settings` record.
OpenAir Database ID	Enter your unique OpenAir account number. See the help topic Finding the Database ID for Your OpenAir Account. **Important:** This setting is available only if NetSuite OpenAir SRP Integration (Bundle ID: 369637) 4.0 or later version is installed in your NetSuite account. The OpenAir Within NetSuite (NetSuite Single Sign-On Integration) and Real-Time Record Import use NetSuite as OIDC Provider and certificate-based authentication instead of SuiteSignOn if an **OpenAir URL** and an **OpenAir Database ID** are specified on the `OpenAir: SRP Integration Settings` record.



OpenAir: SRP Integration Settings

OpenAir SRP Integration Settings Record

NAME *

OpenAir SRP Integration Settings Record

□ INACTIVE
□ ENABLE CANADIAN EXPENSE TAX OVERRIDE
□ EXCHANGE RATE INTEGRATION - ONEWORLD
□ OPENAIR FOREX TIME ZONE ADJUST
□ ENABLE OPENAIR INVOICE TAX EXPORT
□ ENABLE SFA INTEGRATION
□ EXPORT CUSTOMERS BY DEFAULT
□ EXPORT PROJECTS BY DEFAULT
□ EXPORT EMPLOYEES BY DEFAULT
□ EXPORT VENDORS BY DEFAULT
□ EXPORT SERVICE ITEMS BY DEFAULT
□ EXPORT SO LINE DEFAULT
□ ENABLE BILLING RULES FOR SO DISCOUNTS
□ CREATE REV REC RULES ON SALES ORDERS
□ ENABLE OPENAIR TASK INTEGRATION
□ EXPORT SUPPORT CASES TO OPENAIR
□ EXPENSE REPORT - VENDOR BILL INTEGRATION
□ EXP REP TO VENDOR BILL TAXES
□ ENABLE OPENAIR PURCHASE INTEGRATION
□ VENDOR BILL TO OPENAIR PO INTEGRATION
□ VENDOR CREDIT TO OPENAIR PO INTEGRATION

OPENAIR URL

https://company-id.app.openair.com/

OPENAIR DATABASE ID

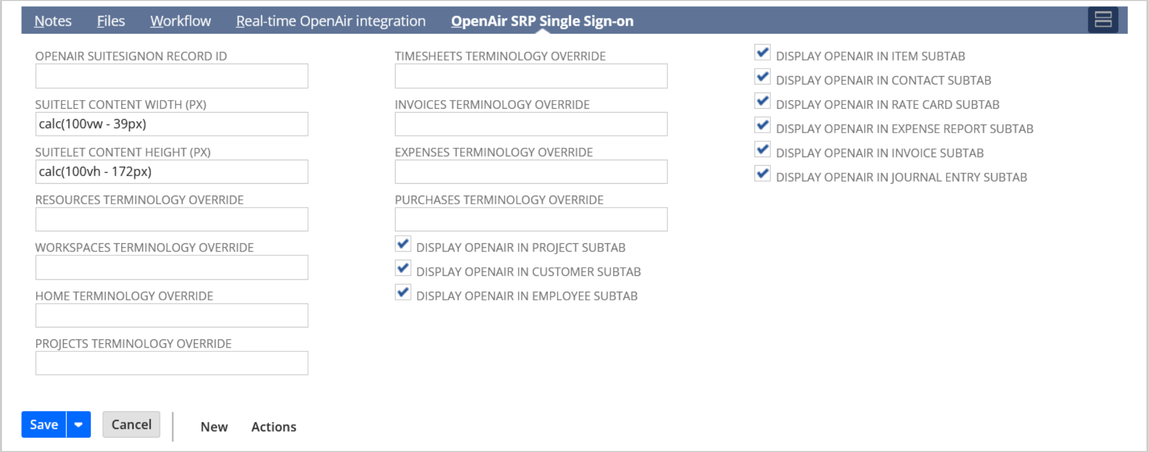
[redacted]

NetSuite OpenAir SRP Single Sign-On

You can use the preferences in this section to control some of the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature settings, if NetSuite OpenAir SRP Single Sign-on (Bundle ID 369619) bundle is installed in your account. For more information about setting up the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature, see Enabling Access to OpenAir From Within NetSuite.

Preference	Setting
OpenAir SuiteSignOn Record ID	Enter the SuiteSignOn record ID for OpenAir. This setting is required to enable the previous version of the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature that relies on the NetSuite Outbound Single Sign-On (SuiteSignOn) feature.

Preference	Setting
Suitelet Content Width (px)	Enter a value to set the width of the Suitelet iFrame for the pages available from the PSA center tab menu. ✓ **Tip:** To utilize the whole of the available page space, set **Suitelet content width (px)** in function of the browser window width. For example, `calc(100vw – 39px)`.
Suitelet Content Height (px)	Enter a value to set the height of the Suitelet iFrame for the pages available from the PSA center tab menu. ✓ **Tip:** To utilize the whole of the available page space, set **Suitelet content height (px)** in function of the browser window height. For example, `calc(100vh – 184px)`.
<OpenAir Module> Terminology Override	Enter a different name for each OpenAir module if you want a different name to be used for this module in NetSuite.
Display OpenAir in <NetSuite Record Type> subtab	Check the box to show the matching OpenAir record as a subtab on NetSuite records of this type. These setting are used to set up the upgraded version of the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature that uses the NetSuite as OIDC Provider feature instead of NetSuite Outbound Single Sign-On (SuiteSignOn) feature. ⚠ **Important:** These settings are available only if NetSuite OpenAir SRP Single Sign-on (Bundle ID: 369619) 7.0 or later version is installed in your NetSuite account.



OPENAIR SUITESIGNON RECORD ID

SUITELET CONTENT WIDTH (PX): calc(100vw - 39px)

SUITELET CONTENT HEIGHT (PX): calc(100vh - 172px)

RESOURCES TERMINOLOGY OVERRIDE

WORKSPACES TERMINOLOGY OVERRIDE

HOME TERMINOLOGY OVERRIDE

PROJECTS TERMINOLOGY OVERRIDE

TIMESHEETS TERMINOLOGY OVERRIDE

INVOICES TERMINOLOGY OVERRIDE

EXPENSES TERMINOLOGY OVERRIDE

PURCHASES TERMINOLOGY OVERRIDE

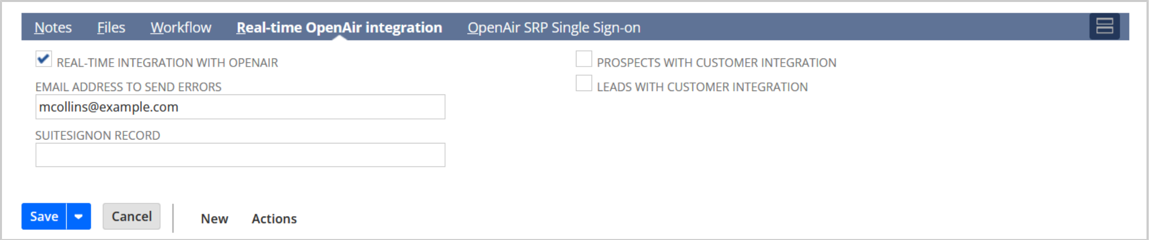
☑ DISPLAY OPENAIR IN PROJECT SUBTAB
☑ DISPLAY OPENAIR IN CUSTOMER SUBTAB
☑ DISPLAY OPENAIR IN EMPLOYEE SUBTAB
☑ DISPLAY OPENAIR IN ITEM SUBTAB
☑ DISPLAY OPENAIR IN CONTACT SUBTAB
☑ DISPLAY OPENAIR IN RATE CARD SUBTAB
☑ DISPLAY OPENAIR IN EXPENSE REPORT SUBTAB
☑ DISPLAY OPENAIR IN INVOICE SUBTAB
☑ DISPLAY OPENAIR IN JOURNAL ENTRY SUBTAB

NetSuite OpenAir SRP Real-Time Add-On

You can use the preferences in this section to control some of the OpenAir real-time integration settings, if NetSuite OpenAir SRP Real-time Add-on (Bundle ID 369630) bundle is installed in your account. For more information about setting up the Real-Time Record Import feature, see Configuring Real-Time Record Import from NetSuite into OpenAir.

Preference	Setting
Real-Time Integration with OpenAir	Check this box to enable the real-time import of records from NetSuite in to OpenAir.

Preference	Setting
Email Address to Send Errors	Enter the email address integration error notifications should be sent to.
SuiteSignOn Record	Enter the SuiteSignOn record ID for OpenAir. This setting is required to enable the previous version of the Real-Time Record Import feature that relies on the NetSuite Outbound Single Sign-On (SuiteSignOn) feature.
Leads with Customer Integration	Check this box to enable additional customer integration features for use with leads. **Note:** Make sure the following settings are enabled on the OpenAir NetSuite Connector administration form, before you check this box in NetSuite: ■ Import all NetSuite contacts associated with NetSuite customers marked 'Export to NS OpenAir' ■ Import all NetSuite contacts associated with NetSuite projects marked 'Export to NS OpenAir' Make sure that the real-time integration script is deployed for Leads (`customdeploy_oa_send_lead`).
Prospects with Customer Integration	Check this box to enable additional customer integration features for use with prospects. **Note:** Make sure the following settings are enabled on the OpenAir NetSuite Connector administration form, before you check this box in NetSuite: ■ Import all NetSuite contacts associated with NetSuite customers marked 'Export to NS OpenAir' ■ Import all NetSuite contacts associated with NetSuite projects marked 'Export to NS OpenAir' Make sure that the real-time integration script is deployed for Prospects (`customdeploy_oa_send_prospect`).



☑ REAL-TIME INTEGRATION WITH OPENAIR

EMAIL ADDRESS TO SEND ERRORS: mcollins@example.com

SUITESIGNON RECORD

☐ PROSPECTS WITH CUSTOMER INTEGRATION

☐ LEADS WITH CUSTOMER INTEGRATION

Terminology

The default or account-specific terminology in OpenAir and NetSuite may be different. To change some of the terminology used in NetSuite, sign in to NetSuite as an account administrator, and go to Setup > Company > Rename Records/Transactions.

Configuring OpenAir for the Integration

OpenAir Professional Services configures your OpenAir account initially so that you can use the integration. Thereafter, you can control your integration settings using the OpenAir NetSuite Connector UI, if you need to make any changes or if you want to take advantages of new integration features. You should contact OpenAir Professional Services if you need to make changes to your integration configuration, and that you test any configuration changes on sandbox accounts before implementing them on your production accounts.

To access the OpenAir NetSuite Connector UI, go to Administration > NetSuite Connector.

Warning: As announced in the October 8, 2022 OpenAir release notes, the New NetSuite Connector UI feature is required to use OpenAir NetSuite Connector. The NetSuite <> OpenAir integration does not work otherwise. For more information about switching to the new OpenAir NetSuite Connector UI, see Switching to the New OpenAir NetSuite Connector UI.

The OpenAir NetSuite Connector configuration and administration includes the following elements:

- **Main OpenAir NetSuite Connector page** — The main OpenAir NetSuite Connector page lets you specify the credentials used to connect to your NetSuite account, specify the data to import from NetSuite to OpenAir and to export from OpenAir to NetSuite, and manage other settings.
  - The main OpenAir NetSuite Connector page includes several tabs, and uses a graphical user interface instead of advanced tag notation to allow more control over customizing the integration. For an overview of the feature, see Navigating the OpenAir NetSuite Connector UI.

    The following help topics describe how to configure OpenAir NetSuite Connector for the integration when the New NetSuite Connector UI feature is enabled for your account:

    - Connecting OpenAir to NetSuite for the Integration
    - Adding and Managing Integration Workflows
    - Mapping OpenAir Fields and NetSuite Fields
    - Scheduling the Integration to Run Automatically
    - Enabling the OpenAir Within NetSuite Feature in OpenAir
    - Enabling the Real-Time Record Import Feature in OpenAir
    - Configuring Real-Time Record Export from OpenAir to NetSuite
- **NetSuite Connector administration form** — The NetSuite Connector administration form lets you select the workflows to include in your integration and additional features, depending on your business needs. For more information, see NetSuite Connector Administration Settings.
- **Workflow settings** — You can control settings specific to each workflow. On the NetSuite Connector Mapping screen, select a workflow and click **Settings** in the top right corner to view the workflow settings form. The Mapping screen only shows the **Settings** link if there are any specific settings for the selected workflow. See Workflow Settings.
- **Settings history page** — Review the history of integration configuration changes. See Reviewing NetSuite Connector Settings History.
- **Status page** — Review the history and status of integration runs. See Reviewing the Integration Run History and Status.
- **Health Check** — Review best practice recommendations for your OpenAir NetSuite Connector configurations. See OpenAir NetSuite Connector Health Check.
- **Access Control** — OpenAir account administrators can control who has access to OpenAir NetSuite Connector and to specific integration features. See Integration Role Permissions in OpenAir.

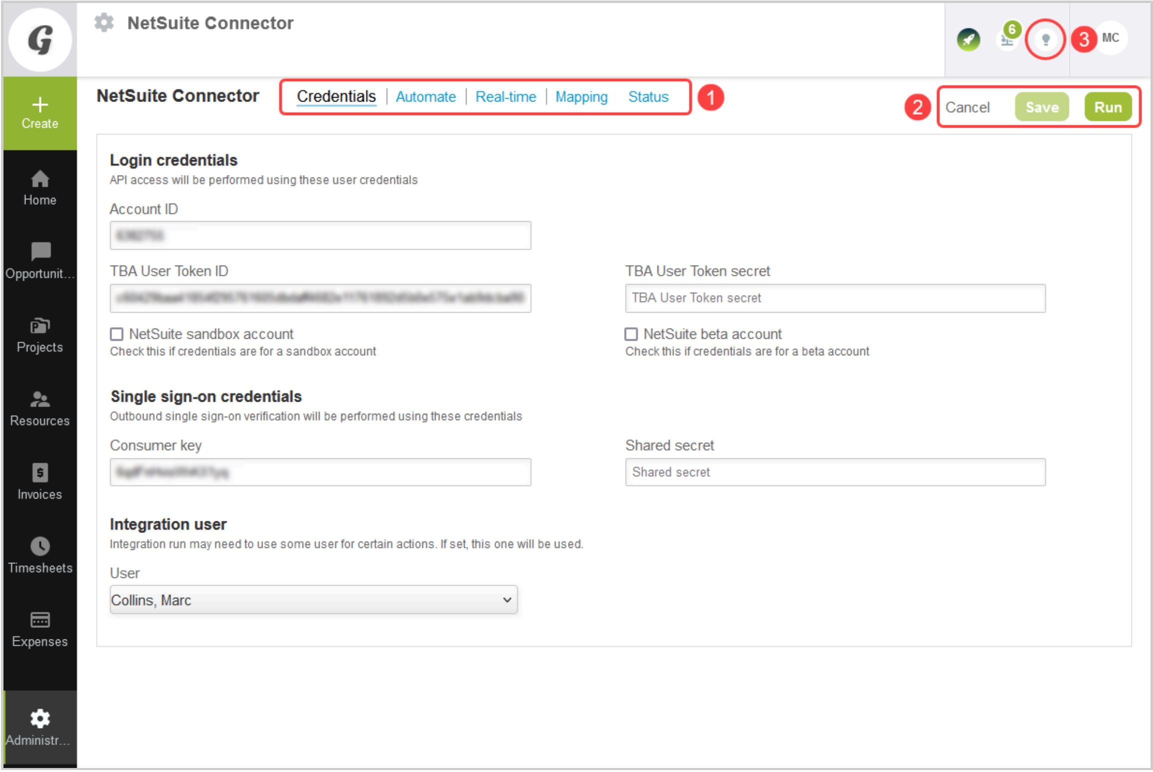
⚠ **Important:** Review the following guidelines:

- Review the list of enhancements for each OpenAir release to take advantage of the latest OpenAir NetSuite Connector features.
- You should test any configuration changes on sandbox accounts before implementing them on your production accounts.
- You should contact OpenAir Professional Services if you need to make changes to your integration configuration.

Navigating the OpenAir NetSuite Connector UI

The following table gives an overview of the different features on the OpenAir NetSuite Connector page.

1	**Tabs** — The OpenAir NetSuite Connector page contains five tabs: - **Credentials** — Use this tab to set up sign-in and single sign-on credentials for the integration. See Connecting OpenAir to NetSuite for the Integration. - **Automate** — Use this tab to schedule specific part of your integration to run automatically at specific times. See Scheduling the Integration to Run Automatically. - **Real-time** — Use this tab to set up the type of records you want to integrate in real time after the record is saved or approved in OpenAir. See Configuring Real-Time Record Export from OpenAir to NetSuite. - **Mapping** — Use this tab to set up your integration import and export workflows and define field mappings for each workflow. See Adding and Managing Integration Workflows and Setting Field Mapping Definitions. - **Status** — The Status screen opens in a separate browser tab or window. Use this screen to view the history of integration runs and detailed information about each run. See Reviewing the Integration Run History and Status.
2	**Actions** — The OpenAir NetSuite Connector page contains the **Save** and **Run** actions. - Click **Save** to apply and save any changes to the credentials, automate, real-time, or mapping configuration tabs . - Click **Run** to run the integration manually for selected workflows. See Running the Bulk Import and Export Integration on Demand.
3	**Contextual Tips** — Additional functionality and information is available from the OpenAir NetSuite Connector Contextual Tips.



NetSuite Connector — Credentials | Automate | Real-time | Mapping | Status

Login credentials

API access will be performed using these user credentials

Account ID: [redacted]

TBA User Token ID: [redacted]

TBA User Token secret: TBA User Token secret

□ NetSuite sandbox account

Check this if credentials are for a sandbox account

□ NetSuite beta account

Check this if credentials are for a beta account

Single sign-on credentials

Outbound single sign-on verification will be performed using these credentials

Consumer key: [redacted]

Shared secret: Shared secret

Integration user

Integration run may need to use some user for certain actions. If set, this one will be used.

User: Collins, Marc

OpenAir NetSuite Connector Contextual Tips

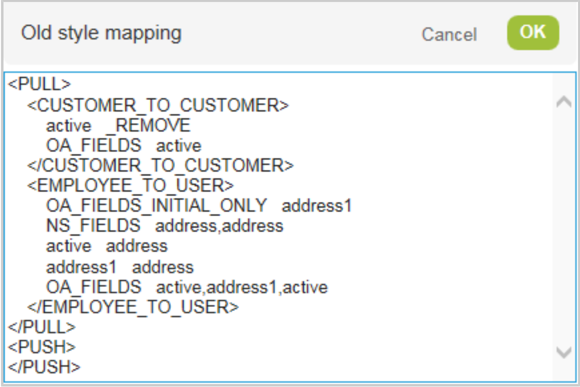
From the OpenAir NetSuite Connector page, you can click the Tips button to access some of the OpenAir NetSuite Connector functionality or to view additional information. The following functionality is available:

- Click **Health Check** to view recommended actions to improve your OpenAir NetSuite Connector configuration and to take full advantage of its features. See OpenAir NetSuite Connector Health Check.
- Click **NetSuite Connector administration** to view the NetSuite Connector administration form and configure the integration to meet your business requirements. See NetSuite Connector Administration Settings.
- Click **Show NetSuite Connector settings history** to view the history of integration configuration changes. See Reviewing NetSuite Connector Settings History.
- Click **OpenAir NetSuite Connector Guide** to view the documentation for the OpenAir NetSuite integration in the OpenAir Help center.
- Click **Synchronize NetSuite custom fields** to update the NetSuite custom fields available for field mapping on the OpenAir NetSuite Connector page. See also Setting Field Mapping Definitions.
- Click **Show progress** to open a new browser tab or window and view the progress of the active integration run, including information about exported and imported records.

Note: Review the following guidelines:

- Wait a short period of time before viewing the progress of your integration run. It takes a few seconds to initialize the integration process. Progress information will become available from the moment the import and export of records start.
- Press Ctrl + F5 to refresh the page and see changes in progress. The page does not refresh automatically.
- If you open the page or refresh the page after the integration is finished, the progress shows the following message: “No integration is running”.

- Click **Show relevant old advanced mapping** to view the advanced tag notation code corresponding to the field mapping configured using the mapping user interface on the OpenAir NetSuite Connector page.



```
<PULL>
  <CUSTOMER_TO_CUSTOMER>
    active   _REMOVE
    OA_FIELDS   active
  </CUSTOMER_TO_CUSTOMER>
  <EMPLOYEE_TO_USER>
    OA_FIELDS_INITIAL_ONLY   address1
    NS_FIELDS   address,address
    active   address
    address1   address
    OA_FIELDS   active,address1,active
  </EMPLOYEE_TO_USER>
</PULL>
<PUSH>
</PUSH>
```

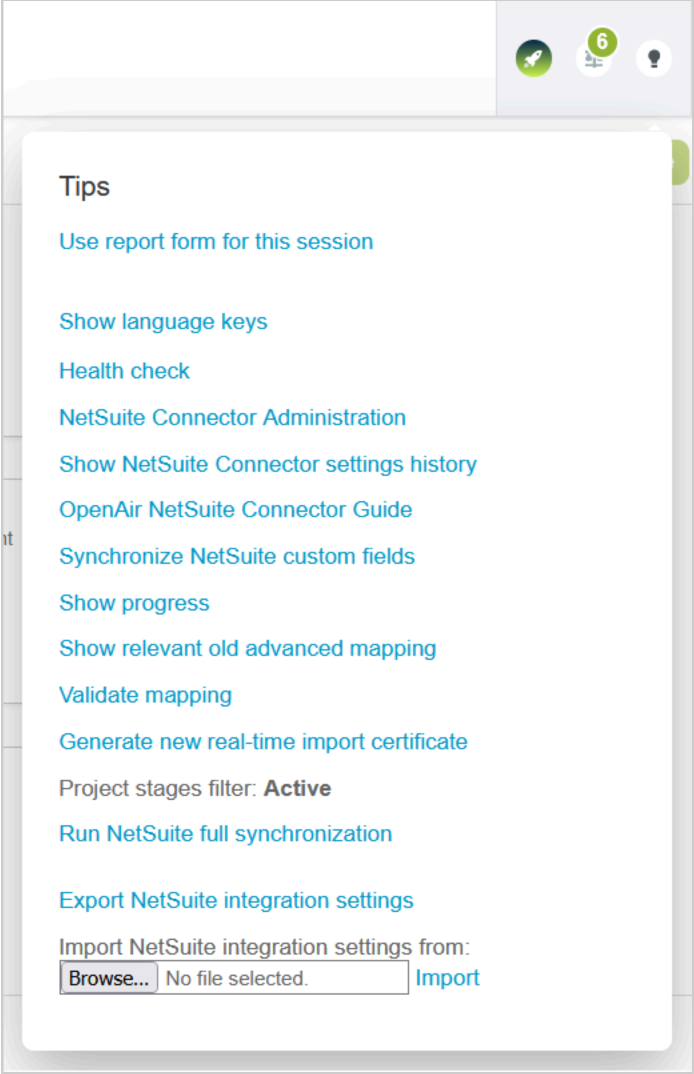
Note: To see the mapping tags, first enable the workflow and save all changes.

- Click **Validate mapping** to verify that all enabled mapping rows for all enabled workflows meet strict validation criteria. See Validating the Mapping Manually.
- Click **Generate a new real-time import certificate** to generate a new certificate for real-time record import and save it to NetSuite and OpenAir. See Real-Time Record Import Certificate and Certificate Rotation.
- **Workflow specific information**.
- **Run NetSuite full synchronization** — Depending on your integration configuration settings, this link is available and lets you run the integration and include either all records last modified on or after the date specified across all selected workflows, or all records across all selected workflows regardless of when the record was last modified. See Expose NetSuite full synchronization control since date and Running a Full Synchronization on Demand.

> **Note:** You cannot use this feature if an integration is already running and will receive the message “ERROR: Another process is running the NetSuite integration. Sync run is not queued.”
>
> To see this link, check the **Expose NetSuite full synchronization control since date** box on the NetSuite Connector administration form.

- **Export NetSuite integration settings** — Exports integration settings as a JSON file. Both active and inactive workflows are exported, including advanced mappings and, where applicable, filters. Requires administrator privileges.
- **Import NetSuite integration settings from:** — Imports settings stored in a JSON file to the integration. Both active and inactive workflows are imported, including advanced mappings and, where applicable, filters. Requires administrator privileges.

> **Note:** The Export and Import functions can be used to transfer integration settings between accounts, archive settings, and migrate accounts to the new NetSuite Connector UI.


6
Tips
Use report form for this session
Show language keys
Health check
NetSuite Connector Administration
Show NetSuite Connector settings history
OpenAir NetSuite Connector Guide
Synchronize NetSuite custom fields
Show progress
Show relevant old advanced mapping
Validate mapping
Generate new real-time import certificate
Project stages filter: Active
Run NetSuite full synchronization
Export NetSuite integration settings
Import NetSuite integration settings from:
Browse... No file selected.
Import

NetSuite Connector Administration Settings

You can use the NetSuite Connector administration form to select the workflows, filtering and features to include in your integration, depending on your business needs.

The NetSuite Connector administration form includes the following sections. For more information about the settings in each section of the form, see the corresponding section of this guide.

- **Integration settings** — This section of the form only displays a message.
- Imports — Control the import workflows (NetSuite > OpenAir) available in the main OpenAir NetSuite Connector UI.
- Exports — Control the export workflows (OpenAir > NetSuite) available in the main OpenAir NetSuite Connector UI.
- Additional integrations — Control the additional integrations available in the main OpenAir NetSuite Connector UI.
- Auto recovery — Control the OpenAir NetSuite Connector autorecovery features.
- Legacy compatibility — Control OpenAir NetSuite Connector legacy compatibility settings.
- Other features — Control other OpenAir NetSuite Connector features.
- Performance features — Control OpenAir NetSuite Connector performance features.
- Custom Imports — Specify custom import workflows available in the main OpenAir NetSuite Connector UI.

> **Note:** A new approach to custom import workflows was introduced in the April 15, 2023 OpenAir release. The new approach is more flexible than custom imports defined on the OpenAir NetSuite Connector administration form and does not rely on NetSuite saved searches. See Custom Import and .

- Exchange rate integration — Control the OpenAir NetSuite Connector exchange rate integration.

All account administrators and employees with the “View and edit integration settings” role permission can view the NetSuite Connector administration form, but only designated employees can change the settings and save the form.

> **Tip:** If a setting label shows in blue color, click the label to view the relevant help topic for this setting in the OpenAir Help Center (requires the View Help Center role permission).

To access the NetSuite Connector administration form:

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button, then **NetSuite Connector administration**

To let a designated employee edit settings on the NetSuite Connector administration form:

1. Contact OpenAir Customer Support and request the NetSuite Connector Administration Form Editor Permission optional feature.
2. In OpenAir, go to Administration > Global settings > Users > [*Select an employee*] > Demographic.
3. Check the **View and edit NetSuite Connector administration form** box under the Optional features section. This employee demographic setting is only available if the employee is an account administrator or has the “View and edit integration settings” role permission.

4. Click **Save**.

> ⚠ **Important:** Make sure you read the documentation and consider changes carefully before you save the NetSuite Connector administration form. The integration may stop working, or may not work as expected and cause data corruption if configured incorrectly.
>
> An optional feature disables the Save button if there are any active scheduled or real-time integrations configured for your account. This forces authorized administrators to deactivate all scheduled and real-time integrations before they can make any changes to the NetSuite Connector administration form. To add this additional level of protection, contact OpenAir Customer Support and ask for the following feature: Prevent Saving NetSuite Connector Administration Form if Workflows are Scheduled or Set for Real-Time Integration.

Imports

This section of the NetSuite Connector administration form lets authorized integration administrators select the import workflows (NetSuite > OpenAir) to include in the integration.

The following import workflows are always enabled and appear in gray on the form:

- Employees (Employees)
- Customers (Customers)
- Projects (Projects)
- Services (Service Items)
- Expense Items (Expense Categories)
- Contacts (Contacts)

Other import workflows in this section of the NetSuite Connector administration form are optional.

- Employees (Vendors)
- Vendors (Vendors)
- Products (Purchase Items)
- Customer POs (Sales Orders)
- Reimbursements (Expense Report Reimbursements)
- Reimbursements (Expense Report Reimbursements)
- POs (Purchase Orders)
- PO (Vendor Bills)
- Retainers (Applied Customer Deposits)
- Tasks (Support Cases)
- POs (Vendor Credits)
- Custom Import

Employees (Vendors)

When enabled, user accounts are established in OpenAir for vendors who transfer from NetSuite as users.

Note: This box cannot be cleared if the Employees (Vendors) import workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

Vendors (Vendors)

When enabled, vendor accounts are established in OpenAir for vendors who transfer from NetSuite as vendors.

Important: Requires the Employees (Vendors) workflow to be enabled. See Employees (Vendors).

Note: This box cannot be cleared if the Vendors (Vendors) import workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

Products (Purchase Items)

Enable push of the following ‘Export to OpenAir Product' NetSuite items to OpenAir: Inventory, Kit/ Package, Non-inventory, Other Charge, and Service.

Note: This box cannot be cleared if the Products (Purchase Items) import workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

Customer POs (Sales Orders)

Enable push of NetSuite sales orders to OpenAir Customer POs.

Note: This box cannot be cleared if the Customer POs (Sales Orders) import workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

Reimbursements (Expense Report Reimbursements)

Enable push of NetSuite expense report/vendor bill payment information to OpenAir as an OpenAir reimbursement.

Note: This box cannot be cleared if the Reimbursements (Expense Report Reimbursements) import workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

Payments (Invoice Payments)

Enable push of NetSuite invoice payment information to OpenAir and create payment transaction on the OpenAir invoice.

Note: This box cannot be cleared if the Payments (Invoice Payments) import workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

POs (Purchase Orders)

Enable NetSuite purchase orders transfer to OpenAir. Must also enable NetSuite product integration and vendor to OpenAir vendor integration. (See below.)

Important: Requires the following workflow to be enabled:

- Products (Purchase Items) — See Products (Purchase Items).
- Vendors (Vendors) — See Vendors (Vendors).

Mutually exclusive with the PO (Vendor Bills) workflow. See PO (Vendor Bills).

Note: This box cannot be cleared if the POs (Purchase Orders) import workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

PO (Vendor Bills)

Enable transfer of NetSuite vendor bills to OpenAir. Must also enable NetSuite product integration and vendor to OpenAir vendor integration. (See below)

Important: Requires the following workflow to be enabled:

- Products (Purchase Items) — See Products (Purchase Items).
- Vendors (Vendors) — See Vendors (Vendors).

Mutually exclusive with the POs (Purchase Orders) workflow. See POs (Purchase Orders).

Note: This box cannot be cleared if the PO (Vendor Bills) import workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

Retainers (Applied Customer Deposits)

Enable push of NetSuite customer deposits to OpenAir and create retainer for invoices.

Note: This box cannot be cleared if the Retainers (Applied Customer Deposits) import workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

Tasks (Support Cases)

When enabled, NetSuite support cases can be imported as OpenAir project tasks.

> **Note:** This box cannot be cleared if the Tasks (Support Cases) import workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

POs (Vendor Credits)

Enable a Vendor Credit integration workflow to retrieve NetSuite vendor credit details for OpenAir as negative purchase items.

For more information, see POs (Vendor Bills) Import Workflow.

> **Note:** This box cannot be cleared if the POs (Vendor Credits) import workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

Custom Import

Check this box to enable custom import workflows. Custom import workflows let you import data from NetSuite service items or custom record types into a supported record type in OpenAir using your own field mapping definitions. You can add custom import workflows to workflow groups and include them in your scheduled integration runs, or when running the integration manually.

Custom import workflow field mapping supports filters and advanced mappings.

By default, a maximum of 4 custom export or import workflows can be active at any one time. To review your requirements, and increase the maximum number of active custom export or import workflows accordingly, contact OpenAir Customer Support.

For more information, see Custom Import Workflows.

> **Note:** The **Custom import** box cannot be cleared if a custom import workflow is listed on, or can be added to the Mapping tab on the main NetSuite Connector page. To disable custom import workflows, first delete all custom import workflows permanently in Administration > NetSuite Connector > Mapping.
>
> You must also check the **Enable Direct TBA** box to enable custom imports. See Enable Direct TBA.

Exports

This section lets you select the export workflows to include in your integration.

The following export workflows are always enabled and appear in gray on the form:

- Expense Reports (Expense Reports)
- Invoices (Invoices)

> **Note:** Invoice approvals must be enabled in OpenAir to be able to export invoices — The Invoices (Invoices) workflow is hidden in the NetSuite Connector UI otherwise. To enable approvals for invoices, go to Administration > Global settings > Organization > Approval Options and check the **Enable approvals for invoices** box.

Other export workflows in this section are optional.

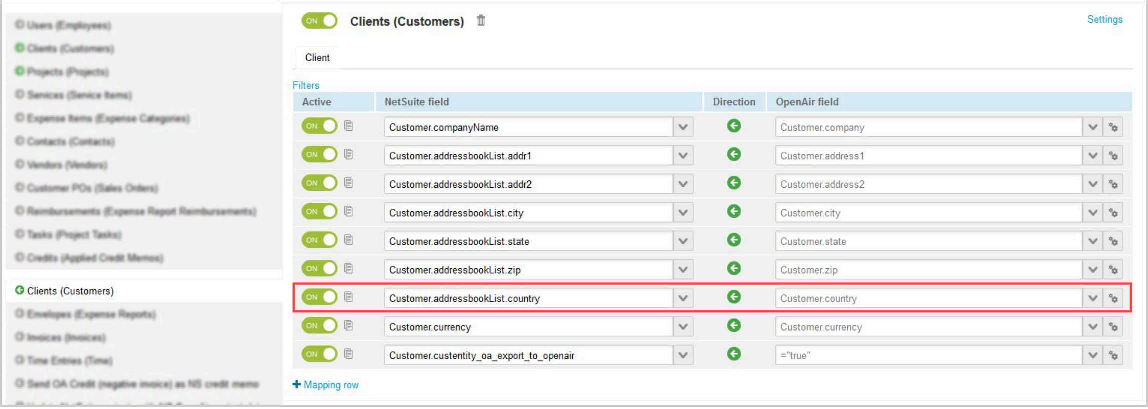
- Customers (Customers)
- Projects (Projects)
- Products (Purchase Items)
- Purchase request (Purchase requisition)
- Time Entries (Time)
- Recognition Transactions (Revenue Recognition Transactions)
- Loaded Cost (Custom Record: <CustomRecordTypeID>)
- Recognition Transactions (Custom Record: OpenAir: Revenue Transactions for ARM) and Recognition Rules (Custom Record: OpenAir: Revenue Rules for ARM)
- Update NetSuite Projects with OpenAir Project Data
- WBS (Opportunity)
- Custom Export

Customers (Customers)

When enabled, a new Customers (Customers) workflow is available in the NetSuite Connector Mapping screen, which allows you to add Customers to NetSuite from OpenAir.

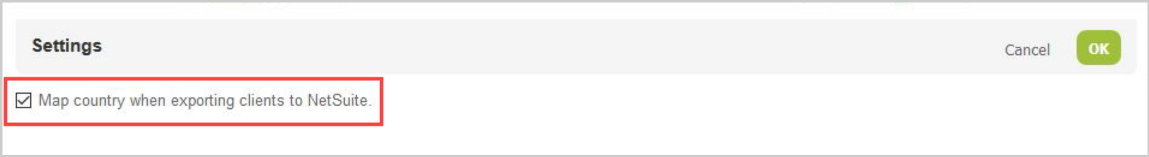
> **Important:** This feature is mutually exclusive with DEPRECATED — Enable 3-way Salesforce.com, NS OpenAir and NetSuite integration.

> **Note:** This box cannot be cleared if the Customers (Customers) export workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.



Active	NetSuite field	Direction	OpenAir field
ON	Customer.companyName	←	Customer.company
ON	Customer.addressbookList.addr1	←	Customer.address1
ON	Customer.addressbookList.addr2	←	Customer.address2
ON	Customer.addressbookList.city	←	Customer.city
ON	Customer.addressbookList.state	←	Customer.state
ON	Customer.addressbookList.zip	←	Customer.zip
ON	Customer.addressbookList.country	←	Customer.country
ON	Customer.currency	←	Customer.currency
ON	Customer.custentity_oa_export_to_openair	←	="true"

The mapping row for the Country field is disabled by default. To enable it, click **Settings** and select **Map country when exporting customers to NetSuite**.



☑ Map country when exporting clients to NetSuite.

Note: It is not possible to map the Country field manually.

Enabling the Customers (Customers) workflow also adds an **Export to NetSuite** option to each Customer form.

When the integration runs, it performs an Add request to NetSuite (in the same manner as DEPRECATED — Enable 3-way Salesforce.com, NS OpenAir and NetSuite integration) for each customer marked **Export to NetSuite**.

The Customers (Customers) workflow supports real-time export. You can set NetSuite Connector to export automatically new/existing project records created/updated in OpenAir to NetSuite when hitting **Save** on the Customer form.

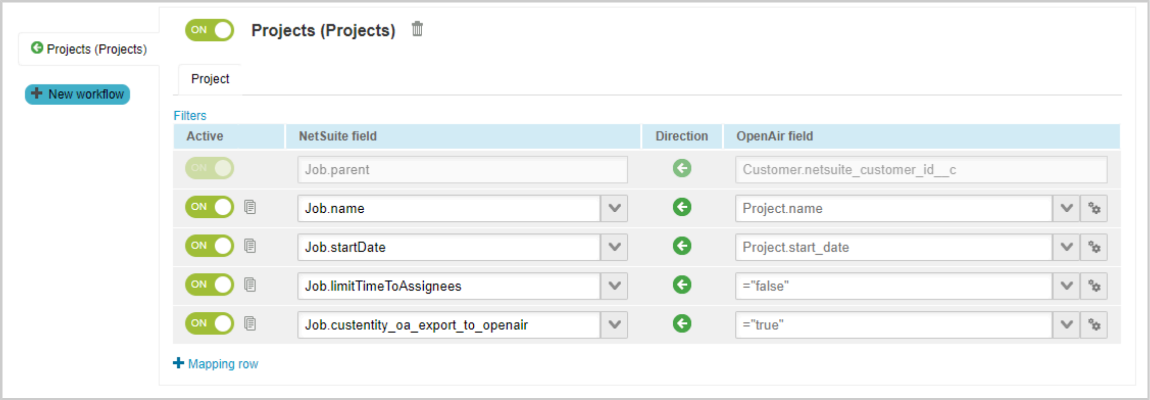
Note: A warning displays when users attempt to export manually an already integrated Customer record.

Projects (Projects)

When enabled, a new Projects (Projects) workflow is made available in the NetSuite Connector Mapping screen, which allows you to add Projects to NetSuite from OpenAir.

Important: This feature is mutually exclusive with DEPRECATED — Enable 3-way Salesforce.com, NS OpenAir and NetSuite integration.

Note: This box cannot be cleared if the Projects (Projects) export workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.



Active	NetSuite field	Direction	OpenAir field
ON	Job.parent	←	Customer.netsuite_customer_id__c
ON	Job.name	←	Project.name
ON	Job.startDate	←	Project.start_date
ON	Job.limitTimeToAssignees	←	="false"
ON	Job.custentity_oa_export_to_openair	←	="true"

Enabling the Projects (Projects) workflow also adds an **Export to NetSuite** option to each Project Properties form.

When the integration runs, it performs an Add request to NetSuite (in the same manner as DEPRECATED — Enable 3-way Salesforce.com, NS OpenAir and NetSuite integration) for each project marked **Export to NetSuite**.

The Projects (Projects) workflow supports real-time export. You can set NetSuite Connector to export automatically new/existing project records created/updated in OpenAir to NetSuite when hitting **Save** on the Project form.

Purchase request (Purchase requisition)

When enabled, purchase requisitions can be created in OpenAir and then sent to NetSuite for fulfillment.

For more information, see Working with Purchase Requisitions in OpenAir.

Note: This box cannot be cleared if the Purchase request (Purchase requisition) export workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

Time Entries (Time)

When enabled, time entries on OpenAir timesheets are exported to NetSuite as Time Tracking.

Note: This box cannot be cleared if the Time Entries (Time) export workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

Recognition Transactions (Revenue Recognition Transactions)

When enabled, revenue recognition from OpenAir is recognized in NetSuite. The Invoices box must be checked on the NetSuite integration form.

Note: This box cannot be cleared if the Recognition Transactions (Revenue Recognition Transactions) export workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

Loaded Cost (Custom Record: <CustomRecordTypeID>)

When enabled, this integration will export/update the OpenAir Loaded Cost to a specified Custom Record in NetSuite.

This feature requires a new Custom Record and specified Custom Fields to be created. Discuss with OpenAir Professional Services before enabling this feature.

Note: This box cannot be cleared if the Loaded Cost (Custom Record: <CustomRecordTypeID>) export workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

Recognition Transactions (Custom Record: OpenAir: Revenue Transactions for ARM) and Recognition Rules (Custom Record: OpenAir: Revenue Rules for ARM)

When enabled, Advanced Revenue Management-related workflows can be added to your NetSuite Connector integration.

For more information, see OpenAir Revenue Recognition with NetSuite Advanced Revenue Management.

Important: This setting enables the following two export workflows:

- Recognition Transactions (Custom Record: OpenAir: Revenue Transactions for ARM)
- Recognition Rules (Custom Record: OpenAir: Revenue Rules for ARM)

Note: This box cannot be cleared if the Recognition Transactions (Custom Record: OpenAir: Revenue Transactions for ARM) export workflow or the Recognition Rules (Custom Record: OpenAir: Revenue Rules for ARM) export workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable these workflows, first delete them in Administration > NetSuite Connector > Mapping.

Update NetSuite Projects with OpenAir Project Data

When enabled, project data is exported from OpenAir to NetSuite.

Note: This box cannot be cleared if the Update NetSuite Projects with OpenAir Project Data export workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

WBS (Opportunity)

When enabled, a link to a NetSuite opportunity exports to an OpenAir project and displays in a custom field on project properties. OpenAir work breakdown structure information can be added and transfers to items on the NetSuite opportunity.

Important: This workflow is mutually exclusive with NS OpenAir project to NetSuite opportunity integration. See Enable OpenAir project to NetSuite opportunity integration.

Note: This box cannot be cleared if the WBS (Opportunity) export workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this workflow, first delete it in Administration > NetSuite Connector > Mapping.

Important: If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:

- Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
- Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Custom Export

Check this box to enable custom export workflows. Custom export workflows let you export data from a supported OpenAir record type to a record type in NetSuite, including custom record types, using your own custom workflows and field mapping definitions. You can add custom export workflows to workflow groups and include them in your scheduled integration runs, or when running the integration manually.

Note: This box cannot be cleared if a custom export workflow is listed on, or can be added to the Mapping tab on the main NetSuite Connector page. To disable custom export workflows, first delete all custom export workflows permanently in Administration > NetSuite Connector > Mapping.

Custom export workflow field mapping supports filters and advanced mappings. However, it currently supports only one pullback row for the NetSuite record ID.

By default, a maximum of 4 custom export or import workflows can be active at any one time. To review your requirements, and increase the maximum number of active custom export or import workflows accordingly, contact OpenAir Customer Support.

If the Business Intelligence Connector feature is enabled for your OpenAir account, you can add custom workflows to export OpenAir published report data to a NetSuite custom record type. You can then use NetSuite features to combine your OpenAir report data and NetSuite data or view them side to side in your custom dashboards and KPIs.

For more information, see Custom Export Workflows.

Note: Custom export workflows require the following NetSuite Connector administration setting to be enabled for your account.

- Enable Direct TBA.

Additional integrations

This section of the NetSuite Connector administration form lets authorized integration administrators enable the following additional integration workflows and features:

- Use OAuth 2.0 for NetSuite Single Sign-On Integration and Real-Time Record Import
- Import all NetSuite contacts associated with NetSuite projects marked 'Export to NS OpenAir'
- Import all NetSuite contacts associated with NetSuite customers marked 'Export to NS OpenAir'
- Enable feature to set import errors on NetSuite records
- Enable NetSuite task import and update
- Enable NetSuite subsidiary import
- Enable NetSuite single sign-on
- Enable NetSuite real-time integration
- Enable NetSuite intercompany journal creation for recognition transaction and invoice export
- Enable NetSuite credit memo integration for invoices
- Enable OpenAir project to NetSuite opportunity integration
- Enable OpenAir Expense report to NetSuite vendor bill integration

Use OAuth 2.0 for NetSuite Single Sign-On Integration and Real-Time Record Import

Checking this box is part of the steps to remove the dependency on the NetSuite Outbound Single Sign-On (SuiteSignOn) feature. For upgrade instructions, see Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn.

Important: Switching to using OAuth 2.0 instead of SuiteSignOn for NetSuite Single Sign-On Integration and Real-Time Record Import is not reversible. After you check this box and click **Save**, you cannot undo the change and go back to using SuiteSignOn.

Changes to the integration configuration in NetSuite are required before you can check this box. For upgrade instructions, see Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn.

Plan to upgrade your integration configuration before the NetSuite 2024.2 release to ensure service continuity. The functionality relying on NetSuite Outbound Single Sign-On (SuiteSignOn) will stop working with the NetSuite 2024.2 upgrade.

You should test all integration configuration changes on sandbox NetSuite and OpenAir accounts before implementing them in your production accounts.

NetSuite plans to end support for the Outbound Single Sign-On (SuiteSignOn) feature with the NetSuite 2024.1 release for non-production accounts and NetSuite 2024.2 for production accounts.

By default, the following OpenAir NetSuite Connector features relied on the SuiteSignOn feature as outbound single sign-on method:

- OpenAir Within NetSuite (OpenAir NetSuite Single Sign-On Integration) — When enabled, this feature gives users access to OpenAir from within the NetSuite UI. See OpenAir Within NetSuite.
- Real-time Record Import — When enabled, OpenAir NetSuite Connector imports records from NetSuite into OpenAir in real time when the record is saved in NetSuite. See Real-Time Record Import.

Upgraded versions of these OpenAir NetSuite Connector features provide the same functionality and replace the dependency on the SuiteSignOn feature in NetSuite with more secure, modern, and reliable authentication methods:

- The upgraded OpenAir Within NetSuite feature uses NetSuite as OIDC Provider as outbound single sign-on method. For more information about NetSuite as OIDC Provider, see NetSuite as OIDC Provider (External link to NetSuite SuiteAnswer | Answer ID: 98241).
- The upgraded Real-Time Record Import feature uses certificate-based authentication.

> **Note:** The older Real-Time Record Import version, which relied on the SuiteSignOn feature, also required the Access Within NetSuite (OpenAir NetSuite Single Sign-On Integration) feature. The upgraded Real-Time Record Import version no longer requires the OpenAir Within NetSuite feature. Each feature can now be enabled separately.

OpenAir changes supporting the upgraded OpenAir Within NetSuite and Real-time Record Import features were delivered as part of the OpenAir 2023.2 release on October 7, 2023.

NetSuite bundles supporting the upgraded OpenAir Within NetSuite and Real-time Record Import features were released on February 7, 2024. The upgrade requires the following bundles and release versions:

- NetSuite OpenAir SRP Integration (Bundle ID: 369637) 4.0
- NetSuite OpenAir SRP Single Sign-On (Bundle ID: 369619) 7.0
- NetSuite OpenAir SRP Real-time Add-on (Bundle ID 369630) 3.0

Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn

Use the following steps to upgrade both, or either one of, the OpenAir Within NetSuite and Real-time Record Import features. The upgrade removes the dependency on the NetSuite Outbound Single Sign-On (SuiteSignOn) feature and uses more secure, modern, and reliable authentication methods.

> **Important:** Plan to upgrade your integration configuration before the NetSuite 2024.2 release to ensure service continuity. The functionality relying on NetSuite Outbound Single Sign-On (SuiteSignOn) will stop working with the NetSuite 2024.2 upgrade.
>
> You should test all integration configuration changes on sandbox NetSuite and OpenAir accounts before implementing them in your production accounts.

To upgrade OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn

1. Update the NetSuite OpenAir SRP Integration (Bundle ID: 369637) to version 4.0 or later.

This bundle version:

- Adds fields to the `OpenAir: SRP Integration Settings` custom record type. These fields are required to set up the upgraded versions of the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) and Real-Time Record Import features.
- Adds the following permission to the `NSOA Connector` role:
  - □ OIDC Provider Setup (Full)
  - □ OAuth 2.0 Authorized Applications Management (Full)
  - □ Log in using OAuth 2.0 Access Tokens (Full)
  - □ Certificate management (Full)

> ⚠ **Important:** Update NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) to Version 4.0 or later and make sure the update completes before you update NetSuite OpenAir SRP Single Sign-On bundle (Bundle ID 369619) to version 7.0 or NetSuite OpenAir SRP Real-time Add-on bundle (Bundle ID 369630) to version 3.0.

2. Update the preferences for the NetSuite <> OpenAir integration.
   a. Go to Customization > Lists, Records and Fields > Record Types > [**List** `OpenAir: SRP Integration Settings`] > [**Edit** *OpenAir SRP Integration Settings Record*].
   b. Enter the following information:
      - **OpenAir URL** — The URL for your OpenAir account. The URL must include the account-specific domain name for your OpenAir account. For more information about your account-specific domain name, see the help topic Your OpenAir Account URLs.
      - **OpenAir Database ID** — Your unique OpenAir account number. See the help topic Finding the Database ID for Your OpenAir Account.
   c. Click **Save**.
3. If you are using the **OpenAir Within NetSuite** feature, update the integration configuration for this feature. See Upgrading OpenAir Within NetSuite Feature Configuration in NetSuite.
4. If you are using the **Real-Time Record Import** feature, update the integration configuration for this feature. See Updating the Real-Time Record Import Feature Configuration in NetSuite.
5. Enable the upgraded OpenAir Within NetSuite and Real-Time Record Import features using OAuth 2.0 in OpenAir.
   a. In OpenAir, go to Administration > NetSuite Connector > Credentials.
   b. Click the Tips utility button, then **NetSuite Connector administration**.
   c. Check the **Use OAuth 2.0 for NetSuite Single Sign-On Integration and Real-Time Record Import (Configuration Changes Required Before Enabling – See Documentation)** box under the Additional Integrations section.
   d. Click **Save**.

      > ⚠ **Important:** Switching to using OAuth 2.0 instead of SuiteSignOn for NetSuite Single Sign-On Integration and Real-Time Record Import is not reversible. After you check this box and click **Save**, you cannot undo the change and go back to using SuiteSignOn.

   e. Click **Close** to return to the OpenAir NetSuite Connector Credentials tab.
6. If you are using the **OpenAir Within NetSuite** feature, enter the NetSuite OAuth 2.0 Credentials in OpenAir.
   a. On the OpenAir NetSuite Connector Credentials tab, under **NetSuite OAuth 2.0 credentials**, enter the **Consumer key** and **Consumer secret** which you copied from

NetSuite when creating an integration record to use the NetSuite as OIDC Provider feature for OpenAir Within NetSuite (Upgrading OpenAir Within NetSuite Feature Configuration in NetSuite – Step 4).

b. Click **Save**.

7. If you are using the **Real-Time Record Import** feature, verify that the certificate required for real-time record imports is valid. See Real-Time Record Import Certificate and Certificate Rotation.

> **Note:** OpenAir uses certificates to validate real-time record import requests. If you followed the above steps in the correct order, OpenAir NetSuite Connector generates a certificate and saves it to NetSuite and OpenAir automatically. Subsequently, OpenAir NetSuite Connector manages the certificate rotation automatically. See Real-Time Record Import Certificate and Certificate Rotation.

Upgrading OpenAir Within NetSuite Feature Configuration in NetSuite

If you are using the OpenAir Within NetSuite feature, follow these steps to replace SuiteSignOn with NetSuite as OIDC Provider as the outbound single sign-on method enabling access to OpenAir from within the NetSuite UI.

> **Important:** The following steps assume that NetSuite OpenAir SRP Integration (Bundle ID: 369637) 4.0 or later version is installed in your NetSuite account and you have updates the preferences for the NetSuite <> OpenAir integration. See Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn.

To upgrade OpenAir Within NetSuite feature configuration in NetSuite:

1. In NetSuite, enable the OAuth 2.0 and NetSuite as OIDC feature. See Enable the NetSuite as OIDC Provider Feature (External link to NetSuite SuiteAnswers | Answer ID: 98270).
2. If you are using your own custom role for the OpenAir integration instead of the “NSOA Connector” role created by the NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle, make sure that this custom role includes the following permissions and permission levels in the **Setup** secondary subtab. For more information, see Creating a Custom Role for the Integration in NetSuite.
   - OIDC Provider Setup (Full)
   - OAuth 2.0 Authorized Applications Management (Full)
   - Log in using OAuth 2.0 Access Tokens (Full)
3. Update the NetSuite OpenAir SRP Single Sign-On (Bundle ID: 369619) bundle to version 7.0 or later.

   This bundle version:
   - Updates SuiteScript 2 versions of Suitelet scripts to enable access to OpenAir modules in the PSA center tab using NetSuite as OIDC Provider.
   - Updates the `OpenAir: Home module portlet v2.0` Portlet script to enable access to the OpenAir dashboard as a custom portlet on the NetSuite dashboard using NetSuite as OIDC Provider.
   - Adds the script `OpenAir: Load subtab` to enable access to the OpenAir record as a subtab on the matching record in NetSuite for selected NetSuite record types using NetSuite as OIDC Provider.
   - Switches to using SuiteScript 2 versions of all scripts in the bundle automatically.

> ⚠ **Important:** Update NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) to Version 4.0 or later before you update NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619) to this version. NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) 4.0 updates the `OpenAir: SRP Integration Settings` custom record type with settings required for using the NetSuite as OIDC Provider for the OpenAir Within NetSuite feature.
>
> SuiteScript 1.0 versions of scripts in the NetSuite OpenAir SRP Single Sign-On (Bundle ID: 369619) bundle support only the SuiteSignOn method. SuiteScript 2 versions of scripts support both the SuiteSignOn and the NetSuite as OIDC Provider methods. OpenAir will end support for SuiteScript 1.0 versions of scripts in the NetSuite OpenAir SRP Single Sign-On (Bundle ID: 369619) bundle some time after the NetSuite 2024.2 release, when support for the SuiteSignOn feature ends (date to be confirmed).

4. Create a custom integration record for the OpenAir Within NetSuite feature and set it up for OpenAir NetSuite Connector to use NetSuite as OIDC Provider for outbound single sign-on. See Create Integration Records for Applications that Use NetSuite as OIDC Provider for Outbound Single Sign-on (External link to NetSuite SuiteAnswers | Answer ID: 98323).
    a. Use the following settings under the OAuth 2.0 section of the integration record:
        - **Authorization code grant** — Check the box. This is required for the NetSuite as OIDC Provider integration to work.
        - **Redirect URI** — Enter one of the following URIs, depending on the type of OpenAir account type you are setting up for the integration:
            - Production account — `https://auth.openair.com/oidc/callback/internal_netsuite_<netsuite_account_id>`
            - Sandbox account — `https://auth.sandbox.openair.com/oidc/callback/internal_netsuite_<netsuite_account_id>`

          where <`netsuite_account_id`> is your NetSuite Account ID

          > ⚠ **Important:** All alphabetical characters in <`netsuite_account_id`> must be lowercase. If your NetSuite Account ID includes uppercase alphabetical characters, replace them with the lowercase equivalent. When setting up the integration between OpenAir and NetSuite sandbox accounts, the NetSuite sandbox account ID typically includes the characters `SB` and should be replaced with the lowercase equivalent `sb`. For example, if your NetSuite Account ID is `345678_SB2`, the Redirect URI is `https://auth.sandbox.openair.com/oidc/callback/internal_netsuite_345678_sb2`.
        - **OAuth2.0 consent policy** — You should select `Never Ask` so that the integration is autoapproved by an administrator. With other settings, users need to authorize the integration before being able to view OpenAir within NetSuite UI.

    b. Copy the **Consumer key / Client ID** and **Consumer secret / Client secret** and store the information in a safe place. You will need to enter this information on the OpenAir NetSuite Connector credentials form in OpenAir for the NetSuite as OIDC Provider integration to work (Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn – Step 6).
5. Give users access to the NetSuite as OIDC Provider integration for the OpenAir Within NetSuite feature.
    a. Go to Setup > Integration > NetSuite as OIDC Provider Setup.
    b. Click the name of the custom integration record you created for the OpenAir Within NetSuite feature (Step 4).

The Application Settings: <*Integration Name*> window appears.

c. On the Entities tab and Employees subtab, check the **Select All** box to give all users access to the NetSuite as OIDC Provider integration for the OpenAir Within NetSuite feature.

d. On the Roles tab, check the **Select All** box to give all roles access to the NetSuite as OIDC Provider integration for the OpenAir Within NetSuite feature.

e. Click **Save**.

6. Add the **Custom Record Entries** permission with **View** permission level or higher to the relevant roles in NetSuite. Without this permission, users cannot access OpenAir from the NetSuite UI.

   For each role, do the following:

   a. Go to Setup > Users/Roles > User Management > Manage Roles > [*Select a role*].

   b. Click **Edit**.

   c. In the Permissions subtab, click the **Lists** secondary subtab.

   d. Add a row and select `Custom Record Entries` under **Permission**, and `View` under **Level**.

   e. Click **Add**.

   f. Click **Save**.

> **Note:** If you are using standard roles, you will need to customize standard roles, and assign these new custom roles to users to let these users access OpenAir from the NetSuite UI. For more information about managing roles in NetSuite, see Customizing or Creating NetSuite Roles (External link to NetSuite SuiteAnswers | Answer ID: 9896).

7. Update the preferences for the NetSuite <> OpenAir integration.

   a. Go to Customization > Lists, Records and Fields > Record Types > [**List** `OpenAir: SRP Integration Settings`] > [**Edit** *OpenAir SRP Integration Settings Record*].

   b. In the NetSuite OpenAir SRP Single Sign-On subtab, check any of the boxes listed in the following table to show the OpenAir record as a subtab on the matching record in NetSuite. The table lists the integrations settings for each of the supported NetSuite record types and the corresponding types of OpenAir record shown in the OpenAir subtab.

Setting	NetSuite record type	OpenAir record type
Display OpenAir in Contact subtab	Contact	Contact
Display OpenAir in Customer subtab	Customer	Customer
Display OpenAir in Employee subtab	Employee	Employee
Display OpenAir in Expense report subtab	Expense Report	Expense Report
Display OpenAir in Invoice subtab	Invoice	Invoice
Display OpenAir in Journal entry subtab	Journal Entry	Revenue Recognition Transaction
Display OpenAir in Rate card subtab	`OpenAir: Project Rate Card` (custom record type)	Rate Card
Display OpenAir in Project subtab	Project	Project
Display OpenAir in Item subtab	Service Item	Category

> **Note:** Some of the configuration-based customization available when using the SuiteSignOn feature is not currently available when using the NetSuite as OIDC Provider feature. You would need to deploy custom SuiteScripts to achieve the following functionality.
>
> - Ability to show the Timesheets module on Support Case records as an OpenAir: Timesheets subtab in NetSuite when using the Tasks (Support Cases) Import Workflow.
> - Ability to set up custom OpenAir subtabs on NetSuite records and displaying OpenAir content other than those listed in the table above. With the SuiteSignOn feature, you could set up custom subtabs and use connection points to associate each custom subtab with a NetSuite record type and use integration variables to specify the OpenAir content shown in this subtab.

c. Click **Save**.

Updating the Real-Time Record Import Feature Configuration in NetSuite

If you are using the Real-Time Record Import feature, follow these steps to use certificate-based authentication instead of the SuiteSignOn feature for real-time imports.

> **Important:** The following steps assume that NetSuite OpenAir SRP Integration (Bundle ID: 369637) 4.0 or later version is installed in your NetSuite account and you have updates the preferences for the NetSuite <> OpenAir integration. See Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn.

To update the Real-Time Record Import feature configuration in NetSuite:

1. If you are using your own custom role for the OpenAir integration instead of the “NSOA Connector” role created by the NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle, make sure that this custom role includes the **Certificate management** permission (**Full** permission level) in the **Setup** secondary subtab. For more information, see Creating a Custom Role for the Integration in NetSuite.
2. Update the NetSuite OpenAir SRP Real-time Add-on (Bundle ID 369630) bundle to version 3.0 or later.

   This bundle version:

   - Updates the `OpenAir: Send Record to OpenAir v2.0` user event script to import records from NetSuite into OpenAir in real-time using certificate-based authentication.
   - Adds the `custcertificate_oa_realtime` certificate record. The default certificate `openair-certificate.pem` associated with the `custcertificate_oa_realtime` certificate record is not valid. It is a required placeholder that will be replaced later. See Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn – Step 7.

> ⚠ **Important:** Update NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) to Version 4.0 or later before you update NetSuite OpenAir SRP Real-time Add-on (Bundle ID 369630) to this version. NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) 4.0 updates the `OpenAir: SRP Integration Settings` custom record type with settings required for using certificate-based authentication for the Real-Time Record Import feature.
>
> NetSuite OpenAir SRP Real-time Add-on (Bundle ID 369630) Version 3.0 or later is required and `OpenAir: Send Record to OpenAir v2.0` must be deployed to enable real-time record import from NetSuite into OpenAir using certificate-based authentication.
>
> `OpenAir: Send Record to OpenAir` (SuiteScript 1.0 version) supports only the SuiteSignOn method. `OpenAir: Send Record to OpenAir v2.0` (SuiteScript 2.0 version) supports both the SuiteSignOn and certificate-based authentication methods. OpenAir will end support for this SuiteScript 1.0 version some time after the NetSuite 2024.2 release, when support for the SuiteSignOn feature ends (date to be confirmed).

3. If you currently have the `OpenAir: Send Record to OpenAir` user event script (SuiteScript 1.0 version) deployed for real-time record import, switch to `OpenAir: Send Record to OpenAir v2.0` (SuiteScript 2 version). See Switching to SuiteScript 2 Versions of Integration Bundle Scripts.
4. Update the preferences for the NetSuite <> OpenAir integration.
    a. Go to Customization > Lists, Records and Fields > Record Types > [**List** `OpenAir: SRP Integration Settings`] > [**Edit** *OpenAir SRP Integration Settings Record*].
    b. Check the **Real-time integration with OpenAir** box in the NetSuite OpenAir SRP Real-Time Add-On subtab.
    c. Click **Save**.

Real-Time Record Import Certificate and Certificate Rotation

OpenAir uses certificates to validate real-time record import requests. Certificates are valid for a finite period. After the initial set up, OpenAir NetSuite Connector generates a new certificate and saves it to NetSuite and OpenAir automatically one to two weeks before the previous certificate expires.

You can use the OpenAir NetSuite Connector Health Check feature to verify whether there is a valid real-time record import certificate in your account, and when the current certificate expires if valid. See OpenAir NetSuite Connector Health Check.

You can trigger the certificate generation and rotation from the contextual tips menu at any time.

To generate a new certificate for real-time record imports:

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button then click **Generate new real-time import certificate**. OpenAir NetSuite Connector generates a new certificate, uploads it in the `custcertificate_oa_realtime` certificate record in NetSuite, and saves it in OpenAir after it is successfully uploaded in NetSuite.

Note: Whereas other OpenAir changes supporting the upgraded OpenAir Within NetSuite and Real-time Record Import features were delivered as part of the OpenAir 2023.2 release on October 7, 2023, the automated certificate rotation and health check was released on February 7, 2024.

Import all NetSuite contacts associated with NetSuite projects marked 'Export to NS OpenAir'

When checked, OpenAir NetSuite Connector imports all NetSuite contact records associated with the project records you import from NetSuite into OpenAir.

Note: When importing a contact associated with a NetSuite project into OpenAir, the OpenAir contact is associated with the customer record associated with the OpenAir project.

Import all NetSuite contacts associated with NetSuite customers marked 'Export to NS OpenAir'

When checked, OpenAir NetSuite Connector imports all NetSuite contact records associated with the customer records you import from NetSuite into OpenAir.

Enable feature to set import errors on NetSuite records

When enabled, the corresponding NetSuite record is updated with any error message that occurs when importing the record into OpenAir. The error message is similar to the error message seen in the integration log with the addition of a date and time stamp based on Eastern Time.

Note: If a prior run of the integration had set an error message in NetSuite and the next run of the integration successfully imports the record into OpenAir, the corresponding NetSuite record will be updated to remove the error message.

⚠ **Important:** If this features is enabled and there is a high volume of records with errors, the integration will run noticeably slower.

When enabled, NetSuite support cases can be imported as OpenAir project tasks.

For more information, see Tasks (Support Cases) Import Workflow.

Enable NetSuite task import and update

When enabled, NetSuite project tasks are imported into OpenAir, resulting in OpenAir project tasks. This also enables the **Task update** section on the NetSuite integration form to configure the updating of NetSuite project task records with OpenAir project task data.

Note: This box cannot be cleared if the Tasks (Project Tasks) import workflow or the Update NetSuite tasks with OpenAir task data export workflow is listed on the Mapping tab on the main NetSuite Connector page. To disable this additional integration option, first delete the Tasks (Project Tasks) import workflow and the Update NetSuite tasks with OpenAir task data export workflow in Administration > NetSuite Connector > Mapping.

Enable NetSuite subsidiary import

Enable import of NetSuite subsidiary information and designate the category to which NetSuite subsidiaries should be mapped. Use together with Enable NetSuite OneWorld integration optimizations.

Enable NetSuite single sign-on

Check this box to enable the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature in OpenAir. You can use the feature to give users of both NetSuite and OpenAir access to OpenAir directly from within the NetSuite user interface.

For an overview of the feature, see OpenAir Within NetSuite.

Additional set up is required to enable the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature:

- In NetSuite – See Enabling Access to OpenAir From Within NetSuite.
- In OpenAir – See Enabling the OpenAir Within NetSuite Feature in OpenAir.

> **Note:** This box cannot be cleared if NetSuite single sign-on or OAuth 2.0 credentials are set on the OpenAir NetSuite Connector Credentials page. To disable this additional integration option, first clear the NetSuite single sign-on or OAuth 2.0 credentials in Administration > NetSuite Connector > Credentials.

Enable NetSuite real-time integration

Check this box to enable the Real-Time Record Import or the Real-Time Record Export features in OpenAir. You can use these features to:

- Import a NetSuite record automatically into OpenAir when the record is saved by any user in NetSuite. For an overview of the Real-Time Record Import feature, including a list of supported record types, see Real-Time Record Import.
- Export an OpenAir record automatically to NetSuite when the record is approved or saved by any user in OpenAir. For an overview of the Real-Time Record Export feature, including a list of supported record types, see Real-Time Record Export.

Additional set up is required to enable:

- Real-Time Record Import in NetSuite – See Configuring Real-Time Record Import from NetSuite into OpenAir.
- Real-Time Record Import in OpenAir – See Enabling the Real-Time Record Import Feature in OpenAir.
- Real-Time Record Export in NetSuite – *No setup required*.
- Real-Time Record Export in OpenAir – See Configuring Real-Time Record Export from OpenAir to NetSuite.

> **Note:** This box cannot be cleared if any real-time integrations are set in the Real-time tab of the main OpenAir NetSuite Connector page. To disable this additional integration option, first delete all active or inactive real-time integrations in Administration > NetSuite Connector > Real-time.

Enable NetSuite intercompany journal creation for recognition transaction and invoice export

Enable transfer of OpenAir revenue recognition transactions and invoices to NetSuite. Run a script to automatically create intercompany journal entries in NetSuite. Cannot be used with sales order integration.

Enable NetSuite credit memo integration for invoices

Enable NetSuite credit memos applied to OpenAir invoices push to OpenAir. OpenAir creates a negative invoice for the credit memo and the total credited amount displays on the original OpenAir invoice.

Enable OpenAir project to NetSuite opportunity integration

An opportunity is associated with a project in NetSuite. That association is made either manually or by a custom workflow and scripting per the customer's business process. The opportunity should be populated in the list custom field created by the integration bundle; in the custentity_oa_opportunity_id custom field.

When this integration point is enabled in OpenAir, the integration will pull the project into OpenAir.

If the NetSuite project is associated with a NetSuite opportunity, the OpenAir project custom field netsuite_project_opportunity_id will have the internal ID of the NetSuite opportunity.

Any user with the proper permissions can create/update the OpenAir project's work breakdown structure.

When ready, a user with the user-level permission “The user can update a NetSuite opportunity from a project work breakdown structure” will see a new link in the tips of the project task list or outline view. The link is not available on the Gantt chart view. The link is called “Update the NetSuite opportunity line items”. When the link in the tips is clicked, the NetSuite opportunity will be updated at the line-level.

Note: The link can be moved and renamed with “Enable NetSuite opportunity export link in project navigation bar.

The update will roll up the default category/activity/service used on timesheets and sum the user planned hours from the task. The default category/activity/service maps to a NetSuite service item and the quantity on the NetSuite opportunity is the sum of the user planned hours for that service across the entire OpenAir project's work breakdown structure.

The opportunity may be updated as many times as needed.

Important: This workflow is mutually exclusive with WBS (Opportunity). See WBS (Opportunity).

Important: If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:

- Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
- Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Enable OpenAir Expense report to NetSuite vendor bill integration

When enabled, vendor expense reports in OpenAir transfer to NetSuite as vendor bills.

Note: This box cannot be cleared if expense reports are set to be exported as vendor bills for any employee in OpenAir. To disable this additional integration option, first clear the **Export expense reports from this employee as NetSuite vendor bills** box in Administration > Users > Employees > [*Select an employee*] > Demographic for all employees concerned.

Auto recovery

This section of the NetSuite Connector administration form lets authorized integration administrators control autorecovery features used in the integration. The following settings are available:

- Verify customer export
- Verify invoice export
- Verify credit (negative invoice) export
- Verify expense report export
- Verify project export
- Verify recognition transaction export
- Verify timesheet export

Verify customer export

Check this box to enable the autorecovery feature for customer records exported to NetSuite as part of an interrupted export process.

OpenAir marks a record as exported and shows the ID of the corresponding record in NetSuite on the entity property form if the record exported successfully to NetSuite. If the export process is interrupted before OpenAir receives a response from NetSuite, records may be exported to NetSuite but not marked as exported in OpenAir. In this case, the record will be exported again the next time the integration workflow runs creating duplicate records in NetSuite.

The autorecovery feature prevents the creation of duplicated records by identifying records included for export in the interrupted integration run, and verifying if the record was already exported to NetSuite. If the record exists in NetSuite, OpenAir NetSuite Connector updates the exported record in OpenAir. Otherwise, the record will be exported again.

Important: To use the autorecovery feature with the Customers (Customers) export workflow, you must update NetSuite OpenAir SRP Integration (Bundle ID: 369637) to version 3.0 or later.

Verify invoice export

Check this box to enable the autorecovery feature for invoice records exported to NetSuite as part of an interrupted export process.

OpenAir marks a record as exported and shows the ID of the corresponding record in NetSuite on the entity property form if the record exported successfully to NetSuite. If the export process is interrupted before OpenAir receives a response from NetSuite, records may be exported to NetSuite but not marked as exported in OpenAir. In this case, the record will be exported again the next time the integration workflow runs creating duplicate records in NetSuite.

The autorecovery feature prevents the creation of duplicated records by identifying records included for export in the interrupted integration run, and verifying if the record was already exported to NetSuite. If the record exists in NetSuite, OpenAir NetSuite Connector updates the exported record in OpenAir. Otherwise, the record will be exported again.

⚠ **Important:** To use the autorecovery feature with the Invoices (Invoices) export workflow, you must update NetSuite OpenAir SRP Integration bundle (Bundle ID: 369637) to version 2.09 or later.

Verify credit (negative invoice) export

Check this box to enable the autorecovery feature for credit (negative invoice) records exported to NetSuite as part of an interrupted export process.

OpenAir marks a record as exported and shows the ID of the corresponding record in NetSuite on the entity property form if the record exported successfully to NetSuite. If the export process is interrupted before OpenAir receives a response from NetSuite, records may be exported to NetSuite but not marked as exported in OpenAir. In this case, the record will be exported again the next time the integration workflow runs creating duplicate records in NetSuite.

The autorecovery feature prevents the creation of duplicated records by identifying records included for export in the interrupted integration run, and verifying if the record was already exported to NetSuite. If the record exists in NetSuite, OpenAir NetSuite Connector updates the exported record in OpenAir. Otherwise, the record will be exported again.

⚠ **Important:** To use the autorecovery feature with the Send OpenAir Credit (negative invoice) as NetSuite Credit Memo export workflow, you must update NetSuite OpenAir SRP Integration bundle (Bundle ID: 369637) to version 3.05 or later.

Verify expense report export

Check this box to enable the autorecovery feature for expense report records exported to NetSuite as part of an interrupted export process.

OpenAir marks a record as exported and shows the ID of the corresponding record in NetSuite on the entity property form if the record exported successfully to NetSuite. If the export process is interrupted before OpenAir receives a response from NetSuite, records may be exported to NetSuite but not marked as exported in OpenAir. In this case, the record will be exported again the next time the integration workflow runs creating duplicate records in NetSuite.

The autorecovery feature prevents the creation of duplicated records by identifying records included for export in the interrupted integration run, and verifying if the record was already exported to NetSuite. If the record exists in NetSuite, OpenAir NetSuite Connector updates the exported record in OpenAir. Otherwise, the record will be exported again.

⚠ **Important:** To use the autorecovery feature for expense reports exported from OpenAir to NetSuite as expense reports, you must update NetSuite OpenAir SRP Integration bundle (Bundle ID: 369637) to version 2.09 or later.

To use the autorecovery feature for expense reports exported from OpenAir to NetSuite as vendor bills, you must update NetSuite OpenAir SRP Integration bundle (Bundle ID: 369637) to version 3.0 or later.

Verify project export

Check this box to enable the autorecovery feature for project records exported to NetSuite as part of an interrupted export process.

OpenAir marks a record as exported and shows the ID of the corresponding record in NetSuite on the entity property form if the record exported successfully to NetSuite. If the export process is interrupted before OpenAir receives a response from NetSuite, records may be exported to NetSuite but not marked as exported in OpenAir. In this case, the record will be exported again the next time the integration workflow runs creating duplicate records in NetSuite.

The autorecovery feature prevents the creation of duplicated records by identifying records included for export in the interrupted integration run, and verifying if the record was already exported to NetSuite. If the record exists in NetSuite, OpenAir NetSuite Connector updates the exported record in OpenAir. Otherwise, the record will be exported again.

> ⚠ **Important:** To use the autorecovery feature with the Projects (Projects) export workflow, you must update NetSuite OpenAir SRP Integration bundle (Bundle ID: 369637) to version 3.0 or later.

Verify recognition transaction export

Check this box to enable the autorecovery feature for recognition transaction records exported to NetSuite as part of an interrupted export process.

OpenAir marks a record as exported and shows the ID of the corresponding record in NetSuite on the entity property form if the record exported successfully to NetSuite. If the export process is interrupted before OpenAir receives a response from NetSuite, records may be exported to NetSuite but not marked as exported in OpenAir. In this case, the record will be exported again the next time the integration workflow runs creating duplicate records in NetSuite.

The autorecovery feature prevents the creation of duplicated records by identifying records included for export in the interrupted integration run, and verifying if the record was already exported to NetSuite. If the record exists in NetSuite, OpenAir NetSuite Connector updates the exported record in OpenAir. Otherwise, the record will be exported again.

> ⚠ **Important:** To use the autorecovery feature with the Recognition Transactions (Revenue Recognition Transactions) export workflow, you must update NetSuite OpenAir SRP Integration bundle (Bundle ID: 369637) to version 2.09 or later.

Verify timesheet export

Check this box to enable the autorecovery feature for timesheet and time entry records exported to NetSuite as part of an interrupted export process.

OpenAir marks a record as exported and shows the ID of the corresponding record in NetSuite on the entity property form if the record exported successfully to NetSuite. If the export process is interrupted before OpenAir receives a response from NetSuite, records may be exported to NetSuite but not marked as exported in OpenAir. In this case, the record will be exported again the next time the integration workflow runs creating duplicate records in NetSuite.

The autorecovery feature prevents the creation of duplicated records by identifying records included for export in the interrupted integration run, and verifying if the record was already exported to NetSuite. If the record exists in NetSuite, OpenAir NetSuite Connector updates the exported record in OpenAir. Otherwise, the record will be exported again.

> ⚠ **Important:** To use the autorecovery feature with the Time Entries (Time) export workflow, you must update NetSuite OpenAir SRP Integration bundle (Bundle ID: 369637) to version 2.09 or later.

Legacy compatibility

This section of the NetSuite Connector administration form lets authorized integration administrators control legacy compatibility features for the integration. The following settings are available:

- DEPRECATED — Enable 3-way Salesforce.com, NS OpenAir and NetSuite integration
- Hide workflow groups
- Do not protect workflows

DEPRECATED — Enable 3-way Salesforce.com, NS OpenAir and NetSuite integration

⚠ **Important:** The 3-way Salesforce, OpenAir and NetSuite integration for customers and projects is deprecated and will be removed at a future date to be announced.

You should use the Projects (Projects) and Customers (Customers) export workflows instead. See Replacing the Deprecated 3-Way Salesforce, OpenAir and NetSuite Integration With the Customers (Customers) or Projects (Projects) Export Workflows.

The **Enable 3-way Salesforce.com, OpenAir and NetSuite integration** setting has been moved to the Legacy compatibility section of the NetSuite Connector administration form. You may continue using the 3-way Salesforce, OpenAir and NetSuite integration if it is already enabled for your account. However, it can no longer be enabled.

Enable Salesforce.com information exports from OpenAir to NetSuite for customers and projects. An OpenAir action can be run to unexport that information so you can delete it manually in NetSuite and re-export it from OpenAir.

Hide workflow groups

Check this box to disable the schedule Integration runs by workflow groups feature. You must first delete all workflow groups defined for your account on the NetSuite Connector Automate tab.

By default, NetSuite Connector lets you define workflow groups and schedule the automated integration runs for a specific group of workflows instead of all enabled workflows. See Scheduling the Integration to Run Automatically.

Note: You should define and use workflow groups instead of **[All]** in your integration schedules. Workflow groups let you run exactly the workflows you want when you need them to run and give you more control when setting up or maintaining your integration. The NetSuite Connector health check displays a warning if this setting is enabled. See OpenAir NetSuite Connector Health Check.

Do not protect workflows

This legacy compatibility setting can be used to disable the protected workflows feature. See Protected workflows.

Check this box if you need to modify workflows that are selected for active scheduled integration runs or set for real-time integration. To disable or delete a workflow, you must still first deactivate all scheduled integration runs and real-time integrations that use this workflow.

Note: You should not check this box and use the default workflow protection to avoid any accidental integration issues that could be caused if changes are made to a workflow when it is in use. The NetSuite Connector health check displays a warning if this setting is enabled. See OpenAir NetSuite Connector Health Check.

Other features

This section of the NetSuite Connector administration form lets authorized integration administrators enable other features used in the integration. The following settings are available:

- Use 2023.1 communication endpoint
- Show account information in integration run results
- Set time zone used by integration user setup on NetSuite to
- Map NetSuite transaction number to OpenAir number when importing vendor bills or purchase orders
- Expose NetSuite full synchronization control since date
- Enable strict mapping validation
- Enable NetSuite opportunity export link in project navigation bar
- Enable NetSuite OneWorld integration optimizations
- Enable Direct TBA
- Do not use OpenAir project exchange rate on invoice and recognition transaction export
- Allow objects with a NetSuite ID to be exported during integration run
- NetSuite custom fields for resource allocation records in field mapping for custom export
- Apply invoice line in NetSuite to expense using OpenAir ticket id
- Disable NetSuite ID change acknowledgment for customers and projects

Use 2023.1 communication endpoint

OpenAir supports NetSuite 2020.1 and 2023.1 communication endpoints.

Check the **Use 2023.1 communication endpoint** box to use the NetSuite 2023.1 communication endpoint. Otherwise, OpenAir NetSuite Connector uses the NetSuite 2020.1 endpoint by default.

After you change this setting and save the form, click the Tips menu icon and click **Synchronize NetSuite custom fields**.

Warning: The NetSuite 2016.2 communication endpoint is available without support until the NetSuite 2023.2 release, at which point it will be retired. See Support for Existing WSDL Versions (external link to NetSuite SuiteAnswers | Answer ID: 10817).

The NetSuite 2016.2 communication endpoint can no longer be used for the NetSuite <> OpenAir integration, and the New NetSuite Connector UI is required to use OpenAir NetSuite Connector with the NetSuite 2020.1 communication endpoint.

Show account information in integration run results

When enabled, your account information will be included in the integration run results.

Set time zone used by integration user setup on NetSuite to

Depending on the time zone settings in NetSuite and OpenAir, conversion problems may result in a date mismatch between NetSuite and OpenAir for some integration record. To prevent any date mismatch, enable this option and select a time zone to be used for the integration from the dropdown list. You should select the same time zone as the time zone set in the preferences for the integration user in NetSuite.

Map NetSuite transaction number to OpenAir number when importing vendor bills or purchase orders

> ⚠ **Important:** You should not enable this feature unless having the same reference numbers in NetSuite and OpenAir is a hard requirement. This feature could cause problems with purchase request and purchase order numbers in OpenAir. Contact OpenAir Professional Services to discuss possible implications before you enable this feature.

Check this box to map the purchase order number (**PO #**) or the vendor bill or credit reference number (**Reference No.**) in NetSuite to the purchase request name (**Purchase request #**) and purchase order name (**PO #**) in OpenAir when importing purchase order, vendor bills or vendor credits from NetSuite into OpenAir as purchase orders.

When checked, the **Purchase request #** and **PO #** for the purchase orders and purchase requests created in OpenAir are the same as the purchase order number (**PO #**) or the vendor bill or credit reference number (**Reference No.**) in NetSuite under the following conditions.

- The **PO #** [`tranID`] or the vendor bill or credit **Reference No.** [`tranID`] must use the following format — `<prefix><number>`, where:
    - `<prefix>` is an optional prefix. If used, it must contain only non-numeric characters. The purchase request number (**Purchase request #**) and purchase order number (**PO #**) in OpenAir include the prefix set in the mapping definition table (`NS-`, `NS-VB-`, or `NS-VC-`, by default) followed by the `<prefix>`.
    - `<number>` is an integer greater than `0` and less than or equal to `1000000000`.
        - It must contain only numeric characters.
        - The `<number>` may contain leading zeros. Leading zeros will be stripped in the purchase request number (**Purchase request #**) and purchase order number (**PO #**) in OpenAir.
    - `tranID` may include another part starting with a non-numeric character immediately to the right of `<number>` is ignored. For example, if the NetSuite **PO #** is `purchase126A1B2`, the **PO #** for the purchase order imported into OpenAir will be `NS-purchase126`.
- The number part of the **Purchase request #** or **PO #** in OpenAir must be unique.
    - When a purchase request record and a purchase order record are created in OpenAir by the integration, the number part of the **Purchase request #** or **PO #** is generated automatically if a **Purchase request #** or **PO #** with the NetSuite `<number>` already exists in OpenAir, or if the `<number>` is less than or equal to the last number generated by OpenAir. For example, if the NetSuite **PO #** is `purchase126` and the last **PO #** generated in OpenAir was `PO-126`, the **PO #** for the purchase order imported into OpenAir will be `NS-purchase127`.
    - When a purchase request record and a purchase order record are updated in OpenAir by the integration, the number part of the **Purchase request #** or **PO #** is generated automatically if **Purchase request #** or **PO #** with duplicate number part exists in OpenAir.

Note: By default, OpenAir generates the **Purchase request #** or **PO #** automatically when a purchase order or purchase request record is created. The **Purchase request #** and **PO #** values include a constant prefix and a unique number:

- The **prefix** [`purchaserequest.prefix` or `purchaseorder.prefix`] is set on the purchase request layout (Administration > Application Settings > Purchases > Purchase request layout) and purchase order layout (Administration > Application Settings > Purchases > Purchase request layout) in OpenAir. All purchase requests and purchase orders created manually in OpenAir
- The **number** is generated automatically with an increment of 1 when a record is created. You can set the starting number on the purchase request and purchase order layouts. This ensures that each purchase request or purchase order have a unique **Purchase request #** or **PO #**.

When purchases requests and purchase orders are created in OpenAir by the POs (Purchase Orders), POs (Vendor Bills), or POs (Vendor Credits) import workflow, by default (when the **Map NetSuite transaction number to OpenAir number when importing vendor bills or purchase orders** is not checked):

- The **prefix** is set by the field mapping definition. By default, the prefix is `NS-` when importing NetSuite purchase orders into OpenAir, `NS-VB-` when importing NetSuite vendor bills into OpenAir as purchase orders, or `NS-VC-` when importing NetSuite vendor credits into OpenAir as purchase orders.
- The **number** is generated automatically in the same way as when records are created manually.

Expose NetSuite full synchronization control since date

When enabled, the **Run NetSuite full synchronization** link is available from the tips menu.

You can specify a date to restrict the synchronization to records last modified on or after this date:

- If you specify a date, clicking **Run NetSuite full synchronization** will run the synchronization and include all records last modified on or after midnight Pacific Time (`00:00:00.000-08:00`/`00:00:00.000-07:00`) of the date specified across all selected workflows.

  Note: During winter months, the Pacific Time Zone is eight hours behind Coordinated Universal Time (UTC-8) and is referred to as Pacific Standard Time (PST). In summer months, the Pacific Time Zone is seven hours behind Coordinated Universal Time (UTC-7) and is referred to as Pacific Daylight Time (PDT).

- Otherwise, clicking **Run NetSuite full synchronization** will run the synchronization and include all records across all selected workflows regardless of when the record was last modified.

Enable strict mapping validation

This will enforce front-end strict mapping validation for field mapping definitions. See Enforcing Front-End Mapping Validation.

Enable NetSuite opportunity export link in project navigation bar

Check this box only if Enable OpenAir project to NetSuite opportunity integration is also enabled.

When enabled, the “Update the NetSuite opportunity line items” link is moved from the Tips section into the header section next to the Edit project link and the link is renamed to Finalize WBS. The link is visible with all screens of a project.

Enable NetSuite OneWorld integration optimizations

Enable OneWorld integration optimizations to speed up data transfer in OneWorld accounts.

Enable Direct TBA

Double the speed of your integration with an optimized link between OpenAir and NetSuite.

Direct token-based authentication (TBA) utilizes a more robust communication channel between OpenAir and NetSuite to deliver faster integration runs. It also simplifies the sign-in credentials administration.

To enable the feature for your account, configure NetSuite for direct TBA, and enable the feature in the NetSuite integration administration form in OpenAir.

To configure NetSuite for direct TBA:

1. In NetSuite, go to Customization > SuiteBundler > Search & Install bundles > List.
2. Update the NetSuite OpenAir SRP Integration bundle to the latest version.
3. Go to Setup > Users/Roles > Access Tokens > New.
4. Set the **Application Name** to `Internal NS Application OPENAIR`.
5. Select the **User** — This is the **Name** on the employee record for the TBA user you chose.
6. Select the **Role** — You can use the NSOA Connector role created by the NetSuite OpenAir SRP Integration bundle, or the custom role you created for the integration. See Creating a Custom Role for the Integration in NetSuite.

   > **Important:** You should not associate the access token with the Administrator role. Some role permissions may have an undesirable effect on the integration. For example, OpenAir transactions may be posted to a closed accounting period in NetSuite if the “Override Period Restrictions” role permission is enabled for the TBA user.
   >
   > See also Frequently Asked Questions (FAQs).

7. NetSuite generates the **Token Name** automatically. You can change the token name or keep the default value.
8. Click **Save**.

   NetSuite generates the **User Token ID** and **User Token Secret** and displays the information on the Access Token page.

9. Copy the **User Token ID** and **User Token Secret** and store the information in a safe place. You will need to enter this information when configuring OpenAir for the integration.

   > ⚠ **Important:** The User Token ID and User Token Secret are displayed only one time. You will not be able to retrieve this information after you close, refresh or leave the page. You can generate a new User Token ID and User Token Secret at any time. To generate a new User Token ID and User Token Secret, revoke the existing token and create a new one.

To enable direct TBA in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button, then click **NetSuite Connector Administration**. The OpenAir NetSuite Connector administration form appears.
3. Check the **Enable Direct TBA** box under the Other features section.
4. Click **Save**.
5. On the NetSuite Connector Credentials page, enter the **TBA User Token ID** and **TBA User Token secret** that you obtained when configuring NetSuite for direct TBA. Note that the **Application ID**, **TBA Consumer Key**, and **TBA Consumer Secret** fields no longer show on the Credentials page.

Do not use OpenAir project exchange rate on invoice and recognition transaction export

This feature will suppress the usage of the OpenAir project exchange rate on invoice and revenue recognition transaction export. The exchange rate used on the invoice and journal entry in NetSuite will be sourced from NetSuite. The project exchange rate in OpenAir will no longer be required when exporting invoices and revenue recognition transactions.

Allow objects with a NetSuite ID to be exported during integration run

When checked, all records with an associated NetSuite ID will be exported during the integration run.

NetSuite custom fields for resource allocation records in field mapping for custom export

A current limitation in NetSuite web services communication endpoint does not allow OpenAir NetSuite Connector to list the custom CRM field for NetSuite resource allocation records automatically for mapping.

When you create a custom export workflow exporting OpenAir records to NetSuite as resource allocation records, only the standard fields for NetSuite resource allocation records, and the custom CRM field holding the internal ID of the matching OpenAir record [`_oa_resource_allocation_id`] are available for mapping by default.

To make other custom CRM field for resource allocation records available for mapping:

1. On the OpenAir NetSuite Connector administration form, enter the list of custom CRM field IDs in the **NetSuite custom fields for resource allocation records in field mapping for custom export** box, separated by commas.

2. Click **Save**.
3. Click **Close**.
4. Click the Tips button, then Synchronize NetSuite custom fields.

For more information about custom export workflows, see Custom Export Workflows.

For more information about the Resource Allocations feature in NetSuite, see Resource Allocations (external link to NetSuite SuiteAnswers | Answer ID: 31868).

For more information about creating a custom CRM field, see Creating Custom CRM Fields (external link to NetSuite SuiteAnswers | Answer ID: 10082).

Apply invoice line in NetSuite to expense using OpenAir ticket id

Check this box to add the **OpenAir Expense Report Line ID** custom field to expense lines in NetSuite and populate it automatically with the internal ID of the OpenAir receipt when exporting an expense report from OpenAir to NetSuite. NetSuite Connector will use the **OpenAir Expense Report Line ID** custom field, if it is populated and visible on the expense report form in NetSuite, to match the relevant expense line when exporting the invoice from OpenAir to NetSuite.

This setting resolves an issue that prevented NetSuite Connector to match the expense line resulting in failed invoice exports from OpenAir to NetSuite. Enabling this setting will ensure that the **OpenAir Expense Report Line ID** is populated automatically on the appropriate NetSuite transaction record in future expense report exports. To troubleshoot previous failed invoice exports, update the **OpenAir Expense Report Line ID** custom field manually in NetSuite for each expense lines on each expense report using the internal ID of the corresponding OpenAir receipt, then export the invoices again.

Important: Make sure the **OpenAir Expense Report Line ID** custom field is visible on the default Expense report form in NetSuite.

1. In NetSuite, sign in as an Administrator and go to Customization > Forms > Transaction forms.
2. Click the name of the transaction form with **type** Expense report and **Preferred** box checked.
3. On the Screen fields subtab, click the Expenses subsection further to the right.
4. Check the **show** box for the **OpenAir Expense Report Line ID** custom field.
5. Click **Save**.

Disable NetSuite ID change acknowledgment for customers and projects

Check this box to disable the safeguard preventing the NetSuite ID from being changed or deleted accidentally. You should not check this box — a wrong NetSuite ID can impact adversely other parts of the integration.

Note: The safeguard adds a two-step process to save a customer or project record in OpenAir that was already integrated when the value of the NetSuite ID custom field was changed. If the NetSuite ID was changed and you click **Save**, an error message appears and the **Acknowledge NetSuite ID change** box is added to the form. You must check the **Acknowledge NetSuite ID change** box and click **Save** again.

This safeguard can be useful when the real-time integration is used to export customer and project records to NetSuite, for example. If a user creates a new record, the save operation exports the record to NetSuite and the NetSuite ID is recorded in the OpenAir database. Without reloading the page, the NetSuite ID field still shows blank, however, and if the user clicks **Save** again accidentally and the safeguard is disabled, the NetSuite ID will be deleted.

Performance features

This section of the NetSuite Connector administration form lets authorized integration administrators control performance features used in the integration. The following settings are available:

- Exclude reporting values from Update NetSuite tasks with OpenAir task data
- Exclude reporting values from Update NetSuite projects with OpenAir project data
- Use direct filters

Exclude reporting values from Update NetSuite tasks with OpenAir task data

Check this box to limit the range of OpenAir information available for mapping to the information in the `project_task` table and speed up the “Update NetSuite tasks with NS OpenAir task data” export workflow.

When checked:

- The dropdown values for the OpenAir field on the mapping definition table are sourced directly from the task record. This limits the range of OpenAir information available for mapping but the workflow runs faster.

  Otherwise, the dropdown values for the OpenAir field are sourced from a detailed report that is generated internally before any information can be exported. It makes a rich range of OpenAir information available for mapping but the workflow runs much slower.
- You can use advanced mapping for the Update NetSuite tasks with OpenAir task data integration workflow.

You should check this box unless you need the additional information available in the detailed report and performance is not a concern for your integration.

Important: The mapping must be redefined entirely when you change this setting. This is because the integration workflow uses two completely different sets of information from different resources for the OpenAir fields when report values are included or excluded.

Delete the “Update NetSuite tasks with NS OpenAir task data” workflow before you change this setting. If you change the “Exclude report values from Update NetSuite tasks with OpenAir task data” setting and attempt to save the NetSuite Connector administration form, an error message will prompt you to delete the workflow — disabling the workflow is not sufficient.

Exclude reporting values from Update NetSuite projects with OpenAir project data

Check this box to limit the range of OpenAir information available for mapping to the information in the `project` table and speed up the “Update NetSuite projects with NS OpenAir project data” export workflow.

When checked:

- The dropdown values for the OpenAir field on the mapping definition table are sourced directly from the project record. This limits the range of OpenAir information available for mapping but the workflow runs faster.

  Otherwise, the dropdown values for the OpenAir field are sourced from a detailed report that is generated internally before any information can be exported. It makes a rich range of OpenAir information available for mapping but the workflow runs much slower.
- You can use advanced mapping for the Update NetSuite projects with OpenAir project data integration workflow.

You should check this box unless you need the additional information available in the detailed report and performance is not a concern for your integration.

Important: The mapping must be redefined entirely when you change this setting. This is because the integration workflow uses two completely different sets of information from different resources for the OpenAir fields when report values are included or excluded.

Delete the “Update NetSuite projects with NS OpenAir project data” workflow before you change this setting. If you change the “Exclude report values from Update NetSuite projects with OpenAir project data” setting and attempt to save the NetSuite Connector administration form, an error message will prompt you to delete the workflow — disabling the workflow is not sufficient.

Use direct filters

Check this box to use direct filters for export workflows and speed up your integration.

When checked, OpenAir NetSuite Connector retrieves only those records matching the filter conditions from the database. Otherwise, OpenAir NetSuite Connector retrieves all records and evaluates the each record against the filter conditions.

Important: Make sure all existing filters are valid before enabling direct filters. Previously, custom filters containing syntax errors were ignored — all records matching the default filter conditions were exported. When using direct filters, an error is returned and the export fails if there is a syntax error in your custom filter. You should test this feature on a sandbox account to validate existing filters before you enable it on your production account.

Note: Direct filtering is always used for:

- Update NetSuite Projects with OpenAir Project Data export workflow. See Update NetSuite Projects with OpenAir Project Data Export Workflow.
- Update NetSuite Tasks with OpenAir Task Data export workflow. See Update NetSuite Tasks with OpenAir Task Data.
- Custom workflows exporting published OpenAir report data to NetSuite custom records, independently of the **Use direct filters for export workflows** administration setting. For more information about published report custom exports, see .

The **Use direct filters** setting has no impact on these workflows.

Custom Imports

You can define up to 10 custom imports from NetSuite to OpenAir. To import NetSuite Record, specify the NetSuite saved search ID, the NetSuite record type, the NetSuite custom record type ID (if importing custom records) and the NetSuite OpenAir record type you want to map the NetSuite record to.

Note: A new approach to custom import workflows was introduced in the April 15, 2023 OpenAir release. You should replace any custom imports defined in this section of the OpenAir NetSuite Connector form with this new approach. It is more flexible and does not rely on NetSuite saved searches. See Custom Import and Custom Import Workflows.

Exchange rate integration

After you have enabled and configured multiple currencies in both NetSuite and OpenAir, you can use the exchange rate integration to:

- Import exchange rates from NetSuite in to OpenAir daily, weekly, semimonthly or monthly, so the exchange rates in OpenAir are the same as in NetSuite on the import date. To do so, check the **Enable NetSuite exchange rate integration**, and set the **Day** and time (**Hour** and **Minute**).

  The import time should be 10-20 minutes after the exchange rate are updated in NetSuite to ensure that exchange rates match between the two systems.

  - In OpenAir, exchange rates are updated one time each day at approximately 3 a.m. Eastern Time (UTC−5).
  - In NetSuite, exchange rates are updated one time each day at approximately 6 a.m. in one of four available time zones when using the Currency Exchange Rate Integration feature. For more information about the Currency Exchange Rate Integration feature in NetSuite, see Currency Exchange Rate Integration (external link to NetSuite SuiteAnswers | Answer ID: 7730).

Note: To maximize the amount of time in the day when foreign currency exchange rates in OpenAir and NetSuite match, you should do both of the following:

1. In NetSuite, set the **Update Time Zone** to U.S. Eastern Time (UTC-5) on the base currency record.
2. On the OpenAir NetSuite Connector Administration form, under the Exchange rate integration section, set the import day and time to every day 10-20 minutes after 6 a.m..

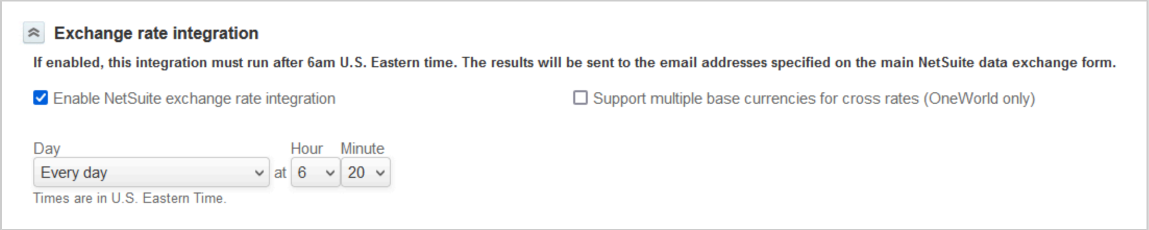
If you select a different Update Time Zone on the base currency record in NetSuite, adjust the import time accordingly. Note that by doing so, the amount of time in the day when foreign currency exchange rates in OpenAir and NetSuite match will be smaller.

- Support multiple base currencies for cross rates, if you have a NetSuite OneWorld account. To do so, check the **Support multiple base currencies for cross rates (OneWorld only)** box.

  OpenAir calculates the cross rates between two non-base currencies against the root subsidiary base currency in NetSuite. NetSuite does not calculate cross rates between two non-base currencies, only base currency to foreign currency rates.

Important: Make sure you enable support for multiple base currencies for cross rates in NetSuite. To do so:

1. Do one of the following:
   - Go to Setup > Company > General Preferences and click the Custom Preferences tab, if you are using the SuiteScript 1 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
   - Go to Customization > Lists, Records and Fields > Record Types > [*View list forOpenAir: SRP Integration Settings*] > [*View or Edit OpenAir SRP Integration Settings Record*], if you are using the SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
2. Check the **Exchange Rate Integration - OneWorld** box.



Exchange rate integration

If enabled, this integration must run after 6am U.S. Eastern time. The results will be sent to the email addresses specified on the main NetSuite data exchange form.

☑ Enable NetSuite exchange rate integration

□ Support multiple base currencies for cross rates (OneWorld only)

Day: Every day at Hour: 6 Minute: 20

Times are in U.S. Eastern Time.

Workflow Settings

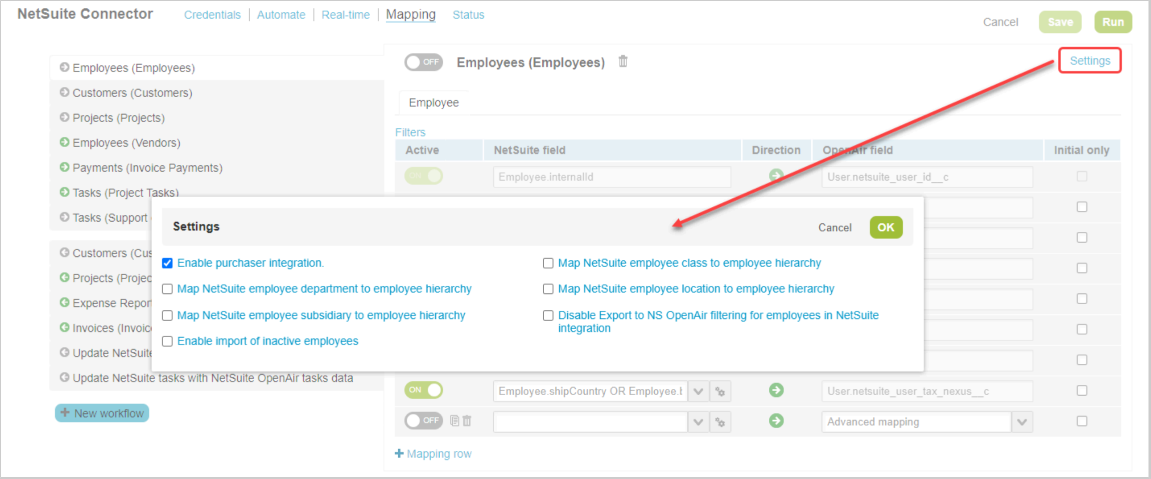
You can control settings specific to each integration workflow from the OpenAir NetSuite Connector Mapping tab.

To view the settings specific to an integration workflow:

1. In OpenAir, go to Administration > NetSuite Connector > Mapping.
2. Click **Settings** in the top right corner. The workflow settings form appears.

> **Note:** The Mapping screen only shows the **Settings** link if there are any specific settings for the selected workflow.

> **Tip:** If a setting label shows in blue color, click the label to view the relevant help topic for this setting in the OpenAir Help Center (requires the View Help Center role permission).



☑ Enable purchaser integration.

☐ Map NetSuite employee department to employee hierarchy

☐ Map NetSuite employee subsidiary to employee hierarchy

☐ Enable import of inactive employees

☐ Map NetSuite employee class to employee hierarchy

☐ Map NetSuite employee location to employee hierarchy

☐ Disable Export to NS OpenAir filtering for employees in NetSuite integration

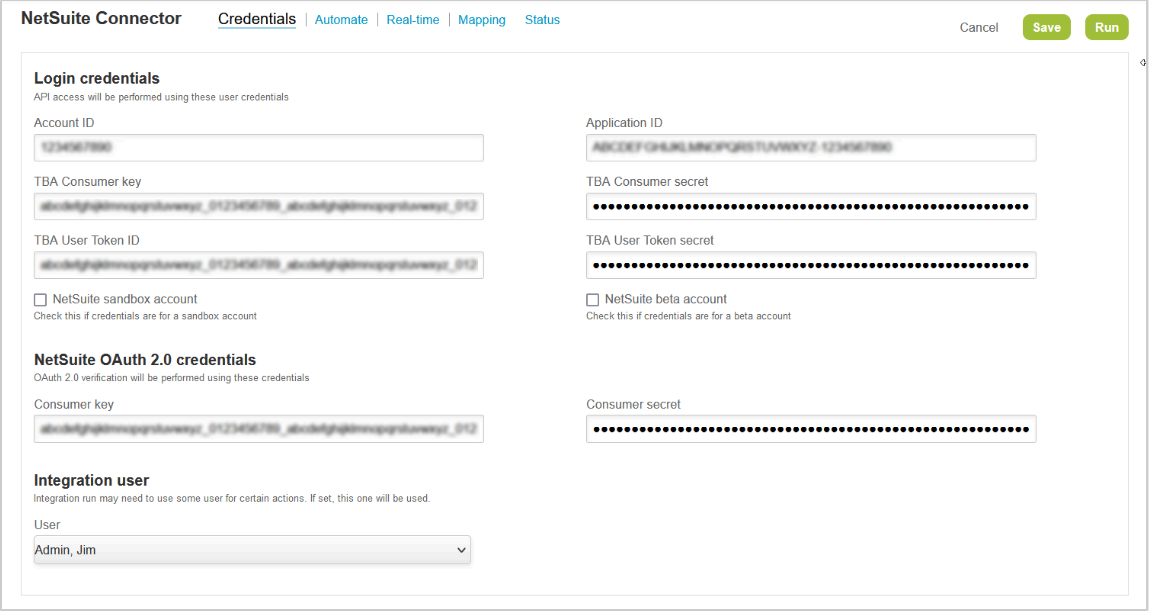
Workflow-specific settings exist for the following standard integration workflows:

- Import Workflows:
  - Employees (Employees) — See Employees (Employees) Import Workflow Settings
  - Customers (Customers) — See Customers (Customers) Import Workflow Settings
  - Projects (Projects) — Projects (Projects) Import Workflow Settings
  - Services (Service Items) — See Services (Service Items) Import Workflow Settings
  - Vendors (Vendors) — See Vendors (Vendors) Import Workflow Settings
  - Customer POs (Sales Orders) — See Customer POs (Sales Orders) Import Workflow Settings
  - Credits (Applied Credit Memos) — See Credits (Applied Credit Memos) Import Workflow Settings
- Export Workflows:
  - Customers (Customers) — See Customers (Customers) Export Workflow Settings
  - Expense Reports (Expense Reports) — See Expense Reports (Expense Reports) Export Workflow Settings.
  - Invoices (Invoices) — See Invoices (Invoices) Export Workflow Settings
  - Time Entries (Time) — See Time Entries (Time) Export Workflow Settings
  - Recognition Transactions (Revenue Recognition Transactions) — See Recognition Transactions (Revenue Recognition Transactions) Export Workflow Settings
  - Send OA Credit (negative invoice) — See Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Workflow Settings

- Update NetSuite projects with OpenAir project data — See Update NetSuite projects with OpenAir project data Export Workflow Settings
- WBS (Opportunity) — See WBS (Opportunity) Export Workflow Settings

Connecting OpenAir to NetSuite for the Integration

OpenAir NetSuite Connector uses the information on the Credentials tab to establish a secure connection with NetSuite. You need to create an integration record for the NetSuite <> OpenAir integration and enable at least one employee to use the integration in NetSuite first, before you can enter the information on this form. See Configuring NetSuite for the Integration.



Login credentials
API access will be performed using these user credentials

Account ID
[redacted]

Application ID
[redacted]

TBA Consumer key
[redacted]

TBA Consumer secret
••

TBA User Token ID
[redacted]

TBA User Token secret
••

☐ NetSuite sandbox account
Check this if credentials are for a sandbox account

☐ NetSuite beta account
Check this if credentials are for a beta account

NetSuite OAuth 2.0 credentials
OAuth 2.0 verification will be performed using these credentials

Consumer key
[redacted]

Consumer secret
••

Integration user
Integration run may need to use some user for certain actions. If set, this one will be used.

User
Admin, Jim

To connect OpenAir to NetSuite for the integration:

1. In OpenAir, go to Administration > NetSuite Connector > Credentials.
2. Enter your NetSuite **Account ID**. The value you enter must be exactly as it is shown on the SOAP web services preference page in NetSuite (Setup > Integration > SOAP Web Services Preferences).
3. Enter the **Application ID**, **TBA Consumer Key**, and **TBA Consumer Secret** which you copied from NetSuite when creating an integration record in NetSuite for the OpenAir integration. See Creating an Integration Record for the OpenAir Integration.
4. Enter the **TBA User Token ID** and **TBA User Token secret** corresponding to the access token created for you to use the integration. See Creating and Assigning an Access Token for the Integration.

   ✓ **Tip:** NetSuite users with the User Access Token permission can view their existing access tokens in NetSuite. To view your access tokens, click **Manage Access Tokens** in the **Settings** portlet,

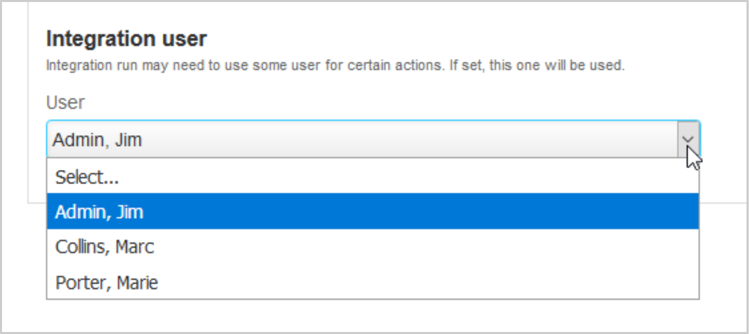
5. Clear both the **NetSuite sandbox account** and **NetSuite beta account** boxes if you are setting up the integration with your NetSuite production account or your NetSuite sandbox account. Check the **NetSuite beta account** box to connect to your beta account.

Important: Do not check the **NetSuite sandbox account** of you are setting up the integration with a NetSuite sandbox account. Before the 2018.1 NetSuite release, you needed to check the **NetSuite sandbox account** box to connect OpenAir to your NetSuite sandbox account as separate domains were used for NetSuite sandbox and production accounts. With effect from the 2018.1 NetSuite release, the account identifier (**Account ID**) is the only thing that differentiates your NetSuite sandbox accounts from your NetSuite production account. Typically, your sandbox Account ID is your production Account ID followed by a the suffix _SB and a number.

6. (Optional) If the OpenAir NetSuite single sign-on integration is enabled, the Credentials form includes the “Single sign-on credentials” or "NetSuite OAuth 2.0 credentials" section. Enter the **Consumer key** and the **Shared secret** or **Consumer secret** generated when you enabled the NetSuite single sign-on integration in NetSuite. For more information about enabling the NetSuite single sign-on integration, see Enabling Access to OpenAir From Within NetSuite and Enabling the OpenAir Within NetSuite Feature in OpenAir.

7. (Optional) You can select an active administrator as the designated **Integration user** for all integration runs.

   - The integration user must be an administrator and an active user. Only active administrators are listed in the dropdown options.
   - OpenAir NetSuite Connector will use the selected integration user’s permissions for all scheduled, manual or real-time integration runs.
   - When the integration creates or updates a record in OpenAir, the audit trail for that record shows the selected integration user as the author of the changes. The audit trail also shows the record was created or updated using the NetSuite integration.
   - The selected integration user’s permissions are protected — the user cannot be set as inactive and the user’s role cannot be changed.



Integration user

Integration run may need to use some user for certain actions. If set, this one will be used.

User

Admin, Jim

To clear the designated integration user, choose the **Select...** option from the **Integration user** dropdown options. When there are no Integration user selected, OpenAir NetSuite Connector reverts back to the default behavior, as follows:

- The owner of manual integration runs is the user who triggers the integration.
- The owner of scheduled or real-time integration runs is the user who set the schedule or real-time integration, if that user record exists, or another administrator otherwise.
- The owner of the integration run shows as the author of the changes in OpenAir record audit trails.

8. Click **Save**.

Adding and Managing Integration Workflows

To view and manage the list of workflows configured for the NetSuite <> OpenAir integration, in OpenAir, go to Administration > NetSuite Connector > Mapping.

1 **List of integration workflows** — The list in the left pane shows all configured integration workflows in the order of execution. Workflow names include the OpenAir record type followed by the NetSuite record type in round brackets.

The arrow icon next to the workflow name indicates if it is an import or export workflows and if the workflow is active or not.

- Arrow pointing right ➡ — Import workflow (workflow importing records from NetSuite to OpenAir).
- Arrow pointing left ⬅ — Export workflow (workflow exporting records from OpenAir to NetSuite).
- Green icon background ➡ — Active workflow (the workflow can be included in your integration runs).
- Gray icon background ➡ — Inactive workflow (the workflow cannot be included in your integration runs).

Click a workflow name to show the field mapping and other controls for this workflow. The selected workflow is indicated with a white color background (instead of gray). For more information about field mappings, see Mapping OpenAir Fields and NetSuite Fields.

2 **Active switch** — Click the switch next to the workflow name to activate ON ● or to deactivate ● OFF the selected workflow.

You can only include an integration workflow in an integration run if this workflow is active. Conversely, you cannot deactivate an integration workflow if this workflow is included in a workflow group selected for an active scheduled integration run, or if the corresponding OpenAir record type is selected for an active real-time integration. See Protected workflows.

3 **Removing a workflow from the list** — To remove the workflow from the list, click the delete icon 🗑 .

You cannot remove a workflow from the list if this workflow is included in a workflow group selected for an active scheduled integration run, or if the corresponding OpenAir is selected for an active real-time integration. See Protected workflows.

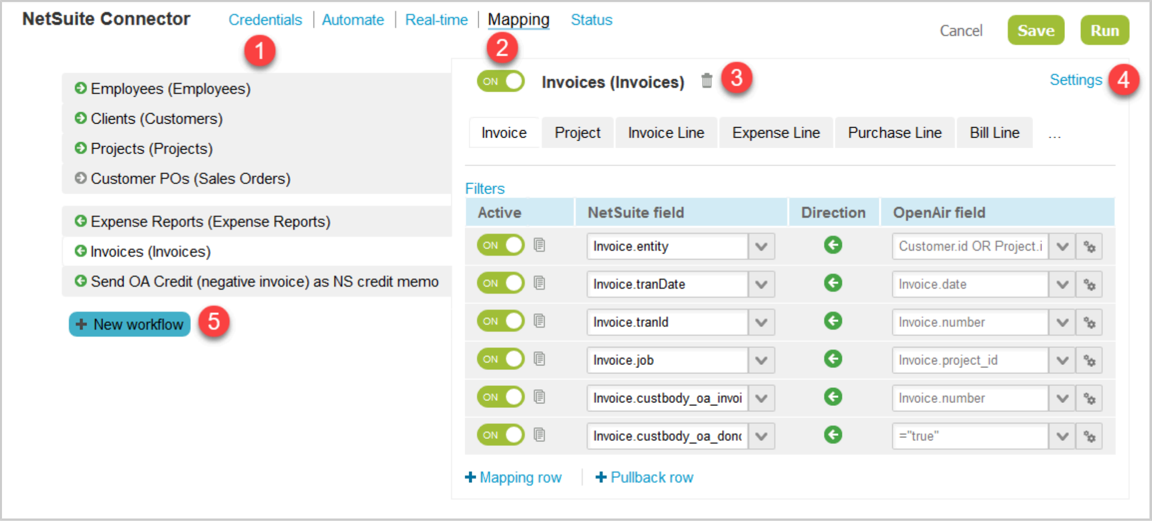
4 **Workflow settings** — To view configuration settings and features specific to the selected workflow, click **Settings**. See Workflow Settings.

You cannot change any of the workflow settings if this workflow is included in a workflow group selected for an active scheduled integration run, or if the corresponding OpenAir record type is selected for an active real-time integration. See Protected workflows.

The NetSuite Connector administration form also includes settings that can be used to enhance the performance of particular workflows. See Performance features.

5 **Adding an integration workflow** — To add an integration workflow, click **New workflow** then click the name of the workflow you want to add from the list.

- The arrow icon next to each workflow name indicates if it is an import or export workflow (See item 1 in this table).
- You can only add workflows if the workflow is enabled on the OpenAir NetSuite Connector administration form. Others are not listed. See NetSuite Connector Administration Settings.
- After you add a workflow:
    - The workflow appears in its predefined position on the list, according to the order of execution.
    - The workflow is inactive. You can click the active switch to activate the workflow
    - The field mapping definition table shows the default configuration with all default field mapping rows set to active.



NetSuite Connector Credentials | Automate | Real-time | Mapping Status Cancel Save Run

- Employees (Employees)
- Clients (Customers)
- Projects (Projects)
- Customer POs (Sales Orders)
- Expense Reports (Expense Reports)
- Invoices (Invoices)
- Send OA Credit (negative invoice) as NS credit memo

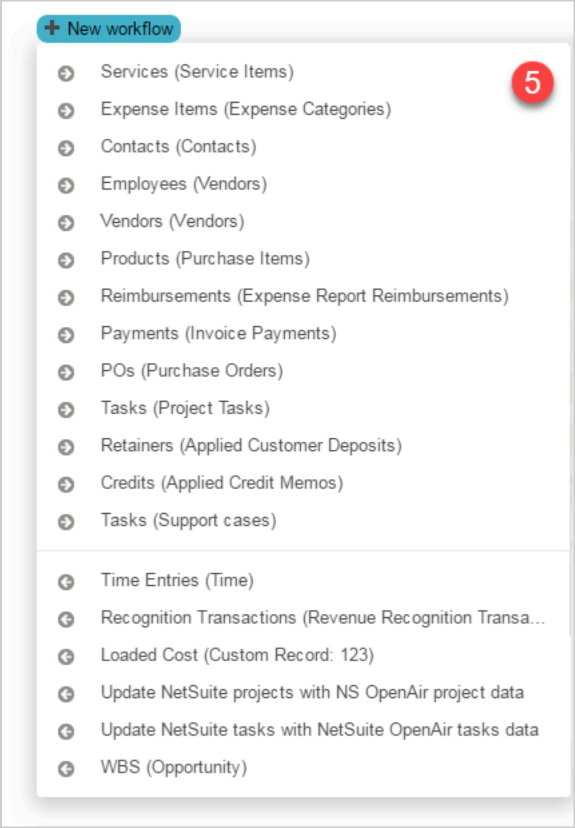
ON Invoices (Invoices) Settings

Invoice | Project | Invoice Line | Expense Line | Purchase Line | Bill Line | ...

Filters

Active	NetSuite field	Direction	OpenAir field
ON	Invoice.entity	←	Customer.id OR Project.i
ON	Invoice.tranDate	←	Invoice.date
ON	Invoice.tranId	←	Invoice.number
ON	Invoice.job	←	Invoice.project_id
ON	Invoice.custbody_oa_invoi	←	Invoice.number
ON	Invoice.custbody_oa_donc	←	="true"

+ Mapping row | + Pullback row



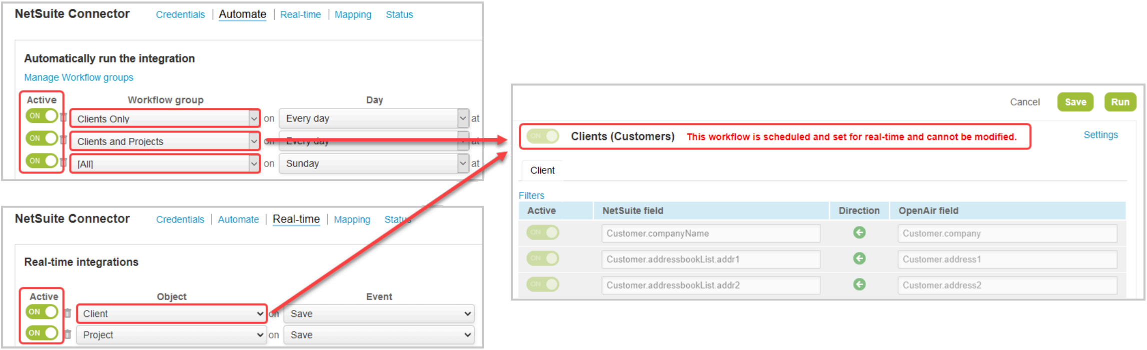
- Services (Service Items)
- Expense Items (Expense Categories)
- Contacts (Contacts)
- Employees (Vendors)
- Vendors (Vendors)
- Products (Purchase Items)
- Reimbursements (Expense Report Reimbursements)
- Payments (Invoice Payments)
- POs (Purchase Orders)
- Tasks (Project Tasks)
- Retainers (Applied Customer Deposits)
- Credits (Applied Credit Memos)
- Tasks (Support cases)
- Time Entries (Time)
- Recognition Transactions (Revenue Recognition Transa…
- Loaded Cost (Custom Record: 123)
- Update NetSuite projects with NS OpenAir project data
- Update NetSuite tasks with NetSuite OpenAir tasks data
- WBS (Opportunity)

Protected workflows

A workflow cannot be modified if it is included in an active scheduled integration run or set for an active real-time integration. It is protected against any changes that may impact your integration — you cannot modify its mappings or settings, and you cannot disable the workflow. To modify a protected workflow, you must first deactivate all scheduled integration runs and real-time integrations that protect this workflow. This is to avoid any accidental integration issues that could be caused if changes are made to a workflow when it is in use.

Important: If you set **Workflow group** to the metavalue [All] for one of your scheduled integration runs, all enabled workflows are protected. As soon as you enable a workflow and save the NetSuite Connector configuration, this workflow becomes protected and cannot be edited.

A legacy compatibility setting lets you disable the protected workflows feature. Disabling protected workflows is not recommended. The option lets you retain the ability to modify the workflow settings and mappings if your scheduled integration runs include all workflows. See Do not protect workflows.


NetSuite Connector
Credentials | Automate | Real-time | Mapping Status
Manage Workflow groups
NetSuite Connector
Credentials | Automate | Real-time | Mapping Status
Cancel
Save
Run
Settings
Client
Filters

Automatically run the integration

Active	Workflow group		Day	
ON	Clients Only	on	Every day	at
ON	Clients and Projects	on	Every day	at
ON	[All]	on	Sunday	at

Real-time integrations

Active	Object		Event
ON	Client	on	Save
ON	Project	on	Save

ON Clients (Customers) This workflow is scheduled and set for real-time and cannot be modified.

Active	NetSuite field	Direction	OpenAir field
ON	Customer.companyName	←	Customer.company
ON	Customer.addressbookList.addr1	←	Customer.address1
ON	Customer.addressbookList.addr2	←	Customer.address2

Mapping OpenAir Fields and NetSuite Fields

OpenAir NetSuite Connector lets you map OpenAir fields to NetSuite fields for each integration workflow. To view or modify field mappings, go to Administration > NetSuite Connector > Mapping and select a workflow. See also Adding and Managing Integration Workflows.

Field mapping definitions are organized into mapping group subtabs. Each mapping group subtab corresponds to a specific record type or hierarchy (category) with a different field mapping definition table. The following table provides a quick reference to field mapping definitions on the Mapping tab.

You cannot make any change to the field mapping for an integration workflow if this workflow is included in a workflow group selected for an active scheduled integration run, or if the corresponding OpenAir record type is selected for an active real-time integration. See Protected workflows.

1	**Mapping group subtabs** — Field mapping definitions are organized into mapping groups. Each mapping group corresponds to a specific record type or hierarchy (category). If there are more mapping group subtabs than can be displayed on the screen, click the more icon to view the hidden groups. Mapping group subtab are listed left to right in their order of execution. Click a subtab to view the field mapping definitions for that mapping group. **Note:** Each mapping group corresponds to an integration point tag in the legacy advanced tag notation. Pullback integration point tags are included in the same mapping group as PUSH integration point tags. See Advanced Mapping and Filter Notation Reference.
2	**Filters** — You can define filters to integrate only the records that meet specific criteria. Click **Filters** to view or set filters for the mapping group. The **Filters** link appears in bold if filters are set. Filters use the advanced tag notation. See Using Filters.
3	**Active switch and actions** — The first column in the field mapping definition table lets you activate and deactivate, copy, or delete field mapping rows.

- Click the switch under the **Active** column header to activate ON or to deactivate OFF the field mapping defined in that row.
- Click the copy icon ⎘ to duplicate the mapping row and add it as a new custom field mapping.
- Click the delete icon 🗑 to delete the mapping row.

4 **NetSuite field** — The second column in the field mapping definition table shows the NetSuite fields. Select a field from the available dropdown options ⌄. If the NetSuite field column is the source data (for an import workflow from NetSuite to OpenAir), you can also enter a value. For example, enter `="1"` to set the source data as a constant (1 in this case).

NetSuite fields are listed using the following format: `<SoapObjectName>.<fieldName>`, where:

- `<SoapObjectName>` is the name of the NetSuite SOAP web services object and uses Pascal case convention.
- `<fieldName>` is the name of the field and uses Camel case convention.

Refer to the NetSuite SOAP Schema Browser for information about NetSuite fields available for mapping — See Schema Reference Documentation.

> **Note:** When importing NetSuite transaction records, or pulling back information from NetSuite transaction records, the NetSuite fields are listed as `Transaction.<fieldName>` when in fact the SOAP web services object corresponding to the related searches are used to fetch the information from NetSuite.

5 **Direction** — The second column in the field mapping definition table shows an arrow which indicates whether the mapped field data is imported, exported or pulled back:

- Arrow pointing right on green color background → — the mapped field data is imported from NetSuite into OpenAir.
- Arrow pointing left on green color background ← — the mapped field data is exported from OpenAir to NetSuite.
- Arrow pointing right on orange color background → — the mapped field data is pulled back from NetSuite when a record is created.

6 **OpenAir field** — The fourth column in the field mapping definition table shows the NetSuite fields. Select a field from the available dropdown options ⌄. If the OpenAir field column is the source data (for an export workflow from OpenAir to NetSuite), you can also enter a value. For example, enter `="1"` to set the source data as a constant (1 in this case).

OpenAir fields are listed using the following format: `<Table_name>.<field_name>`, where:

- `<Table_name>` is the name of the table in the OpenAir database.
- `<field_name>` is the name of the field in the OpenAir database.

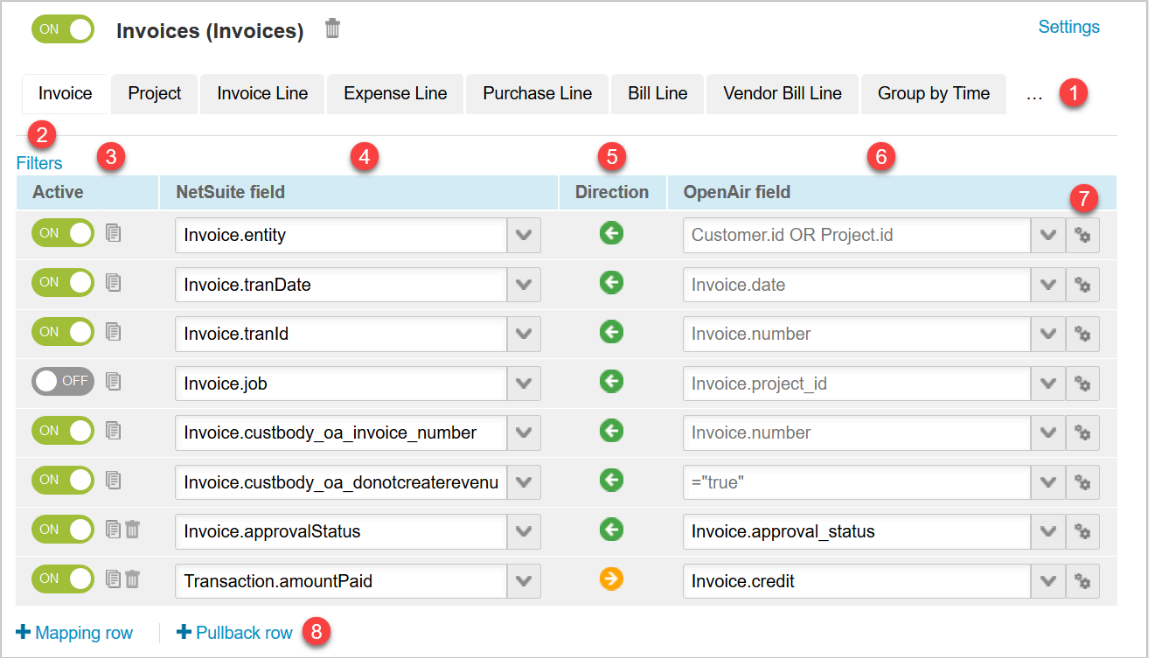
Refer to the Data Dictionary for information about standard OpenAir fields available for mapping — See Schema Reference Documentation.

OpenAir custom fields are available for mapping. Advanced field mapping may be required to ensure that custom field values are formatted correctly, depending on the custom field type. See OpenAir Custom Fields and Field Mapping.

> **Note:** The case convention for both table and field names in OpenAir uses lowercase and underscore characters. However, the table names are listed with an uppercase first letter in the field mapping definition tables.

7 **Advanced mapping** — To view or set your field mapping definitions, click the advanced mapping icon next to the dropdown field for the source data ⚙. See Using the Advanced Tag Notation.

	Note: The source data is the data in the **OpenAir field** column if you are viewing the field mapping definitions for an export workflow (from OpenAir to NetSuite), or in the **NetSuite field** column if you are viewing the field mapping definitions for an import workflow (from NetSuite to OpenAir).
	Initial only — The field mapping definitions for import workflows (workflows importing record data from NetSuite into OpenAir) also include a fifth column. Check the box in the **initial only** column if you want to import the field data only the first time a specific record is imported from NetSuite into OpenAir. The field mapping row is used when the integration creates a record in OpenAir but not when it updates a record already imported.
8	**Adding a field mapping row or pullback row** — After you add an integration workflow, the field mapping definition table shows the default configuration with all default field mapping rows set to active. You can add new mapping rows and create custom field mapping definitions to suit your business requirements. See Setting Field Mapping Definitions. - Click **+ Mapping row** to add a new row and define a custom field mapping. - Click **+ Pullback row** to add a pullback row. Pullback rows define field values to be pulled back when a record is created in NetSuite using the integration. You can add a pullback row only for relevant export workflows and for relevant mapping groups.



Active	NetSuite field	Direction	OpenAir field
ON	Invoice.entity	←	Customer.id OR Project.id
ON	Invoice.tranDate	←	Invoice.date
ON	Invoice.tranId	←	Invoice.number
OFF	Invoice.job	←	Invoice.project_id
ON	Invoice.custbody_oa_invoice_number	←	Invoice.number
ON	Invoice.custbody_oa_donotcreaterevenu	←	="true"
ON	Invoice.approvalStatus	←	Invoice.approval_status
ON	Transaction.amountPaid	→	Invoice.credit

Setting Field Mapping Definitions

After you add an integration workflow on the OpenAir NetSuite Connector Mapping tab, the field mapping definition table shows the default configuration with all default field mapping rows set to active. You can customize the field mapping definitions for each integration workflow to suit your business requirements. For example, you can deactivate default field mapping rows or add custom mapping rows and set your own custom field mapping definitions.

For an overview of what you can or cannot change on the field mapping definition table, see Field Mapping Restrictions.

You can also use the strict mapping validation features to ensure your field mapping definitions are free from error. See Validating Your Field Mapping Definitions.

Note: Consult NetSuite SOAP schema browser for reference when defining field mapping for custom export workflows. It includes information about any specific requirements for exporting OpenAir data to NetSuite.

For example, if you are exporting project task records from OpenAir to NetSuite and you want to set the project task start date in NetSuite to match the start date in OpenAir, you need to add the following mapping lines. NetSuite calculates the earliest possible start date based on existing predecessors unless you set `constraintType` to the value `_fixedStart`.

- NetSuite Field: `ProjectTask.startDate` — OpenAir Field: `Project_task.start_date`
- NetSuite Field: `ProjectTask.constraintType` — OpenAir Field: `="_fixedStart"`

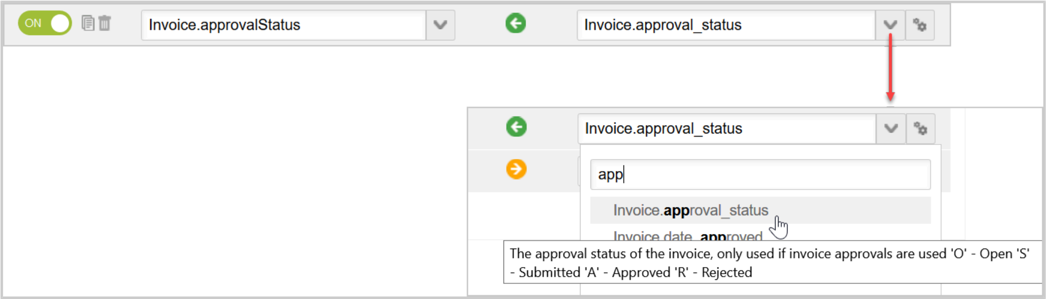
To set a field mapping definition:

1. On the subtab for the relevant integration workflow and mapping group, do one of the following:
   - To modify an existing field mapping definition, locate the relevant field mapping row.
   - To add a new row and set a custom field mapping definition for that row, click **+ Mapping row**.
   - To add a pullback row and set a custom field mapping definition for that row, click **+ Pullback row**.

   Note: Pullback rows define field values to be pulled back when a record is created in NetSuite using the integration. You can add a pullback row only for relevant export workflows and for relevant mapping groups.

2. On the relevant field mapping row, do one of the following:
   - Click the dropdown field in the column for the source data, then click to select the field you want to include. You can use the following guidelines to locate the correct field
     - Type a few characters to filter the dropdown options. The list shows only the field names including these characters.
     - Point to a field to view a tooltip describing the field. Tooltips are available only for OpenAir fields and take the description from the OpenAir Data Dictionary. For a description of NetSuite fields, refer to the NetSuite documentation.
     - If you are looking for a NetSuite custom field and this field does not appear in the dropdown list, click the Tips button then **Synchronize NetSuite custom fields** to update the NetSuite custom fields available for field mapping definition.
   - Click the advanced mapping icon ⚙ to view or enter the advanced tag notation for the source data.

> **Note:** The source data is the OpenAir field, if you are setting a field mapping definition for an export workflow (from OpenAir to NetSuite), or the NetSuite field, if you are setting a field mapping definition for an export workflow (from NetSuite to OpenAir).



The approval status of the invoice, only used if invoice approvals are used 'O' - Open 'S' - Submitted 'A' - Approved 'R' - Rejected

3. Click the dropdown field in the column for the target data, then click to select the field you want to include.

> **Note:** The target data is the NetSuite field, if you are setting a field mapping definition for an export workflow (from OpenAir to NetSuite), or the OpenAir field, if you are setting a field mapping definition for an export workflow (from NetSuite to OpenAir).
>
> You cannot change the target to a different value on a default mapping row. If you change it and save the OpenAir NetSuite Connector configuration, a new custom field mapping row is added with the modified field mapping definition, and the default field mapping row is restored to the default field mapping definition.

4. (Optional) For import workflow field mapping definitions, check the box in the **initial only** column if you want to import the field data only the first time a specific record is imported from NetSuite into OpenAir. The field mapping row is used when the integration creates a record in OpenAir but not when it updates a record already imported.
5. Click the switch under the **Active** column header to activate ON or to deactivate OFF the field mapping row.
6. Click **Save** to save the OpenAir NetSuite Connector configuration.

Field Mapping Restrictions

Different restrictions apply to different types of field mapping rows:

1	**Required field mappings** are set by default and cannot be disabled or modified. The entire row appears faded with text in light gray color on the mapping screen.
2	**Default field mappings** are set by default. They cannot be deleted but allow the following changes: ▪ Enable/Disable the field mapping by clicking the toggle in the **Active** column. ▪ Set the source to a different value or use advanced mapping. ▪ Check the **Initial only** option to use the field mapping to update the target field value in the initial integration run only. Default field mappings cannot be deleted and the OpenAir field displays in light gray text for these rows.

3 If you set the target to a different value and save the configuration. The default field mapping will be restored to its original definition and the repurposed default field mapping will be saved as a new custom field mapping.

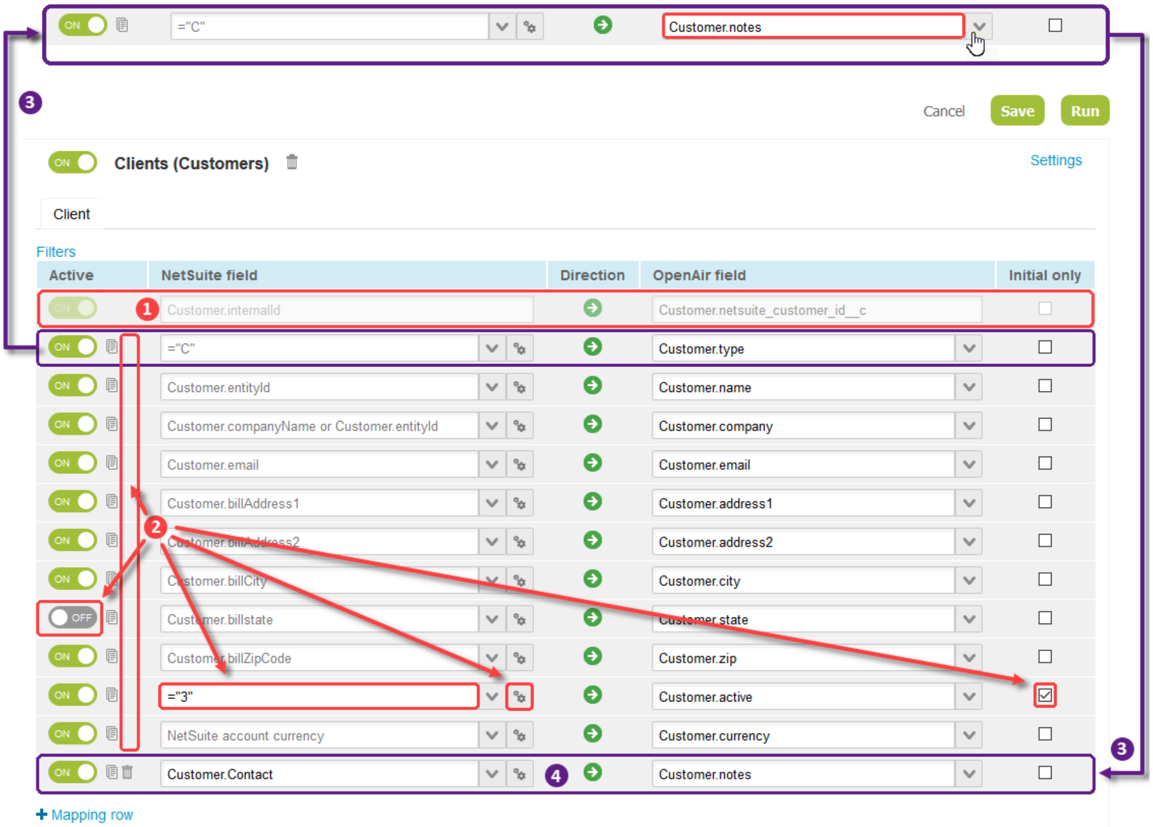
> **Important:** Refresh the screen to display the changes on the mapping screen.

4 You can add new mapping rows and define **custom field mappings**. The following field mappings are invalid and will be ignored:

- The target field is blank.
- The target field is a constant.
- Duplicate target fields. If two or more otherwise valid field mappings have the same target field, only the last one will be used. Any field mapping defined in of the rows above with duplicate target fields will be ignored.

> **Note:** Pullback field mappings are defined in the same way as custom field mappings.

> **Tip:** Use the strict mapping validation features to ensure your mappings are free from error before saving them. See Validating Your Field Mapping Definitions.



ON	="C"	→	Customer.notes	□

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Customer.internalId	→	Customer.netsuite_customer_id__c	□
ON	="C"	→	Customer.type	□
ON	Customer.entityId	→	Customer.name	□
ON	Customer.companyName or Customer.entityId	→	Customer.company	□
ON	Customer.email	→	Customer.email	□
ON	Customer.billAddress1	→	Customer.address1	□
ON	Customer.billAddress2	→	Customer.address2	□
ON	Customer.billCity	→	Customer.city	□
OFF	Customer.billstate	→	Customer.state	□
ON	Customer.billZipCode	→	Customer.zip	□
ON	="3"	→	Customer.active	☑
ON	NetSuite account currency	→	Customer.currency	□
ON	Customer.Contact	→	Customer.notes	□

Validating Your Field Mapping Definitions

You can use the strict mapping validation features to ensure your field mapping definitions are free from error.

In addition to the restrictions enforced by default on field mapping definitions, you can use the following methods to validate your field mapping definitions:

- Click **Validate mapping** in the Tips menu. See Validating the Mapping Manually.
- Enable Strict mapping validation on the NetSuite Connector administration form. See Enforcing Front-End Mapping Validation.

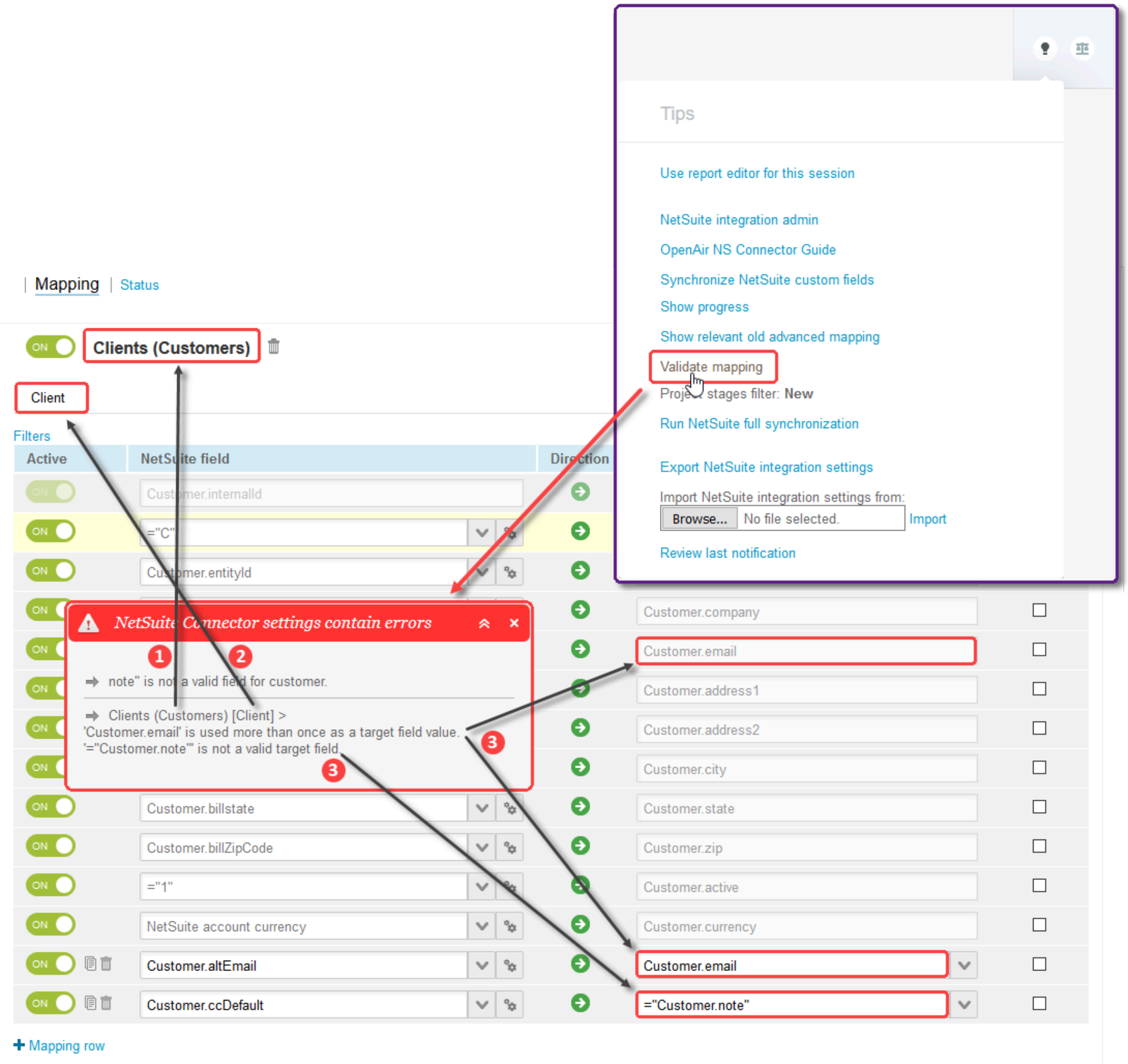
Validating the Mapping Manually

Click **Validate mapping** in the Tips menu to check all active mappings. This feature lets you:

- Verify that all enabled mapping rows for all enabled workflows meet strict validation criteria.
- Validate the mapping without having to save your NetSuite Connector configuration.
- Identify any errors with validation messages indicating:

1	The workflow — for example: Customers (Customers)
2	The mapping group tab in square brackets — for example: [Customers]
3	For every mapping row: - the target field — for example: `'Customer.email'` - an error message — for example: ”`is used more than once as a target field value`”

Note: Only the active field mapping rows for active workflows are included in the validation. The validation skips any workflow or mapping row with the **Active** switch off OFF.



Tips

Use report editor for this session

NetSuite integration admin

OpenAir NS Connector Guide

Synchronize NetSuite custom fields

Show progress

Show relevant old advanced mapping

Validate mapping

Project stages filter: New

Run NetSuite full synchronization

Export NetSuite integration settings

Import NetSuite integration settings from: Browse... No file selected. Import

Review last notification

NetSuite Connector settings contain errors

→ note" is not a valid field for customer.

→ Clients (Customers) [Client] >
'Customer.email' is used more than once as a target field value.
'="Customer.note"' is not a valid target field.

Active	NetSuite field	Direction		
ON	Customer.internalId	→		
ON	="C"	→		
ON	Customer.entityId	→		
ON		→	Customer.company	☐
ON		→	Customer.email	☐
ON		→	Customer.address1	☐
ON		→	Customer.address2	☐
ON		→	Customer.city	☐
ON	Customer.billstate	→	Customer.state	☐
ON	Customer.billZipCode	→	Customer.zip	☐
ON	="1"	→	Customer.active	☐
ON	NetSuite account currency	→	Customer.currency	☐
ON	Customer.altEmail	→	Customer.email	☐
ON	Customer.ccDefault	→	="Customer.note"	☐

Enforcing Front-End Mapping Validation

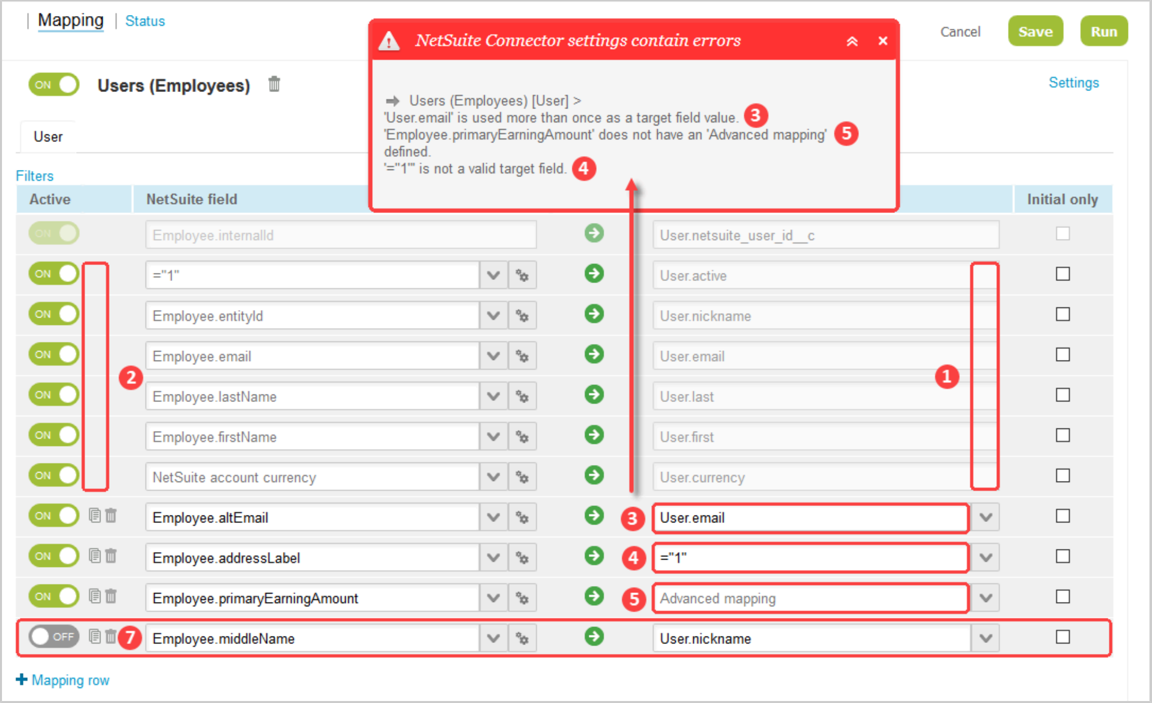
The Strict Mapping Validation feature validates the field mapping definition whenever you save the integration configuration on the OpenAir NetSuite Connector page.

To enable the feature, go to the NetSuite Connector administration and check the **Enable strict mapping validation** box.

The following table summarizes the validation and error prevention in place when the Strict Mapping Validation feature is enabled.

1	You cannot change the target field for default field mapping rows — The field is read-only text field instead of a dropdown field.
2	You cannot duplicate of default field mapping rows — The copy icon 🗐 is no longer available.

3	Each target field in the field mapping definition table must be unique — A validation error message appears if you attempt to save the configuration and the same target field is used in two or more rows of the field mapping definition table .
4	Target fields must not be set to a constant value — A validation error message appears if you attempt to save the configuration and a target field is set to a constant.
5	Target field must not be blank — The target field shows "Advanced mapping" by default and a validation error message appears if you attempt to save the configuration and the advanced mapping is not defined for that field mapping row.
6	Use valid quotation marks only — A validation error message appears if you attempt to save the configuration and an advanced mapping or filter definition contains quotation marks that are not valid. **Note:** Using curly quotation marks instead of straight quotation marks can cause the integration workflow to fail. Invalid quotation marks can be included accidentally if you copy advanced mapping or filter definitions from a word processor, for example, and paste them in OpenAir.
7	The validation ignores inactive field mapping rows (with the **Active** toggle switch off OFF).



NetSuite Connector settings contain errors

Users (Employees) [User] >
'User.email' is used more than once as a target field value.
'Employee.primaryEarningAmount' does not have an 'Advanced mapping' defined.
'="1"' is not a valid target field.

Active	NetSuite field		Initial only
ON	Employee.internalId	User.netsuite_user_id__c	□
ON	="1"	User.active	□
ON	Employee.entityId	User.nickname	□
ON	Employee.email	User.email	□
ON	Employee.lastName	User.last	□
ON	Employee.firstName	User.first	□
ON	NetSuite account currency	User.currency	□
ON	Employee.altEmail	User.email	□
ON	Employee.addressLabel	="1"	□
ON	Employee.primaryEarningAmount	Advanced mapping	□
OFF	Employee.middleName	User.nickname	□

Using Filters

You can define filters to integrate only the records that meet specific criteria. Click **Filters** to view or set filters for the mapping group using the filter builder. The **Filters** link appears in bold if filters are set.

Filter Builder

You can use a filter builder to define filter conditions for each import and export workflow. The filter builder validates your filter syntax, and simplifies it by removing the need for field declaration and opening and closing filter block statements.

The filter builder popup window appears when you click **Filters** on the Mapping tab.

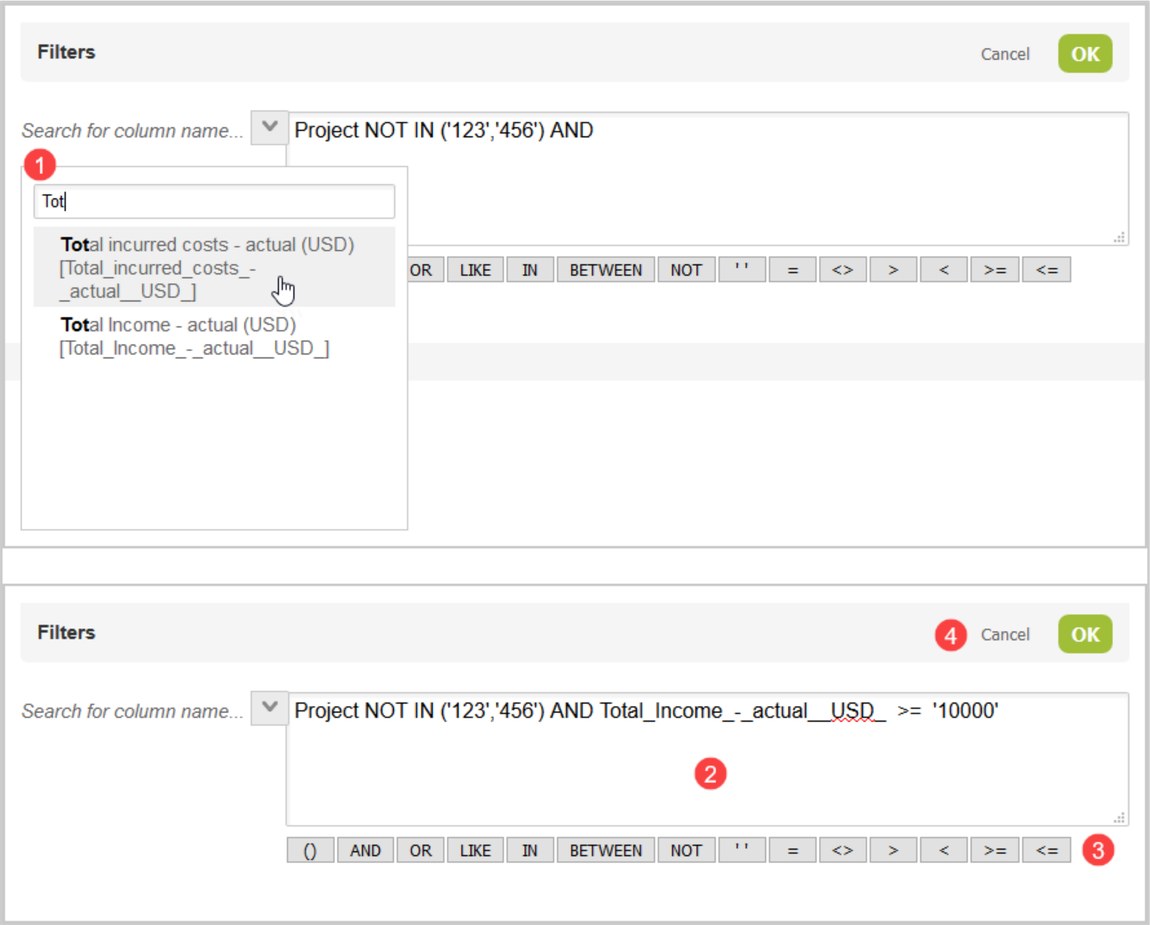
1. Type any characters to filter the list of fields available. Click the field you want to filter by.
2. You can view your filter conditions as you build them in the main panel. Click a field, a relational operator, logical operator, or other syntax element to insert it.
3. Use comparison and logical operators and other syntax elements to define and combine your filter conditions.
   - Available comparison operators: = (equal to), <> and != (not equal to), > (greater than), < (less than) , >= (greater than or equal to), <= (less than or equal to).

     > **Note:** The following operators cannot be used with string values: > (greater than), < (less than) , >= (greater than or equal to), <= (less than or equal to).

   - Available logical operators:
     - `AND` — All the conditions separated by `AND` must be `true` for the expression to be `true`.
     - `OR` — One of the conditions separated by `OR` must be `true` for the expression to be `true`.
     - `LIKE` — The first operand must match a pattern defined by the second operand. The pattern can include wildcards: an underscore (_) is a substitute for one character, a percentage symbol (%) is a substitute for any number of characters. For example, `User LIKE '_o%n'` matches all values with `'o'` in the second position and ending with `'n'`. The `LIKE` can only be used with string values. It cannot be used with numeric values or dates.
     - `IN` — The first operand is equal to one of a comma-separated list of values. The list is between an opening and closing parentheses. For example: `User IN ('John','Jon','Jonathan','Gordon')`.
     - `BETWEEN` — The first operand must be within a given range. The range is defined by a begin value and end value separated by `AND`. Begin and end values are included in the range. Values can be numbers, text or dates. For example, `Age BETWEEN '18' AND '65'` is equivalent to `Age >= '18' AND Age <= '65'`. The `LIKE` cannot be used with string values. It can only be used with numeric values or dates.
     - `NOT` — The conditions preceded by `NOT` must be `false` for the expression to be `true`. Can be used with operators `LIKE`, `IN`, and `BETWEEN` to exclude values matching a pattern, in a list, or within a given range. For example, `Age NOT BETWEEN '18' AND '65'` is equivalent to `Age < '18' AND Age > '65'`.

     > **Note:** The following operators are supported only when using direct filters: `LIKE`, `IN`, `BETWEEN`.
     >
     > In addition to extended operator support, direct filters preprocess filter conditions and speed up export workflows. For more information about direct filters, see Use direct filters.

   - Other syntax elements:
     - `()` — Parentheses must be used to define a list of values. Parentheses can also be used to modify operator precedence.
     - `''` — All values and patterns must be contained within a pair of single quotation marks.
4. Click **OK** to validate your filter conditions against any syntax error and save your filter conditions if valid.


Filters
Cancel
OK
Search for column name...
1
Tot
Total incurred costs - actual (USD)
[Total_incurred_costs_-_actual__USD_]
Total Income - actual (USD)
[Total_Income_-_actual__USD_]
OR
LIKE
IN
BETWEEN
NOT
' '
=
<>
>
<
>=
<=
Filters
4
Cancel
OK
Search for column name...
2
()
AND
OR
LIKE
IN
BETWEEN
NOT
' '
=
<>
>
<
>=
<=
3

Project NOT IN ('123','456') AND

Project NOT IN ('123','456') AND Total_Income_-_actual__USD_ >= '10000'

Using the Advanced Tag Notation

You can use the OpenAir NetSuite Connector advanced tag notation for your field mapping definitions.

To view or enter a field mapping definition using the advanced tag notation, click the Advanced icon ⚙ next to the source dropdown field on the relevant field mapping row. For advanced tag notation reference, see Advanced Mapping and Filter Notation Reference.

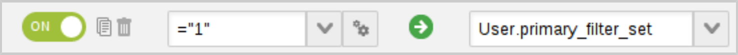
Note: Integration point tags are not required, as the advanced mapping is set in the context of the integration workflow and mapping group.

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

Examples of Advanced Mappings

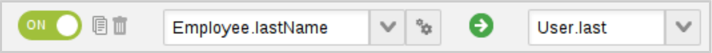
Value to value (fixed)



="1"

User.primary_filter_set

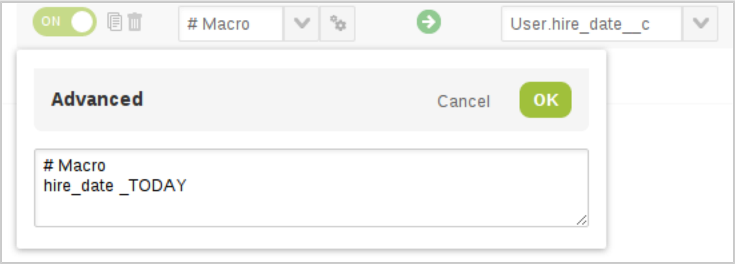
Value to value (from schema)



Employee.lastName

User.last

Value to value (macros)

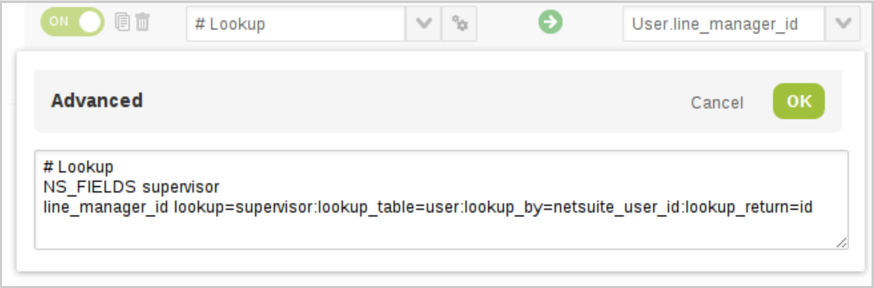


Macro

User.hire_date__c

```
# Macro
hire_date _TODAY
```

Value to value (lookup)

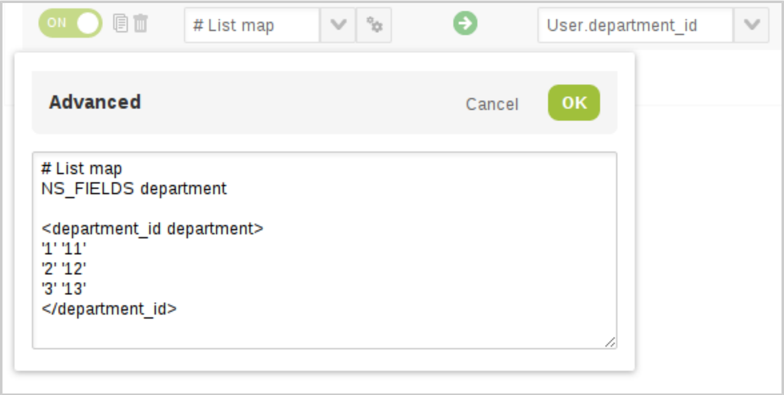


Lookup

User.line_manager_id

```
# Lookup
NS_FIELDS supervisor
line_manager_id lookup=supervisor:lookup_table=user:lookup_by=netsuite_user_id:lookup_return=id
```

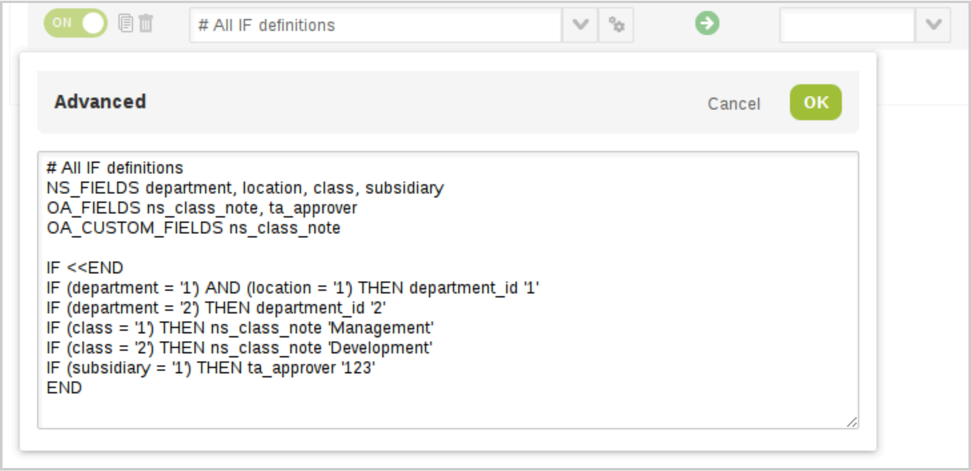
Value to value (list map)



```
# List map
NS_FIELDS department

<department_id department>
'1' '11'
'2' '12'
'3' '13'
</department_id>
```

Values to values (conditional branching / IF ... THEN)

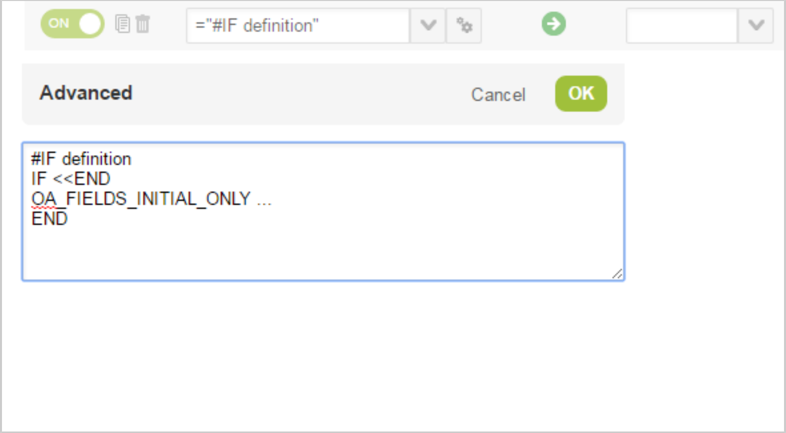
Important: There cannot be more than one conditional branching block on the same mapping group tab. Include all conditional branching (`IF ... THEN ...`) mapping definitions on the same mapping definition row. Leave the destination field empty on that row.



```
# All IF definitions
NS_FIELDS department, location, class, subsidiary
OA_FIELDS ns_class_note, ta_approver
OA_CUSTOM_FIELDS ns_class_note

IF <<END
IF (department = '1') AND (location = '1') THEN department_id '1'
IF (department = '2') THEN department_id '2'
IF (class = '1') THEN ns_class_note 'Management'
IF (class = '2') THEN ns_class_note 'Development'
IF (subsidiary = '1') THEN ta_approver '123'
END
```

Mapping an initial-only checkbox in a row

Note: To define an “Initial Only” checkbox in a field mapping row, you must define it using either an OA_FIELDS_INITIAL_ONLY or NS_FIELDS_INITIAL_ONLY tag in the advanced mapping window, as shown below:



```
#IF definition
IF <<END
OA_FIELDS_INITIAL_ONLY ...
END
```

OpenAir Custom Fields and Field Mapping

OpenAir custom fields are available for mapping. Advanced field mapping may be required to ensure that custom field values are formatted correctly, depending on the custom field type. You should always verify the format of custom field values before you attempt to include a custom field in your field mappings.

Review the following examples:

- **URL custom fields** – URL custom field values in NetSuite include the URL only. In OpenAir, URL custom field values include both the visible link text and the link URL. If the NetSuite URL field value is `https://www.example.com/example`, the corresponding URL field value in OpenAir should be `[Example of visible link text](https://www.example.com/example)`. To import URL custom field from NetSuite into OpenAir, use the following advanced field mapping:

```
target_field '[Example of visible link text](',source_field,')'
```

- **Project pick list custom fields** – List/record custom field values in NetSuite include the internal ID of the selected object only. In OpenAir, Pick list custom field values sourcing list items from projects include the internal IDs of the customer [`customer_id`] and the project [`project_id`] separated by a colon [`:`]. To import the list/record custom field holding the internal ID of a project from NetSuite to OpenAir, use an advanced field mapping similar to the following example:

```
target_field lookup=source_field:lookup_table=project:lookup_by=netsuite_project_id:lookup_return=cus
tomer_id,':',lookup=source_field:lookup_table=project:lookup_by=netsuite_project_id:lookup_return=id
```

Note: You can use the inspect functionality in your browser to verify the format of pick list custom field values. See Verifying the Format of OpenAir Pick List Custom Field Values

Verifying the Format of OpenAir Pick List Custom Field Values

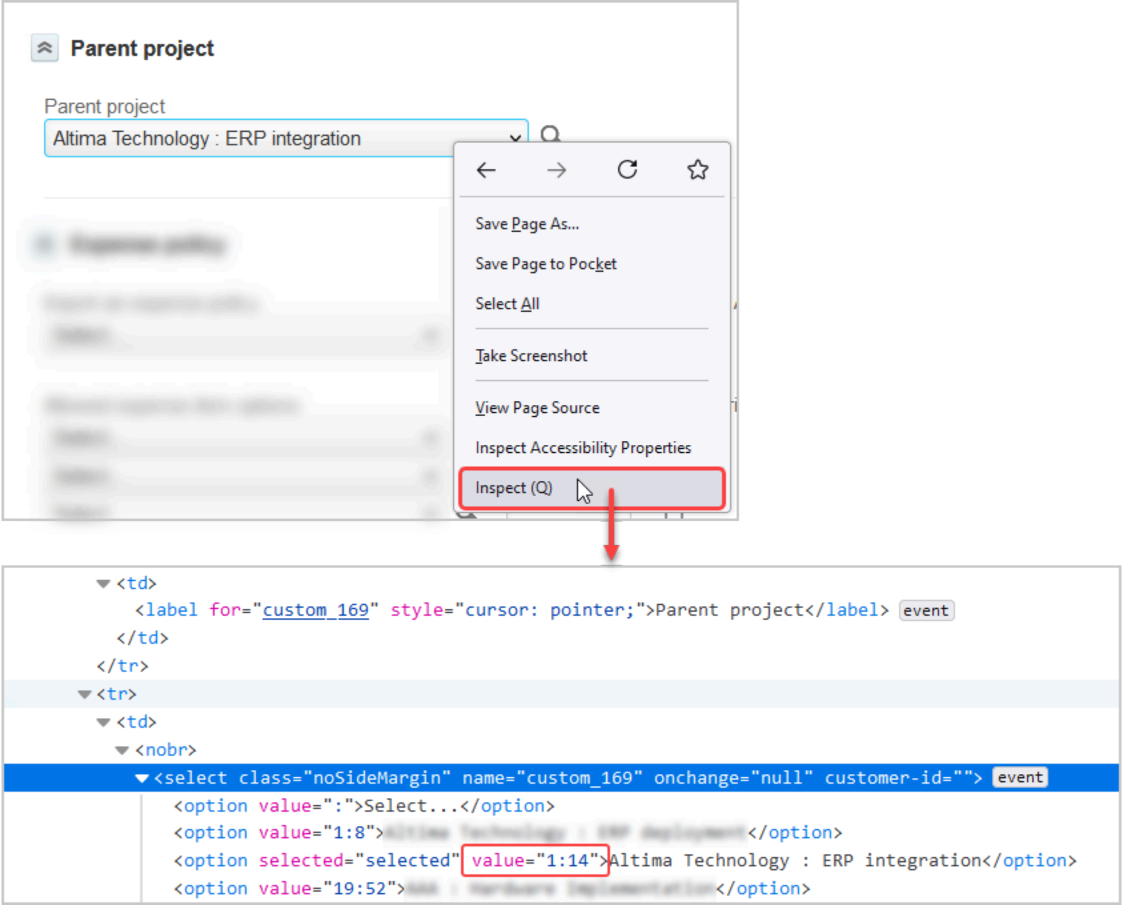
You can use the inspect functionality in your browser to verify the format of pick list custom field values.

To verify the format of OpenAir custom field values

1. In OpenAir, go to the record properties form that includes the custom field.
2. Right-click the pick list custom field value.

   A context menu appears.
3. Click **Inspect**.

   The Developer Tools frame or window appears and shows the page HTML. Each `option` element has a `value` attribute showing how the value is formatted.



```
    ▼<td>
       <label for="custom_169" style="cursor: pointer;">Parent project</label> event
     </td>
   </tr>
  ▼<tr>
    ▼<td>
      ▼<nobr>
        ▼<select class="noSideMargin" name="custom_169" onchange="null" customer-id=""> event
           <option value=":">Select...</option>
           <option value="1:8">[redacted]</option>
           <option selected="selected" value="1:14">Altima Technology : ERP integration</option>
           <option value="19:52">[redacted]</option>
```

Scheduling the Integration to Run Automatically

OpenAir NetSuite Connector lets you schedule the integration to run automatically for selected workflow groups on specific days and times.

To schedule integration runs, in OpenAir, go to Administration > NetSuite Connector > Automate.

The form shows a list of scheduled integration runs in a tabular form with information about the workflow group, the day and time for the integration run.

- You can add scheduled integration runs. See Adding a Scheduled Integration Run.
- You can create workflow groups to include only the integration workflows you need in each scheduled integration run. See Creating and Managing Workflow Groups.

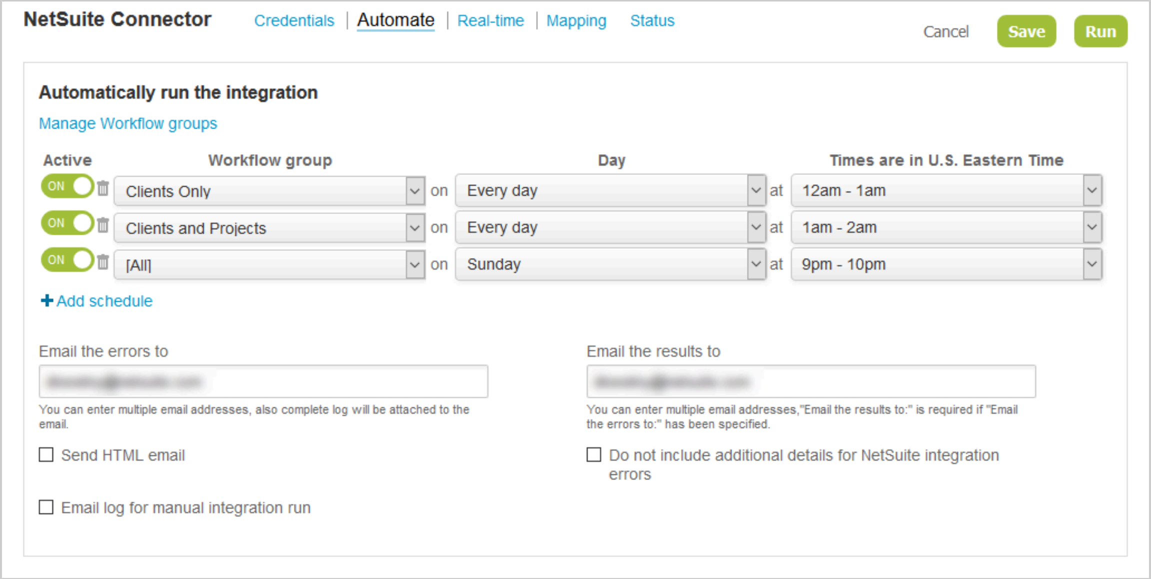
Tip: Define and use workflow groups to gain more control when setting up or maintaining your integration. If you do not select a workflow group, the scheduled integration run includes all integration workflows active at the time of the run. For more information about activating integration workflows, see Adding and Managing Integration Workflows.

- You can activate, deactivate or delete a scheduled integration run at any time:
  - To activate or deactivate a scheduled integration run, click the **Active** switch then click **Save**.
  - To delete a scheduled run, click the delete icon 🗑 then click **Save**.

If a workflow is part of a group selected for an active scheduled integration run, it is protected — you cannot modify the field mappings or the settings for the workflow, and you cannot deactivate the workflow. See Protected workflows.

You should define and use workflow groups in your integration schedules. However, a legacy compatibility administration setting lets you disable the Workflow Group feature. See Hide workflow groups.

You can also configure OpenAir NetSuite Connector to send result and error notification email for each scheduled and manual integration run.



Automatically run the integration

Manage Workflow groups

Active	Workflow group		Day		Times are in U.S. Eastern Time
ON 🗑	Clients Only	on	Every day	at	12am - 1am
ON 🗑	Clients and Projects	on	Every day	at	1am - 2am
ON 🗑	[All]	on	Sunday	at	9pm - 10pm

+Add schedule

Email the errors to

[redacted]

You can enter multiple email addresses, also complete log will be attached to the email.

☐ Send HTML email

☐ Email log for manual integration run

Email the results to

[redacted]

You can enter multiple email addresses,"Email the results to:" is required if "Email the errors to:" has been specified.

☐ Do not include additional details for NetSuite integration errors

Adding a Scheduled Integration Run

You can schedule the integration to run for a specific workflow group on a specific day and at a specific time. See also Creating and Managing Workflow Groups.

To add a scheduled integration run:

1. In OpenAir, go to Administration > NetSuite Connector > Automate.
2. Click **Add schedule**.
3. Select a **Workflow group**. This will restrict the scheduled integration run to the workflows included in that group. The options include:
   - Workflow groups defined for your account.

- The metavalue **[All]** — all the workflows enabled on the NetSuite Connector Mapping screen.

> **Tip:** **Workflow group** defaults to **[All]** when you add a new schedule. Define and use workflow groups instead of **[All]** in your integration schedules to gain more control when setting up or maintaining your integration.

4. Select the **Day** and **Time frame** for the run. Scheduling options include:
   - Days of the week, every day, specific days of the month including the 1st – 31st, last day of the month, 1st and 15th of the month, and 15th and last day of the month.
   - Hourly time ranges — OpenAir NetSuite Connector will determine the optimum time the run occurs within that hourly time range.

   > **Note:** Scheduled runs begin during the time frame specified but may end after that time frame.
   >
   > You can schedule up to 24 integration runs per day. However, scheduled runs cannot overlap. An error message will appear if there are any potential problems with scheduled runs.
   >
   > If you use scheduled integration runs for Project Tasks or Customers, you can improve the performance of these integration runs by enabling monitored fields. See Enabling Monitored Fields for Import Workflows.

5. Click **Save**.

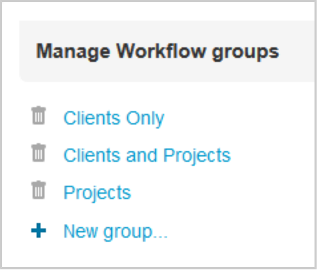
Creating and Managing Workflow Groups

You can create workflow groups to include only the integration workflows you need in each scheduled integration run. You can include any active workflows in your workflow groups.

After you create workflow groups, you can use them in your integration schedules instead of including all the workflows that are active at the time of the integration run. This lets you gain more control when setting up or maintaining your integration. See also Adding a Scheduled Integration Run.

To create or modify workflow groups:

1. In OpenAir, go to Administration > NetSuite Connector > Automate.
2. Click **Manage workflow groups**. The manage workflow groups dialog appears.



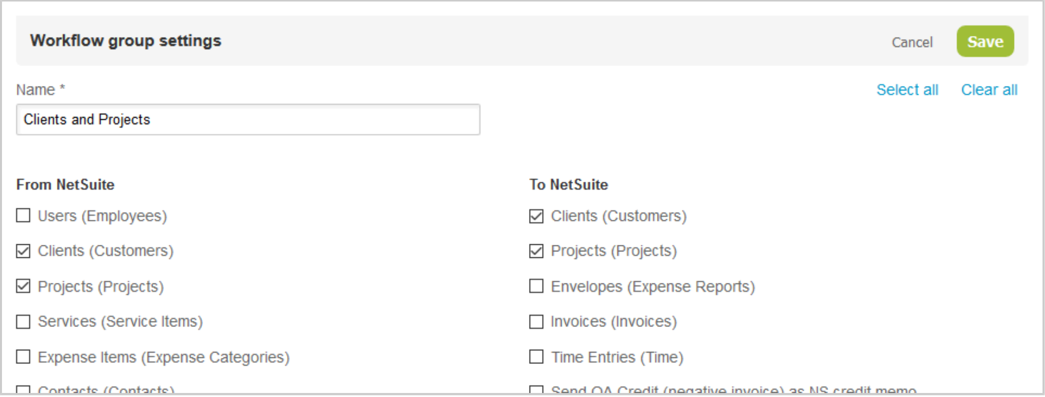
3. Do one of the following:

- Click **New group ...** to create a new workflow group.
- Click a workflow group name to view or modify it.

The workflow group settings dialog appears.

4. Enter a **Name** for the workflow group, if creating a new group.
5. Check the boxes for the workflows you want to include. You can only select workflows that are currently active. For more information about activating integration workflows, see Adding and Managing Integration Workflows.

> ✓ **Tip:** Click **Select all** to check all boxes and clear the box for each workflow you want to exclude, or click **Clear all** to clear all boxes and check the box for each workflow you want to include.

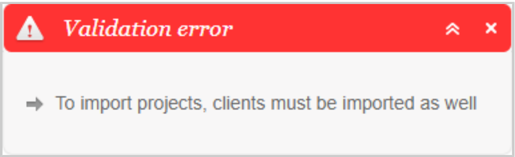


Name *: Clients and Projects

From NetSuite	To NetSuite
□ Users (Employees)	☑ Clients (Customers)
☑ Clients (Customers)	☑ Projects (Projects)
☑ Projects (Projects)	□ Envelopes (Expense Reports)
□ Services (Service Items)	□ Invoices (Invoices)
□ Expense Items (Expense Categories)	□ Time Entries (Time)
□ Contacts (Contacts)	□ Send OA Credit (negative invoice) as NS credit memo

6. Click **Save**.

   OpenAir NetSuite Connector validates any workflow dependencies before saving the workflow group settings. A validation error message appears if a workflow is required but not included in the workflow group. For example, if you create a workflow group to import projects from NetSuite, customers must also be imported.

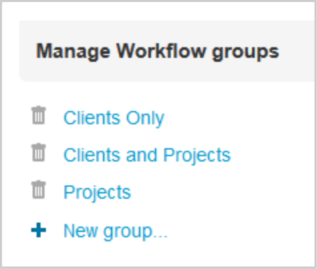


→ To import projects, clients must be imported as well

7. Click **Cancel** or outside the dialog to close it and return to the Automate tab.

To delete a workflow group:

1. In OpenAir, go to Administration > NetSuite Connector > Automate.
2. Click **Manage workflow groups**. The manage workflow groups dialog appears.



- Clients Only
- Clients and Projects
- Projects
- + New group...

3. To delete a workflow group, click the delete icon 🗑 next to the workflow group name.
4. Click **Cancel** or outside the dialog to close it and return to the Automate tab.

Configuring Result and Error Notification Email

You can also configure OpenAir NetSuite Connector to send notification email after scheduled and manual integration runs.

You can specify different recipients for result email, sent after each run, and error email, sent only if there were errors during the run.

To configure result and error notification email:

1. In OpenAir, go to Administration > NetSuite Connector > Automate.
2. **Email the results to** — Enter a comma–separated list of email addresses you want to send result notification email to after each integration run.
3. **Email the errors to** — Enter a comma–separated list of email addresses you want to send error notification email to after an integration run if there were errors during the run.

> **Note:** You cannot send error notification email without sending result notification email. Error notification email have the subject line “ERROR” and include the integration run log in attachment.

4. Check the **Send HTML email** box for enhanced email formatting (if your email client supports HTML emails). HTML notification email show errors in red color.
5. Check the **Email log for manual integration run** box to send the full integration run to result and error notification email recipients after each manual integration run. Notification email are sent only for scheduled integration run otherwise.
6. Click **Save**.

Enabling the OpenAir Within NetSuite Feature in OpenAir

After you setup the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature in NetSuite, you need to enable it in OpenAir and allow NetSuite single sign-on access for users before they can access OpenAir from within NetSuite.

For steps to set up the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) using NetSuite as OIDC Provider in NetSuite, see Enabling Access to OpenAir From Within NetSuite.

Warning: The steps in this help topic describe how to enable the upgraded version of the OpenAir Within NetSuite feature, which uses NetSuite as OIDC Provider as outbound single sign-on method instead of SuiteSignOn, in OpenAir.

Until the OpenAir 2023.2 release, OpenAir Within NetSuite relied on the NetSuite Outbound Single Sign-On (SuiteSignOn) feature as outbound single sign-on method. NetSuite plans to end support for SuiteSignOn with the NetSuite 2024.2 for production accounts. The functionality relying on SuiteSignOn may stop working at any time after its end-of-support date.

If you are using the previous feature version relying on NetSuite Outbound Single Sign-On (SuiteSignOn), upgrade before the NetSuite 2024.2 release to ensure service continuity. For upgrade instructions, see Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn.

You should test all integration configuration changes on sandbox NetSuite and OpenAir accounts before implementing them in your production accounts.

To enable the NetSuite Single Sign-On integration in OpenAir:

1. Enable the upgraded OpenAir Within NetSuite and Real-Time Record Import features using OAuth 2.0 in OpenAir.
   a. In OpenAir, go to Administration > NetSuite Connector > Credentials.
   b. Check the **Use OAuth 2.0 for NetSuite Single Sign-On Integration and Real-Time Record Import (Configuration Changes Required Before Enabling – See Documentation)** box under the Additional Integrations section.
   c. Click **Save**.

   Important: Switching to using OAuth 2.0 instead of the SuiteSignOn is not reversible. After you check this box and click **Save**, you cannot undo the change and go back to using SuiteSignOn.

   d. Click **Close** to return to the OpenAir NetSuite Connector Credentials tab.

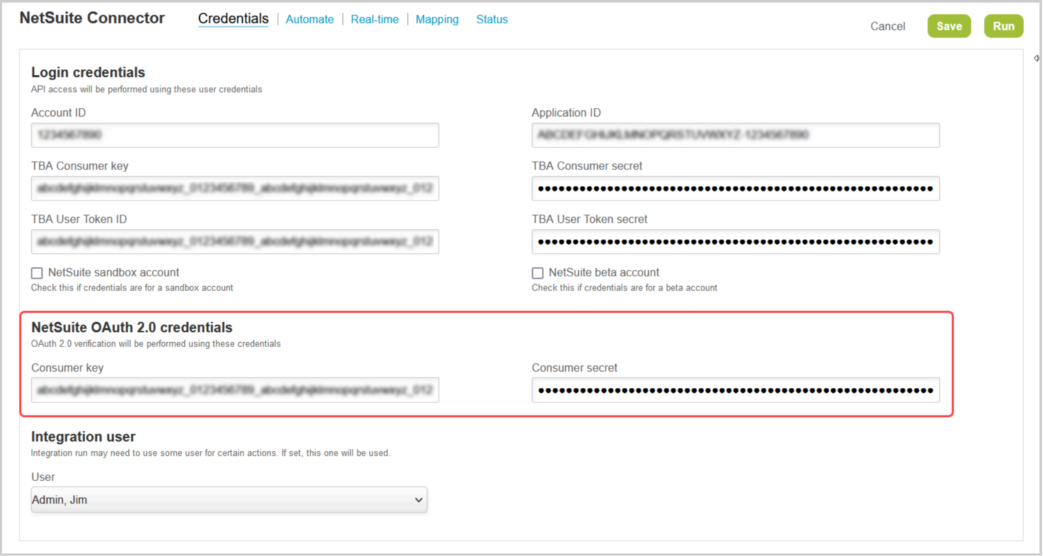
   Tip: If you are setting up the upgraded versions of both the OpenAir Within NetSuite and Real-Time Record Import features at the same time, do this step after completing the steps to enable the Real-Time Record Import feature. OpenAir NetSuite Connector then generates a certificate for real-time imports and saves it to NetSuite and OpenAir automatically.

2. In OpenAir, go to Administration > NetSuite Connector > Credentials.
3. Click the Tips button then **NetSuite Connector administration**.

   The OpenAir NetSuite Connector administration form appears.

4. Check the **Enable NetSuite Single Sign-On** box under the Other Features section of the form. See Enable NetSuite single sign-on.
5. Click **Save**.
6. Click **Close** to return to the OpenAir NetSuite Connector Credentials form.

7. Under **NetSuite OAuth 2.0 credentials**, enter the **Consumer Key** and the **Shared Secret** you copied when creating an integration record for the NetSuite as OIDC Provider integration. See Enabling Access to OpenAir From Within NetSuite.



Login credentials

API access will be performed using these user credentials

Account ID: [redacted]

Application ID: [redacted]

TBA Consumer key: [redacted]

TBA Consumer secret: ••

TBA User Token ID: [redacted]

TBA User Token secret: ••

□ NetSuite sandbox account
Check this if credentials are for a sandbox account

□ NetSuite beta account
Check this if credentials are for a beta account

NetSuite OAuth 2.0 credentials

OAuth 2.0 verification will be performed using these credentials

Consumer key: [redacted]

Consumer secret: ••

Integration user

Integration run may need to use some user for certain actions. If set, this one will be used.

User: Admin, Jim

8. Click **Save**.

9. Run the integration for the Employees (Employees) import workflow, to ensure that all relevant user records are integrated. The OpenAir employee record must have the correct NetSuite Employee ID for single sign-on. See Running the Bulk Import and Export Integration on Demand and Employees (Employees) Import Workflow.

10. For all relevant users, do the following:

    a. Go to Administration > Global Settings > Users > Employees > [*Select an employee*] > Demographic

    b. Check the **Allow NetSuite Single Sign-On** box.

    > **Note:** If you cannot locate the **Allow NetSuite Single Sign-On** box, check that the custom field is active and not hidden in the custom field properties and form permissions for the Employee demographic form.

    c. Click **Save**.

> ✓ **Tip:** Use the bulk user change wizard to enable NetSuite single sign-on access for all relevant users. See the help topic Making Changes to Multiple Employee Records at the Same Time.

Enabling the Real-Time Record Import Feature in OpenAir

After you setup the Real-Time Record Import feature in NetSuite, you need to enable it in OpenAir.

For an overview of the feature, see Real-Time Record Import.

> **Warning:** The steps in this help topic describe how to enable and set up the upgraded version of the Real-Time Record Import feature, which uses certificate-based authentication instead of SuiteSignOn to validate real-time record import requests.
>
> Until the OpenAir 2023.2 release, Real-Time Record Import relied on the NetSuite Outbound Single Sign-On (SuiteSignOn) feature.
>
> NetSuite plans to end support for SuiteSignOn with the NetSuite 2024.2 for production accounts. The functionality relying on SuiteSignOn may stop working at any time after its end-of-support date. Upgrade before the NetSuite 2024.2 release to ensure service continuity. For upgrade instructions, see Upgrading OpenAir NetSuite Connector Features to Remove the Dependency on SuiteSignOn.
>
> You should test all integration configuration changes on sandbox NetSuite and OpenAir accounts before implementing them in your production accounts.

To enable the Real-Time Record Import Feature in OpenAir:

1. In OpenAir, go to Administration > NetSuite Connector > Real-time.
2. Click the Tips button then **NetSuite Connector administration**.

   The OpenAir NetSuite Connector administration form appears.
3. Check the **Enable NetSuite real-time integration** box under the Other Features section of the form. See Enable NetSuite real-time integration.
4. Check the **Use OAuth 2.0 for NetSuite Single Sign-On Integration and Real-Time Record Import** box. See Use OAuth 2.0 for NetSuite Single Sign-On Integration and Real-Time Record Import.

   > **Note:** This step is also required if you are setting up the OpenAir Within NetSuite feature. If you are setting both OpenAir Within NetSuite and Real-Time Record Import features at the same time, do this step after completing all other steps to enable the Real-Time Record Import feature. OpenAir NetSuite Connector then generates a certificate for real-time imports and saves it to NetSuite and OpenAir automatically.
5. Click **Save**.
6. Verify that a valid certificate was generated and saved to NetSuite and OpenAir. In some cases – if you checked the **Use OAuth 2.0 for NetSuite Single Sign-On Integration and Real-Time Record Import** box before other steps required to set up the Real-Time Record Import in NetSuite, for example, you may need to generate the certificate in OpenAir. See Real-Time Record Import Certificate and Certificate Rotation.

Configuring Real-Time Record Export from OpenAir to NetSuite

You can configure OpenAir NetSuite Connector to export an OpenAir record automatically to NetSuite when the record is approved or saved by any user in OpenAir.

Real-time export from OpenAir to NetSuite runs in the background. When saving, approving or running an action on records, either individually or in bulk, users no longer have to wait until the export finishes or for a confirmation notification to continue working. An email is sent to the administrator in case of an integration error.

Note: This feature requires a NetSuite SuiteCloud Plus license and the **Multithreading** feature.

The Real-time integrations tab shows a list of OpenAir record types configured for real-time export in a tabular form with information about the event that triggers the export.

The following OpenAir records can be exported to NetSuite upon approval:

- Invoices
- Expense Reports
- Timesheets
- Purchase requests

The following OpenAir records can be exported to NetSuite upon saving:

- Customers
- Projects
- Tasks

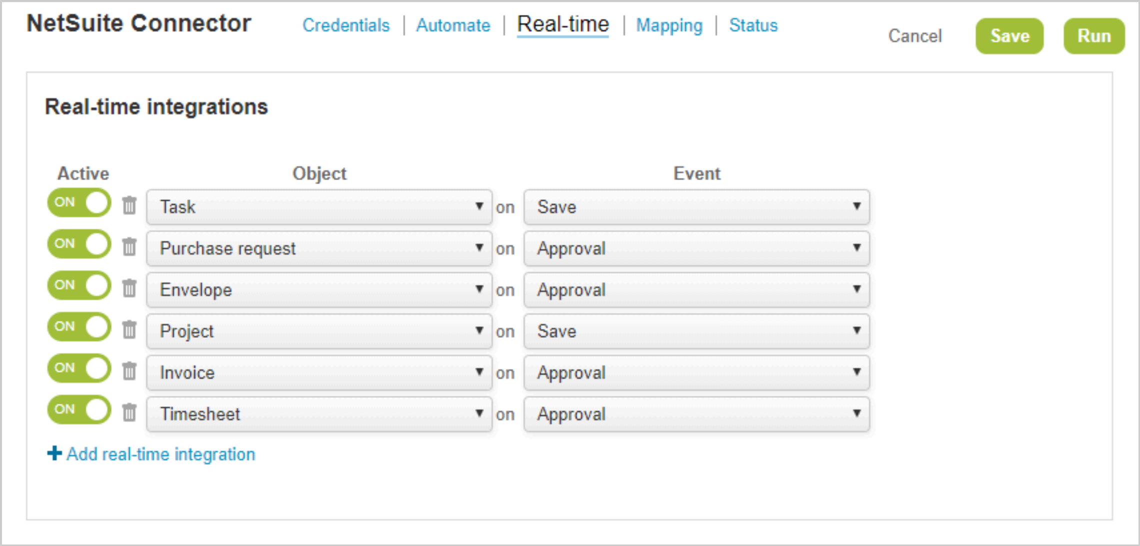
For all OpenAir record types except Project and Task records, the real-time export only creates a record in NetSuite when the record is approved or saved for the first time. For Project and Task records, the real-time integration can create and update the record in NetSuite.

If an OpenAir record type is selected for real-time integration, the corresponding workflow is protected — you cannot modify the field mappings or the settings for the workflow, and you cannot deactivate the workflow. See Protected workflows.

To configure real-time record export from OpenAir to NetSuite:

1. In OpenAir, go to Administration > NetSuite Connector > Real-time.
2. Click the Tips button then **NetSuite Connector administration**.

   The OpenAir NetSuite Connector administration form appears.
3. Check the **Enable NetSuite real-time integration** box under the Other Features section of the form. See Enable NetSuite real-time integration.
4. Click **Save**.
5. Click **Close** to return to the OpenAir NetSuite Connector Real-Time form.
6. Click **Add real-time integration**.
7. Select the OpenAir record type from the **Object** dropdown list.
8. Select the **Event** that triggers the real-time record export.
9. Click **Save**.


NetSuite Connector
Credentials | Automate | Real-time | Mapping | Status
Cancel
Save
Run

Real-time integrations

Active	Object		Event
ON	Task	on	Save
ON	Purchase request	on	Approval
ON	Envelope	on	Approval
ON	Project	on	Save
ON	Invoice	on	Approval
ON	Timesheet	on	Approval

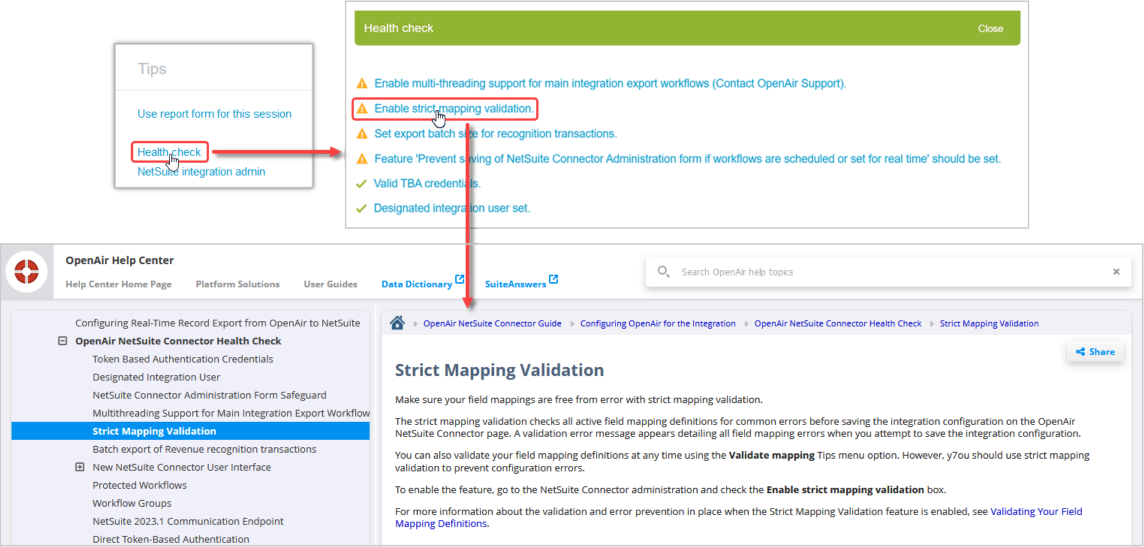
+ Add real-time integration

OpenAir NetSuite Connector Health Check

You can use the Health Check feature to review best practice setup for your OpenAir NetSuite Connector configuration. Follow the guideline actions to take full advantage of OpenAir NetSuite Connector features.

To display the health check guidelines, go to the OpenAir NetSuite Connector page and click **Health check** in the Tips menu.

- A cross mark icon ✕ indicates a required settings that is either missing or not valid.
- A warning icon ⚠ indicates setup actions you should take.
- A check mark icon ✓ indicates setup that is completed.
- Click the link for each guideline to view the relevant help topic in the OpenAir Help Center. Requires the View Help Center role permission.



⚠ Enable multi-threading support for main integration export workflows (Contact OpenAir Support).

⚠ Enable strict mapping validation.

⚠ Set export batch size for recognition transactions.

⚠ Feature 'Prevent saving of NetSuite Connector Administration form if workflows are scheduled or set for real time' should be set.

✓ Valid TBA credentials.

✓ Designated integration user set.

OpenAir Help Center
Help Center Home Page
Platform Solutions
User Guides
Data Dictionary
SuiteAnswers
Search OpenAir help topics
Configuring Real-Time Record Export from OpenAir to NetSuite
OpenAir NetSuite Connector Health Check
Token Based Authentication Credentials
Designated Integration User
NetSuite Connector Administration Form Safeguard
Multithreading Support for Main Integration Export Workflow
Strict Mapping Validation
Batch export of Revenue recognition transactions
New NetSuite Connector User Interface
Protected Workflows
Workflow Groups
NetSuite 2023.1 Communication Endpoint
Direct Token-Based Authentication
OpenAir NetSuite Connector Guide
Configuring OpenAir for the Integration
OpenAir NetSuite Connector Health Check
Strict Mapping Validation
Share

Strict Mapping Validation

Make sure your field mappings are free from error with strict mapping validation.

The strict mapping validation checks all active field mapping definitions for common errors before saving the integration configuration on the OpenAir NetSuite Connector page. A validation error message appears detailing all field mapping errors when you attempt to save the integration configuration.

You can also validate your field mapping definitions at any time using the **Validate mapping** Tips menu option. However, y7ou should use strict mapping validation to prevent configuration errors.

To enable the feature, go to the NetSuite Connector administration and check the **Enable strict mapping validation** box.

For more information about the validation and error prevention in place when the Strict Mapping Validation feature is enabled, see Validating Your Field Mapping Definitions.

The health check verifies the following configuration options:

- Token Based Authentication Credentials

- Real-Time Record Import Certificate and Certificate Rotation
- Designated Integration User
- NetSuite Connector Administration Form Safeguard
- Multithreading Support for Main Integration Export Workflows
- Strict Mapping Validation
- Batch export of Revenue recognition transactions
- New NetSuite Connector User Interface
- Protected Workflows
- Workflow Groups
- NetSuite 2023.1 Communication Endpoint
- Direct Token-Based Authentication
- Deprecated 3–way Salesforce, OpenAir and NetSuite Integration

Token Based Authentication Credentials

OpenAir NetSuite Connector uses token-based authentication (TBA) to access NetSuite. Valid **TBA User token ID** and **TBA User token secret** are required on the OpenAir NetSuite Connector Credentials form for the integration to work.

If OpenAir NetSuite Connector reports an authentication error or if the OpenAir NetSuite Connector Health Check reports that the TBA credentials are not valid, do the following:

1. Verify that an integration record was created for the OpenAir integration in NetSuite. To view integration records, sign in to NetSuite as an account administrator, then go to Setup > Integration > Manage Integrations. If it does not already exist, create an integration record. See Creating an Integration Record for the OpenAir Integration.
2. Verify that an access token was created for the main integration user and role. To view your access tokens, sign in to NetSuite as the main integration user, go to the Settings portlet, then click **Manage access tokens**. If it does not already exist, create an access token for the main integration user and role. See Creating and Assigning an Access Token for the Integration.
3. In OpenAir, go to Administration > NetSuite Connector > Credentials and verify the following information:
   a. **Account ID** — Verify the value matches the account ID for your NetSuite account.
   b. **Application ID**, **Consumer key** and **Consumer secret** — This information is displayed when you create an integration record in NetSuite. You may need to generate a new Consumer Key and Consumer Secret as this information is only displayed one time. See Creating an Integration Record for the OpenAir Integration.
   c. **User token ID** and **User token secret** — Copy the **User token ID** and the **User Token secret** from the Access Token page in NetSuite, and paste it on the OpenAir NetSuite Connector Credentials tab in OpenAir.

   See also Connecting OpenAir to NetSuite for the Integration.

Real-Time Record Import Certificate and Certificate Rotation

The upgraded version of the Real-Time Record Import feature uses certificate-based authentication instead of SuiteSignOn to validate real-time record import requests.

Certificates are valid for a finite period. After the initial set up, OpenAir NetSuite Connector generates a new certificate and saves it to NetSuite and OpenAir automatically one to two weeks before the previous certificate expires.

You can use the OpenAir NetSuite Connector Health Check feature to verify whether there is a valid real-time record import certificate in your account, and when the current certificate expires if valid.

- If the health check shows "No valid certificate available for real-time import", use the following steps to generate and save the certificate.
- Otherwise, the health check shows "Current real-time import certificate is valid until `<expiry date>`".

You can trigger the certificate generation and rotation from the contextual tips menu at any time.

To generate a new certificate for real-time record imports:

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button then click **Generate new real-time import certificate**. OpenAir NetSuite Connector generates a new certificate, uploads it in the `custcertificate_oa_realtime` certificate record in NetSuite, and saves it in OpenAir after it is successfully uploaded in NetSuite.

Note: This health check recommendation shows only if the Enable NetSuite single sign-on or Enable NetSuite real-time integration and the Use OAuth 2.0 for NetSuite Single Sign-On Integration and Real-Time Record Import boxes are checked on the OpenAir NetSuite Connector administration form.

However, certificates are generated and saved automatically in NetSuite and OpenAir only if the upgraded version of the Real-Time Record Import feature is set up in NetSuite. For more information about setting up the upgraded version of the Real-Time Record Import feature in NetSuite, see Configuring Real-Time Record Import from NetSuite into OpenAir.

Certificate Generation Troubleshooting

The NetSuite OpenAir SRP Real-time Add-on bundle (Bundle ID 369630) 3.0 or later version adds the `custcertificate_oa_realtime` certificate record. The default certificate `openair-certificate.pem` associated with the `custcertificate_oa_realtime` certificate record is not valid. It is a required placeholder and it is replaced by a new certificate as part of the initial setup and periodically thereafter as part of the automatic certificate rotation.

If attempting to generate the certificate using the above steps results in the error `Certificate "custcertificate_oa_realtime" not found`, the certificate record may have been accidentally deleted in NetSuite. In this case, recreate the certificate record.

To recreate the certificate record in NetSuite:

1. In NetSuite, go to Documents > Files > File Cabinet.
2. Locate the file cabinet directory for the OpenAir certificates (`SuiteBundles > Bundle 369630 > OpenAir Certificates`).
3. Next to `openair-certificate.pem`, click **Download** to save the certificate file to your computer.
4. Go to Setup > Company > Certificates.
5. Click **Create New**.

The new certificate window appears.

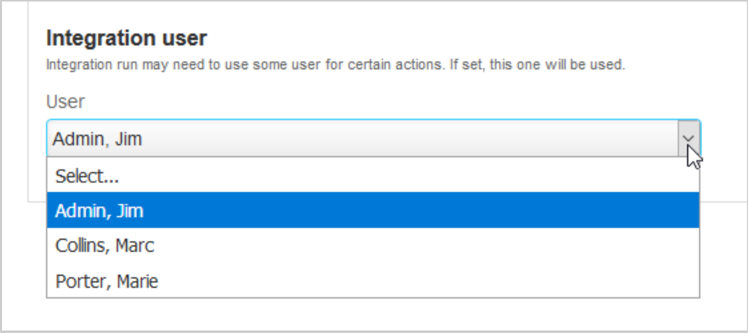
6. Enter the following certificate details
   - **Name**: `OpenAir real-time`
   - **ID**: `_oa_realtime`
7. On the Files tab, click **Choose a file** and select the `openair-certificate.pem` you saved to your computer earlier.
8. Click **Save**.

   After you save the certificate in NetSuite, try generating the certificate in OpenAir.

Designated Integration User

Select an active administrator as the designated integration user and owner of integration runs. The designated integration user must be an administrator and an active user.

- To set the integration user, go to Administration > Global Settings > Account > Integration: NetSuite > Credentials, select an Integration User from the dropdown options and click Save. Only administrators are listed in the dropdown options.
- To clear the designated integration user, choose the Select... option from the Integration user dropdown options.



Integration user

Integration run may need to use some user for certain actions. If set, this one will be used.

User

Admin, Jim

If you select an integration user:

- OpenAir NetSuite Connector will use the selected integration user’s permissions for all scheduled, manual or real-time integration runs.
- When the integration creates or updates a record in OpenAir, the audit trail for that record shows the selected integration user as the author of the changes.
- The audit trail also shows the record was created or updated using the NetSuite integration.
- The selected integration user’s permissions are protected — the user cannot be set as inactive and the user’s role cannot be changed.

If you do not select an integration user, OpenAir NetSuite Connector reverts back to the default behavior:

- The owner of manual integration runs is the user who triggers the integration.
- The owner of scheduled or real-time integration runs is the user who set the schedule or real-time integration, if that user record exists, or another administrator otherwise.
- The owner of the integration run shows as the author of the changes in OpenAir record audit trails.

NetSuite Connector Administration Form Safeguard

An optional feature disables the Save button on the NetSuite Connector administration form if there are any active scheduled or real-time integrations configured for your account. This forces authorized account administrators to deactivate all scheduled and real-time integrations before they can make any changes to the NetSuite Connector administration form.

This feature provides additional protection on top of other form validation in place to avoid common errors when editing the NetSuite Connector administration settings. You should use this safeguard to avoid any accidental integration issues.

To enable this feature, contact OpenAir Customer Support and request the following feature: Prevent Saving NetSuite Connector Administration Form if Workflows are Scheduled or Set for Real-Time Integration.

Multithreading Support for Main Integration Export Workflows

Increase throughput by enabling the main integration to use multiple threads concurrently.

OpenAir NetSuite Connector can use multiple threads to export records from OpenAir to NetSuite as part of main integration runs.

To enable multithreading support for the main integration, contact OpenAir Customer Support.

- Up to 10 threads can be allocated for main integration runs.
- Multithreading is supported only for export from OpenAir to NetSuite and not for import from NetSuite to OpenAir.

To enable multithreading support for the main integration, contact OpenAir Customer Support.

⚠ **Important:** To enable multithreading support, you must have an active NetSuite SuiteCloud Plus license. NetSuite SuiteCloud Plus allows up to 10 threads to be allocated concurrently for communication with NetSuite web services.

If you do not have an active NetSuite SuiteCloud Plus, the OpenAir NetSuite integration is limited to a single thread communication with NetSuite web services at all time.

Strict Mapping Validation

Make sure your field mappings are free from error with strict mapping validation.

The strict mapping validation checks all active field mapping definitions for common errors before saving the integration configuration on the OpenAir NetSuite Connector page. A validation error message appears detailing all field mapping errors when you attempt to save the integration configuration.

You can also validate your field mapping definitions at any time using the **Validate mapping** Tips menu option. However, y7ou should use strict mapping validation to prevent configuration errors.

To enable the feature, go to the NetSuite Connector administration and check the **Enable strict mapping validation** box.

For more information about the validation and error prevention in place when the Strict Mapping Validation feature is enabled, see Validating Your Field Mapping Definitions.

Batch export of Revenue recognition transactions

If the Recognition Transactions (Revenue Recognition Transactions) workflow is enabled, you can export revenue recognition in batches to run your revenue recognition exports up to two times faster.

This feature lets you integrate OpenAir revenue recognition transactions to NetSuite recognition transaction in batches instead of one by one. The maximum batch size is 100 records. The feature will significantly reduce the run time of the Recognition Transactions (Revenue Recognition Transactions) workflow exporting large numbers of journal entries to NetSuite.

To enable this feature, enter the **NetSuite recognition transaction batch size** batch size in the workflow settings for the Recognition Transactions (Revenue Recognition Transactions) workflow.

See also Exporting Revenue Recognition Transactions to NetSuite in Batches.

New NetSuite Connector User Interface

As announced in the October 8, 2022 OpenAir release notes, the New NetSuite Connector UI feature is required to use OpenAir NetSuite Connector. The NetSuite <> OpenAir integration does not work otherwise. See also NetSuite 2023.1 Communication Endpoint.

Warning: The NetSuite 2016.2 communication endpoint is available without support until the NetSuite 2023.2 release, at which point it will be retired. See Support for Existing WSDL Versions (external link to NetSuite SuiteAnswers | Answer ID: 10817).

The NetSuite 2016.2 communication endpoint can no longer be used for the NetSuite <> OpenAir integration, and the New NetSuite Connector UI is required to use OpenAir NetSuite Connector with the NetSuite 2020.1 communication endpoint.

The New NetSuite Connector UI was introduced in the October 17, 2015 OpenAir release as an optional feature. It changed the way you set up and manage the NetSuite <> OpenAir integration with greater control over settings and field mapping definitions for each workflow and field mapping group (or integration point). The New NetSuite Connector UI optional feature lets you map NetSuite fields to OpenAir fields using dropdown lists. It also lets you set advanced mappings for each field mapping row.

For more information about the New NetSuite Connector UI feature, see Navigating the OpenAir NetSuite Connector UI.

For more information about enabling the New NetSuite Connector UI feature for your account, see Switching to the New OpenAir NetSuite Connector UI.

Switching to the New OpenAir NetSuite Connector UI

As announced in the October 8, 2022 OpenAir release notes, the New NetSuite Connector UI feature is required to use OpenAir NetSuite Connector. The NetSuite <> OpenAir integration does not work otherwise. If the feature is not enabled for your account and you are already using OpenAir NetSuite Connector, use the following steps to enable it and continue working with OpenAir NetSuite Connector.

Warning: The NetSuite 2016.2 communication endpoint is available without support until the NetSuite 2023.2 release, at which point it will be retired. See Support for Existing WSDL Versions (external link to NetSuite SuiteAnswers | Answer ID: 10817).

The NetSuite 2016.2 communication endpoint can no longer be used for the NetSuite <> OpenAir integration, and the New NetSuite Connector UI is required to use OpenAir NetSuite Connector with the NetSuite 2020.1 communication endpoint.

To switch to the new OpenAir NetSuite Connector UI:

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button, then click **Export NetSuite integration settings**. A dialog appears.
3. Select **Save file** and click **OK** to download the NetSuite Connector settings export file to your computer. The name of export file is `netsuite_settings_<yyyy_mm_dd_hh_mm_ss>.json` where `<yyyy_mm_dd_hh_mm_ss>` includes information about the date and time of the export.
4. Contact OpenAir Customer Support and request the New NetSuite Connector UI feature.
5. After the New NetSuite Connector UI feature is enabled for your account, go to the OpenAir NetSuite Connector UI in OpenAir.
6. Click the Tips button, then click **Browse** under **Import NetSuite integration settings**.
7. Select the `netsuite_settings_<yyyy_mm_dd_hh_mm_ss>.json` you downloaded previously to your computer, then click **Import**.

   > **Note:** If you are importing NetSuite Connector settings from a different OpenAir account, you will need to configure custom imports and exchange rate integrations settings manually for the account you are importing the NetSuite Connector settings to. To do so, click the Tips button, then click **NetSuite Connector administration** and configure the settings under Custom imports and Exchange rate integration. For more information, see Custom Imports and Exchange rate integration.

8. Verify that your integration works as expected.

Protected Workflows

You should protect workflows to avoid any accidental integration issues that could be caused if changes are made to a workflow when it is in use.

By default, NetSuite Connector restricts changes to workflows that are selected for scheduled integration runs as part of a workflow group or that are set for real-time integration. See Protected workflows.

A legacy compatibility administration setting lets you allow changes to these workflows. However, it should not be used. See Do not protect workflows.

Workflow Groups

You should define and use workflow groups instead of **[All]** in your integration schedules. Workflow groups let you run exactly the workflows you want when you need them to run and give you more control when setting up or maintaining your integration. See Creating and Managing Workflow Groups.

A legacy compatibility administration setting lets you disable the Scheduled Integration by Workflow Groups feature. However, it should not be used. See Hide workflow groups.

NetSuite 2023.1 Communication Endpoint

Use the NetSuite 2023.1 communication endpoint for the NetSuite <> OpenAir integration and take advantage of its features.

OpenAir supports NetSuite 2020.1 and 2023.1 communication endpoints. NetSuite Connector uses the NetSuite 2020.1 endpoint by default unless the option to use the 2023.1 endpoint is enabled.

To use the NetSuite 2023.1 communication endpoint for the integration:

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button, then **NetSuite Connector administration**.
3. Check the **Use 2023.1 communication endpoint** box.
4. Click **Save**.
5. Click the Tips menu icon and click **Synchronize NetSuite custom fields**.

To revert back to using the NetSuite 2020.1 endpoint, follow the above procedure and clear the **Use 2023.1 communication endpoint** box.

> **Warning:** The NetSuite 2016.2 communication endpoint is available without support until the NetSuite 2023.2 release, at which point it will be retired. See Support for Existing WSDL Versions (external link to NetSuite SuiteAnswers | Answer ID: 10817).
>
> The NetSuite 2016.2 communication endpoint can no longer be used for the NetSuite <> OpenAir integration, and the New NetSuite Connector UI is required to use OpenAir NetSuite Connector with the NetSuite 2020.1 communication endpoint.

Direct Token-Based Authentication

Double the speed of your integration with an optimized link between OpenAir and NetSuite.

Direct token-based authentication (TBA) utilizes a more robust communication channel between OpenAir and NetSuite to deliver faster integration runs. It also simplifies the sign-in credentials administration.

For information about enabling Direct TBA, see Enable Direct TBA.

Deprecated 3–way Salesforce, OpenAir and NetSuite Integration

OpenAir has ended support for the 3-way Salesforce, OpenAir and NetSuite integration for customers and projects.

You should use the Projects (Projects) and Customers (Customers) export workflows instead. See Replacing the Deprecated 3-Way Salesforce, OpenAir and NetSuite Integration With the Customers (Customers) or Projects (Projects) Export Workflows.

The **Enable 3-way Salesforce.com, OpenAir and NetSuite integration** setting has been moved to the Legacy compatibility section of the NetSuite Connector administration form. You may continue using the 3-way Salesforce, OpenAir and NetSuite integration if it is already enabled for your account. However, it can no longer be enabled.

Replacing the Deprecated 3-Way Salesforce, OpenAir and NetSuite Integration With the Customers (Customers) or Projects (Projects) Export Workflows

OpenAir has ended support for the 3-way Salesforce, OpenAir and NetSuite integration for customers and projects. You should use the Projects (Projects) and Customers (Customers) export workflows instead.

If you are using the deprecated 3-Way Salesforce, OpenAir and NetSuite integration to export customer or project records from OpenAir to NetSuite, you should change your OpenAir NetSuite Connector configuration before the feature is removed from OpenAir.

To replace the 3-Way Salesforce, OpenAir and NetSuite integration with the Customers (Customers) or Projects (Projects) export workflows:

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button, then **NetSuite Connector administration**.
3. On the NetSuite Connector administration form, do the following:
   - Clear the **DEPRECATED Enable 3-way Salesforce.com, NS OpenAir and NetSuite integration** box under the Legacy compatibility section.
   - Check the applicable boxes under the Exports section:
     - **Customers (Customers)** if you are exporting customer records from OpenAir to NetSuite.
     - **Projects (Projects)** if you are exporting project records from OpenAir to NetSuite.
4. Click **Save**. A message appears confirming that your NetSuite Connector administration settings were changed successfully.
5. Click **Close** to return to the main NetSuite Connector page.
6. Click **Mapping**.
7. Click **Customers (Customers)** or **Projects (Projects)**. The field mapping for the Customers (Customers) or Projects (Projects) export workflow appears.

   Note: Repeat this and the following steps for each workflow, if applicable.

   Note the following changes. With the 3-Way Salesforce, OpenAir and NetSuite integration enabled, the mapping included a default field mapping row for the custom fields in NetSuite and OpenAir holding the Salesforce record ID. With the Customers (Customers) or Projects (Projects) export workflow enabled instead of the 3-Way Salesforce, OpenAir and NetSuite integration, the field mapping row needs to be mapped again or removed — it is no longer a default mapping row.

   - Customers (Customers) field mapping

NetSuite field	Direction	OpenAir field
`Customer.custentity_oa_sfa_id`	<	`Customer.customer_sf_id`

   - Projects (Projects) field mapping

NetSuite field	Direction	OpenAir field
`Job.custentity_oa_sfa_id`	<	`Project.project_sf_id`

8. Do one of the following:
   - Select the applicable **OpenAir field** for this field mapping row, if you want to retain this field mapping in your integration. Refer to the tables in the previous step for the custom field name.
   - Switch the **Active** toggle off for this field mapping row, or delete the field mapping row, otherwise.
9. Make sure the workflow is active (the switch next to the workflow name is on) and click **Save**.

   Note: Saving the workflow mapping adds the **Export Customer to NetSuite** or **Export Project to NetSuite** check box on the entity form in OpenAir. This box must be checked to export a new customer or project record from OpenAir to NetSuite. For more information, see Customers (Customers) and Projects (Projects).

Note: The 3-Way Salesforce, OpenAir and NetSuite integration requires the `Customer.customer_sf_id` or `Project.project_sf_id` custom field to be populated to export a new OpenAir customer and project record from OpenAir to NetSuite. You may use your own API integration application to populate this custom field, and continue using the deprecated 3-Way Salesforce, OpenAir and NetSuite integration to export customer and project record to NetSuite until the functionality is removed from OpenAir.

Reviewing NetSuite Connector Settings History

You can view the history of integration configuration changes using the OpenAir NetSuite Connector Settings History page.

To enable the NetSuite Connector Settings History feature, contact OpenAir Customer Support.

To view the NetSuite Connector Settings History page, go to Administration > NetSuite Connector, click the Tips menu, then **View history settings**.

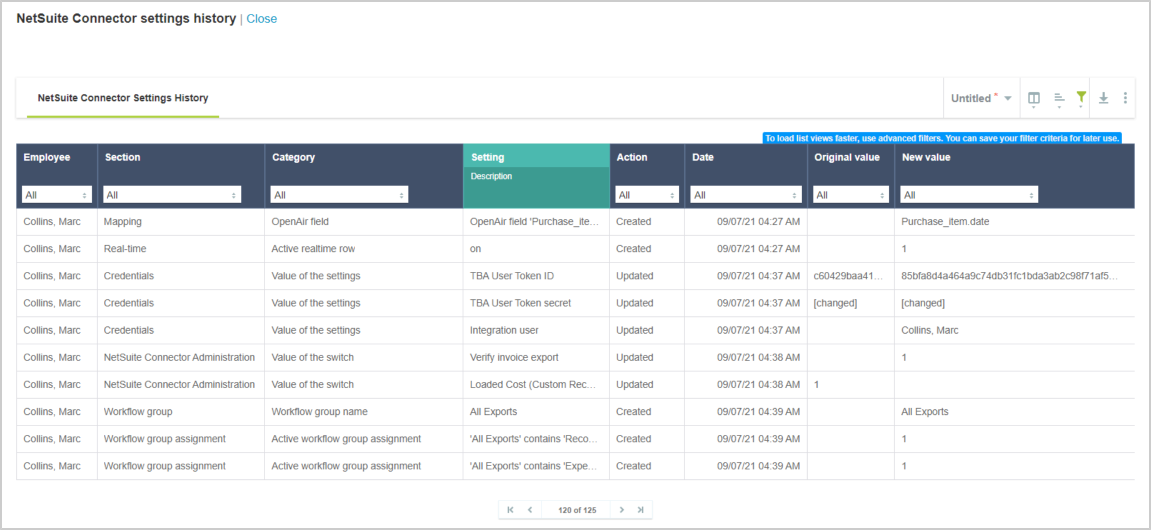
The history settings page opens in a new browser tab or window. It uses the redesigned user experience list view layout and features. The list shows audit trail information transformed for better readability.

Note: A simple change can create multiple rows in the settings history. For example, adding a workflow to the list of workflows also adds rows changes for all the mapping groups and mapping definition rows created when adding this workflow.

The audit trail information shown on the NetSuite Connector Settings History in this OpenAir release is not fully optimized for readability and will be refined in a future release.

The list view can include the following columns

- Name of the **Employee** who made the change.
- Name of the **Proxy Employee** — The name of the employee who made the change while signed in by proxy as the **Employee**.
- **Section** of the NetSuite Connector UI where the changes was made — The different sections correspond to the different tab on the New NetSuite Connector UI (Credentials, Automate, Real-time, Mapping), to a specific feature (Workflow group and Workflow group assignment), or to the NetSuite Connector administration and workflow settings forms(NetSuite Connector Administration). See Section and Setting Description for Each Type of Change.
- **Category** of the setting changed — Settings are grouped into categories that give additional information to help identify what was changed. See Category and Value for Each Type of Change.
- The values in the **Tab** column shows the integration workflow and mapping group impacted by changes made to field mapping definitions.
- Internal **ID** of the **Setting** changed.
- **Description** of the **Setting** changed — Additional information to help identify what was changed. See Section and Setting Description for Each Type of Change.
- **Action** performed — Values include: Created, Updated, Deleted.
- **Date** the change was made.
- **Original value** before the change. See Category and Value for Each Type of Change.
- **New value** after the change. See Category and Value for Each Type of Change.
- **Change made via** — How the change was made. For example, the change could be made using the OpenAir NetSuite Connector UI, by importing settings from a file.



NetSuite Connector settings history | Close

NetSuite Connector Settings History

To load list views faster, use advanced filters. You can save your filter criteria for later use.

Employee	Section	Category	Setting Description	Action	Date	Original value	New value
All	All	All		All	All	All	All
Collins, Marc	Mapping	OpenAir field	OpenAir field 'Purchase_ite…	Created	09/07/21 04:27 AM		Purchase_item.date
Collins, Marc	Real-time	Active realtime row	on	Created	09/07/21 04:27 AM		1
Collins, Marc	Credentials	Value of the settings	TBA User Token ID	Updated	09/07/21 04:37 AM	c60429baa41…	85bfa8d4a464a9c74db31fc1bda3ab2c98f71af5…
Collins, Marc	Credentials	Value of the settings	TBA User Token secret	Updated	09/07/21 04:37 AM	[changed]	[changed]
Collins, Marc	Credentials	Value of the settings	Integration user	Updated	09/07/21 04:37 AM		Collins, Marc
Collins, Marc	NetSuite Connector Administration	Value of the switch	Verify invoice export	Updated	09/07/21 04:38 AM		1
Collins, Marc	NetSuite Connector Administration	Value of the switch	Loaded Cost (Custom Rec…	Updated	09/07/21 04:38 AM	1	
Collins, Marc	Workflow group	Workflow group name	All Exports	Created	09/07/21 04:39 AM		All Exports
Collins, Marc	Workflow group assignment	Active workflow group assignment	'All Exports' contains 'Reco…	Created	09/07/21 04:39 AM		1
Collins, Marc	Workflow group assignment	Active workflow group assignment	'All Exports' contains 'Expe…	Created	09/07/21 04:39 AM		1

Section and Setting Description for Each Type of Change

The following table lists the sections of the NetSuite Connector UI as referred to in the settings history list view, the type of change they refer to, and the additional information given in the Setting Description column to help identify what was changed.

Section	Change Type	Information in the Setting Description
Automation	Changes to scheduled integration runs configured on the Automate tab.	The new day and time selected for the schedule row impacted by the change — for example, `on Wednesday at 1am - 2am`.
Credentials	Changes to the configuration settings on the Credentials tab or to the email notification settings on the Automate tab.	The label of the setting impacted by the change, exactly as it shows on the form.
Mapping	Changes to a mapping definition row on the Mapping tab.	The OpenAir field and NetSuite field selected for the mapping definition row impacted by the change, or the advanced mapping set for that row, if any — for example `OpenAir field '="10"', NetSuite field 'Customer.reminderDays', no advanced mapping`. **Note:** The **Tab** column shows the integration workflow and mapping group impacted by the change.
Real-time	Changes to real-time exports from OpenAir to NetSuite configured on the Real-time tab.	The new object and event selected for the real-time integration row impacted by the change — for example, `on Envelope approval`. **Note:** The object name is given in the default OpenAir terminology, not in the account terminology.
NetSuite Connector Administration	Changes to settings on the NetSuite Connector	The label of the setting impacted by the change, exactly as it shows on the form.

Section	Change Type	Information in the Setting Description
	administration form or in the Workflow settings form for specific integration workflows.	
Tab	Changes to the workflow list on the Mapping tab. Mapping group subtabs (integration points) are created or deleted automatically when adding or removing an integration workflow.	The display names of the integration workflow and mapping group impacted by the change — for example `Customer POs (Sales Orders) -> Date/Milestone Fixed Fee` where “Customer POs (Sales Order)” is the integration workflow and “Date/Milestone Fixed Fee” is the mapping group.
Workflow	Changes to the workflow list on the Mapping tab.	The display names of the integration workflow impacted by the change.
Workflow group	Changes to the list of workflow groups available on the Automate tab.	The names of the workflow group impacted by the change.
Workflow group assignments	Changes to the list of integration workflows included in a workflow group (Automate tab).	The display names of the workflow and workflow group impacted by the change: `'Workflow group name' contains 'Workflow display name'`.

Category and Value for Each Type of Change

The following table lists the change categories, a description of the change for each category, the location in the NetSuite Connector UI where the change can be made, and a reference for the values shown in the **Original value** and **New value** columns.

Category	Change	UI Location	Original Value / New Value
Value of the setting	Setting on the Credentials or Automate tab.	Field on the Credentials tab or email notification fields on the Automate tab.	For check boxes, 1 if checked, blank otherwise. For dropdown fields, as it appears in the selected dropdown option. For text boxes, as entered on the form or `[Changed]` if the text box content is masked.
Value of the switch	Administration setting value or workflow setting value.	Check box or dropdown field on the NetSuite Connector administration form (Tips menu option) or the Workflow settings popup form (Mapping tab).	For check boxes, 1 if checked, blank otherwise. For dropdown fields, as it appears in the selected dropdown option.
Active real-time row	Boolean value indicating whether the real-time integration is active.	**Active** switch on the real-time integration row (Real-time tab).	1 if active, blank otherwise.
Object for Real-time	The OpenAir record type to be exported to NetSuite in real time.	**Object** dropdown field on the real-time integration row (Real-time tab).	The name of the OpenAir record type in the default OpenAir terminology.
Event for Real-time	The event triggering the export of the OpenAir record to NetSuite.	**Event** dropdown field on the real-time integration row (Real-time tab).	As it appears in the selected dropdown option.

Category	Change	UI Location	Original Value / New Value
Workflow group name	The display name of the workflow group	**Name** text box on the Workflow group popup form (Automate tab).	The **Name** as entered on the Workflow Group popup form.
Active workflow group assignment	Boolean value indicating whether or not the integration workflow is included in the workflow group.	Check box for the integration workflow on the Workflow group popup form (Automate tab).	1 if active, blank otherwise.
Active workflow	Boolean value indicating whether or not the integration workflow is active.	**Active** switch at the top mapping definition table (Mapping tab).	1 if active, blank otherwise.
Workflow export/ import	String value indicating whether the integration workflow imports records from NetSuite into OpenAir or exports records from OpenAir to NetSuite.	Arrow on gray or green color background in the workflow list or in the **Direction** column on the mapping definition tables for the integration workflow (Mapping tab).	`from_ns` if the workflow imports records from NetSuite into OpenAir, `to_ns` if the workflow exports records from OpenAir to NetSuite.
Workflow display name	The display name of the integration workflow.	Workflow list and title at the top of the mapping definition pane (Mapping tab). For custom export workflows, the **Display name** text box on the Custom Workflow popup form, when creating a new custom export workflow (Mapping Tab).	For custom export workflows, the **Workflow display name** as entered when creating a new custom export workflow. Not shown otherwise.
Is custom workflow	Boolean value indicating whether or not the integration workflow is a custom export workflow.	If the integration workflow is a custom export workflow, selecting the workflow then clicking **Settings** on the Mapping tab shows a popup window with the custom workflow details. See Viewing Custom Export Workflows Settings.	1 if the integration workflow is a custom export workflows, blank otherwise.
Workflow name	The name of the integration workflow.	Workflow list and title at the top of the mapping definition pane (Mapping tab). For custom export workflows, the **Workflow name** text box on the Custom Workflow popup form, when creating a new custom export workflow (Mapping Tab).	For custom export workflows, the **Workflow name** as entered when creating a new custom export workflow. For standard workflows, the workflow name is not shown or use anywhere else on the UI.
Workflow selected	Boolean value indicating whether or not the integration workflow is	Workflow list (Mapping tab).	1 if included on the workflow list, blank otherwise.

Category	Change	UI Location	Original Value / New Value
	included on the workflow list on the Mapping tab.		
Workflow sequence	Updated by OpenAir when a custom workflow is added, activated, deactivated or deleted.	N/A	A integer number indicating the place of the custom workflow in the running order. **Note:** Custom export workflows run after all standard workflows.
Tab export/import	String value indicating whether the mapping group subtab is part of an integration workflow importing records from NetSuite into OpenAir or exporting records from OpenAir to NetSuite.	Arrow on gray or green color background in the workflow list or in the **Direction** column on the mapping definition tables for the integration workflow (Mapping tab).	Always blank
Tab display name	The display name of the mapping group subtab (or integration point).	Mapping group subtab (Mapping tab).	For custom export workflows, the **Workflow display name** as entered when creating a new custom export workflow. Not shown otherwise.
Tab filters	Filter conditions for the mapping group subtab (or integration point), if any and if available.	Pop up form that appears when clicking the Filters link at the top of the mapping definition table for the integration workflow mapping group, if available (Mapping tab).	The field mapping as defined using advanced filter notation.
Is custom tab	Boolean value indicating whether or not the mapping group subtab is part of a custom workflow.	N/A	1 if the mapping group subtab is part of a custom export workflows, blank otherwise.
Tab name	The name of the mapping group subtab (or integration point).	Mapping group subtab (Mapping tab).	As listed in the documentation (except it is displayed in lower case). See Integration Point Tags. For custom export workflows, the **Workflow name** as entered when creating a new custom export workflow.
Integration method	The integration method used for the mapping group subtab (or integration point). Determines whether the custom workflow exporting OpenAir published reports to a NetSuite custom record type only creates records, or replaces all records in NetSuite.	**Integration method** dropdown field on the Custom Workflow popup form, when creating a new custom export workflow (Mapping Tab).	As it appears in the selected dropdown option.

Category	Change	UI Location	Original Value / New Value
NetSuite record	For custom export workflows, the name of the NetSuite record type or custom record type.	**NetSuite record** dropdown field (or **Specify** dropdown field in the case of custom record types) on the Custom Workflow popup form, when creating a new custom export workflow (Mapping Tab).	As it appears in the selected dropdown option.
OpenAir record	For custom export workflows, the name of the OpenAir record type or OpenAir published report.	**OpenAir record** dropdown field (or **Specify** dropdown field in the case of published reports) on the Custom Workflow popup form, when creating a new custom export workflow (Mapping Tab).	As it appears in the selected dropdown option.
Workflow	The internal ID of the integration workflow the mapping group (or integration point) is part of.	N/A	The internal ID of the integration workflow.
Tab sequence	N/A	N/A	Always blank
Mapping row active	Boolean value indicating whether or not the mapping definition row is active.	**Active** column on the mapping definition table for the integration workflow (Mapping tab).	1 if active, blank otherwise.
Advanced mapping	Advanced mapping for the mapping definition row, if any.	Popup form that appears when clicking the advanced mapping icon ⚙ next to the dropdown field for the source data on the mapping definition tables for the integration workflow (Mapping tab).	The field mapping as defined using advanced mapping notation.
Initial only row	Boolean value indicating whether or not the mapping definition row should be used only the first time a record is imported from NetSuite into OpenAir (only when the OpenAir record is created).	**Initial only** check box on the mapping definition table for integration workflows importing records from NetSuite into OpenAir (Mapping tab).	1 if checked, blank otherwise.
Tab id	The internal ID of the mapping group (or integration point) the mapping definition row applies to.	Mapping group subtab (Mapping tab).	The internal ID of the mapping group subtab.
NetSuite field	The NetSuite field selected for the mapping definition row.	**NetSuite field** column on the mapping definition tables for the integration workflow (Mapping tab).	As it appears in the selected dropdown option.
OpenAir field	The OpenAir field selected for the mapping definition row.	**OpenAir field** column on the mapping definition	As it appears in the selected dropdown option.

Category	Change	UI Location	Original Value / New Value
		tables for the integration workflow (Mapping tab).	
Pullback	Boolean value indicating whether or not the mapping definition row is a pullback row.	**Add Pullback row** link and **Direction** column with an arrow on orange background ➔ on the mapping definition tables for integration workflows exporting records from OpenAir to NetSuite (Mapping tab).	1 if the mapping definition row is a pullback row, blank otherwise.
Scheduled workflow group	Workflow group the scheduled integration run is restricted to, if any.	**Workflow** dropdown field on the schedule row (Automate tab).	The selected workflow group display name, or blank if the metavalue [All] is selected.
Scheduled hour	Hour of the scheduled integration run. The	**Times are in U.S. Eastern Time** dropdown field on the schedule row (Automate tab).	As it appears in the selected dropdown option.
Scheduled day	Day of the scheduled integration run.	**Day** dropdown field on the schedule row (Automate tab).	As it appears in the selected dropdown option.
Scheduled active	Boolean value indicating whether or not the scheduled integration run is active.	**Active** switch on the schedule row (Automate tab).	1 if active, blank otherwise.

Reviewing the Integration Run History and Status

You can view the history of integration runs with all the details using the OpenAir NetSuite Connector Status page.

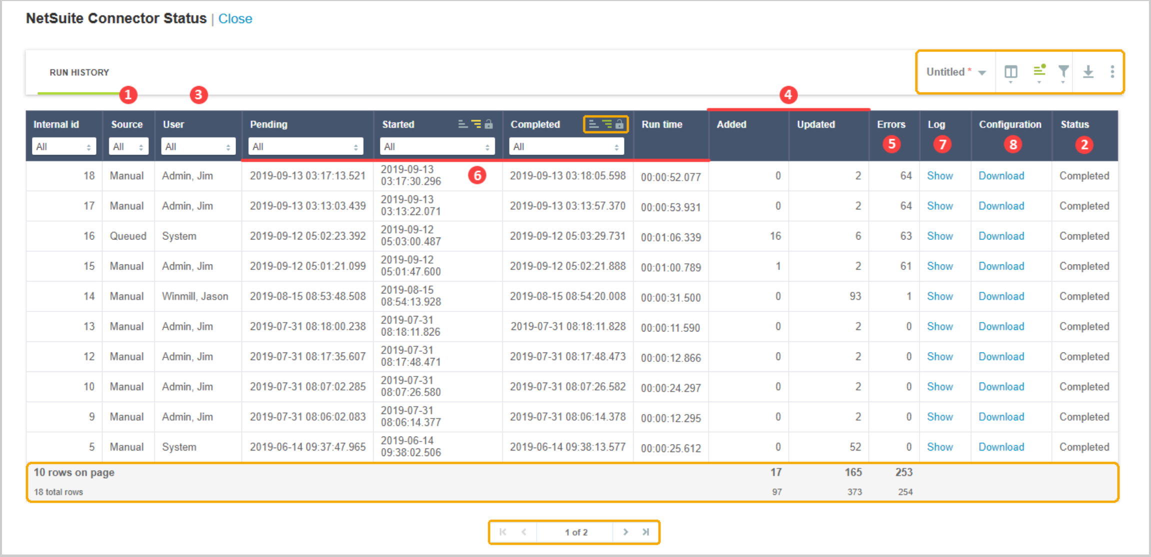
To view the NetSuite Connector Status page, go to Administration > NetSuite Connector > Status.

The Status page opens in a new browser tab or window and shows a list view of integration runs. It uses the redesigned user experience list view layout and features, including advanced filters and column filters for the following columns: Internal ID, Source, Workflow group, User, Pending, Started, Completed.

The list view includes the following information:

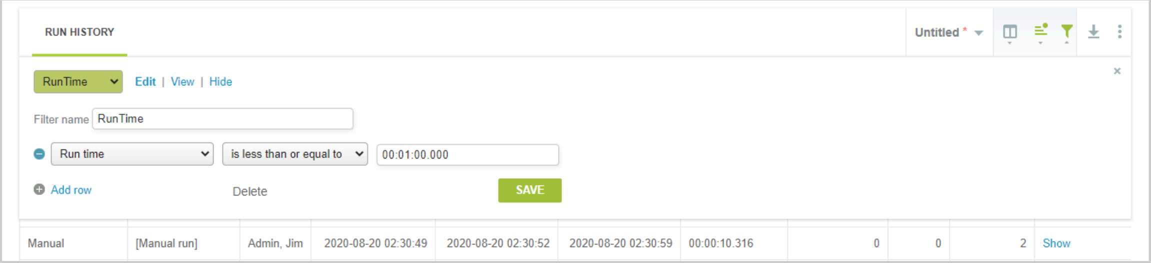
1	**Source** — How the integration run was started (Automated/Manual/Queued/Real-time import/Scripting). You can use a column filter to refine the listed integration runs by source.
	Workflow group — The selected workflow group for scheduled integration runs. You can use a column filter to refine the listed integration runs by workflow group.
2	**Status** — The status of the integration run (Pending/Running/Completed)
3	**User** — The name of the user who started the integration run or "System" if the integration run was started automatically. You can use a column filter to refine the listed integration runs by User.

4	**Added** and **Updated** — The number of records that were added and updated in the integration run.
5	**Errors** — The number of errors that occurred in the integration run.
6	**Pending**, **Started**, **Completed**, and **Run time** — Time stamps indicating when the integration run was queued, when it started and when it completed, as well as the duration of the integration run. You can use column filters to refine the listed integration runs by pending, started and completed date.
7	**Log** — Link to the log for that particular integration run. Click **Show** to display the log.
8	**Download** — Link to download the configuration details used by that particular integration run. Click **Download** to save the configuration for that integration run as a JSON file on your computer.



Internal id	Source	User	Pending	Started	Completed	Run time	Added	Updated	Errors	Log	Configuration	Status
All	All	All	All	All	All							
18	Manual	Admin, Jim	2019-09-13 03:17:13.521	2019-09-13 03:17:30.296	2019-09-13 03:18:05.598	00:00:52.077	0	2	64	Show	Download	Completed
17	Manual	Admin, Jim	2019-09-13 03:13:03.439	2019-09-13 03:13:22.071	2019-09-13 03:13:57.370	00:00:53.931	0	2	64	Show	Download	Completed
16	Queued	System	2019-09-12 05:02:23.392	2019-09-12 05:03:00.487	2019-09-12 05:03:29.731	00:01:06.339	16	6	63	Show	Download	Completed
15	Manual	Admin, Jim	2019-09-12 05:01:21.099	2019-09-12 05:01:47.600	2019-09-12 05:02:21.888	00:01:00.789	1	2	61	Show	Download	Completed
14	Manual	Winmill, Jason	2019-08-15 08:53:48.508	2019-08-15 08:54:13.928	2019-08-15 08:54:20.008	00:00:31.500	0	93	1	Show	Download	Completed
13	Manual	Admin, Jim	2019-07-31 08:18:00.238	2019-07-31 08:18:11.826	2019-07-31 08:18:11.828	00:00:11.590	0	2	0	Show	Download	Completed
12	Manual	Admin, Jim	2019-07-31 08:17:35.607	2019-07-31 08:17:48.471	2019-07-31 08:17:48.473	00:00:12.866	0	2	0	Show	Download	Completed
10	Manual	Admin, Jim	2019-07-31 08:07:02.285	2019-07-31 08:07:26.580	2019-07-31 08:07:26.582	00:00:24.297	0	2	0	Show	Download	Completed
9	Manual	Admin, Jim	2019-07-31 08:06:02.083	2019-07-31 08:06:14.377	2019-07-31 08:06:14.378	00:00:12.295	0	2	0	Show	Download	Completed
5	Manual	System	2019-06-14 09:37:47.965	2019-06-14 09:38:02.506	2019-06-14 09:38:13.577	00:00:25.612	0	52	0	Show	Download	Completed
10 rows on page							17	165	253			
18 total rows							97	373	254			

1 of 2



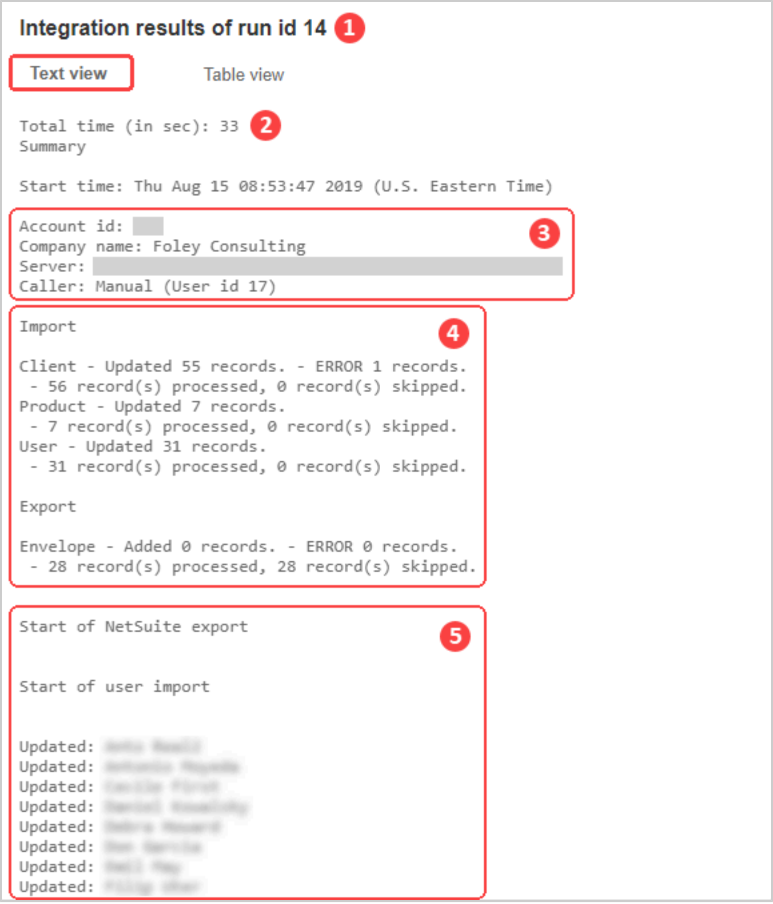
Filter name RunTime

Run time | is less than or equal to | 00:01:00.000

Manual	[Manual run]	Admin, Jim	2020-08-20 02:30:49	2020-08-20 02:30:52	2020-08-20 02:30:59	00:00:10.316	0	0	2	Show

The integration run log is displayed in a popup window and includes two tabs:

- Click the Text View tab to display summary information about the integration run, which includes:
  1. The unique integration run identifier
  2. The time taken to complete
  3. The integration run
  4. Account / connection details
  5. Summary of records integrated
  6. Integration messages



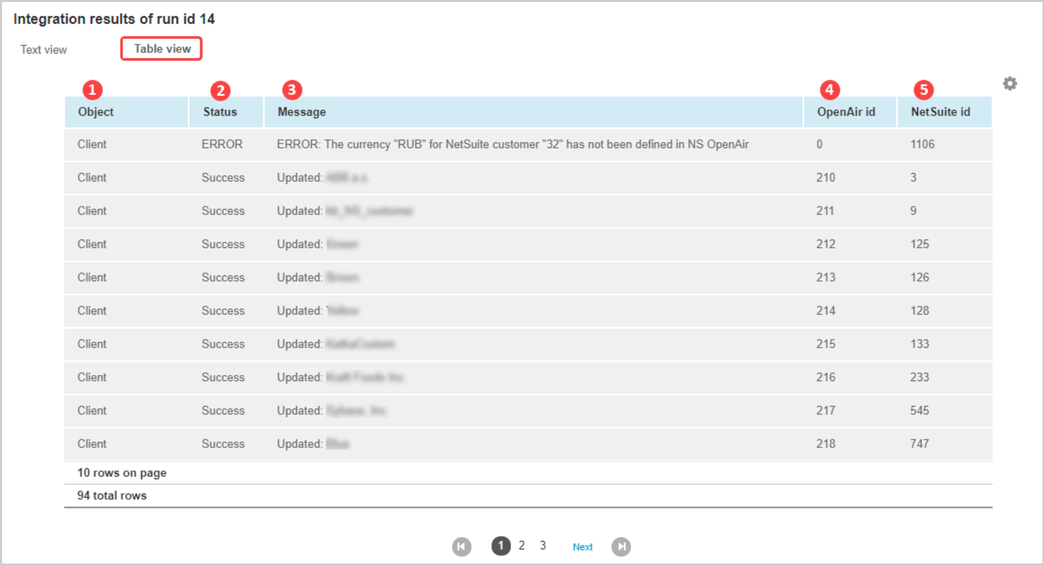
Integration results of run id 14

Text view   Table view

```
Total time (in sec): 33
Summary

Start time: Thu Aug 15 08:53:47 2019 (U.S. Eastern Time)

Account id: [redacted]
Company name: Foley Consulting
Server: [redacted]
Caller: Manual (User id 17)

Import

Client - Updated 55 records. - ERROR 1 records.
 - 56 record(s) processed, 0 record(s) skipped.
Product - Updated 7 records.
 - 7 record(s) processed, 0 record(s) skipped.
User - Updated 31 records.
 - 31 record(s) processed, 0 record(s) skipped.

Export

Envelope - Added 0 records. - ERROR 0 records.
 - 28 record(s) processed, 28 record(s) skipped.

Start of NetSuite export


Start of user import


Updated: [redacted]
Updated: [redacted]
Updated: [redacted]
Updated: [redacted]
Updated: [redacted]
Updated: [redacted]
Updated: [redacted]
Updated: [redacted]
```

- Click the **Table View** tab to display a list view with key details of records integrated in the run, which includes:

  1. **Object** — The type of record integrated
  2. **Status** — The outcome of the integration or the current status if the integration has not completed.
  3. **Message** — Information message
  4. **OpenAir id** — OpenAir unique identifier for the integrated record
  5. **NetSuite id** — NetSuite unique identifier for the integrated record



Integration results of run id 14

Object	Status	Message	OpenAir id	NetSuite id
Client	ERROR	ERROR: The currency "RUB" for NetSuite customer "32" has not been defined in NS OpenAir	0	1106
Client	Success	Updated: [redacted]	210	3
Client	Success	Updated: [redacted]	211	9
Client	Success	Updated: [redacted]	212	125
Client	Success	Updated: [redacted]	213	126
Client	Success	Updated: [redacted]	214	128
Client	Success	Updated: [redacted]	215	133
Client	Success	Updated: [redacted]	216	233
Client	Success	Updated: [redacted]	217	545
Client	Success	Updated: [redacted]	218	747

10 rows on page

94 total rows

1 2 3 Next

The usual list view features are supported for both the status screen and the logs for individual runs. For example, you can use the download feature to save the list view data as a CSV, HTML or PDF file.

Integration Role Permissions in OpenAir

Administrators can set up OpenAir integration roles with precise privileges. The integration role permissions are:

- View the NetSuite integration
- View and run the NetSuite integration
- View and edit the NetSuite integration settings
- Allow employee to export customers to NetSuite
- Allow employee to export invoices to NetSuite
- Allow employee to export Expense reports to NetSuite
- Allow employee to export project data to NetSuite
- Allow employee to export timesheets to NetSuite
- Allow employee to export recognition transactions to NetSuite
- Allow employee to export purchase requests to NetSuite
- Allow employee to view last NetSuite error on expense reports
- Allow employee to view last NetSuite error on invoices
- Allow employee to view last NetSuite error on timesheets

To create Integration User Roles in OpenAir

1. Go to Administration > Global Settings > Users > Roles.
2. Click the **Create** button and click **Role** under New.
3. Name the role, and select the role permissions this role will use. Integration role permissions are found under **Integration**.

4. Click **Save** or **Save & create another**.

Running the NetSuite <> OpenAir Integration

> **Note:** You should review the fundamentals of the NetSuite <> OpenAir integration before you start using OpenAir NetSuite Connector to exchange data between OpenAir and NetSuite. See How the NetSuite <> OpenAir Integration Works.

After the NetSuite <> OpenAir Integration is fully configured in both NetSuite and OpenAir, you can run the integration manually or trigger the integration from OpenAir user scripts.

- Running the Bulk Import and Export Integration on Demand
- Running a Full Synchronization on Demand
- Exporting a Single Record from OpenAir to NetSuite on Demand
- Running the NetSuite <> OpenAir Integration From OpenAir User Scripts

You can select which NetSuite records to import into OpenAir. See Marking NetSuite Records for Import into OpenAir.

You can also schedule the integration to run automatically, configure NetSuite and OpenAir for the real-time integration. The help topics are covered under the configuration chapters:

- Scheduling the Integration to Run Automatically
- Configuring Real-Time Record Export from OpenAir to NetSuite
- Configuring Real-Time Record Import from NetSuite into OpenAir

Running the Bulk Import and Export Integration on Demand

You can trigger the integration manually to import and export records in bulk on demand. You must be an OpenAir account administrator or have the “View and run the NetSuite integration” role permissions to do so.

- You can select the integration workflows you want to include in each manual run. OpenAir NetSuite Connector validates any workflow dependencies before running the integration. A validation error message appears if a workflow is required but not included in the run. For example, if you configure the manual run to import projects from NetSuite, customers must also be imported.
- Only the integrated records modified since the last integration run are included. OpenAir NetSuite Connector compares the **Updated** timestamp on the NetSuite record and the OpenAir record to determine which records to include in the integration.

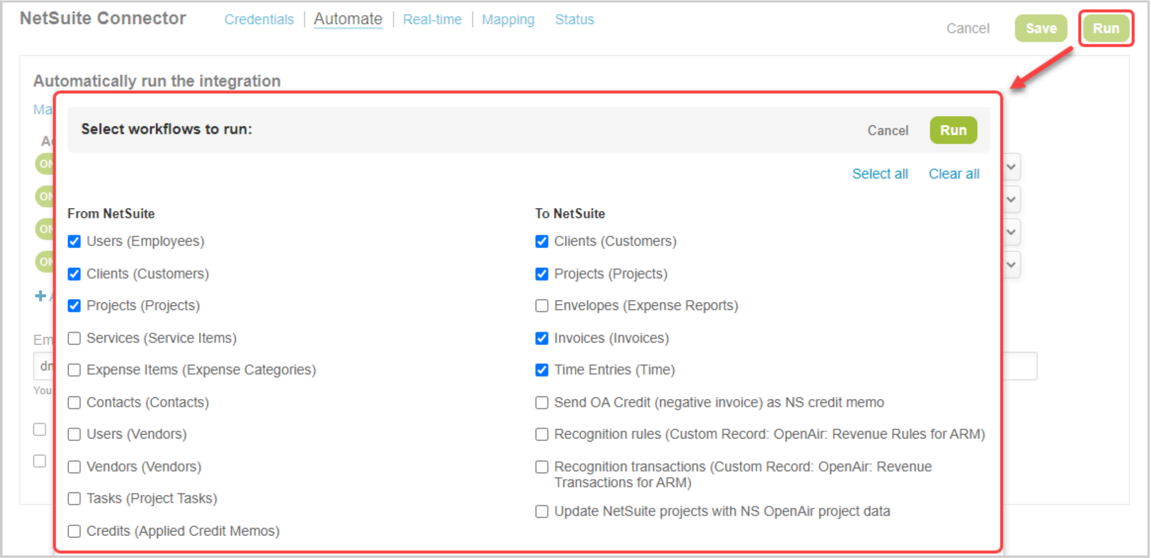
  You can run a full synchronization to synchronize all integrated records for all or selected workflows. See Running a Full Synchronization on Demand.

To run the bulk integration on demand:

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click **Run**.

   A popup window appears.

3. Check the boxes for the integration workflows you want to include in the integration run.
4. Click **Run** to launch the integration run for the selected integration workflows.



NetSuite Connector
Credentials | Automate | Real-time | Mapping | Status
Cancel
Save
Run
Automatically run the integration

Select workflows to run: Cancel Run

Select all Clear all

From NetSuite	To NetSuite
☑ Users (Employees)	☑ Clients (Customers)
☑ Clients (Customers)	☑ Projects (Projects)
☑ Projects (Projects)	□ Envelopes (Expense Reports)
□ Services (Service Items)	☑ Invoices (Invoices)
□ Expense Items (Expense Categories)	☑ Time Entries (Time)
□ Contacts (Contacts)	□ Send OA Credit (negative invoice) as NS credit memo
□ Users (Vendors)	□ Recognition rules (Custom Record: OpenAir: Revenue Rules for ARM)
□ Vendors (Vendors)	□ Recognition transactions (Custom Record: OpenAir: Revenue Transactions for ARM)
□ Tasks (Project Tasks)	□ Update NetSuite projects with NS OpenAir project data
□ Credits (Applied Credit Memos)	

Running a Full Synchronization on Demand

You can run a full synchronization on demand to synchronize integrated records for all or selected workflows. Depending on your integration configuration, the full synchronization includes either all records last modified on or after the date specified, or all records regardless of when the record was last modified. You must be an OpenAir account administrator or have the “View and run the NetSuite integration” role permissions to do so, and the Expose NetSuite full synchronization control must be checked on the OpenAir NetSuite Connector administration form – See Expose NetSuite full synchronization control since date.

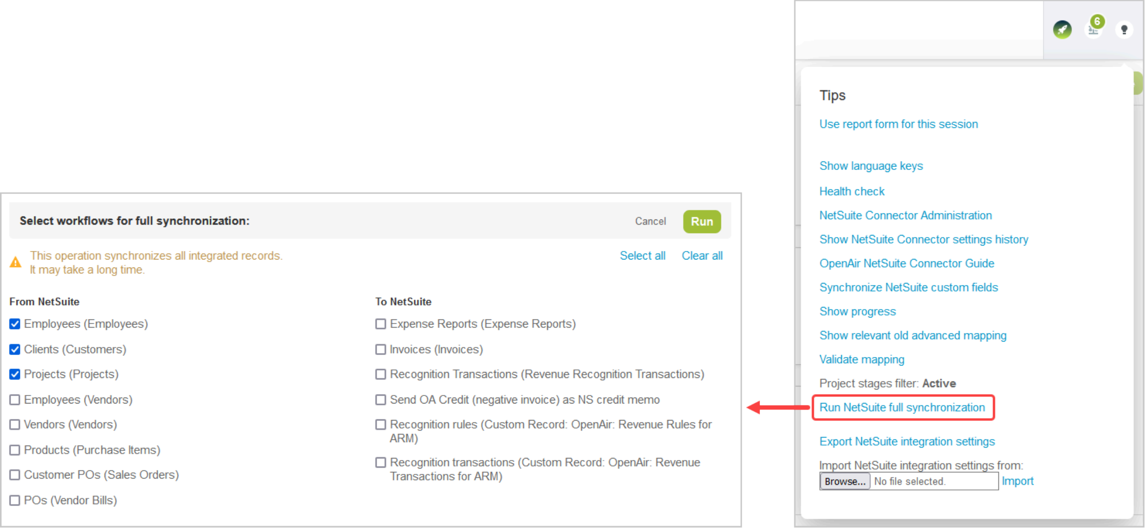
You can select the integration workflows you want to run a full synchronization for. OpenAir NetSuite Connector validates any workflow dependencies before running the integration. A validation error message appears if a workflow is required but not included in the run. For example, if you configure a full synchronization for projects imported from NetSuite, you must also include imported customers in the full synchronization run.

> **Important:** This operation synchronizes all integrated records and any new records marked for integration for the selected workflows.
>
> Full synchronization runs may take a long time. Consider either of the following options before launching a full synchronization run.
>
> - Running the Bulk Import and Export Integration on Demand.
> - Limiting the full synchronization to records last modified on or after a specific date. See Expose NetSuite full synchronization control since date.

To run a full synchronization on demand:

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button, then click **Run NetSuite full synchronization** from the Tips menu.

   A popup window appears
3. Check the boxes for the integration workflows you want to include in the integration run.
4. Click **Run** to launch the integration run for the selected integration workflows.



Select workflows for full synchronization: Cancel Run

This operation synchronizes all integrated records.
It may take a long time.

Select all Clear all

From NetSuite

- ☑ Employees (Employees)
- ☑ Clients (Customers)
- ☑ Projects (Projects)
- □ Employees (Vendors)
- □ Vendors (Vendors)
- □ Products (Purchase Items)
- □ Customer POs (Sales Orders)
- □ POs (Vendor Bills)

To NetSuite

- □ Expense Reports (Expense Reports)
- □ Invoices (Invoices)
- □ Recognition Transactions (Revenue Recognition Transactions)
- □ Send OA Credit (negative invoice) as NS credit memo
- □ Recognition rules (Custom Record: OpenAir: Revenue Rules for ARM)
- □ Recognition transactions (Custom Record: OpenAir: Revenue Transactions for ARM)

Tips

Use report form for this session

Show language keys

Health check

NetSuite Connector Administration

Show NetSuite Connector settings history

OpenAir NetSuite Connector Guide

Synchronize NetSuite custom fields

Show progress

Show relevant old advanced mapping

Validate mapping

Project stages filter: **Active**

Run NetSuite full synchronization

Export NetSuite integration settings

Import NetSuite integration settings from:
Browse... No file selected. Import

Marking NetSuite Records for Import into OpenAir

You can select which NetSuite records to import into OpenAir. The integration bundle adds the **Export to OpenAir** custom field for each NetSuite record type that can be imported into OpenAir. Only those records with the **Export to OpenAir** box checked are imported into OpenAir as part of the Bulk Import and Export On Demand, Scheduled Bulk Import and Export, and Real-Time Record Import, unless otherwise specified in the workflow settings on the OpenAir NetSuite Connector UI.

- You can include the **Export to OpenAir** custom field on your record forms and lists.
- You can configure NetSuite to check the **Export to OpenAir** box by default for new records using the **Export <Record Type> by Default** NetSuite OpenAir SRP Integration custom preference. See Configuring Integration Preferences and Terminology Preferences in NetSuite.
- You can configure OpenAir NetSuite Connector to import NetSuite records that are not marked for export to OpenAir as well as those marked for export to OpenAir using the **Disable Export to OpenAir filtering for <Record Type> in NetSuite integration** workflow setting. See Workflow Settings.

Note: The **Export to OpenAir** check box is not available for expense categories.

For more information, see Working with Expense Categories in OpenAir.

To mark a NetSuite record for import into OpenAir

1. In NetSuite, go to the record you want to import into OpenAir.
2. Check the **Export to OpenAir** box on the NetSuite record. Conversely, clear the **Export to OpenAir** box if you do not want the NetSuite record to be imported into OpenAir.
3. Click **Save**.

> ✓ **Tip:** Add the **Export to OpenAir** field as a column to your list views in NetSuite and use inline-editing to check or clear the **Export to OpenAir** directly from your list views.

Exporting a Single Record from OpenAir to NetSuite on Demand

You can run the integration manually to export a single OpenAir record to NetSuite on demand. You must be an OpenAir account administrator or have the “View and run the NetSuite integration” role permissions to do so.

Single record export on demand follows the field mapping and settings configured for the relevant integration workflow. See Adding and Managing Integration Workflows.

Single record export on-demand can be used with the following OpenAir record types. See the description integration workflows for any applicable restrictions.

- Expense Reports — See Expense Reports (Expense Reports) Export Workflow.
- Invoices — See Invoices (Invoices) Export Workflow.
- Revenue Recognition Transactions — See Recognition Transactions (Revenue Recognition Transactions) Export Workflow.
- Customers — See Customers (Customers) Export Workflow.
- Timesheets — See Time Entries (Time) Export Workflow.
- Purchase Requests — See Purchase Requests (Purchase Requisitions) Export Workflow.
- Projects — See Projects (Projects) Export Workflow and Update NetSuite Projects with OpenAir Project Data.
- Project Tasks — See Update NetSuite Tasks with OpenAir Task Data.
- Work Breakdown Structure — See WBS (Opportunity) Export Workflow.

To export a single record from OpenAir to NetSuite on demand:

1. In OpenAir, go to the record you want to export.
2. Click the Tips button then **Export to NetSuite** or **Send to NetSuite**.

Running the NetSuite <> OpenAir Integration From OpenAir User Scripts

You can use the following user scripting functions to trigger an integration run:

- `NSOA.NSConnector.integrateAllNow()` — Use this function to import and export records in bulk from your scheduled scripts. The run will include all integration workflows that are active at the time the run is triggered. See User Scripting.
- `NSOA.NSConnector.integrateWorkflowGroup(name)` — Use this function to import and export records in bulk from your scheduled scripts. The run will include only integration workflows in the workflow group specified by name. See User Scripting.
- `NSOA.NSConnector.integrateRecord()` — Use this function to export a single OpenAir record from your form scripts. See User Scripting.

Working with Integrated Records in NetSuite and OpenAir

After Professional Services have configured your NetSuite and OpenAir accounts, you are ready to begin working with records and data in each application. Refer to the following:

- Sharing Core Data - Review records to verify that core data exists in both applications. This allows you to exchange information and manage projects between the two applications. While Professional Services does your initial set up, you need to maintain core data as you make changes or enhancements to your account.
- Working with Records in NetSuite - Review the records available for transfer from NetSuite to OpenAir, why you use them, and how you export them.
- Working with Data in OpenAir - Review the records available for transfer from OpenAir to NetSuite, why you use them, and how you export them.

Each is explained in more detail in the following sections.

Sharing Core Data

When Professional Services configures your NetSuite and OpenAir accounts for the integration, the records necessary to ensure the integration functions properly are present in both applications. As your company grows and adds more customers, projects, and employees to your accounts or as you implement new features or make changes, you must add to that initial set up of core data and include records that reflect the growth and modifications made. By maintaining core data, you keep the foundation you need to successfully run the integration.

Shared core data may include information about customers, contacts, projects, employees, purchase items, vendors, and in some cases, vendors who become OpenAir users. It also may include service items for exporting invoices and expense categories or expense items for transferring expense reports. These core data must be in both applications for the same entities before you can exchange related transactional information to and from either application for expense reports, invoices, and revenue recognition transactions. They must also be present to use optional features like sales orders, purchase orders, vendor bills, expense report reimbursements, invoice payments, and customer deposits and credits applied to invoices. When you establish core data in both applications, you can exchange information at key points in the management process.

The following example shows how shared core data provides the framework for data exchange. An OpenAir user who is also a NetSuite employee creates an expense report in OpenAir. It references a customer and project. Expenses include meals and travel, expense categories or expense items that exist in both applications. In fact, all of these records must exist in both applications. Once the expense report is approved in OpenAir, it transfers to NetSuite when you run the integration. Because the core data exists in both applications, the expense report can transfer from OpenAir to NetSuite.

The shared core data in the previous example include a customer, project, user, and an expense report containing expense categories or items. With that common information, you may do one or more of the following:

- Create a NetSuite service order, transfer it to OpenAir, and run billing rules in OpenAir for the project based on the NetSuite service order.
- Calculate revenue transactions in OpenAir for completed activities/items the sales order transferred from NetSuite.
- Pay the expense report in NetSuite and transfer payment information back to the expense report in OpenAir.

- Use the expense report in charges on an OpenAir invoice and transfer it to NetSuite for payment and record keeping.
- Pay or credit invoices in NetSuite and apply those payments and credits to the invoice in OpenAir.
- Update general ledgers in NetSuite and create reports.

Core Data Records

The following records are common to both applications. While Professional Services makes sure that the records for core data match when they configure your account, your account will grow and change over time as you implement enhancements or make changes that require additional records. Review the record types below to understand how core records work together between the applications to make the integration run smoothly. Records included in core data are:

- Employees and Vendors established and transferred from NetSuite to OpenAir. Employee and user records must be in both applications before you can transfer expense reports, timesheets, or use time and expenses in billing rules. Vendors used in purchase orders must be in both applications before you can exchange purchase order information. See Working with Employee Records in OpenAir and Working with Vendors in OpenAir.
- Customers established and transferred from NetSuite to OpenAir. Customer records must be in both applications before you can select them on expenses, timesheets, or projects and create subsequent billing rules and invoices. Working with Customer Records in OpenAir.
- Projects established and transferred from NetSuite to OpenAir. Project records must be in both applications before you can select them on expenses or timesheets or use them in subsequent purchase orders, billing rules, and invoices. If your account is multi-currency, you must also go into the Edit project or Properties form in OpenAir and type in an exchange rate. If it is not, this step is not necessary. See Working with Project Records in OpenAir.
- Items for services established and transferred from NetSuite to OpenAir. Service items from NetSuite are Activities in OpenAir. These records must be in both applications before you can use them on invoices or revenue recognition transactions. NetSuite Inventory, Kit/Package, Non-inventory, Other Charge, and Service items must be in both applications before you can use them on purchase orders. See Working with Item Records in OpenAir.
- Expense categories established and transferred from NetSuite to OpenAir. Expense categories from NetSuite are Expense items in OpenAir. These records need to be in both applications before you can select them on receipts and transfer them in expense reports. See Working with Expense Categories in OpenAir.

Customizing OpenAir List Views to Display NetSuite IDs

In OpenAir, you can easily customize list views to display a column for associated NetSuite IDs. This provides a quick way to determine which entities already exist in both applications since the IDs are on the list view. Also, when you transfer a new record that has a name that is similar to an existing record, you can easily find the new record because it displays the NetSuite ID.

To customize OpenAir list views to display a NetSuite ID:

1. Click the arrow on the top left menu bar and select Customize list view. A pop-up window displays with available items.
2. Scroll down and highlight the NetSuite ID information you want to display.
3. Click Add selected items.
4. Highlight the item to move it to the most convenient display location.

5. Click OK.

Adding an OpenAir Record to Match a NetSuite Record

If a record exists in NetSuite and not in OpenAir, simply mark it for export to OpenAir. When you run the integration, it establishes the matching record in OpenAir. The integration transfers the information from NetSuite and creates a record in OpenAir that includes the assigned NetSuite ID number. See Exporting NetSuite Records to OpenAir and Running the Integration.

Adding a NetSuite Record to Match an OpenAir Record

If a record already exists in OpenAir, but doesn't exist in NetSuite, you need to create it in NetSuite and transfer it to OpenAir. To use it in the integration, it must exist in both applications, and in OpenAir, it must display a NetSuite ID on the appropriate form. You create a matching record in NetSuite and transfer it to OpenAir when you run the integration. See Exporting NetSuite Records to OpenAir and Running the Integration.

The following is an example of what you do to add an employee record in NetSuite to match an existing user record in OpenAir.

To establish an employee record in NetSuite and transfer it to OpenAir:

1. Create a record in NetSuite for the user. The name you enter in NetSuite should be identical to the name in the OpenAir user account.
2. Run the integration. Refer to Running the Integration. OpenAir searches for the User record and adds a NetSuite User ID to the existing OpenAir account.
3. Navigate to OpenAir > Global Settings > Users and click on the User ID.Edit user demographic information form displays.
4. Scroll to the bottom of the form and verify that a NetSuite Employee ID displays.

> ⚠ **Important:** If the NetSuite record you created did not exactly match the OpenAir user account, a new user record is created in OpenAir. Now you have two records for the same user with slightly different information. Combine them into one record, keeping the original OpenAir information intact. Add the NetSuite Employee ID to the original OpenAir user account.

To add the NetSuite ID to the original OpenAir record:

1. Open the new record that was created when you ran the integration.
2. Copy the NetSuite ID. (In the previous example, it is a NetSuite Employee ID.)
3. Open the original OpenAir record. (In the previous example, it is a user account.)
4. Paste the ID into the NetSuite ID field. (In the previous example, it is a NetSuite Employee ID.)
5. Delete the newly created record. You don't need it because the NetSuite ID was entered into the original OpenAir record.

When you create one of the following record types in NetSuite and run the integration, the specified OpenAir form contains the following NetSuite ID.

NetSuite Record	Transfers to OpenAir Form	NetSuite ID Added
Vendor	Depending on your implementation:	NetSuite Vendor ID

NetSuite Record	Transfers to OpenAir Form	NetSuite ID Added
	Administration > Application Settings > Purchases > Vendors Administration > Global Settings > Users > Employees and select a User ID	
Customer	Administration > Global Settings > Customers > Customers and select a Nickname	NetSuite Customer ID
Projects	Projects > Projects and select a Project and then click Edit project or Properties link	NetSuite Job ID (You must also type in an exchange rate in the NetSuite Project Exchange Rate field and save the project.)
Contacts	Administration > Global Settings > Customers > Contacts and select a contact Name	NetSuite Contact ID
Item	Administration > Application Settings > Invoices > Activities and select an Activity	NetSuite Item ID
Resale or Purchase Item	Administration > Application Settings > Purchases > Products	NetSuite Purchase Item ID
Expense Category	Administration > Application Settings > Expenses > Expense items and select an Expense item	NetSuite ExpenseCategory ID
Opportunity	Projects > Projects and select a Project and then click Edit project or Properties link	NetSuite Job ID and NetSuite Opportunity ID

Working with Records in NetSuite

The integration is flexible in that you can choose how you want to use it and when you want to use it. It provides the overall structure to capture data in NetSuite and export that data to OpenAir. Lists of NetSuite records available for export as well as export methods follow.

NetSuite Records Available for Export to OpenAir

The following are records you can export from NetSuite to OpenAir. In many cases, you must export records that include core data to be able to use the information in the integration. You can export each record individually or choose to export all records within the record types.

- Employees - Must establish employee records in NetSuite and transfer them to OpenAir before you can transfer a user's expense report or timesheet back to NetSuite, or use the user's time and expenses in OpenAir billing rules for invoice charges. A NetSuite Employee ID custom field displays on the User Edit demographic information form.
- Vendors - Must establish vendor records in NetSuite and transfer them to OpenAir either as a vendor account or as a user account. Depending on your implementation, a NetSuite Vendor ID custom field displays on the Vendor form or the User Edit demographic information form. See Working with Vendors in OpenAir.
- Customers - Must establish customer records in NetSuite and transfer them to OpenAir before you can select them on expenses, timesheets, or projects and create subsequent billing rules and invoices. A customer must exist in NetSuite before a contact associated with the customer can be transferred to

OpenAir. A NetSuite Customer ID custom field displays on the Customer Edit demographic information form.

- Projects - Must establish project records in NetSuite and transfer them to OpenAir before you can select them on expenses or timesheets or use them in subsequent purchase orders, billing rules and invoices. A project must exist in NetSuite before a contact or sales order associated with the project can be transferred to OpenAir. A NetSuite Job ID custom field displays on the Edit project or Properties form.

  Creating multi-currency projects and invoices is a standard feature of OpenAir, but needs to be enabled in NetSuite. To enable multi-currency customer support in the integration, update NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) to version 1.31 or later, and enable multi-currency customers in NetSuite.

  Multi-currency projects, invoices and other sales transactions can then be synchronized between NetSuite and OpenAir.

- Contacts - Must establish a contact record in NetSuite before you can use it in OpenAir. A NetSuite Contact ID custom field displays on the Edit contact form.
- Opportunities - Must establish an opportunity in NetSuite and associate it with a NetSuite project before you can transfer it to OpenAir. The customer must exist in OpenAir. NetSuite Job ID and NetSuite Opportunity ID custom fields display on the Edit project or Properties form.
- Items for Services or Activities - Must establish item records in NetSuite before you can use them on OpenAir timesheets, invoices or revenue recognition transactions. A NetSuite Item ID custom field displays on the Edit activity form. When you transfer an opportunity to OpenAir to develop a project's work breakdown structure for opportunity items, the OpenAir Timesheets activities must exist as items in NetSuite and contain rate information. See Configuring the Opportunity to Project Integration.
- Items for Purchase - Must establish items in NetSuite before you can use them in purchase orders that transfer to OpenAir. The purchase item can be an Inventory, Kit/Package, Non-inventory, Other Charge, or Service item. A custom field for NetSuite Purchase Item ID is added to the OpenAir Product form. In addition, if the revenue recognition integration is enabled, the following custom fields are also added: NetSuite Product Subtype, NetSuite Income Account ID, NetSuite Product Type, and NetSuite Deferred Revenue Account ID. A check box field indicated if the NetSuite Product Is Fulfillable.
- Expense Categories - While these are not defined as a record in NetSuite, you transfer expense categories to OpenAir so you can select them on receipts and transfer them back to NetSuite on expense reports. A NetSuite ExpenseCategory ID displays on the Edit expense item form. See Working with Expense Categories in OpenAir and Setting up the NetSuite Integration in OpenAir.
- Sales Orders - Must establish sales order records in NetSuite and transfer them to OpenAir before you can use them on OpenAir billing rules, OpenAir recognition rules, or invoices. Depending on your integration, the following custom fields are added to the OpenAir Customer POs form: NetSuite Discount Customer PO, NetSuite Sales Order Line ID, NetSuite Project ID, NetSuite Sales Order Net Amount, and NetSuite Sales Order ID, NetSuite Sales Order Line Quantity, and NetSuite Item ID. Other check box fields include: Calculate Quantity From % Invoiced, NetSuite Sales Order Line Discount Customer PO, and NetSuite Sales Order Header Discount Customer PO. An Activities form may include the following check box: NetSuite description item use for charges in invoice export. This check box is used only with the “Enable charge details on invoices to map to NetSuite description items when using the sales order integration” feature. See Configuring the Sales Order Integration.
- Purchase Orders and Vendor Bills - Must establish vendors and purchase items in NetSuite and transfer them to OpenAir before you can use them on purchase orders and vendor bills transferred to OpenAir.
- Expense Report Reimbursements - When expense reports transfer from OpenAir to NetSuite, payments made in NetSuite can transfer back to OpenAir and display as reimbursements. The NetSuite Payment ID displays in Notes.
- Invoice Payments - When invoices transfer from OpenAir to NetSuite, payments made in NetSuite can transfer back to OpenAir as invoice payments. The NetSuite Payment ID displays in Notes.

- Customer Deposits Applied to Invoices - When invoices transfer from OpenAir to NetSuite, customer deposits applied in NetSuite can transfer back to OpenAir as retainers. The NetSuite customer deposit ID displays in Notes.
- Credit Memos Applied to Invoices - When invoices transfer from OpenAir to NetSuite, a credit memo applied in NetSuite can transfer back to OpenAir as a negative invoice. If you enter an OpenAir charge type, it also transfers back to NetSuite and displays on the NetSuite Invoice line item. Each line of the NetSuite credit memo becomes a negative charge in OpenAir and OpenAir creates a new negative invoice. The total credited amount displays on the original OpenAir invoice as the reason for credit. The credit memo import respects the value set on the OpenAir charge type for each negative charge created in OpenAir.
- Import Taxes Applied to Exported Invoices - When invoices transfer from OpenAir to NetSuite, taxes on the total amount are applied in NetSuite and can transfer back to OpenAir. The total tax amount displays on the OpenAir invoice.

> ⚠ **Important:** If the SuiteTax is enabled for your NetSuite account, NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) to version 2.11 or later is required in order for the tax amount calculated in NetSuite to be imported into OpenAir. With earlier versions, the tax amount calculated in NetSuite with SuiteTax cannot be imported back into OpenAir.
>
> Note that SuiteTax compatibility testing is currently in progress. You should test the NetSuite <> OpenAir integration with the SuiteTax feature on sandbox accounts before you enable the feature in your NetSuite production account. See Known Limitations of SuiteTax (external link to NetSuite SuiteAnswers). See also Legacy Tax or SuiteTax.

Working with Data in OpenAir

Working with data in OpenAir through the integration is flexible. Choose how and when to use it. Once you transfer records from NetSuite to OpenAir, add information to those records, manage projects in OpenAir, and submit timesheets and expenses for those projects. Later, transfer that information back to NetSuite, either on a case-by-case basis or all records of a type. Lists of OpenAir records available for export as well as different export methods follow.

OpenAir Data Available for Export to NetSuite

Data from OpenAir that transfers to NetSuite includes:

- Approved expense reports - When OpenAir user records have a NetSuite Employee ID and expense items have a NetSuite Expense Category ID, you can transfer approved expense reports from these users for the expense items to NetSuite.
- Approved timesheets - When OpenAir user records have a NetSuite Employee ID, customer records in OpenAir have a NetSuite Customer ID, projects have a NetSuite Job ID, and activities associated with tasks have a NetSuite Item ID, you can transfer approved timesheets from these users to NetSuite.
- Invoices - When customer records in OpenAir have a NetSuite Customer ID, projects have a NetSuite Job ID, and billing rule charges are based on sales orders transferred from NetSuite and edited in OpenAir, you can include those customers, projects, and charges on invoices and export them to NetSuite. You can also initialize OpenAir invoices to gather data from the NetSuite Sales Order form like you would use the Next Bill button in NetSuite to pre-populate an invoice. Additional charge details can also transfer to NetSuite with the OpenAir invoice. See Configuring the Sales Order Integration.

  Creating multi-currency projects and invoices is a standard feature of OpenAir, but needs to be enabled in NetSuite. To enable multi-currency customer support in the integration, update NetSuite

OpenAir SRP Integration bundle 369637 (previously 2851) to version 1.31 or later, and enable multi-currency customers in NetSuite.

Multi-currency projects, invoices and other sales transactions can then be synchronized between NetSuite and OpenAir.

- Revenue recognition transactions - When customer records in OpenAir have a NetSuite Customer ID, projects have a NetSuite Job ID, and billing rule charges are based on sales orders transferred from NetSuite and edited in OpenAir, you can include those customers, projects, and charges on revenue recognition transactions and transfer them to NetSuite.
- Opportunity items - An opportunity associated with a project transfers to OpenAir when customer records in OpenAir have a NetSuite Customer ID and projects have a NetSuite Job ID and a NetSuite Opportunity ID. Items on a project's work breakdown structure can transfer back to the NetSuite opportunity. To use this feature, click the “Update the NetSuite opportunity line items” link on either the OpenAir Projects Phases/Tasks: List or Outline forms. A check box does not display under “Send data to NetSuite” on the NetSuite integration form. See Configuring the Opportunity to Project Integration.
- Customers — The Customers (Customers) export workflow lets you create/update Customer records in OpenAir first and then export them to NetSuite. The workflow supports both integration on-demand and real-time integration. See Customers (Customers).
- Projects — The Projects (Projects) export workflow lets you create/update Project records in OpenAir first and then export them to NetSuite. The workflow supports both integration on-demand and real-time integration. See Projects (Projects).

Note: Names for OpenAir records differ slightly from those in NetSuite. See Terminology Mapping for the Integration for a terminology mapping table.

Configuring Optional Features

This chapter provides instructions for implementing the optional features in the integration. Since the development of the integration is ongoing, new features and functionality are being added. Even though these features may not have been in your initial set up, they are available and you can use them. It also provides guidance for using them, especially if the workflow is unfamiliar. Remember that you can always contact OpenAir Customer Support for more information and assistance. To enable new features, request that the latest OpenAir Bundle be installed. See Contacting NetSuite and OpenAir and open a support ticket.

To set up some optional features, you must configure specific settings in both applications. To set up other features, you must run the integration to establish and exchange data. Because of that, these procedures address steps you take in each application. You should sign in to both your OpenAir account and your NetSuite account while you implement each of the following features.

> **Note:** The structure and content of this guide are currently under review. The review aims to consolidate information and create self-contained reference for each integration workflows and other integration features. Content for the some of the following topics below has been reviewed and relocated. Alternative reference are given where the content has been relocated.

- Configuring the Classifications to Hierarchies Integration — Content relocated to Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.
- Configuring the Expense Report to Vendor Bill Integration — Content relocated to Expense Reports (Expense Reports) Export Workflow.
- Configuring the Invoices Integration — Content relocated to Invoices (Invoices) Export Workflow.
- Configuring the Opportunity to Project Integration
- Configuring the Projects Integration — Content relocated to:
  - Projects (Projects) Import Workflow.
  - Tasks (Project Tasks) Import Workflow
  - Invoices (Invoices) Export Workflow
  - Update NetSuite Projects with OpenAir Project Data Export Workflow
  - Update NetSuite Tasks with OpenAir Task Data
- Configuring the Purchases/Vendor Bills Integration — Content relocated to:
  - POs (Purchase Orders) Import Workflow
  - POs (Vendor Bills) Import Workflow
- Configuring the Vendor Credit Integration — Content relocated to POs (Vendor Credits) Import Workflow.
- Configuring Revenue Recognition
- OpenAir Revenue Recognition with NetSuite Advanced Revenue Management
- Configuring the Sales Order Integration
- (DEPRECATED) Configuring the Salesforce.com Integration for Customers and Projects
- Configuring the Tax Export Integration — Content relocated to Enable tax export on expense reports.
- Configuring Tax Override for Canadian Expense Reports — Content relocated to Allow manual override of Canadian taxes on Expense report export.
- Configuring the Time Entries Export Integration
- Configuring Work In Progress Accounting and Intercompany Journal Entries

- Configuring the Support Case Integration — Content relocated to Tasks (Support Cases) Import Workflow.
- Configuring Purchaser Integration — Content relocated to Enable purchaser integration.
- Configuring Integration for Percent Complete Rules with NetSuite Advanced Revenue Management

After both software applications have been configured properly, you can transfer data and use these specific features.

Configuring the Classifications to Hierarchies Integration

See Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

The structure and content of this guide are currently under review. The review aims to consolidate information and create self-contained reference for each integration workflows and other integration features. This topic has been removed and replaced with the topic referenced above.

Configuring the Expense Report to Vendor Bill Integration

See Expense Reports (Expense Reports) Export Workflow.

The structure and content of this guide are currently under review. The review aims to consolidate information and create self-contained reference for each integration workflows and other integration features. This topic has been removed and replaced with the topic referenced above.

Configuring the Invoices Integration

See Invoices (Invoices) Export Workflow.

The structure and content of this guide are currently under review. The review aims to consolidate information and create self-contained reference for each integration workflows and other integration features. This topic has been removed and replaced with the topic referenced above.

Configuring the Opportunity to Project Integration

You can create an opportunity in NetSuite and transfer an associated project to OpenAir. Once the project is in OpenAir, develop a work breakdown structure for the project that includes phases/tasks and hours and transfer the default timesheet activities and planned hours as items back to the opportunity in NetSuite, where you can then edit and further enhance them. Total planned hours are combined and a total for each item displays on the NetSuite Opportunity form. Pricing is applied in NetSuite.

Using the OpenAir Projects application helps you better define an opportunity and scope out a potential project. A configuration setting can be enabled for your OpenAir account to determine the activity from tasks/phases in the work breakdown structure. You may add tasks and users to a project in OpenAir as needed and transfer the updates to the opportunity in NetSuite as many times as required. Updates are made on a project-by-project basis for each opportunity using a link in tips on either the OpenAir Projects Phases/Tasks: List or Outline forms. These updates do not function as part of an integration run and cannot be scheduled.

Note: Since the integration uses planned hours instead of assigned hours, you can assign planned hours to a user or you can choose to assign planned hours per task or use task assignment at the project level.

This integration uses service items from NetSuite and the related OpenAir Timesheet or Invoices activities, so these records need to be present in both applications. The corresponding NetSuite service items must contain rate information. The integration uses the default rate the first time items are added to the opportunity. Subsequent updates respect the rates if they were changed. Customers must also exist in both applications as well. See Sharing Core Data.

To enable the Opportunity to Project Integration in NetSuite:

1. Make sure you are using the latest version of the NetSuite OpenAir SRP Integration bundle 369637 (previously 2851). It should be 1.19 or higher. A custom field is added to the Projects form for OpenAir: Opportunity. To associate a project with an opportunity, select the opportunity number using the drop-down list. This task can be automated using a SuiteScript. See below.
2. Ensure rate information is associated with items. Go to Lists > Accounting > Items and click on an item. Click on Pricing to edit or add rate information.
3. (Optional) Contact OpenAir Professional Services and request that a SuiteScript be deployed to automatically create a project from an opportunity and associate the two. If deployed, a custom field is added to the Opportunity form for OpenAir: Create Project.

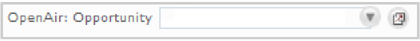
To enable the Opportunity to Project Integration in OpenAir:

1. On the NetSuite Connector administration form in OpenAir , check the **Enable OpenAir project to NetSuite opportunity integration** box.
2. Specify which OpenAir users have permission to perform updates to NetSuite opportunities on the OpenAir user demographic form. Select the check box under Projects Options for: The user can update a NetSuite opportunity from a project work breakdown structure. The following link is added to Projects: Update the NetSuite opportunity line items. It displays on the Projects forms for task list or task outline in tips.
3. (Optional) If desired, contact OpenAir Customer Support and request the account configuration setting to be enabled: Enable the feature to determine the activity to be set in transactions by searching the task and phase work breakdown structure. This allows you to set an activity at the phase/task level and have it apply also at the task level.

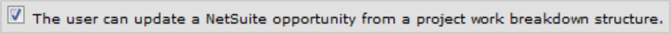
Implementing the Opportunity to Project Integration

The bundle adds custom fields to both applications. They include:

- NetSuite project form:

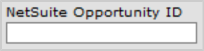


- OpenAir user demographic form:



☑ The user can update a NetSuite opportunity from a project work breakdown structure.

- OpenAir project properties form:



NetSuite Opportunity ID

To implement the Opportunity to Project integration:

1. In NetSuite, go to Transactions > Sales > Create Opportunities. Identify the Company and add a Project. Complete other required information. Click Save.
2. In NetSuite, go to Lists > Relationships > Projects > List and locate the project that was created from the opportunity. Find the OpenAir: Opportunity custom field. You may need to change your project form to display the field. Use the drop-down list and select the opportunity number of the opportunity you just created. Click Save.

   > **Note:** You can contact OpenAir Professional Services and request that a SuiteScript be deployed to automatically create a project from an opportunity and associate the two.

3. In OpenAir, run the integration. See Running the Integration.
4. In OpenAir, go to Projects > Projects and select the project associated with the opportunity you just created. The Internal ID of the NetSuite Opportunity is listed on the Properties form. Add or update any additional project information and Save.
5. In OpenAir, add work breakdown structure information such as phases/tasks, duration, and user assignments.

   > **Note:** Any hours assigned to a project task that is not associated with a Timesheets or Invoices activity are skipped. Only activities with NetSuite IDs transfer to items in the NetSuite opportunity. See Sharing Core Data.

6. To transfer the project information back to the NetSuite opportunity, click the link in tips for: Update the NetSuite opportunity line items

   > **Note:** If you do not see the link in tips to Update the NetSuite opportunity line items, contact your Administrator. Permission to perform an update of NetSuite opportunity line items is provided on the OpenAir user demographic form.

7. To view the updated items on the NetSuite opportunity, in NetSuite go to Transactions > Sales > Create Opportunities > List and select the opportunity.

Configuring the Projects Integration

See the following topics:

- Projects (Projects) Import Workflow.
- Tasks (Project Tasks) Import Workflow
- Invoices (Invoices) Export Workflow
- Update NetSuite Projects with OpenAir Project Data Export Workflow

- Update NetSuite Tasks with OpenAir Task Data

The structure and content of this guide are currently under review. The review aims to consolidate information and create self-contained reference for each integration workflows and other integration features. This topic has been removed and replaced with the topics referenced above.

Configuring the Purchases/Vendor Bills Integration

See the following topics:

- POs (Purchase Orders) Import Workflow
- POs (Vendor Bills) Import Workflow

The structure and content of this guide are currently under review. The review aims to consolidate information and create self-contained reference for each integration workflows and other integration features. This topic has been removed and replaced with the topics referenced above.

Configuring the Vendor Credit Integration

See POs (Vendor Credits) Import Workflow.

The structure and content of this guide are currently under review. The review aims to consolidate information and create self-contained reference for each integration workflows and other integration features. This topic has been removed and replaced with the topics referenced above.

Configuring Revenue Recognition

When you amortize revenue and set up rules for recognizing it on a project-by-project basis, you can exchange this information between the applications. If you use revenue recognition in OpenAir, you can send revenue recognized from paid invoices to NetSuite. To use the revenue recognition integration, you must enable the feature in both accounts.

OpenAir revenue recognition can also be configured to work with NetSuite's Advanced Revenue Management. Please see OpenAir Revenue Recognition with NetSuite Advanced Revenue Management for more details.

⚠ **Important:** This section relates to the **Recognition Transactions (Revenue Recognition Transactions)** workflow sending transactional data to NetSuite as Journal Entries. If you want to integrate your revenue recognition rules and recognition transactions to NetSuite for use in Advanced Revenue Management, see OpenAir Revenue Recognition with NetSuite Advanced Revenue Management.

To enable Revenue Recognition in NetSuite:

1. Go to Setup > Company > Enable Features > Accounting and select the check box for Revenue Recognition under Advanced Features.

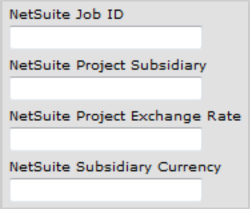
2. Customize your invoice to show the Delay Rev Rec field. Go to Transactions > Sales > Create Invoices. Select Customize > Screen Fields > Columns and select the check box to show the Deferred Revenue field.
3. Identify the Deferred Revenue account on all service items. Go to Lists > Accounting > Items. For each service item, select the Deferred Revenue Account on the Rev Rec/Amort subtab.
4. Contact your NetSuite Professional Services representative to complete any other setup steps required in NetSuite to correctly distribute deferred revenue and unbilled receivables over the life of a project.

To enable Revenue Recognition in OpenAir:

1. On the NetSuite Connector administration form, check the following box under Exports:
   - **Recognition Transactions (Revenue Recognition Transactions)**.
2. In the NetSuite Connector UI, add the following workflows on the Mapping tab:
   - **Invoices (Invoices)** export.
   - **Recognition Transactions (Revenue Recognition Transactions)** export.

Implementing Revenue Recognition

Once you run the integration, custom fields display on both the Edit project and Edit activities forms in OpenAir. Refer to the following examples.



NetSuite Job ID

NetSuite Project Subsidiary

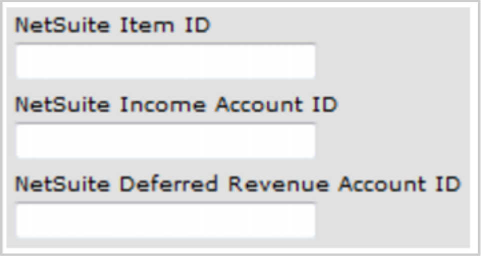
NetSuite Project Exchange Rate

NetSuite Subsidiary Currency

The following describes custom fields created for different types of NetSuite accounts:

Custom Field	NetSuite Account Type	Custom Fields Description
NetSuite Project ID	All NetSuite accounts	Displays the internal ID from NetSuite.
NetSuite Project Subsidiary/ NetSuite Project Subsidiary ID	One World, Multi-currency, and Revenue Recognition	Displays the name/ID of the subsidiary that contains the project in NetSuite. These fields do not function in the integration, but are present to help you define the NetSuite Project Exchange Rate. (See below.)
NetSuite Project Exchange Rate	Multi-currency	Displays a blank NetSuite Project Exchange Rate field, a required field that is NOT related to the exchange rate for a project. When the corresponding NetSuite account is a OneWorld account, this exchange rate converts from the project currency to the subsidiary base currency. If not a OneWorld account, this exchange rate converts from the project currency to the system-wide base currency. It is a user-defined exchange rate for exporting invoices and revenue recognition transactions to NetSuite, and it is used for both the invoices and journal entries that are created in NetSuite when revenue recognition transactions are sent.

Custom Field	NetSuite Account Type	Custom Fields Description
NetSuite Subsidiary Currency	One World, Multi-currency, and Revenue Recognition	Displays the currency of the subsidiary that contains the project in NetSuite.


NetSuite Item ID

NetSuite Income Account ID

NetSuite Deferred Revenue Account ID

The NetSuite Item ID displays the item ID of the corresponding NetSuite service item. The other two custom fields display the account IDs of the related service items in NetSuite. The fields must be populated with valid NetSuite IDs. To verify account IDs in NetSuite, go to Lists > Accounting > Items.

To send revenue recognition transaction data from OpenAir to NetSuite:

1. Before you begin, if you have been using the integration prior to enabling the revenue recognition feature, rerun the integration to capture all NetSuite information again:
   - Uncheck the selections under Get Data from NetSuite and Save.
   - Then, re-select the entities and click Save & Run. The integration will then get the latest information from NetSuite and populate the new custom fields in the associated projects and activities forms in OpenAir.
2. In OpenAir, go to Projects > Projects and select a project. If the project has been pulled into OpenAir from NetSuite, ensure it has been opened and saved in the Edit Project form.
3. If the project is multi-currency, click the Edit project link and type the NetSuite Project Exchange Rate and click Save.

   > **Note:** The value you enter is the rate you want to use to recognize the revenue. Invoices are transferred to NetSuite in the default foreign currency specified in Administration > Global Settings > Organization > Currencies > Multi-currency using the NetSuite Project Exchange Rate specified on the Edit project form.

4. Select Financials > Billing, and click Run billing.

   > **Note:** Billing rules must exist. They can be created for each project or copied from another project. For example, if you are using OpenAir for Timesheets, you could run a Time billing rule.

5. Select the desired items and click Create project billing transactions and charges.
6. Select Financials > Recognition and click Run recognition.

   > **Note:** Recognition rules must exist. They can be created for each project or copied from another project. For example, if you are using OpenAir for Timesheets, you might run a Time project billing rule.

7. Select the check boxes of the desired items and click Create recognition transactions.

8. To export an individual revenue recognition transaction, use the link in hints to Export to NetSuite. To export all revenue recognition transactions, run the integration.
9. Go to Invoices > New invoice. Select the project and select the charges to invoice. Click Create the invoice.
10. Submit/Approve the invoice and enter payment information. An approved invoice and its payment information can be exported to NetSuite.

    > **Note:** You can submit the invoice using the link in hints or run the integration. The transfer of revenue recognition transaction data and invoice information are done separately and can be done independently.

11. Run the integration. See Working with Integrated Records in NetSuite and OpenAir.

    A journal entry for the revenue recognition information displays in NetSuite.

> **Note:** You can create custom mappings to map straight from the purchase_item table in revenue recognition to the NetSuite journal entry. See Integration Points Exporting OpenAir Information to NetSuite. You use the following:
>
> - `REVENUE_PURCHASE_TO_JOURNAL` (for header of the journal entry from the purchase item)
> - `JOURNAL_LINE_CREDIT_PURCHASE` (for credit line of the journal entry from the purchase item)
> - `JOURNAL_LINE_DEBIT_PURCHASE` (for debit line of the journal entry from the purchase item)

> **Tip:** Enable your revenue recognition exports to run up to two times faster by exporting recognition transaction records in batches to NetSuite. See Exporting Revenue Recognition Transactions to NetSuite in Batches.

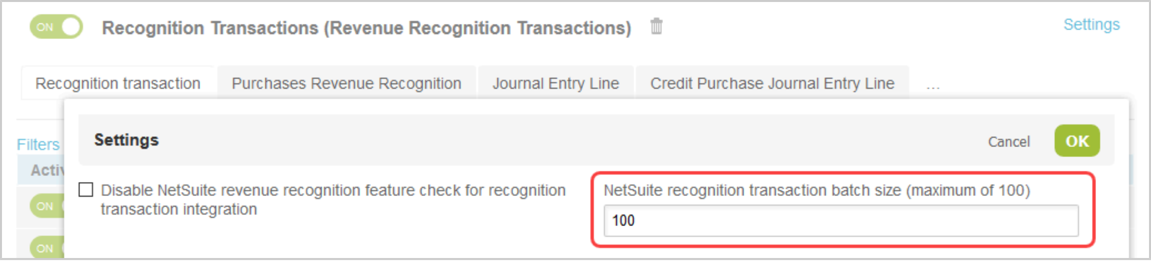
Exporting Revenue Recognition Transactions to NetSuite in Batches

You can enable your revenue recognition exports to run up to two times faster by exporting recognition transaction records in batches to NetSuite.

This feature lets you integrate OpenAir revenue recognition transactions to NetSuite recognition transaction in batches instead of one by one. The maximum batch size is 100 records. The feature will significantly reduce the run time of the Recognition Transactions (Revenue Recognition Transactions) workflow exporting large numbers of journal entries to NetSuite.

To set the recognition transaction record batch size:

1. In OpenAir, go to Administration > NetSuite Connector > Mapping.
2. Click the Recognition Transactions (Revenue Recognition Transactions) workflow.
3. Click **Settings**.
4. Enter the required value for the **NetSuite recognition transaction batch size (maximum of 100)**.
5. Click **Save**.


ON
Recognition Transactions (Revenue Recognition Transactions)
Settings
Recognition transaction
Purchases Revenue Recognition
Journal Entry Line
Credit Purchase Journal Entry Line
...
Filters
Acti
ON
ON

Settings

Cancel OK

☐ Disable NetSuite revenue recognition feature check for recognition transaction integration

NetSuite recognition transaction batch size (maximum of 100): 100

Important: If you have scripts running when records are created in your NetSuite account, reduce the batch size to avoid your scripts timing out. Test this feature in your sandbox account to determine the best batch size for your NetSuite customizations.

Note: Exporting revenue recognition transaction in batches may delay the first web request. This is expected behavior. The subsequent web requests will follow immediately after the first one.

OpenAir Revenue Recognition with NetSuite Advanced Revenue Management

Introduction

OpenAir can integrate your revenue recognition rules and recognition transactions to NetSuite for use in Advanced Revenue Management.

To learn more about OpenAir Revenue Recognition Rules, see the **Revenue Recognition** section in the **Projects** chapter of the Administrator Guide.

The following OpenAir Revenue Recognition Rules are currently supported by NetSuite's Advanced Revenue Management:

- **As Billed**
- **Fixed Amount on Date or Milestone**
- **Fixed Amount % Complete**
- **% Complete**
- **Incurred vs. Forecast**
- **Time Project**

Workflow

The first step in sending OpenAir Revenue Recognition Transactions to NetSuite is to configure the OpenAir Advanced Revenue Management plug-in in NetSuite. This process must only be completed once and is valid for all Revenue Recognition Rules supported by Advanced Revenue Management. See Configuring the Integration for Revenue Recognition Rules with NetSuite Advanced Revenue Management

Next, configure a Revenue Recognition Rule in OpenAir which is supported by Advanced Revenue Management. See the **Revenue Recognition** section in the **Projects** chapter of the Administrator Guide for more information.

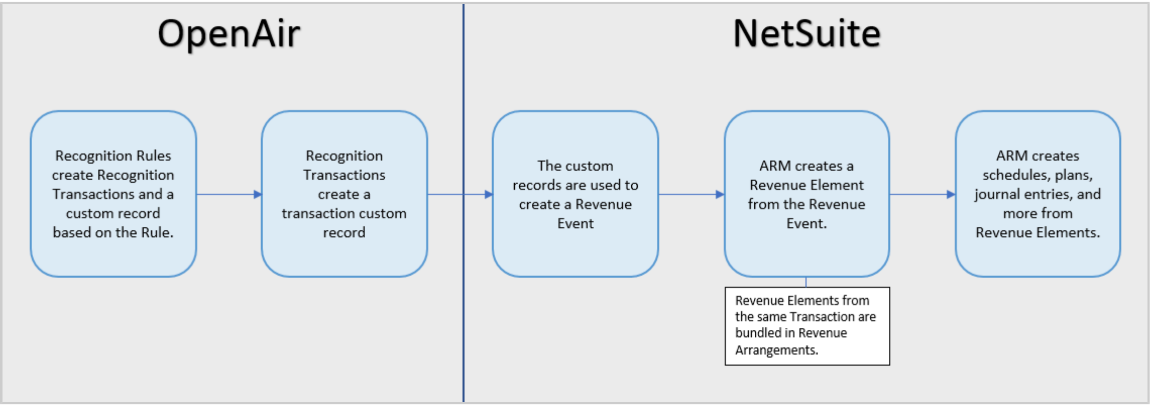
Set up the mappings for the rule on the Mappings page in OpenAir NetSuite Connector.

When you run the integration, OpenAir NetSuite Connector sends instructions for processing your Revenue Recognition Rules to a custom record in NetSuite using the mappings you created.

Note: Creating Revenue Recognition Rules in OpenAir and running the rules are two independent processes. Both must be configured separately.

When OpenAir runs Revenue Recognition Rules, it creates Revenue Recognition Transactions for each line in the rule (As Billed rules work differently, see). When Advanced Revenue Management is triggered in NetSuite, OpenAir NetSuite Connector creates a Revenue Event and a custom record in NetSuite from the Transaction. The custom record says which rule created the Revenue Transaction, and also tells Advanced Revenue Management how to account for the Revenue Event.

Advanced Revenue Management then uses the Revenue Events to create NetSuite Revenue Elements. A Revenue Element is created for each Revenue Recognition Transaction. These Elements are then used by Advanced Revenue Management.


OpenAir
NetSuite
Recognition Rules create Recognition Transactions and a custom record based on the Rule.
Recognition Transactions create a transaction custom record
The custom records are used to create a Revenue Event
ARM creates a Revenue Element from the Revenue Event.
ARM creates schedules, plans, journal entries, and more from Revenue Elements.
Revenue Elements from the same Transaction are bundled in Revenue Arrangements.

Relationship Between Revenue Recognition and Billing

Note: Revenue Recognition and Billing are two independent processes. It is possible to link some types of Revenue Recognition Rules to Billing Rules for a particular project when using Advanced Revenue Management. Billing can be integrated either before or after Revenue Recognition Transactions are sent to NetSuite (“As billed” Revenue Recognition Rules and Transactions are integrated only when the relevant invoice has been exported to NetSuite).

When OpenAir integrates Recognition transactions to Advanced Revenue Management, it also creates custom reports within NetSuite which can be used to help reconcile billing and revenue recognition. These reports are:

- OpenAir: Revenue Rules for ARM
- OpenAir: Revenue Transactions for ARM

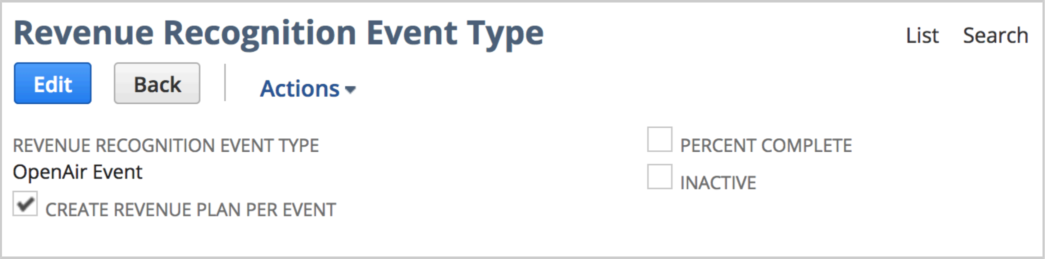
Configuring the Integration for Revenue Recognition Rules with NetSuite Advanced Revenue Management

OpenAir Date or Milestone Fixed Fee, As Billed, Percent Complete, and Incurred vs. Forecast recognition rules can be integrated to NetSuite and used by the Advanced Revenue Management (ARM) feature. When this feature is configured, OpenAir NetSuite Connector sends OpenAir recognition transactions and rules to NetSuite as custom records. The NetSuite OpenAir SRP ARM bundle 370366 (previously 217172) uses these custom records to create revenue arrangements with elements, which Advanced Revenue Management can then use. These values appear in NetSuite reports, and allow you to reconcile your declared revenue against your billing to confirm that your records balance.

To use OpenAir recognition rules with Advanced Revenue Management, you first need to configure the OpenAir Advanced Revenue Management plug-in in NetSuite. Then, configure your OpenAir recognition rules.

To configure the OpenAir Advanced Revenue Management Integration (ARM) plug-in in NetSuite:

1. Enable the Advanced Revenue Management feature in NetSuite. See Advanced Revenue Management for more information.
2. Contact NetSuite Professional Services and request for the limited release Revenue Management plug-in to be enabled on your NetSuite account.
3. Verify that the limited release Revenue Management plug-in was enabled successfully for your NetSuite account. To do so, go to Lists > Accounting > Revenue Recognition Events > New and verify that the fields **Custom Record Type** [`recordtype`] and **Custom Record** [`record`] are available on the form. These field are available when the limited release Revenue Management plug-in is enabled for your account. For more information, see Creating a Custom Revenue Recognition Event (external link to NetSuite SuiteAnswers | Answer ID: 63073) and related topics.
4. Install the NetSuite OpenAir SRP ARM bundle 370366 (previously 217172). OpenAir NetSuite Connector requires version 0.92 or later.
5. Go to Customization > Plugins > Manage plugins and check the **OpenAir: ARM Plugin Implementation** box
6. To begin using this feature, first go to List > Accounting > Revenue Recognition Type and create a **Revenue Recognition Event Type** called **OpenAir Event**.



Revenue Recognition Event Type

REVENUE RECOGNITION EVENT TYPE
OpenAir Event

☑ CREATE REVENUE PLAN PER EVENT

□ PERCENT COMPLETE

□ INACTIVE

Each item which you integrate from NetSuite to OpenAir must have **OpenAir Event** set in the **Create revenue plans on** field.

7. Go to Setup > Accounting > Fair Value Price List and set the Fair Value Price List for this OpenAir Event. You can also select the **Use Sales Price as Fair Value** option in Accounting preferences.

To configure OpenAir Revenue Recognition for use in NetSuite's Advanced Revenue Management:

1. Go to Administration > NetSuite Connector, click the **Tips** button, click NetSuite Connector administration, check the **Recognition rules (Custom Record: OpenAir: Revenue Rules for ARM) | Recognition transactions (Custom Record: OpenAir: Revenue Transactions for ARM)** or **Enable workflows for Advanced Revenue Management feature** box, and click **Save**.

> **Note:** Steps 1 and 2 only need to be performed once. All other steps must be performed for each revenue recognition rule you use with Advanced Revenue Management

2. Enable the **Recognition rules (Custom Record: OpenAir: Revenue Rules for ARM)** and **Recognition transactions (Custom Record: OpenAir: Revenue Transactions for ARM)** workflows in the NetSuite Connector Mapping screen.

These workflows will allow NetSuite to create NetSuite custom records from exported OpenAir revenue recognition rules and transactions correctly.

> **Note:** The default mappings for the **Recognition rules (Custom Record: OpenAir: Revenue Rules for ARM)** includes an additional row if the **Do not use OpenAir project exchange rate on invoice and recognition transaction export** box is NOT checked. This row maps the NetSuite Project Exchange Rate field in the OpenAir Project record to the Exchange Rate in the NetSuite Revenue Element record. When running the integration, the Revenue Elements created in NetSuite will include the exchange rate set on the Project Properties form in OpenAir. The NetSuite Project Exchange Rate field can be set manually or integrated from NetSuite.

3. Go to Projects > Projects > [Select a project] > Financials > Recognition > [Create or select a recognition rule supported by Advanced Revenue Management].
4. In the **Revenue Recognition Rule** form, select a NetSuite billing rule to associate the OpenAir Revenue Recognition Rule with from the **Associated billing rule** dropdown list.
5. Select the **Use this rule with Advanced Revenue Management** option.
6. Click **Save**.

> **Note:** Revenue Recognition rules integrated to NetSuite must have a **Service** selected.

Configuring the Sales Order Integration

You can create sales orders in NetSuite and transfer them to OpenAir for projects that have been previously transferred to OpenAir. OpenAir creates a unique Customer PO for each sales order line item you enter in NetSuite along with associated billing rules and recognition rules.

Edit the billing rules and recognition rules in OpenAir and run them for the project. Create invoices in OpenAir and export them to NetSuite. Export revenue recognition transactions back to NetSuite and create journal entries.

> **Note:** The sales order integration creates billing rules and revenue recognition rules in OpenAir when the OpenAir customer PO corresponding to a NetSuite sales order line item is created, Billing rules are created in OpenAir. Revenue recognition rules are created only if there is a corresponding billing rule on the sales order line.
>
> The sales order integration does not update billing rules and revenue recognition rules associated with an already existing customer PO in OpenAir. If an already integrated sales order line item is updated in NetSuite, the billing rules and revenue recognition rules associated with the corresponding customer PO in OpenAir do not reflect the changes.

Refer to the following additional details about sales orders transfer to OpenAir as POs.

- Discounts or Markups as well as revenue recognition rules on sales order line items transfer to OpenAir along with specific billing rule information.
- A Description on a sales order line item transfers to OpenAir as the Charge Description. Ensure the NetSuite form you are using includes the Description field.
- An Item on a sales order line item transfers to OpenAir as an Activity on a time or fixed fee billing rule. Ensure the NetSuite item is marked for export to OpenAir.

- If you create billing rules, you can also create revenue recognition rules similar to the ones you would create for a project in OpenAir. In addition, use custom mappings that transfer back to NetSuite. See Customizing Revenue Recognition Rules.
- The integration can add invoice details from the OpenAir project and user records to a NetSuite invoice. Create a Description Item in NetSuite and OpenAir and select a check box on the OpenAir Activities form. Set up custom mappings to specify the information you would like transferred and displayed on your NetSuite invoice.
- OpenAir invoices that transfer to NetSuite can have an initialize step during which the integration gathers information from the NetSuite sales order and incorporates it into the exported invoice in a way similar to the Next Bill functionality in NetSuite. Invoice information can then be edited on the NetSuite invoice.

Enabling the Sales Order Feature in NetSuite

To enable the Sales Order feature in NetSuite:

1. In NetSuite, do one of the following:
   - Go to Setup > Company > General Preferences (Custom Preferences), if you are using the SuiteScript 1 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
   - Go to Customization > Lists, Records and Fields > Records > [*View list forOpenAir: SRP Integration Settings*] > [*View OpenAir SRP Integration Settings Record*] , if you are using the SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
2. Select the check boxes for the following:
   - Export SO Line Default.
   - Enable billing rules for SO discounts. This requires NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) Version 1.15 or higher and allows you to assign fixed fee billing rules to discount items in NetSuite on a per project basis.
   - Create rev rec rules on sales orders. This requires NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) Version 1.25 or higher and allows you to create revenue recognition rules in OpenAir when you create NetSuite Sales Orders and corresponding billing rules.
3. Mark all sales order lines that you want to transfer with a valid OpenAir billing type rule: time, fixed fee % complete, or fixed fee on date or milestone.
4. Mark all sales order lines that you want to transfer with a valid OpenAir revenue recognition type rule: % complete, as billed, fixed amount % complete, fixed amount on date or milestone, incurred vs. forecast, or time. You can only have one time revenue recognition rule per project.

Note: Please note the following:

- The Sales Order integration does not use an Export to OpenAir check box on either the sales order header or sales order line, but items have an Export to OpenAir check box. This needs to be selected if you are going to use the item in a sales order you transfer to OpenAir.
- For invoicing, ensure your company has enabled A/R and A/P. Go to Setup > Company > Enable Features > Accounting and verify check boxes for A/R and A/P are selected.

Enabling the Sales Order Feature in OpenAir

There are a number of sales order related features you can choose to enable in the integration. Refer to the following:

To enable the Sales Order feature in OpenAir:

1. On the NetSuite Connector administration form and on the relevant workflow settings forms in OpenAir, check all or some of the following boxes:
    - NetSuite Connector administration — Imports:
        - □ Customer POs (Sales Orders)
    - Customer POs (Sales Orders) import workflow settings form:
        - □ Create time billing rules in sales order integration with a service filter. Map sales order line description to billing rule Charge description
        - □ Enable recognition rule creation in sales order integration
    - Invoices (Invoices) export workflow settings form:
        - □ Adjust quantity based on % invoiced for time charges related to NetSuite sales order lines with quantities that do not reflect hours
        - □ Always respect the NetSuite sales order customer and project when exporting invoices — Allows the transfer of NetSuite sales order information such as Sales Team when you export invoices back to NetSuite.
        - □ Create billing rules for NetSuite discount and markup lines on sales orders
        - □ Enable charges to map to NetSuite description items
        - □ Initialize invoices from corresponding sales orders when exporting invoices to NetSuite
2. In the NetSuite Connector UI, add the following workflows on the Mapping tab:
    - **Customer POs (Sales Orders)** import.
    - **Invoices (Invoices)** export, if you are going to export invoices back to NetSuite.
    - **Recognition Transactions (Revenue Recognition Transactions)** import, if you are going to export revenue transactions back to NetSuite and create related journal entries.

Implementing the Sales Order Integration

You can send NetSuite sales orders to OpenAir using one of the OpenAir Billing Rules: Time, Fixed fee % complete, or Fixed fee on date or milestone. In addition, you can select one of the following OpenAir Revenue Recognition Rules: % complete, as billed, fixed amount on % complete, fixed amount on date or milestone, incurred vs. forecast, and time. You must select a billing rule in order to transfer a revenue recognition rule.

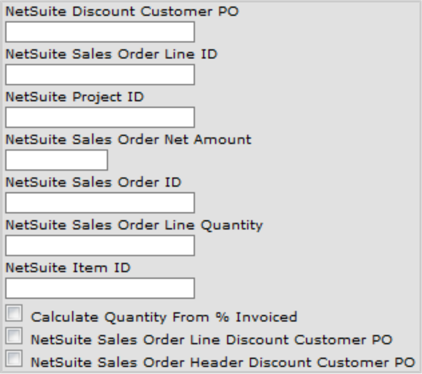
- Generate charges using the Customer PO either by running billing rules in the Projects application or creating charges in the Invoices application. Use the charges you generate to create invoices. Then, you can export the invoices back to NetSuite.
- Recognize revenue by running the revenue recognition rules. Transfer revenue recognized for a project back to NetSuite through the integration. See Customizing Revenue Recognition Rules.

There are several enhancements that allow greater control of the final NetSuite and OpenAir invoice. They include the following.

- When you use the feature to calculate quantity from % invoiced on the NetSuite sales order with a time billing rule, the invoice shows a percentage of the quantity instead of hours when it exports to NetSuite. See Calculating Quantity from % Invoice.
- When you transfer a sales order to OpenAir, the NetSuite company and project information, such as the Sales Team, exports with the invoice and displays in NetSuite. See Exporting NetSuite Sales Order Information with OpenAir Invoice.

- When you use NetSuite discounts and markups on a NetSuite sales order and assign a fixed fee billing rule, the sales order transfers to OpenAir with those discounts and markups. The integration creates a related billing rule and Customer PO that have a monetary value equal to the total value with the discount or markup. You can edit the billing rule and Customer PO and use them to generate an OpenAir charge. The discount or markup charge is based on the billing rule when billing is run for the project. The generation of that charge is based on the amount of the billing rule, like other fixed-fee billing rules. Therefore, you need to adjust the billing rules so that you generate the correct discount amount. When you export the invoice back to NetSuite, the amount includes the discount or markup. Both the OpenAir and NetSuite invoices match. You can ensure the invoice created in OpenAir includes the negative charge that corresponds to the discount and the NetSuite sales order discounts and markup lines are not applied after the invoice exports to NetSuite. Creating OpenAir Billing Rules for NetSuite Discounts and Mark Ups.
- When you use the feature to add detailed information from the OpenAir charges to the description item in the NetSuite invoice, additional charge or slip information about the project and users transfers to the NetSuite invoice. See Exporting OpenAir Charge Detail to NetSuite Description Item.
- When you use the feature to initialize the invoice, the integration gathers information from the NetSuite sales order, places it on the OpenAir invoice and then exports that invoice to NetSuite. This functions like the Next Bill button functions in NetSuite. You open the exported invoice in NetSuite, edit it, and save it. It contains more information than your OpenAir invoice displays. See Initializing an OpenAir Invoice.
- When you include a Description on a sales order item, it transfers to OpenAir as a Charge Description. The Item transfers to OpenAir as an Activity.

Once you enable the features you want in the Sales Order integration and transfer sales orders from NetSuite to OpenAir, custom fields display on the Edit Customer PO form in OpenAir. See below.



NetSuite Discount Customer PO

NetSuite Sales Order Line ID

NetSuite Project ID

NetSuite Sales Order Net Amount

NetSuite Sales Order ID

NetSuite Sales Order Line Quantity

NetSuite Item ID

☐ Calculate Quantity From % Invoiced
☐ NetSuite Sales Order Line Discount Customer PO
☐ NetSuite Sales Order Header Discount Customer PO

Importing NetSuite Sales Orders and Creating OpenAir Invoices

Create sales orders in NetSuite and run the integration to import a Customer PO to OpenAir. To view a Customer PO in OpenAir, go to Administration > Application Settings > Invoices > Customer POs. Click on a Customer PO.

You can also ensure a Customer PO is associated with a project by adding it to the available POs in that project. Go to Projects > Project and select the project designated on the sales order. Click Edit project or Properties to display the project properties form. Scroll down to Customer POs, and click the Edit link. A pop-up window displays available Customer POs. Select the PO, click Add selected, and click OK to add the PO to the project.

From the Customer PO, you can generate an associated billing rule or charge to create sales order charges as well as revenue recognition rules. You must generate billing rules in order to generate revenue recognition rules. These billing rules and recognition rules displays under the Financials tab in the project for which the sales order was created.

- Use the charges to create an OpenAir invoice. When you run the integration, you export the invoices back to NetSuite.
- Use the recognition rules to recognize revenue. When you run the integration, you export the recognized revenue back to NetSuite. See Customizing Revenue Recognition Rules.

Refer to the following procedures for importing NetSuite sales orders to OpenAir Customer POs so you can generate invoices and export them to NetSuite. They include creating charges in both the OpenAir Projects and Invoices applications as well as producing OpenAir invoices:

- Importing Sales Orders from NetSuite to Customer POs in OpenAir
- Creating Charges in the OpenAir Projects Application
- Creating Charges in the OpenAir Invoices Application
- Generating Invoices in OpenAir and Exporting Them to NetSuite

Importing Sales Orders from NetSuite to Customer POs in OpenAir

Use this procedure to transfer Sales Orders from NetSuite to Customer POs in OpenAir.

To import Sales Orders from NetSuite to Customer POs in OpenAir:

1. In NetSuite, go to Lists > Accounting > Items and either create a new Item or edit an existing one. Ensure you select a Default OpenAir Billing Rule and the check box to Export to OpenAir. Under the Basic tab, select the Income Account using the drop-down list as well as the Tax Schedule, and Deferred Revenue Account.

   > **Note:** You may create a time billing rule and transfer it to use in the OpenAir New other rate charge form in the Invoices application. Time billing rules can be a daily rate charge as well as an hourly charge, such as one you might use in a time billing rule in the Projects application.

2. In NetSuite, go to Transactions > Sales > Enter Sales Orders and enter sales order information. The customer and project must already have been transferred to OpenAir.
3. Select an item, one designated for export to OpenAir, from the drop-down list and fill in the information. Click in the OpenAir Billing Rule column and select an OpenAir billing rule from the drop-down list.

   > **Note:** An OpenAir billing rule is not the same as the NetSuite Billing Schedule.

4. If you are using the revenue recognition rule integration, select a OpenAir: Rev Rec Rule from the drop-down list.
5. Ensure the status of the NetSuite Sales Order is pending fulfillment or pending billing.

   > **Note:** When you export an invoice back to NetSuite, it only exports if the sales order is an approved sales order and the sales order is still available for billing. If it is not approved or has been fully billed, you receive an error message.

6. Run the integration. Refer to Working with Integrated Records in NetSuite and OpenAir.

Creating Charges in the OpenAir Projects Application

Use this procedure when you are going to run time billing or fixed fee billing rules available under the Financials tab in the Projects application.

To create charges in the OpenAir Projects application:

1. In OpenAir, go to Projects and select a project. Go to Financials > Billing. Click on the billing rule for the Customer PO to display the information. Edit the dates on the billing rule.

   > ⚠ **Important:** You must change the default date on the time billing rules. It is intentionally set to 01/01/50.

2. If it is a time billing rule, under Billing rates, select the source to “Get the billing rates from.” When the billing rule is a type “O” or other rate, select a rate card or some other non-hourly rate.
3. Click Run billing. Generally, it is not necessary to enter dates in the Date filter form. Click Next to create project billing transactions and charges.
4. Click the button to Create project billing transactions and charges. The charges transfer from the Projects application to the Invoices application in OpenAir. See Generating Invoices in OpenAir and Exporting Them to NetSuite for instructions on using charges to produce an OpenAir invoice and export it NetSuite.

Creating Charges in the OpenAir Invoices Application

Use this procedure when you are going to create a one-off charge. You can use the Invoices application to create charges based on a new other rate as well as a new hourly or flat rate charge. Remember to associate it with the correct Customer PO.

To create charges in the OpenAir Invoices application:

1. In OpenAir, go to Invoices > Charges and select New other rate…
2. Select the Customer : Project using the drop-down list. Select the Activity and User.
3. Select the Customer PO, the one that you imported from the NetSuite Sales Order.
4. Enter Date, Quantity, Rate, and additional Currency and time information.
5. Type Description and Notes.
6. Select the charge stage as Invoice Ready and click Save.

   The charges transfer from the Charges tab to the Invoices tab. See Generating Invoices in OpenAir and Exporting Them to NetSuite for instructions on using charges to produce an OpenAir invoice and export it NetSuite.

Generating Invoices in OpenAir and Exporting Them to NetSuite

Use this procedure to generate an invoice in OpenAir from the charges you created with the Customer PO in either the Projects application or the Invoices application.

To generate invoices in OpenAir and export them to NetSuite:

1. Go to Invoices > Invoices > New invoice. Use the drop-down list to select the Customer : Project : NetSuite Sales Order #. The charges you just generated display in the list view.
2. Select the check boxes of the charges you want to include in the invoice and click the Save button. The invoice is created and displays. It includes the Customer PO and NetSuite Sales Order # as well as other company and customer information. The charges are listed.
3. Click Submit/Approve and ensure the OpenAir invoice is approved.

4. Run the integration or click the Export to NetSuite link. It displays on the invoice form in tips for the approved invoice. Refer to Working with Integrated Records in NetSuite and OpenAir.

> **Note:** If you are exporting invoices to NetSuite through the integration, please note the following:
>
> - If the **Initialize invoices from corresponding sales orders when exporting invoices to NetSuite. This is used with the sales order integration** setting is enabled, the export message indicates that the invoice # is initialized from the NetSuite sales order # and that the export succeeded. If you have enabled importing taxes, the taxes applied are also listed.
> - Charges are grouped automatically by the NetSuite sales order line and not just a sales order ID. A NetSuite sales order can have multiple lines on it and each of those lines becomes a unique customer PO in OpenAir. Also, if the charges on an invoice reference two unique customer POs in OpenAir, the resulting invoice in NetSuite will also have two lines, and each line corresponds to a line on the NetSuite sales order. However, one invoice can only reference one NetSuite sales order so multiple sales orders are not referenced on a single invoice.
> - It is not required that you reference a sales order line on an invoice, but if you do, you can only reference a sales order line one time.
> - You can send an invoice that has more than one charge that references a single sales order line, but those charges are rolled up into one when the invoice is exported to NetSuite. The OpenAir invoice retains the detail. And if the rates used on the charges that are rolled up are different, rates are calculated based on the quantity and amount after roll up.
> - You can close out sales order lines when you send the invoice to NetSuite, indicating the quantity has been billed.

Calculating Quantity from % Invoice

Use this feature for time billing rules when you want to use hours as a container of time and bill a percentage of the whole amount instead of individual hours.

To calculate quantity from % invoice:

1. In NetSuite, go to Transactions > Sales > Enter Sales Orders. Create a new sales order.
2. Select an item and mark the check box to get the OpenAir Quantity as % Invoiced. Fill in the sales order information.
3. Run the integration. Refer to Working with Integrated Records in NetSuite and OpenAir.
4. Go to Administration > Application Settings > Invoices and select Customer POs. Scroll down and ensure the check box is selected to Calculate Quantity From % Invoiced.
5. Go to Projects > Projects and select the project the sales order is associated with.
6. Go to Financials > Billing. Edit the date of the billing rule to the current time frame.

7. Run billing, generate transactions, and create an invoice for the Customer PO that had the check box selected to Calculate Quantity From % Invoiced. Approve the invoice.
8. Run the integration. The invoice you export back to NetSuite contains the desired information.

Creating OpenAir Billing Rules for NetSuite Discounts and Mark Ups

Use this feature to edit billing rules related to NetSuite service order discounts and markups so that accurate charges are included in invoices generated in OpenAir. This ensures that both the NetSuite invoice and the OpenAir invoice reflect the same discounts or markups. Billing rules and Customer POs transfer with NetSuite service order discount and markup information to OpenAir and then transfer back to NetSuite when you export the invoice. You are able to create negative charge amounts in OpenAir as needed. While the integration automatically handles discounts and markups, this feature allows finer manual control of that process.

To create OpenAir billing rules for NetSuite discounts and mark ups:

1. In NetSuite, go to Transactions > Sales > Enter Sales Orders. Create a new sales order.
2. Select an item that is either a discount or markup. Select either a Fixed fee % complete, Fixed fee on date or milestone, or Do not export billing rule.

   > ⚠ **Important:** Ensure you have selected the NetSuite custom preference to enable billing rules for SO discounts. Otherwise, a discount item is not permitted.

3. Approve the sales order.
4. Run the integration. Refer to Working with Integrated Records in NetSuite and OpenAir.
5. Go to Administration > Application Settings > Invoices and select Customer POs. Scroll down and ensure one or both of the check boxes is selected: NetSuite Sales Order Line Discount Customer PO and/or NetSuite Sales Order Header Discount Customer PO.
6. Go to Projects > Projects and select the project the sales order is associated with.
7. Go to Financials > Billing. Edit the billing rule as necessary to reflect the desired discount or markup information.

   > ⓘ **Note:** If discounts should be applied over several periods or as a percentage of billed time, adjust the billing rule accordingly.

8. Run billing, generate transactions, and create an invoice for the Customer PO that had the discount check box selected. Approve the invoice.
9. Run the integration. The invoice you export back to NetSuite contains the desired information.

   > ⓘ **Note:** If you create a sales order to export to OpenAir and there is a discount/markup line following a sales order line, the discount/markup line also exports to OpenAir as a Customer PO and a billing rule is created even though you may have selected the rule: Do not export. This discount/markup line applies to the line above it and is needed to prevent user errors.

Exporting NetSuite Sales Order Information with OpenAir Invoice

When you transfer a NetSuite sales order to OpenAir and create an invoice from it in OpenAir, the NetSuite customer and project information, such as the sales team, can transfer back to NetSuite when you export the invoice from OpenAir. This is similar to the functionality that exists when creating an invoice from a sales order in NetSuite.

To export sales order information with an OpenAir invoice:

1. In NetSuite, go to Transactions > Sales > Enter Sales Orders. Create a new sales order and identify the Sales Team.
2. Approve the sales order.
3. Run the integration. Refer to Working with Integrated Records in NetSuite and OpenAir.
4. In OpenAir, run billing, generate transactions, and create an invoice for the sales order Customer PO. Approve the invoice.
5. Run the integration. The invoice you export back to NetSuite contains the desired information.

> **Note:** Normally, customer and project information come from the OpenAir invoice. When you use this feature, this information comes from the NetSuite sales order at both the header and line levels. If there is a discrepancy between the two invoices, adjust the OpenAir invoice manually.

Exporting OpenAir Charge Detail to NetSuite Description Item

When exporting invoices from OpenAir that are associated with a NetSuite sales order for time or fixed fee billing, you can specify additional information to transfer from OpenAir to the NetSuite invoice. First you create or identify a Description item in NetSuite. Next, you create or identify an existing activity (service/category/item) in OpenAir and select a check box that indicates it is a NetSuite description item used for charges in the invoice export. Then, you set up custom mapping to identify the fields and values you would like to transfer. See Advanced Mapping and Filter Notation Reference and Mapping Description Items When Exporting Invoices.

To export OpenAir charge detail to a NetSuite description items:

1. In NetSuite, go to Lists > Accounting > Items. At the bottom of the list form, find Type and select Description from the drop-down list. This displays the Internal IDs of Description items in your account. Either create a new item or identify the one you want to use and make note of the Internal ID. You will manually add this Internal ID number to an Activity in OpenAir.
2. In OpenAir, go to Administration > Application Settings > Invoices > Activities. Create a new Activity with the name of the Description item from NetSuite.
3. Scroll down the form and enter the Internal ID of the Description item from your NetSuite account in the text box on the OpenAir form for NetSuite Item ID.
4. Continue scrolling down the form and select the check box for: NetSuite description item used for charges in invoice export. This custom field is added to the Activity form when the feature is enabled.
5. In NetSuite, go to Transactions > Sales > Enter Sales Orders. Create a new sales order, enter items, approve it, and export it. You do not enter the NetSuite Description item as one of the items on the sales order.
6. Run the integration. Refer to Working with Integrated Records in NetSuite and OpenAir.
7. In OpenAir, edit the billing rule date, run billing, generate transactions, and create an invoice for the sales order Customer PO. Approve the invoice.
8. In OpenAir, go to Administration > Integration: NetSuite. Scroll down to the Advanced Settings section and type the custom mappings for the information you would like to transfer. See Mapping Description Items When Exporting Invoices. Click Save.

> **Note:** This feature requires that additional custom mappings be specified. The integration must receive the specific definitions you would like to have displayed as description items. Examples of such mappings are provided in Mapping Description Items When Exporting Invoices.

9. Run the integration. Refer to Working with Integrated Records in NetSuite and OpenAir.
10. The invoice transfers to NetSuite and displays the Description item along with the additional information you specified in the Advanced Settings section.

Initializing an OpenAir Invoice

You can enable a feature in the integration that ensures the invoice you export from OpenAir contains information from the associated NetSuite sales order. This functionality is similar to creating a sales order in NetSuite and selecting the Next Bill button. The resulting NetSuite invoice contains pertinent sales order information. Now, when you initialize an OpenAir invoice upon export, the integration gathers information from the NetSuite sales order, adds it to the OpenAir data that is being exported, and sends it all to the NetSuite invoice. You can edit this information in NetSuite and save it just as you do with an invoice created by the Next Bill button. Custom mapping are available to map custom field values from the service sales order to the invoice. See Advanced Mapping and Filter Notation Reference and Mapping Service Sales Order Custom Fields When Exporting Invoices.

Customizing Revenue Recognition Rules

If you are not familiar with Advanced Mapping and Filter Notation Reference, you may want to take a few minutes and review it. This section assumes that you understand the following:

- Integration Point Tags
- Field Declarations
- Advanced Field Mapping Definition

The types of revenue recognition rules you can create in NetSuite on the sales order form are similar to the recognition rules you can create for a project in OpenAir. To use this integration, you select one of the types from a custom column on the NetSuite sales order form. Remember, in order to export a revenue recognition rule from a NetSuite sales order to an OpenAir PO, you must select both a billing rule and one of the supported revenue recognition rule types on the sales order line item.

The NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) creates the OpenAir: Revenue Rules (`customrecord_oa_rev_rec_rules`) custom entity and records in NetSuite defining the supported revenue recognition rule types. The OpenAir: Revenue Rules (`customrecord_oa_rev_rec_rules`) records are listed in the following table.

> **Note:** The **Internal ID** and **OpenAir: Type** must be set as listed in the following table for the integration to work correctly. The revenue recognition rules may not integrate from OpenAir to NetSuite, and the integration may return in error if the revenue recognition rule types are not defined as listed in this table.

Internal ID	Name	OpenAir: Type
1	% Complete	P
2	As Billed	A
3	Fixed Amount for % Complete	FP

Internal ID	Name	OpenAir: Type
4	Fixed Amount for Date or Milestone	FD
5	Incurred vs. Forecast	I
6	Time	T

Each revenue recognition rule type has its own mapping syntax. For the most part, you can map most of the fields on the revenue_recognition_rule table. See the OpenAir data dictionary: `https://<account-domain>/database/single_user.html#revenue_recognition_rule`. Since some of the options normally available on the web application revenue recognition rule form are not actual fields in the revenue_recognition_rule table, custom mapping allows you to set those options through the integration to either override defaults or set other fields. Refer to the following tables to review available custom mappings for each rule type. Then, take a look at the sample code for the OA_FIELDS section. See Code Example for OA_FIELDS.

As Billed Rule

The following custom mappings are available for as billed revenue recognition rules.

Fields from revenue_recognition_rule table	Additional form fields	Correlation with form
All fields from the revenue_recognition_rule table	inc_time	Populate hours in recognition transactions check box

Fixed Amount Rule

You can use all fields from the revenue_recognition_rule table except the following: project_task_id, category_id, category _1_id, category _2_id, category _3_id, category _4_id, category _5_id, cost_center_id, agreement_id, and customerpo_id.

That is because these values are populated in the repeated rows on the form. To view an example of this form in OpenAir, go to Projects > Financials > Recognition and select New > recognition rule.

The following custom mappings are available for fixed amount for % complete as well as fixed amount for date or milestone revenue recognition rules.

Fields from revenue_recognition_rule table	Additional form fields	Correlation with form
Use the form fields to customize the mapping of the first amount row on the revenue recognition rule.	recognition_type_1	Rule Type
	acct_code_1	Accounting code
	amount_1	Amount (money)
	project_task_id_1	Project task
	category_id_1	Service
	category_1_id_1	Service 1
	category_2_id_1	Service 2
	category_3_id_1	Service 3
	category_4_id_1	Service 4

Fields from revenue_recognition_rule table	Additional form fields	Correlation with form
	category_5_id_1	Service 5
	cost_center_id_1	Cost center
	agreement_id_1	Agreement
	customerpo_id_1	Customer PO

Incurred vs. Forecast Rule

The following custom mappings are available for incurred vs. forecast recognition rules.

Fields from revenue_recognition_rule table	Additional form fields	Correlation with form
All fields from revenue_recognition_rule table	inc_approved_time	Approved hours multiplied by user cost check box
	inc_approved_ticket	Approved receipts check box
	inc_fulfilled_purchase_items	Fulfilled purchase items check box
	fore_estimated_time	Estimated hours multiplied by user cost check box
	fore_approvedbooked_time	Approved and booked hours multiplied by user cost check box
	fore_expense	Total planned receipts: (money) check box
	fore_expense_total	Total planned receipts: (money) textbox, the amount
	fore_purchase	Total planned purchase items: (money) check box
	fore_purchase_total	Total planned purchase items: (money) textbox, the amount

Time Rule

All fields from the revenue_recognition_rule table are available for custom mappings in time revenue recognition rules.

Code Example for OA_FIELDS

For custom mappings available for the revenue recognition rule feature, you use an OA_FIELDS section and reference them as you would normally reference true fields. However, since these are not true fields, these form fields are not available in lookups. Refer to the following:

```
<REVREC_RULE_AS_BILLED>
    OA_FIELDS    inc_time, acct_code
    inc_time     '0'
    acct_code    'TESTCODE'
</REVREC_RULE_AS_BILLED>
```

(DEPRECATED) Configuring the Salesforce.com Integration for Customers and Projects

Important: The 3-way Salesforce, OpenAir and NetSuite integration for customers and projects is deprecated and will be removed at a future date to be announced.

You should use the Projects (Projects) and Customers (Customers) export workflows instead. See Replacing the Deprecated 3-Way Salesforce, OpenAir and NetSuite Integration With the Customers (Customers) or Projects (Projects) Export Workflows.

The **Enable 3-way Salesforce.com, OpenAir and NetSuite integration** setting has been moved to the Legacy compatibility section of the NetSuite Connector administration form. You may continue using the 3-way Salesforce, OpenAir and NetSuite integration if it is already enabled for your account. However, it can no longer be enabled.

When you use the Salesforce.com integration with OpenAir, you can pass customer and project information captured in Salesforce.com through OpenAir to NetSuite. A Salesforce.com ID is transferred to NetSuite as part of creating the entity in NetSuite. Once the customer and project are in NetSuite, they are assigned a NetSuite ID, which is subsequently added to the OpenAir customer and project record. While information can be updated and transferred between OpenAir and Salesforce.com, those updates cannot be exported back to NetSuite. To make changes to the NetSuite record, either manually update it in NetSuite, or if necessary, unexport it from OpenAir and then manually delete the record in NetSuite. Unexporting the record in OpenAir allows you to re-export the record to NetSuite from OpenAir. If you unexport the OpenAir record, you must delete the NetSuite ID from the OpenAir custom field and save it. Then, you can make changes to the OpenAir record either manually or from Salesforce.com and export the OpenAir record back to NetSuite. See Unexporting a NetSuite Customer or Project.

For this integration, customers and projects must have a Salesforce ID and cannot already have a NetSuite ID. The following custom field for CUSTENTITY_OA_SFA_ID is created on the NetSuite Customers edit window and Projects edit window.

Unexporting a NetSuite Customer or Project

Once a customer has been transferred from Salesforce.com through OpenAir to NetSuite, there may be a case when you need to delete the customer or project record in NetSuite and transfer the record again from OpenAir. To do this, delete the record in NetSuite, delete the NetSuite ID from the customer or project edit form in OpenAir, go to the OpenAir list view and mark the customer or project for unexport. Run an action in OpenAir and ensure you select the check box to unexport the entity. In the hints of the customer or project edit form in OpenAir, the export link displays again: Export to NetSuite. To transfer the customer or project record again to NetSuite, either click the link or run the integration.

Important: Unexporting a customer or project should only be done with care. Do not use this as a way of passing regular updates to NetSuite. NetSuite is still the system of record in the integration and changes should be made to the NetSuite record.

To unexport a NetSuite Customer or Project:

1. In OpenAir for customers, go to Administration > Global Settings > Customers > Customers and select the customer you want to unexport. For projects, go to Projects > Projects and select the project you want to unexport.

2. Delete the NetSuite ID from the project or customer properties form.

> **Note:** Before you run the integration, ensure the OpenAir project or customer properties forms do not display a NetSuite ID. You must delete it before running the integration.

3. In NetSuite, go to Lists > Relationships and select the customer or project. Click the Edit link for the customer or project.
4. Click Delete and confirm the action.
5. In OpenAir for customers, go to Administration > Global Settings > Customers > Customers. For projects, go to Projects > Projects.
6. On the list view, select the check box next to the customer or project you would like to unexport.

> **Note:** If a Run icon does not display on the list view, customize the list view. Click on the arrow at the top left in the title bar. When a List layout pop-up window displays, select Run an action from Available Items and click Add selected items. Click OK.

7. Click the Run icon. When an Actions pop-up window displays, select the check box for Unexport the customers. Click OK. The NetSuite customer record is unexported.
8. Make changes to the OpenAir record. You can export the customer or project again back to NetSuite using the normal export procedures.

> **Note:** NetSuite assigns a new NetSuite ID number to the customer or project. Future expenses and invoices related to the re-exported customer or project will be associated with this new NetSuite ID number.

Configuring the Tax Export Integration

See Enable tax export on expense reports.

The structure and content of this guide are currently under review. The review aims to consolidate information and create self-contained reference for each integration workflows and other integration features. This topic has been removed and replaced with the topics referenced above.

Configuring Tax Override for Canadian Expense Reports

See Allow manual override of Canadian taxes on Expense report export.

The structure and content of this guide are currently under review. The review aims to consolidate information and create self-contained reference for each integration workflows and other integration features. This topic has been removed and replaced with the topics referenced above.

Configuring the Time Entries Export Integration

The time entries export integration allows you to transfer time from your OpenAir timesheet to NetSuite. By default, time transfers to NetSuite as non-billable. To transfer it as billable time, you must use a custom field and designate it as billable. An OpenAir custom field captures the NetSuite internal ID of each exported time entry. As with other integration features, you can export all timesheets if you select the check box on the NetSuite integration form for Time Entries (Time) or individually using the link that displays in tips on the bottom of the OpenAir timesheet report. This feature does not connect time entries to the NetSuite invoice but to NetSuite time tracking.

Since some companies do not require an activity or service item to be specified on time entries, you also have the option of exporting time entries to NetSuite without an activity or service item. If there is a service item on the OpenAir time entry, it is transferred to NetSuite.

If required, OpenAir Customer Support can assist you in distinguishing time as billable or non-billable using a custom field on the Timesheets grid. However, since most time is billed in OpenAir, you should proceed with caution.

To enable Time Entries Export in NetSuite:

1. Go to Setup > Company > Enable Features > Employees and select the check box for Time Tracking under Time & Expenses.
2. Make sure you are using the latest version of the NetSuite OpenAir SRP Integration bundle 369637 (previously 2851). Custom fields are added to your account for time entry entities. If the following NetSuite custom field does not exist, create a free-form text field for custcol_oa_time_entry_id.

To enable Time Entries Export in OpenAir:

1. On the NetSuite Connector administration form under the Exports section, check the **Time Entries (Time)** box.
2. (Optional) On the Time Entries (Time) workflow settings form, check the **Do not require a service on time entry export to NetSuite** box. This ensures time entries export with or without an activity or service item.
3. In the NetSuite Connector UI, add the following workflows on the Mapping tab:
   - **Time Entries (Time)** export.
4. Go to Administration > Application Settings > Timesheets > Other settings. Check the following boxes:
   - Require customer/project on time entries
   - (Optional) Require an activity on time entries
   - Show the customer/project drop-down on the main grid
   - Show the activity drop-down on the main grid.
5. If you require that some time entries export to NetSuite as billable in NetSuite, contact OpenAir Customer Support and request the following account configuration option:

   List of time entry custom fields to appear to the right of the standard popups on timesheet grid (e.g. custom_4, custom_12) — This holds the information for the custom fields created when the time entries export is enabled.

Implementing the Time Entries Export Integration

Once you enable the time entries feature and the time entry custom fieldsare configured for your account, two fields display on the Timesheets application in OpenAir: NetSuite Billable Time (with Billable and Non-Billable selections) and NetSuite Time Entry ID. When you export the timesheet, the information for each time entry on the OpenAir timesheet transfers to NetSuite. If for some reason a time entry on an OpenAir timesheet does not export, you can fix the error and re-export it. There is no possibility of duplicating time entries that did export to NetSuite. Only those time entries that failed are allowed to export again. The Export to NetSuite link in tips only displays as long as at least one time entry on a timesheet failed. When all time entries are transferred, the link changes and reads: This timesheet was exported to NetSuite.

To set up exporting time entries to NetSuite:

1. In OpenAir, go to Timesheets > Timesheets > New and create a timesheet.
2. Approve the timesheet through the normal OpenAir approval process.
3. Export the timesheet to NetSuite either by running the integration or using the Export to NetSuite link in tips. See Exporting OpenAir Data to NetSuite.

Configuring Work In Progress Accounting and Intercompany Journal Entries

In OneWorld accounts, you now have the ability to transfer revenue and billing between subsidiaries while work is in progress. Create Projects in NetSuite and transfer them to OpenAir. Establish billing rules in OpenAir, generate charges, create invoices, and export them to NetSuite. When payment occurs, transfer payment information to OpenAir. Recognize revenue in OpenAir and export it to NetSuite. Run a scheduled script to generate intercompany journal entries that show incremental revenue accrual on balance sheets while work is in progress.

> ⚠ **Important:** You cannot use this feature with the sales order integration. At this time, these features are incompatible.

When this feature is enabled, projects display the NetSuite subsidiary on the OpenAir Project Properties form as well as the billing rules and revenue recognition rules forms. When configuring the integration to import NetSuite subsidiaries, you specify a category_x field (such as Service_1, Service_2, or Service_3) to which NetSuite subsidiaries should be mapped.

To enable Work In Progress Accounting and Intercompany Journal Entries in NetSuite:

1. Contact NetSuite Customer Support and ensure you are using the latest version of the NetSuite OpenAir SRP Integration bundle 369637 (previously 2851). Then, request that the OpenAir: WIP Accounting bundle 369609 (previously 17953) version 1.0 or higher be installed in your OneWorld account.

   > ⓘ **Note:** This feature is only available in OneWorld accounts.

2. Go to Setup > Company > General Preferences > Custom Preferences.
   - Select the check box to: Enable WIP Accounting.
   - Type an email address of the person who should receive the logs from a schedule script run.
   - Select a user. When the logs are emailed, they appear to be sent from this user.
   - Set a value in the WIP Accounting Remaining Usage.

     > ⓘ **Note:** The WIP Accounting Usage Metering general preference is the lower limit threshold under which the scheduled script will rescheduled itself to run. To derive this number, calculate the number of usage units any single intercompany journal consumes. This requires you to understand the scripts that run to create one intercompany journal and the operations these scripts use. Once you figure out what one takes, the value of the WIP Accounting Usage Metering should be slightly higher than that number.

   - Click Save.
3. Schedule the script. Go to Setup > Customization > Script Deployments.
   - Change the Type filter on the bottom to Scheduled.

- Locate the script: customdeploy_oa_automate_icje and click Edit.
- Enter scheduling information. Or, to run the script now, click the arrow on the Save button and select Save and Execute.
- Emails are sent to the address of the person indicated. They include a link that displays the Intercompany Journal. Click OpenAir: WIP Accounting to see if the script created the Intercompany Journal from an Invoice or a Journal. Both are explained in following sections.

To enable Work In Progress Accounting and Intercompany Journal Entries in OpenAir:

1. Contact OpenAir Customer Support and request the following account configuration option:
   - Enable extended category feature.
2. On the NetSuite Connector administration form, check the following boxes:
   - Under Exports:
     - Recognition Transactions (Revenue Recognition Transactions)
   - Under Additional integrations:
     - Enable NetSuite intercompany journal creation for recognition transaction and invoice export
     - Enable NetSuite subsidiary import — Select the category to which NetSuite subsidiaries should be mapped
   - Under Other features:
     - Enable NetSuite OneWorld integration optimizations
3. In the NetSuite Connector UI, add the following workflows on the Mapping tab:
   - **Invoices (Invoices)** export.
   - **Recognition Transactions (Revenue Recognition Transactions)** export.
4. Ensure the following custom fields are created before you run the integration. Go to Administration > Global Settings > Custom Fields. When you enabled the NetSuite subsidiary import, you designated the category to which NetSuite subsidiaries should be mapped. Ensure that your selection displays under Association on the custom field form:
   - netsuite_ow_subsidiary_id
   - netsuite_ow_subsidiary_interco_acct_id
   - netsuite_ow_subsidiary_currency_id

Implementing Work in Progress Accounting and Intercompany Journal Entries

A number of intercompany journal entry (ICJE) fields are added to both OpenAir forms and NetSuite forms and hold related information. In addition, an integration script searches journal entries and invoice line items. Based on the specified criteria, it creates intercompany journal entries. Review the sections that follow for information on script criteria.

Using a Journal Entry to Generate an Intercompany Journal Entry

The following three conditions must exist for the integration to use a journal entry to generate an automatic intercompany journal entry. The integration automatically populates these fields based on the OpenAir revenue recognition transaction, so there is no user action. Generally, the fields are blank unless the subsidiary of the project on the OpenAir revenue recognition transaction is different than the subsidiary stamped on the revenue recognition transaction. In that case, the integration populates the fields on this table.

NetSuite Custom Field/Name on Form	Required Field Status
custbody_oa_create_icje OpenAir: Create ICJE	Must be checked
custbody_oa_to_subsidiary OpenAir: To Subsidiary	Must have a value
custbody_oa_auto_icje_error ICJE Error	Must NOT be checked

For each journal entry that meets the above criteria, the integration creates an intercompany journal entry and includes the following information:

- Subsidiary — subsidiary of the project
- To Subsidiary — the value is sourced from the OpenAir: To Subsidiary field on the original journal
- Exchange rate — system populated
- Created From Journal ID (custbody_oa_created_from_je_id) — internal ID of originating journal entry
- Created From Journal # (custbody_oa_created_from_je_number) — transaction number of originating journal entry
- Line level of the intercompany journal includes:
  - From subsidiary — debit revenue, credit deferred revenue (WIP) reverses the original journal entry
  - To subsidiary — credit revenue, debit deferred revenue (WIP)

  > **Note:** This moves the revenue from the project subsidiary to the subsidiary of the employees who are doing the work captured in the billing and recognition rules.

- Original journal entry includes the following:
  - OpenAir: Create ICJE field — unchecked
  - Created ICJE # (custbody_oa_created_icje_number) — the transaction number of the intercompany journal that was created
  - Created ICJE ID (custbody_oa_created_icje_id) — the internal ID of the intercompany journal that was created

Using Invoice Items to Generate an Intercompany Journal Entry

The following three conditions must exist for the integration to use invoice line items to generate an intercompany journal entry.

NetSuite Custom Field/Name on Form	Required Field Status
custcol_oa_create_icje	Must be checked

NetSuite Custom Field/Name on Form	Required Field Status
OpenAir: Create ICJE	
custcol_oa_icje_to_subsidiary OpenAir: To Subsidiary	Must have a value
custcol_oa_icje_error ICJE Error	Must NOT be checked

For each invoice line item that meets the above criteria, the integration creates an intercompany journal entry and includes the following information:

- Subsidiary — subsidiary of the project
- To Subsidiary — the value is sourced from the OpenAir: To Subsidiary field on the original invoice line
- Exchange rate — system populated
- Created From Invoice ID (custbody_oa_created_from_iv_id) — internal ID of the invoice that sourced this intercompany journal
- Created From Invoice Line # (custbody_oa_created_from_line_number) — line ID of the invoice that sourced this intercompany journal
- Line level of the intercompany journal includes:
  - From subsidiary — debit deferred revenue (WIP), credit intercompany clearing account
  - To subsidiary — credit intercompany clearing account, debit deferred revenue (WIP)

  > **Note:** This moves the amount billed on that invoice line from the project subsidiary to the subsidiary of the employees who are doing the work captured in the billing and recognition rules.

- Original invoice line item includes the following fields or columns:
  - OpenAir: Create ICJE field — unchecked
  - Created ICJE # (custbody_oa_created_icje_number) — the transaction number of the intercompany journal that was created
  - Created ICJE ID (custbody_oa_created_icje_id) — the internal ID of the intercompany journal that was created

Correcting Errors Logs Generated From Scripts

Scripts run on a NetSuite company schedule. When they run, a log is created and emailed according to the setup specified in Setup > Company > General Preferences > Custom Preferences > OpenAir: WIP Accounting. Monitor these logs for errors, correct the errors, and select the check box for OpenAir: Create ICJE so that the script can attempt to create an intercompany journal entry.

To correct errors:

1. Note the error message and make necessary corrections.
2. Clear the ICJE automation error check box and delete the ICJE automation error message text.
3. Select the check box for OpenAir: Create ICJE.

   During the next run of the scheduled script, this journal entry or invoice line will be evaluated against the criteria, and if it passes, the integration creates an intercompany journal entry.

Note: In NetSuite, scheduled scripts are subject to usage metering since their operations consume certain usage units. Scheduled scripts have an upper limit of 10,000 usage units on any single script run. If there are many intercompany journals to create, this upper limit will be reached. When this happens, the script reschedules itself to run at another time. A log is generated every time a script runs, including when the script is rescheduled.

The WIP Accounting Remaining Usage general preference is used to control when the script will be rescheduled on a run. If there are additional scripts that fire when intercompany journals are created, it is important to take these into account when setting the WIP Accounting Remaining Usage general preference. Once the remaining usage on the schedule script run falls below the threshold indicated by WIP Accounting Remaining Usage, the script will be rescheduled.

Contact OpenAir Professional Services if further customizations for the intercompany journals are desired. These customizations can be done using user event SuiteScript, which have operations that consume usage units. See NetSuite Help for more information on usage metering.

New Fields for Work in Progress Accounting and Intercompany Journal Entry

When you enable this feature, a number of new custom fields display in your NetSuite Account. There is a subtab for OpenAir: WIP Accounting on both the Journal and Intercompany Journal forms.

In addition, the following fields are also added:

NetSuite Custom Fields	Field Name	NetSuite form and Field Information
custrecord_oa_interco_acct	OpenAir: Intercompany Clearing Account	Subsidiary form - Must specify an Intercompany Clearing Account
custbody_oa_to_subsidiary	OpenAir: To Subsidiary	Journal form - List/Record
custbody_oa_icje_currency	OpenAir: ICJE Currency	Journal form - List/Record
custbody_oa_create_icje	OpenAir: Create ICJE	Journal form - Select check box to create intercompany journal entry
custbody_oa_created_from_je_number	Created From Journal #	Journal form - text
custbody_oa_created_from_je_id	Created From Journal ID	Journal form - text
custbody_oa_created_icje_number	Created ICJE #	Journal form - text
custbody_oa_created_icje_id	Created ICJE ID	Journal form - text
custbody_oa_auto_icje_error	ICJE automation error	Journal form - Clear check box to create intercompany journal entry in next script run
custbody_oa_auto_icje_error_msg	ICJE automation error message	Journal form - Clear error message to create intercompany journal entry in next script run
custbody_oa_created_from_iv_id	Created From Invoice ID	Journal form - text
custbody_oa_created_from_iv_number	Created From Invoice #	Journal form - text

NetSuite Custom Fields	Field Name	NetSuite form and Field Information
custbody_oa_created_from_line_number	Created From Invoice Line #	Journal form - text
custcol_oa_create_icje	OpenAir: Create ICJE invoice	Invoice form - Sales Item - Select check box to create intercompany journal entry
custcol_oa_icje_number	OpenAir: Created ICJE #	Invoice form - Sales Item - text
custcol_oa_icje_id	OpenAir: Created ICJE ID	Invoice form - Sales Item - text
custcol_oa_icje_currency	OpenAir: From Subsidiary Currency	Invoice form - Sales Item - Currency list/ record
custcol_oa_icje_clearing_account	OpenAir: ICJE Clearing Account	Invoice form - Sales Item - Subsidiary list/ record
custcol_oa_icje_wip_account	OpenAir: ICJE WIP account	Invoice form - Sales Item - Account list/ record
custcol_oa_icje_to_subsidiary	OpenAir: To Subsidiary invoice	Invoice form - Sales Item - Account list/ record

To set up Work in Progress Accounting and Intercompany Journal Entry:

1. In NetSuite, ensure the Customers, Employees, and Projects have the check box selected to Export to OpenAir. This feature works using the integration. Note the subsidiaries involved because using this feature requires you to edit the Subsidiary form as described in the next step.
2. Go to Setup > Company > Subsidiaries. Edit the form for each of the subsidiaries involved and choose an account for OpenAir: Intercompany Clearing Account. Click Save.
3. Go to Lists > Accounting Items and create a service for sale for the subsidiaries involved. Designate an Income Account and a Deferred Revenue Account for work in progress.
4. In OpenAir, run the integration. See Running the Integration.

   Depending on how your account is configured, the integration imports subsidiaries, employees, customers, projects, services, expense items, contacts, and payment, and exports invoices and recognition transactions.
5. In OpenAir, go to Projects > Projects and select one of the projects involved with this feature. Create tasks and assign employees/users. Select the Service item you created for work in progress. As work is performed on the project tasks, you can bill for that work.
6. In OpenAir Projects, go to Financials > Billing and create billing rules to capture the work in progress. These can be fixed fee or time billing rules.

   > **Note:** Select the Service line associated with the subsidiary.
7. Run billing and click the button to Create project billing transactions and charges.
8. Go to Invoices > Invoices and create a new invoice using those charges. Click the link to Export to NetSuite.
9. In NetSuite, go to Transactions > Sales > Create Invoices > List and locate the invoice you just exported from OpenAir. Click on edit to review the work in progress fields. Notice work reflects employees from different subsidiaries.
10. In OpenAir, go to Invoices > Invoices > Approved and select the invoice. Click the Payments link. Payments information displays. Click Save.
11. To recognize the revenue and transfer it to NetSuite, go to Projects > Projects and select the project. Go to Financials > Recognition.

12. Click New and select an As Billed rule. Select the Service as well as the Service line. Notice the Currency is already selected. Click Save. You can also create other types of recognition rules as fits your situation.
13. Click Run to run the recognition rule and click Next or enter Start date and End date.

    Recognition transactions from the previous invoice display.
14. Click the Create recognition transactions button.

    The total amount included in the recognition transactions displays.
15. Run the integration. The integration exports the revenue recognition transactions to NetSuite. They display in NetSuite as journal entries.
16. In NetSuite, go to Transactions > Financial > Make Journal Entries > List. The exported journal entries display. When the scheduled integration script runs, the journal entries generate intercompany journal entries.

    > **Note:** An email is automatically sent to the account you specified when you set up the feature. It contains a link to the intercompany journal entry. Click on it to display the NetSuite form. Click on the OpenAir: WIP Accounting tab to display information about the transaction such as the related invoice # or journal #.

Configuring the Support Case Integration

See Tasks (Support Cases) Import Workflow.

Configuring Purchaser Integration

See Enable purchaser integration.

The structure and content of this guide are currently under review. The review aims to consolidate information and create self-contained reference for each integration workflows and other integration features. This topic has been removed and replaced with the topics referenced above.

Configuring Integration for Percent Complete Rules with NetSuite Advanced Revenue Management

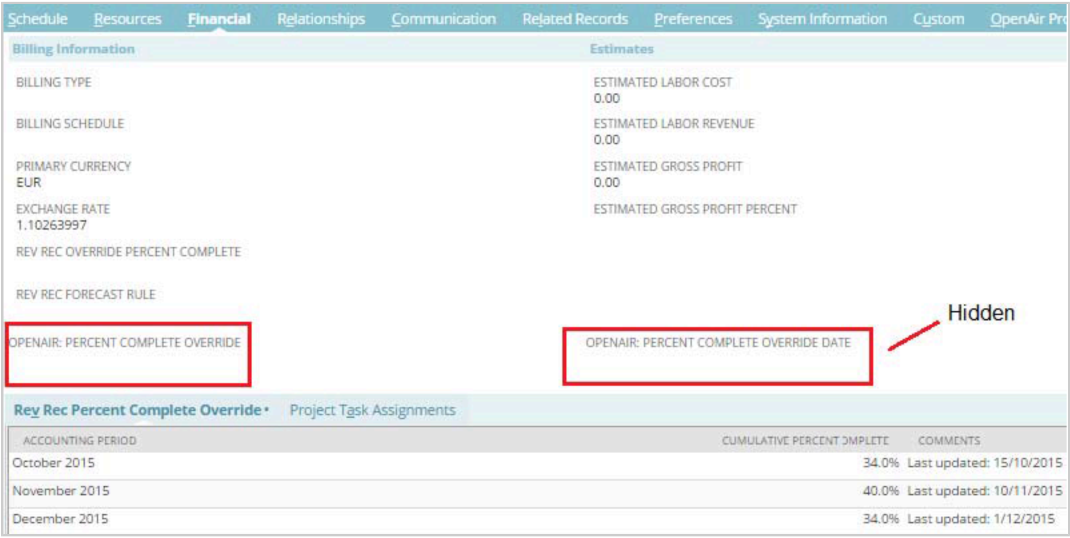
OpenAir percent complete rules can be integrated to NetSuite and then used by the Advanced Revenue Management (ARM) feature for percent-complete revenue recognition in NetSuite.

To enable Percent Complete in Advanced Revenue Management in NetSuite:

1. Enable the Advanced Revenue Management feature.
2. Update the NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) to the latest version. Requires at least version 1.35.

    > **Note:** Only projects that have been integrated from NetSuite to OpenAir will be updated. See Working with Project Records in OpenAir.

The bundle installs two new custom fields and a user event script.



Billing Information

BILLING TYPE

BILLING SCHEDULE

PRIMARY CURRENCY
EUR

EXCHANGE RATE
1.10263997

REV REC OVERRIDE PERCENT COMPLETE

REV REC FORECAST RULE

OPENAIR: PERCENT COMPLETE OVERRIDE

Estimates

ESTIMATED LABOR COST
0.00

ESTIMATED LABOR REVENUE
0.00

ESTIMATED GROSS PROFIT
0.00

ESTIMATED GROSS PROFIT PERCENT

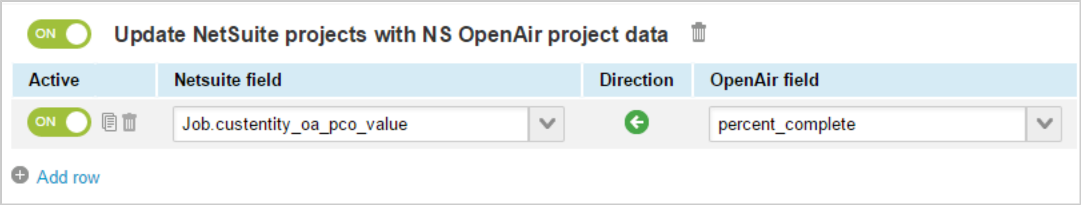
OPENAIR: PERCENT COMPLETE OVERRIDE DATE

Rev Rec Percent Complete Override • Project Task Assignments

ACCOUNTING PERIOD	CUMULATIVE PERCENT COMPLETE	COMMENTS
October 2015	34.0%	Last updated: 15/10/2015
November 2015	40.0%	Last updated: 10/11/2015
December 2015	34.0%	Last updated: 1/12/2015

The script updates the "Rev Rec Percent Complete Overrides" subtab row with the "Cumulative Percent Complete" for the appropriate "Accounting Period" recognized from today's date. The "Cumulative Percent Complete" value is replaced if the accounting period already exists in the list. The "Comments" column is set to the last updated date. The custom fields are cleared after processing.

To enable Percent Complete in Advanced Revenue Management in OpenAir, set the mapping as follows:



Update NetSuite projects with NS OpenAir project data

Active	Netsuite field	Direction	OpenAir field
ON	Job.custentity_oa_pco_value	←	percent_complete

Importing Data From NetSuite Into OpenAir

This section provides reference information for each integration workflows importing information from a specific NetSuite record type into a specific OpenAir record type.

- Employees (Employees) Import Workflow
- Customers (Customers) Import Workflow
- Projects (Projects) Import Workflow
- Services (Services Items) Import Workflow
- Expense Items (Expense Categories) Import Workflow
- Contacts (Contacts) Import Workflow
- Employees (Vendors) Import Workflow
- Vendors (Vendors) Import Workflow
- Products (Purchase Items) Import Workflow
- Customer POs (Sales Orders) Import Workflow
- Reimbursements (Expense Report Reimbursements) Import Workflow
- Payments (Invoice Payments) Import Workflow
- Tasks (Project Tasks) Import Workflow
- POs (Purchase Orders) Import Workflow
- POs (Vendor Bills) Import Workflow
- Retainers (Applied Customer Deposits) Import Workflow
- Credits (Applied Credit Memos) Import Workflow
- Tasks (Support Cases) Import Workflow
- POs (Vendor Credits) Import Workflow
- Custom Import Workflows
- Importing or Mapping NetSuite Classifications to OpenAir Hierarchies

Employees (Employees) Import Workflow

The Employees (Employees) import workflow lets you:

- Create new employee records in NetSuite first and then import them into OpenAir.
- Update employee records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding employee record in OpenAir.

> ⚠ **Important:** NetSuite is the primary application for employee records. Changes made to employee records in OpenAir after the initial import are not exported back to NetSuite.

You can import department, class, location, and subsidiary information for employees from NetSuite into OpenAir as new hierarchies or map them to existing hierarchies in OpenAir. See Importing or Mapping NetSuite Classifications to OpenAir Hierarchies. See also the following workflow settings:

- Map NetSuite employee class to employee hierarchy
- Map NetSuite employee department to employee hierarchy

- Map NetSuite employee location to employee hierarchy
- Map NetSuite employee subsidiary to employee hierarchy

For information about workflow specific settings, see Employees (Employees) Import Workflow Settings.

For information about field mapping definitions for this workflow, see Employees (Employees) Import Field Mapping Definition.

Requirements and Downstream Dependencies

Matching employee records must exist in both NetSuite and OpenAir before you can export expense reports, timesheets, or use time and expenses in billing rules.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
✓	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir employee record after you create or update the employee record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir employee record when the Employee form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir employee record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Employees (Employees) Import Workflow

The Employees (Employees) import workflow is enabled as standard. Use the following steps to set up the workflow.

To set up the Employees (Employees) import workflow in NetSuite

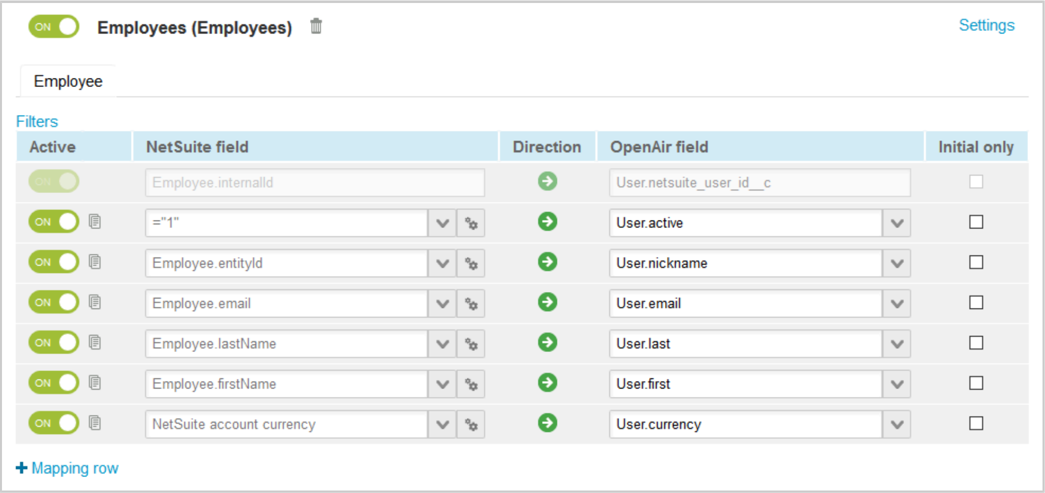
1. (Optional) To mark new employee records for export to OpenAir by default:
    a. In NetSuite, do one of the following:
        - Go to Setup > Company > General Preferences (Custom Preferences), if you are using the SuiteScript 1 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
        - Go to Customization > Lists, Records and Fields > Records > [*View list forOpenAir: SRP Integration Settings*] > [*View OpenAir SRP Integration Settings Record*] , if you are using the SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.

    b. Check the **Export Employees by Default** box.

    After you enable this preference, the **Export to OpenAir** [`custentity_oa_export_to_openair`] box is checked by default when a user creates a new employee record. Users with the relevant permissions can clear the box on individual records if required.

To set up the Employees (Employees) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector > Mapping.
2. Click **New workflow** then **Employees (Employees)**.
3. Click **Employees (Employees)** in the list of integration workflows.
4. Click **Workflow settings** to view and modify the settings specific to this workflow. For more information about the workflow settings, see Employees (Employees) Import Workflow Settings.
5. Modify the field mapping definition table for the Employees (Employees) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Employees (Employees) Import Field Mapping Definition.
6. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON **Employees (Employees)** Settings

Employee

Filters

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Employee.internalId	→	User.netsuite_user_id__c	□
ON	="1"	→	User.active	□
ON	Employee.entityId	→	User.nickname	□
ON	Employee.email	→	User.email	□
ON	Employee.lastName	→	User.last	□
ON	Employee.firstName	→	User.first	□
ON	NetSuite account currency	→	User.currency	□

+ Mapping row

7. Click **Save**.
8. (Optional) Add the Employees (Employees) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Employees (Employees) Import Workflow Settings

You can control Employees (Employees) import workflow settings from the NetSuite Connector UI. On the OpenAir NetSuite Connector Mapping screen, select the Employees (Employees) import workflow and click **Settings** to view the workflow settings form.

The following settings are available for the Employees (Employees) import workflow:

- Enable purchaser integration
- Map NetSuite employee class to employee hierarchy
- Map NetSuite employee department to employee hierarchy
- Map NetSuite employee location to employee hierarchy
- Map NetSuite employee subsidiary to employee hierarchy
- Disable Export to OpenAir filtering for employees in NetSuite integration
- Enable import of inactive employees

Enable purchaser integration

When checked, the Employees (Employees) import workflow creates or updates the purchaser record associated with the imported employee record in OpenAir if the **OpenAir: Employee is NS purchaser** box is checked on the employee record in NetSuite.

The information saved on the purchaser record in OpenAir includes:

- **Associated Employee** [`purchaser.user_id`] — The internal ID of the associated employee record in OpenAir.
- **Purchaser** [`purchaser.name`] — The name of the purchaser is set to the full name of the associated employee.
- **NetSuite Purchase Order Integration Purchaser** [`purchaser.netsuite_purchaser__c`] — Checked when the associated employee record is imported from NetSuite and the **OpenAir: Employee is NS purchaser** box is checked on the employee record in NetSuite.

After you import the NetSuite employee as both an employee and a purchaser in OpenAir, you can:

- Create purchase order or vendor bill transactions associated with this employee in NetSuite, and import purchase order transactions from NetSuite into OpenAir using the POs (Purchase Orders) workflow. This will create a PO record in OpenAir and associate it with the corresponding purchaser record in OpenAir. See Products (Purchase Items) Import Workflow and POs (Vendor Bills) Import Workflow.
- Create invoices in OpenAir and include charges to rebill purchases with purchaser information and export these invoices from OpenAir to NetSuite using the Invoices (Invoices) export workflow. See Invoices (Invoices) Export Workflow.

> **Note:** The **OpenAir: Employee is NS purchaser** [`custentity_oa_ns_purchaser`] custom field is created when you install NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) 1.34 or later version.

Map NetSuite employee class to employee hierarchy

Check this box to map the employee **Class** in NetSuite to an employee hierarchy in OpenAir.

- **NetSuite class** — The class is a type of classification used to identify, categorize and track records such as financials, transactions, and employees in NetSuite. Classes are useful for wider segments within your business. For more information about classes, see Departments and Classes Overview (external link to NetSuite SuiteAnswers | Answer ID: 9844).
- **OpenAir hierarchy** — You can use hierarchies are used to create multilevel classification trees and categorize records in your OpenAir account. Each hierarchy is associated to one of the following record types: employees, customers, or projects. Multiple hierarchies can apply to the same record type. For more information about hierarchies, see Administrator Guide.

After you check this box and you save the integration configuration, OpenAir NetSuite Connector creates the following records in OpenAir:

- The **Employee Class** hierarchy. The Employee demographic form in OpenAir will show dropdown fields for the Employee Class hierarchy if it contains any active nodes — Edit the hierarchy record or use form permissions to hide the Employee Class hierarchy dropdown field on Employee demographic form, if required. You can also set the **Employee Class** as required, and choose to display it as a column on list views.

> **Note:** You can map the NetSuite class to an existing OpenAir hierarchy instead of using the hierarchy created by OpenAir NetSuite Connector. To do so, you need to delete the Employee Class hierarchy created by OpenAir NetSuite Connector and edit the existing hierarchy, hierarchy levels, and nodes in OpenAir. This lets you retain the existing hierarchy you are already using in OpenAir.

- Custom fields for hierarchy and hierarchy nodes records. These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.
  - **NetSuite class hierarchy** `[netsuite_hierarchy_class]` — Check box used to identify if the hierarchy maps to the NetSuite class.
  - **NetSuite ID** `[netsuite_hierarchy_node_id]` — Text custom field used to map each level or node in the OpenAir hierarchy to the corresponding NetSuite class record.
  - **NetSuite class level or hierarchy node** `[netsuite_hierarchy_node_class]` — Check box used to identify if the OpenAir hierarchy level or node maps to a NetSuite class depth level or record.

For more information about importing or mapping NetSuite classifications to OpenAir hierarchies, see Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

> **Important:** If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:
>
> - Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
> - Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Map NetSuite employee department to employee hierarchy

Check this box to map the employee **Department** in NetSuite to an employee hierarchy in OpenAir.

- **NetSuite department** — The department is a type of classification used to identify, categorize and track records such as financials, transactions, and employees in NetSuite. Departments are listed first on transactions, and are useful when designating transactions and employees as part of an internal team. For more information about departments, see Departments and Classes Overview (external link to NetSuite SuiteAnswers | Answer ID: 9844).
- **OpenAir hierarchy** — You can use hierarchies are used to create multilevel classification trees and categorize records in your OpenAir account. Each hierarchy is associated to one of the following record types: employees, customers, or projects. Multiple hierarchies can apply to the same record type. For more information about hierarchies, see Administrator Guide.

After you check this box and you save the integration configuration, OpenAir NetSuite Connector creates the following records in OpenAir:

- The **Employee Department** hierarchy. The Employee demographic form in OpenAir will show dropdown fields for the Employee Department hierarchy if it contains any active nodes — Edit the hierarchy record or use form permissions to hide the Employee Department hierarchy dropdown field on Employee demographic form, if required. You can also set the **Employee Department** as required, and choose to display it as a column on list views.

  > **Note:** You can map the NetSuite department to an existing OpenAir hierarchy instead of using the hierarchy created by OpenAir NetSuite Connector. To do so, you need to delete the Employee Department hierarchy created by OpenAir NetSuite Connector and edit the existing hierarchy, hierarchy levels, and nodes in OpenAir. This lets you retain the existing hierarchy you are already using in OpenAir.

- Custom fields for hierarchy and hierarchy nodes records. These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.
  - **NetSuite department hierarchy** `[netsuite_hierarchy_department]` — Check box used to identify if the hierarchy maps to the NetSuite department.
  - **NetSuite ID** `[netsuite_hierarchy_node_id]` — Text custom field used to map each level or node in the OpenAir hierarchy to the corresponding NetSuite department record.
  - **NetSuite department level or hierarchy node** `[netsuite_hierarchy_node_department]` — Check box used to identify if the OpenAir hierarchy level or node maps to a NetSuite department depth level or record.

For more information about importing or mapping NetSuite classifications to OpenAir hierarchies, see Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

> **Important:** If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:
>
> - Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
> - Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Map NetSuite employee location to employee hierarchy

Check this box to map the employee **Location** in NetSuite to an employee hierarchy in OpenAir.

- **NetSuite location** — The location is a type of classification used to track information about employees and transactions for multiple offices or warehouses in NetSuite. You can associate transactions and employees with each location, and then filter report data by location. For more information about locations, see Locations Overview (external link to NetSuite SuiteAnswers | Answer ID: 9845).
- **OpenAir hierarchy** — You can use hierarchies are used to create multilevel classification trees and categorize records in your OpenAir account. Each hierarchy is associated to one of the following

record types: employees, customers, or projects. Multiple hierarchies can apply to the same record type. For more information about hierarchies, see Administrator Guide.

After you check this box and you save the integration configuration, OpenAir NetSuite Connector creates the following records in OpenAir:

- The **Employee Location** hierarchy. The Employee demographic form in OpenAir will show dropdown fields for the Employee Location hierarchy if it contains any active nodes — Edit the hierarchy record or use form permissions to hide the Employee Location hierarchy dropdown field on Employee demographic form, if required. You can also set the **Employee Location** as required, and choose to display it as a column on list views.

  > **Note:** You can map the NetSuite location to an existing OpenAir hierarchy instead of using the hierarchy created by OpenAir NetSuite Connector. To do so, you need to delete the Employee Location hierarchy created by OpenAir NetSuite Connector and edit the existing hierarchy, hierarchy levels, and nodes in OpenAir. This lets you retain the existing hierarchy you are already using in OpenAir.

- Custom fields for hierarchy and hierarchy nodes records. These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.
  - **NetSuite location hierarchy** [`netsuite_hierarchy_location`] — Check box used to identify if the hierarchy maps to the NetSuite location.
  - **NetSuite ID** [`netsuite_hierarchy_node_id`] — Text custom field used to map each level or node in the OpenAir hierarchy to the corresponding NetSuite location record.
  - **NetSuite location level or hierarchy node** [`netsuite_hierarchy_node_location`] — Check box used to identify if the OpenAir hierarchy level or node maps to a NetSuite location depth level or record.

For more information about importing or mapping NetSuite classifications to OpenAir hierarchies, see Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

> **Important:** If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:
>
> - Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
> - Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Map NetSuite employee subsidiary to employee hierarchy

Check this box to map the employee **Subsidiary** in NetSuite to an employee hierarchy in OpenAir.

- **NetSuite subsidiary** — If you use NetSuite OneWorld, subsidiaries are the primary classification used to organize your NetSuite records, and additional classification records are associated to a subsidiary. Each subsidiary represents a separate company within your global organization. Subsidiaries can be international or domestic. For more information about subsidiaries, see Subsidiaries in OneWorld (external link to NetSuite SuiteAnswers | Answer ID: 9872).
- **OpenAir hierarchy** — You can use hierarchies are used to create multilevel classification trees and categorize records in your OpenAir account. Each hierarchy is associated to one of the following record types: employees, customers, or projects. Multiple hierarchies can apply to the same record type. For more information about hierarchies, see Administrator Guide.

After you check this box and you save the integration configuration, OpenAir NetSuite Connector creates the following records in OpenAir:

- The **Employee Subsidiary** hierarchy. The Employee demographic form in OpenAir will show dropdown fields for the Employee Subsidiary hierarchy if it contains any active nodes — Edit the hierarchy record or use form permissions to hide the Employee Subsidiary hierarchy dropdown field on Employee demographic form, if required. You can also set the **Employee Subsidiary** as required, and choose to display it as a column on list views.

  > **Note:** You can map the NetSuite subsidiary to an existing OpenAir hierarchy instead of using the hierarchy created by OpenAir NetSuite Connector. To do so, you need to delete the Employee Subsidiary hierarchy created by OpenAir NetSuite Connector and edit the existing hierarchy, hierarchy levels, and nodes in OpenAir. This lets you retain the existing hierarchy you are already using in OpenAir.

- Custom fields for hierarchy and hierarchy nodes records. These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.
  - **NetSuite subsidiary hierarchy** [`netsuite_hierarchy_subsidiary`] — Check box used to identify if the hierarchy maps to the NetSuite subsidiary .
  - **NetSuite ID** [`netsuite_hierarchy_node_id`] — Text custom field used to map each level or node in the OpenAir hierarchy to the corresponding NetSuite subsidiary record.
  - **NetSuite subsidiary level or hierarchy node** [`netsuite_hierarchy_node_subsidiary`] — Check box used to identify if the OpenAir hierarchy level or node maps to a NetSuite subsidiary depth level or record.

For more information about importing or mapping NetSuite classifications to OpenAir hierarchies, see Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

⚠ **Important:** If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:

- Form scripts associated with the Project form and triggered by an "On submit", "Before save", or "After save" event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
- Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Disable Export to OpenAir filtering for employees in NetSuite integration

Check this box to import all employees from NetSuite to OpenAir.

- When checked, OpenAir NetSuite Connector imports employee records from NetSuite into OpenAir even if the **Export to OpenAir** [`custentity_oa_export_to_openair`] box is not checked on the NetSuite employee record.
- Otherwise, OpenAir NetSuite Connector imports an employee record only if the **Export to OpenAir** [`custentity_oa_export_to_openair`] box is checked.

By default, employee records are imported from NetSuite into OpenAir if the **Inactive** box is NOT checked on the employee record in NetSuite. You can modify this behavior — See Enable import of inactive employees.

- The **Inactive** box is NOT checked on the employee record in NetSuite. You can modify this behavior — See Enable import of inactive employees.
- The **Export to OpenAir** [`custentity_oa_export_to_openair`] box is checked n the employee record in NetSuite. You can modify this behavior — See Disable Export to OpenAir filtering for employees in NetSuite integration.

Enable import of inactive employees

Check this box to import all employees from NetSuite to OpenAir.

- When checked, OpenAir NetSuite Connector imports employee records from NetSuite into OpenAir even if the **Inactive** box is checked on the NetSuite employee record. The **Active** box is checked or cleared accordingly whether the Employees (Employees) import workflow creates a new employee record or updates an existing employee record in OpenAir. Make sure you use filter conditions so that you only import the inactive records that you need.
- Otherwise, OpenAir NetSuite Connector imports the employee record only if the **Inactive** box is not checked. The **Active** box is checked by default when the Employees (Employees) import workflow creates a new employee record.

By default, employee records are imported from NetSuite into OpenAir if the **Export to OpenAir** [custentity_oa_export_to_openair] box is checked on the employee record in NetSuite. You can modify this behavior — See Disable Export to OpenAir filtering for employees in NetSuite integration.

Employees (Employees) Import Field Mapping Definition

The Employees (Employees) import workflow includes one mapping group:

- Employee [Integration point: EMPLOYEE_TO_USER] — See Employee Mapping Group.

Employee Mapping Group

Use the Employee mapping group to map information on employee records in OpenAir with information on employee records in NetSuite.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes employee name, email, password, OpenAir role and currency. Filters are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Employee [Employee] Schema: Employee (2020.1) | Employee (2023.1) XSD: EmployeeSearchRowBasic platformCommon (2020.1) | platformCommon (2023.1)	Employee [user] Data Dictionary: user	Employee [EMPLOYEE_TO_USER]

Default and Custom Filters

By default, employee records are imported from NetSuite into OpenAir if the following conditions are all met:

- The **Inactive** box is NOT checked on the employee record in NetSuite. You can modify this behavior — See Enable import of inactive employees.

- The **Export to OpenAir** [custentity_oa_export_to_openair] box is checked on the employee record in NetSuite. You can modify this behavior — See Disable Export to OpenAir filtering for employees in NetSuite integration.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definitions for the Employee mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Note: Features controlled by the following workflow settings add additional information associated with the employee when creating or updating a project using the Employees (Employees) import workflow. For more information, see the settings descriptions.

- Enable purchaser integration — Adds purchaser record associated with the employee
- Map NetSuite employee class to employee hierarchy — Can be used to import employee class information as a hierarchy node associated with the employee.
- Map NetSuite employee department to employee hierarchy — Can be used to import employee department information as a hierarchy node associated with the employee.
- Map NetSuite employee location to employee hierarchy — Can be used to import employee location information as a hierarchy node associated with the employee.
- Map NetSuite employee subsidiary to employee hierarchy — Can be used to import employee subsidiary information as a hierarchy node associated with the employee.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

Review the following guidelines:

- The Employees (Employees) import workflow cannot be used to associate an employee work schedule to the employee. Field mappings with OpenAir field `User.workschedule_id` will be ignored.
- If you are using custom field mapping definitions to set a **Password** for the employee in OpenAir, the **Password** is subject to the same minimum password policy requirements enforced when creating or changing a password using the OpenAir UI.

Customers (Customers) Import Workflow

The Customers (Customers) import workflow lets you:

- Create new customer records in NetSuite first and then import them into OpenAir.

> **Note:** You can also use the Customers (Customers) export workflow to create new customer records in OpenAir first and then export them to NetSuite. See Customers (Customers) Export Workflow.

- Update customer records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding customer record in OpenAir.

> **Important:** NetSuite is the primary application for customer records. Changes made to customer records in OpenAir after the initial import are not exported back to NetSuite.

You can also import NetSuite leads and prospects into OpenAir as customers. See the following workflow settings:

- Include NetSuite leads marked 'Export to OpenAir' in the customers import
- Include NetSuite prospects marked 'Export to OpenAir' in the customers import

You can import department, class, location, and subsidiary information for customers from NetSuite into OpenAir as new hierarchies or map them to existing hierarchies in OpenAir. See Importing or Mapping NetSuite Classifications to OpenAir Hierarchies. See also the following workflow settings:

- Map NetSuite customer class to customer hierarchy
- Map NetSuite customer department to customer hierarchy
- Map NetSuite customer location to customer hierarchy
- Map NetSuite customer subsidiary to customer hierarchy

For information about workflow specific settings, see Customers (Customers) Import Workflow Settings.

For information about field mapping definitions for this workflow, see Customers (Customers) Import Field Mapping Definition.

Requirements and Downstream Dependencies

Matching customer records must exist in both NetSuite and OpenAir before you can:

- Import projects associated with these customers from NetSuite into OpenAir.
- Export projects from OpenAir to NetSuite.

- Import contacts associated with these customers or these customers' projects from NetSuite into OpenAir.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
✓	✓	✓	✓	✓	✓

- Run the integration manually, use a scheduled integration run to create or update the OpenAir customer record after you create or update the customer record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir customer record when the Customer form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir customer record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.
- When you install NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) 2.08 or later version on your NetSuite account, NetSuite customer records are imported into OpenAir only if one of the monitored field values have changed. See Monitored Fields.

Setting up the Customers (Customers) Import Workflow

The Customers (Customers) import workflow is enabled as standard. Use the following steps to set up the workflow.

To set up the Customers (Customers) import workflow in NetSuite

1. (Optional) To mark new customer records for export to OpenAir by default:
   a. In NetSuite, do one of the following:
      - Go to Setup > Company > General Preferences (Custom Preferences), if you are using the SuiteScript 1 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
      - Go to Customization > Lists, Records and Fields > Records > [*View list forOpenAir: SRP Integration Settings*] > [*View OpenAir SRP Integration Settings Record*] , if you are using the SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.

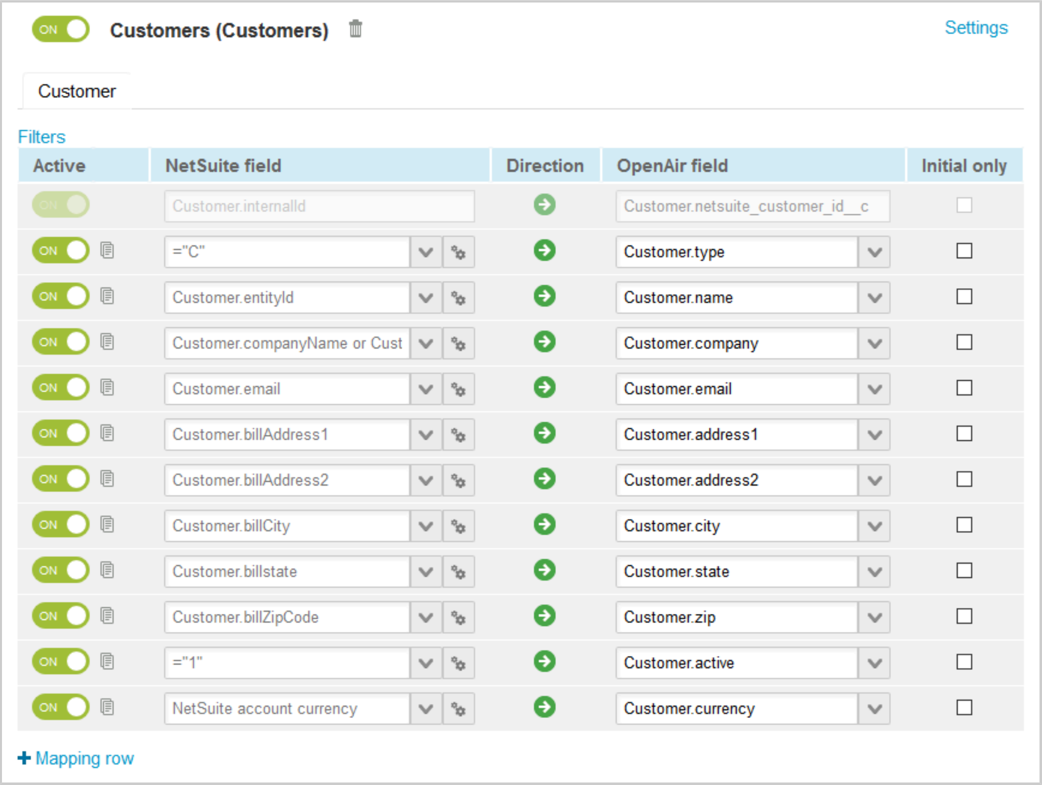
   b. Check the **Export Customers by Default** box.

   After you enable this preference, the **Export to OpenAir** [`custentity_oa_export_to_openair`] box is checked by default when a user creates a new customer record. Users with the relevant permissions can clear the box on individual records if required.

To set up the Customers (Customers) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector > Mapping.
2. Click **New workflow** then **Customers (Customers)**.

3. Click **Customers (Customers)** in the list of integration workflows.
4. Click **Workflow settings** to view and modify the settings specific to this workflow. For more information about the workflow settings, see Customers (Customers) Import Workflow Settings.
5. Modify the field mapping definition table for the Customers (Customers) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Customers (Customers) Import Field Mapping Definition.
6. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Customer.internalId	→	Customer.netsuite_customer_id__c	□
ON	="C"	→	Customer.type	□
ON	Customer.entityId	→	Customer.name	□
ON	Customer.companyName or Cust	→	Customer.company	□
ON	Customer.email	→	Customer.email	□
ON	Customer.billAddress1	→	Customer.address1	□
ON	Customer.billAddress2	→	Customer.address2	□
ON	Customer.billCity	→	Customer.city	□
ON	Customer.billstate	→	Customer.state	□
ON	Customer.billZipCode	→	Customer.zip	□
ON	="1"	→	Customer.active	□
ON	NetSuite account currency	→	Customer.currency	□

7. Click **Save**.
8. (Optional) Add the Customers (Customers) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Customers (Customers) Import Workflow Settings

You can control Customers (Customers) import workflow settings from the NetSuite Connector UI. On the OpenAir NetSuite Connector Mapping screen, select the Customers(Customers) import workflow and click **Settings** to view the workflow settings form.

The following settings are available for the Customers(Customers) import workflow:

- Map NetSuite customer class to customer hierarchy
- Map NetSuite customer department to customer hierarchy
- Map NetSuite customer location to customer hierarchy
- Map NetSuite customer subsidiary to customer hierarchy
- Disable Export to OpenAir filtering for customers in NetSuite integration

- Include NetSuite leads marked 'Export to OpenAir' in the customers import
- Include NetSuite prospects marked 'Export to OpenAir' in the customers import
- Skip merge of matching customers with NetSuite Internal ID

Map NetSuite customer class to customer hierarchy

Check this box to map the customer **Class** in NetSuite to a customer hierarchy in OpenAir.

- **NetSuite class** — The class is a type of classification used to identify, categorize and track records such as financials, transactions, and employees in NetSuite. Classes are useful for wider segments within your business. For more information about classes, see Departments and Classes Overview (external link to NetSuite SuiteAnswers | Answer ID: 9844).

  > **Note:** After you enable classes for your account, you can create class records and use them to categorise transactions and employees. Classifications are not available for customer and project records as standard in NetSuite. However, you can create a List/Record custom entity field to use classes for customers and projects using a list/record field, if required. See Using Classifications for Customers and Projects in NetSuite.

- **OpenAir hierarchy** — You can use hierarchies are used to create multilevel classification trees and categorize records in your OpenAir account. Each hierarchy is associated to one of the following record types: customers, customers, or projects. Multiple hierarchies can apply to the same record type. For more information about hierarchies, see Administrator Guide.

After you check this box and you save the integration configuration, OpenAir NetSuite Connector creates the following records in OpenAir:

- The **Customer Class** hierarchy. The Customer demographic form in OpenAir will show dropdown fields for the Customer Class hierarchy if it contains any active nodes — Edit the hierarchy record or use form permissions to hide the Customer Class hierarchy dropdown field on Customer demographic form, if required. You can also set the **Customer Class** as required, and choose to display it as a column on list views.

  > **Note:** You can map the NetSuite class to an existing OpenAir hierarchy instead of using the hierarchy created by OpenAir NetSuite Connector. To do so, you need to delete the Customer Class hierarchy created by OpenAir NetSuite Connector and edit the existing hierarchy, hierarchy levels, and nodes in OpenAir. This lets you retain the existing hierarchy you are already using in OpenAir.

- Custom fields for hierarchy and hierarchy nodes records. These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.
  - **NetSuite class hierarchy** [`netsuite_hierarchy_class`] — Check box used to identify if the hierarchy maps to the NetSuite class.
  - **NetSuite ID** [`netsuite_hierarchy_node_id`] — Text custom field used to map each level or node in the OpenAir hierarchy to the corresponding NetSuite class record.
  - **NetSuite class level or hierarchy node** [`netsuite_hierarchy_node_class`] — Check box used to identify if the OpenAir hierarchy level or node maps to a NetSuite class depth level or record.

For more information about importing or mapping NetSuite classifications to OpenAir hierarchies, see Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

> ⚠ **Important:** If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:
>
> - Form scripts associated with the Project form and triggered by an "On submit", "Before save", or "After save" event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
> - Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Map NetSuite customer department to customer hierarchy

Check this box to map the customer **Department** in NetSuite to a customer hierarchy in OpenAir.

- **NetSuite department** — The department is a type of classification used to identify, categorize and track records such as financials, transactions, and employees in NetSuite. Departments are useful for wider segments within your business. For more information about departments, see Departments and Classes Overview (external link to NetSuite SuiteAnswers | Answer ID: 9844).

  > **Note:** After you enable departments for your account, you can create department records and use them to categorise transactions and employees. Classifications are not available for customer and project records as standard in NetSuite. However, you can create a List/Record custom entity field to use departments for customers and projects using a list/record field, if required. See Using Classifications for Customers and Projects in NetSuite.

- **OpenAir hierarchy** — You can use hierarchies are used to create multilevel classification trees and categorize records in your OpenAir account. Each hierarchy is associated to one of the following record types: customers, customers, or projects. Multiple hierarchies can apply to the same record type. For more information about hierarchies, see Administrator Guide.

After you check this box and you save the integration configuration, OpenAir NetSuite Connector creates the following records in OpenAir:

- The **Customer Department** hierarchy. The Customer demographic form in OpenAir will show dropdown fields for the Customer Department hierarchy if it contains any active nodes — Edit the hierarchy record or use form permissions to hide the Customer Department hierarchy dropdown field on Customer demographic form, if required. You can also set the **Customer Department** as required, and choose to display it as a column on list views.

  > **Note:** You can map the NetSuite department to an existing OpenAir hierarchy instead of using the hierarchy created by OpenAir NetSuite Connector. To do so, you need to delete the Customer Department hierarchy created by OpenAir NetSuite Connector and edit the existing hierarchy, hierarchy levels, and nodes in OpenAir. This lets you retain the existing hierarchy you are already using in OpenAir.

- Custom fields for hierarchy and hierarchy nodes records. These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.

  - **NetSuite department hierarchy** [netsuite_hierarchy_department] — Check box used to identify if the hierarchy maps to the NetSuite department.
  - **NetSuite ID** [netsuite_hierarchy_node_id] — Text custom field used to map each level or node in the OpenAir hierarchy to the corresponding NetSuite department record.
  - **NetSuite department level or hierarchy node** [netsuite_hierarchy_node_department] — Check box used to identify if the OpenAir hierarchy level or node maps to a NetSuite department depth level or record.

For more information about importing or mapping NetSuite classifications to OpenAir hierarchies, see Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

> **Important:** If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:
>
> - Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
> - Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Map NetSuite customer location to customer hierarchy

Check this box to map the customer **Location** in NetSuite to a customer hierarchy in OpenAir.

- **NetSuite location** — The location is a type of classification used to identify, categorize and track records such as financials, transactions, and employees in NetSuite. Locations are useful for wider segments within your business. For more information about locations, see Locations Overview (external link to NetSuite SuiteAnswers | Answer ID: 9845).

  > **Note:** After you enable locations for your account, you can create location records and use them to categorise transactions and employees. Classifications are not available for customer and project records as standard in NetSuite. However, you can create a List/Record custom entity field to use locations for customers and projects using a list/record field, if required. See Using Classifications for Customers and Projects in NetSuite.

- **OpenAir hierarchy** — You can use hierarchies are used to create multilevel classification trees and categorize records in your OpenAir account. Each hierarchy is associated to one of the following record types: customers, customers, or projects. Multiple hierarchies can apply to the same record type. For more information about hierarchies, see Administrator Guide.

After you check this box and you save the integration configuration, OpenAir NetSuite Connector creates the following records in OpenAir:

- The **Customer Location** hierarchy. The Customer demographic form in OpenAir will show dropdown fields for the Customer Location hierarchy if it contains any active nodes — Edit the hierarchy record or use form permissions to hide the Customer Location hierarchy dropdown field on Customer demographic form, if required. You can also set the **Customer Location** as required, and choose to display it as a column on list views.

> **Note:** You can map the NetSuite location to an existing OpenAir hierarchy instead of using the hierarchy created by OpenAir NetSuite Connector. To do so, you need to delete the Customer Location hierarchy created by OpenAir NetSuite Connector and edit the existing hierarchy, hierarchy levels, and nodes in OpenAir. This lets you retain the existing hierarchy you are already using in OpenAir.

- Custom fields for hierarchy and hierarchy nodes records. These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.
  - **NetSuite location hierarchy** [`netsuite_hierarchy_location`] — Check box used to identify if the hierarchy maps to the NetSuite location.
  - **NetSuite ID** [`netsuite_hierarchy_node_id`] — Text custom field used to map each level or node in the OpenAir hierarchy to the corresponding NetSuite location record.
  - **NetSuite location level or hierarchy node** [`netsuite_hierarchy_node_location`] — Check box used to identify if the OpenAir hierarchy level or node maps to a NetSuite location depth level or record.

For more information about importing or mapping NetSuite classifications to OpenAir hierarchies, see Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

> **Important:** If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:
>
> - Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
> - Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Map NetSuite customer subsidiary to customer hierarchy

Check this box to map the customer **Subsidiary** in NetSuite to a customer hierarchy in OpenAir.

- **NetSuite subsidiary** — If you use NetSuite OneWorld, subsidiaries are the primary classification used to organize your NetSuite records, and additional classification records are associated to a subsidiary. Each subsidiary represents a separate company within your global organization. Subsidiaries can be international or domestic. For more information about subsidiaries, see Subsidiaries in OneWorld (external link to NetSuite SuiteAnswers | Answer ID: 9872).
- **OpenAir hierarchy** — You can use hierarchies are used to create multilevel classification trees and categorize records in your OpenAir account. Each hierarchy is associated to one of the following record types: employees, customers, or projects. Multiple hierarchies can apply to the same record type. For more information about hierarchies, see Administrator Guide.

After you check this box and you save the integration configuration, OpenAir NetSuite Connector creates the following records in OpenAir:

- The **Customer Subsidiary** hierarchy. The Customer demographic form in OpenAir will show dropdown fields for the Customer Subsidiary hierarchy if it contains any active nodes — Edit the hierarchy record or use form permissions to hide the Customer Subsidiary hierarchy dropdown field on Customer demographic form, if required. You can also set the **Customer Subsidiary** as required, and choose to display it as a column on list views.

  > **Note:** You can map the NetSuite subsidiary to an existing OpenAir hierarchy instead of using the hierarchy created by OpenAir NetSuite Connector. To do so, you need to delete the Customer Subsidiary hierarchy created by OpenAir NetSuite Connector and edit the existing hierarchy, hierarchy levels, and nodes in OpenAir. This lets you retain the existing hierarchy you are already using in OpenAir.

- Custom fields for hierarchy and hierarchy nodes records. These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.
  - **NetSuite subsidiary hierarchy** [netsuite_hierarchy_subsidiary] — Check box used to identify if the hierarchy maps to the NetSuite subsidiary .
  - **NetSuite ID** [netsuite_hierarchy_node_id] — Text custom field used to map each level or node in the OpenAir hierarchy to the corresponding NetSuite subsidiary record.
  - **NetSuite subsidiary level or hierarchy node** [netsuite_hierarchy_node_subsidiary] — Check box used to identify if the OpenAir hierarchy level or node maps to a NetSuite subsidiary depth level or record.

For more information about importing or mapping NetSuite classifications to OpenAir hierarchies, see Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

> **Important:** If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:
>
> - Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
> - Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Disable Export to OpenAir filtering for customers in NetSuite integration

Check this box to import all customers from NetSuite to OpenAir.

- When checked, OpenAir NetSuite Connector imports customer records from NetSuite into OpenAir even if the **Export to OpenAir** [custentity_oa_export_to_openair] box is not checked on the NetSuite customer record.

- Otherwise, OpenAir NetSuite Connector imports a customer record only if the **Export to OpenAir** [custentity_oa_export_to_openair] box is checked.

By default, NetSuite lead and prospect records (customer records with the **Stage** field set to Lead or Prospect) are not imported into OpenAir. You can modify this behavior to import leads and prospects — See Include NetSuite leads marked 'Export to OpenAir' in the customers import and Include NetSuite prospects marked 'Export to OpenAir' in the customers import.

Include NetSuite leads marked 'Export to OpenAir' in the customers import

Check this box to import NetSuite leads (customer records with the **Stage** field set to Lead) as well as NetSuite customers (customer records with the **Stage** field set to Customer) into OpenAir as customers.

By default, customer records are imported from NetSuite into OpenAir if the **Export to OpenAir** [custentity_oa_export_to_openair] box is checked on the customer record in NetSuite. You can modify this behavior — See Disable Export to OpenAir filtering for employees in NetSuite integration.

You can also import NetSuite prospects (customer records with the **Stage** field set to Prospect) — See Include NetSuite prospects marked 'Export to OpenAir' in the customers import.

✓ **Tip:** By default, lead and prospect records are imported into OpenAir as customers (customer records of type “C”). To import lead and prospect records into OpenAir as prospects, use a custom field mapping definition to set the customer.type in OpenAir to C (Customer) or P (Prospect) depending on the Customer.stage value in NetSuite (_customer, _lead, or _prospect).

See also:

- Disable Export to OpenAir filtering for customers in NetSuite integration
- Include NetSuite prospects marked 'Export to OpenAir' in the customers import

Include NetSuite prospects marked 'Export to OpenAir' in the customers import

Check this box to import NetSuite prospects (customer records with the **Stage** field set to Prospect) as well as NetSuite customers (customer records with the **Stage** field set to Customer) into OpenAir as customers.

By default, customer records are imported from NetSuite into OpenAir if the **Export to OpenAir** [custentity_oa_export_to_openair] box is checked on the customer record in NetSuite. You can modify this behavior — See Disable Export to OpenAir filtering for employees in NetSuite integration.

You can also import NetSuite leads (customer records with the **Stage** field set to Lead) — See Include NetSuite leads marked 'Export to OpenAir' in the customers import.

✓ **Tip:** By default, lead and prospect records are imported into OpenAir as customers (customer records of type “C”). To import lead and prospect records into OpenAir as prospects, use a custom field mapping definition to set the customer.type in OpenAir to C (Customer) or P (Prospect) depending on the Customer.stage value in NetSuite (_customer, _lead, or _prospect).

Skip merge of matching customers with NetSuite Internal ID

When importing NetSuite customers into OpenAir, OpenAir NetSuite Connector uses the **NetSuite ID** custom field in OpenAir to find the matching OpenAir record. If there are no matching results, it uses the

Customer ID (customer.entityId) on the NetSuite customer record and the **Nickname** (Customer.name) on the OpenAir customer record to find a matching record with a case-insensitive search. If a NetSuite **Customer ID** – OpenAir **Nickname** match exists, OpenAir NetSuite Connector updates the **NetSuite ID** by default.

Check this box to perform an additional check based on the **NetSuite ID** custom field on the matching OpenAir record. When the **Skip merge of matching customers with NetSuite Internal ID** setting is enabled and OpenAir NetSuite Connector finds matching records exist based on NetSuite **Customer ID** – OpenAir **Nickname**:

- If the **NetSuite ID** custom field on the matching OpenAir record is blank, the OpenAir record is updated.
- Otherwise, no changes are made to the OpenAir record — the import workflow ignores this record.

Customers (Customers) Import Field Mapping Definition

The Customers (Customers) import workflow includes one mapping group:

- Customer [Integration point: CUSTOMER_TO_CUSTOMER] — See Customer Mapping Group Tab.

Customer Mapping Group Tab

Use the Customer mapping group to map information on customer records in OpenAir with information on customer records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes company name, email, address and currency. Filters are supported.
- Additional Field Mappings — Best practice guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Customer [Customer] Schema: Customer (2020.1) | Customer (2023.1) XSD: CustomerSearchRowBasic platformCommon (2020.1) | platformCommon (2023.1)	Customer [customer] Data Dictionary: customer	Customer [CUSTOMER_TO_CUSTOMER]

Note: The `Customer` SOAP object is used for customer, lead, and prospect records. This is determined by the property `Customer.stage` with the value `_customer`, `_lead`, or `_prospect`.

Default and Custom Filters

By default, customer records are imported from NetSuite into OpenAir if the following conditions are all met:

- The **Inactive** box is NOT checked on the customer record in NetSuite.
- The **Export to OpenAir** [`custentity_oa_export_to_openair`] box is checked on the customer record in NetSuite. You can modify this behavior — See Disable Export to OpenAir filtering for employees in NetSuite integration.
- The **Stage** is set to “Customer” [`_customer`] on the customer record in NetSuite. You can modify this behavior to import leads and prospects — See Include NetSuite leads marked 'Export to OpenAir' in the customers import and Include NetSuite prospects marked 'Export to OpenAir' in the customers import.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definitions for the Customer mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Note: Features controlled by the following workflow settings add additional information associated with the customer when creating or updating a customer using the Customers (Customers) import workflow. For more information, see the settings descriptions.

- Map NetSuite customer class to customer hierarchy — Can be used to import customer class information as a hierarchy node associated with the customer.
- Map NetSuite customer department to customer hierarchy — Can be used to customer project department information as a hierarchy node associated with the customer.
- Map NetSuite customer location to customer hierarchy — Can be used to import customer location information as a hierarchy node associated with the customer.
- Map NetSuite customer subsidiary to customer hierarchy— Can be used to import customer subsidiary information as a hierarchy node associated with the customer.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

See the OpenAir Help Center for detailed information about additional field mapping definitions for the Customer mapping group tab that may be required or useful in some cases. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Projects (Projects) Import Workflow

The Projects (Projects) import workflow lets you:

- Create new project records in NetSuite first and then import them into OpenAir.

  Note: You can also use the Projects (Projects) export workflow to create new project records in OpenAir first and then export them to NetSuite. See Projects (Projects) Export Workflow.

- Update project records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding project record in OpenAir.

> **Note:** You can also use the Update NetSuite Projects with OpenAir Project Data export workflow to update project records in OpenAir and run the integration to update the project records in NetSuite. See Update NetSuite Projects with OpenAir Project Data Export Workflow.

- Create project templates in OpenAir, select an **OpenAir Project Template** when creating a new project in NetSuite. When you import the project into OpenAir, the OpenAir project contains project-related information copied from the template. The information copied from the project template can include tasks, assignments, billing rules, loaded costs, revenue recognition rules, auto-bill settings, project approvers, and more. See Create new projects based on the project template specified in NetSuite.

You can import department, class, location, and subsidiary information for projects from NetSuite into OpenAir as new hierarchies or map them to existing hierarchies in OpenAir. See Importing or Mapping NetSuite Classifications to OpenAir Hierarchies. See also the following workflow settings:

- Map NetSuite project class to project hierarchy
- Map NetSuite project department to project hierarchy
- Map NetSuite project location to project hierarchy
- Map NetSuite project subsidiary to project hierarchy

You can also import project stage and rate card information or create a workspace automatically for the project in OpenAir when importing a project from NetSuite into OpenAir. See the relevant workflow setting descriptions for more information:

- Map the OpenAir project stage from the NetSuite project
- Map the OpenAir rate card from the NetSuite project
- Enable project workspace creation

For information about workflow specific settings, see Projects (Projects) Import Workflow Settings.

For information about field mapping definitions for this workflow, see Projects (Projects) Import Field Mapping Definition.

Requirements and Downstream Dependencies

To activate the Projects (Projects) import workflow, the Customers (Customers) import workflow must also be active. See Customers (Customers) Import Workflow.

Matching project records must exist in both NetSuite and OpenAir before you can export expense reports and timesheets associated with these projects, use purchase orders, or use time and expenses in your billing rules and generate invoices.

A project must exist in NetSuite before a contact or sales order associated with the project can be imported into OpenAir.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
✓	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir project record after you create or update the project record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir project record when the Project form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir project record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Projects (Projects) Import Workflow

The Projects (Projects) workflow is enabled as standard. Use the following steps to set up the workflow.

To set up the Projects (Projects) import workflow in NetSuite

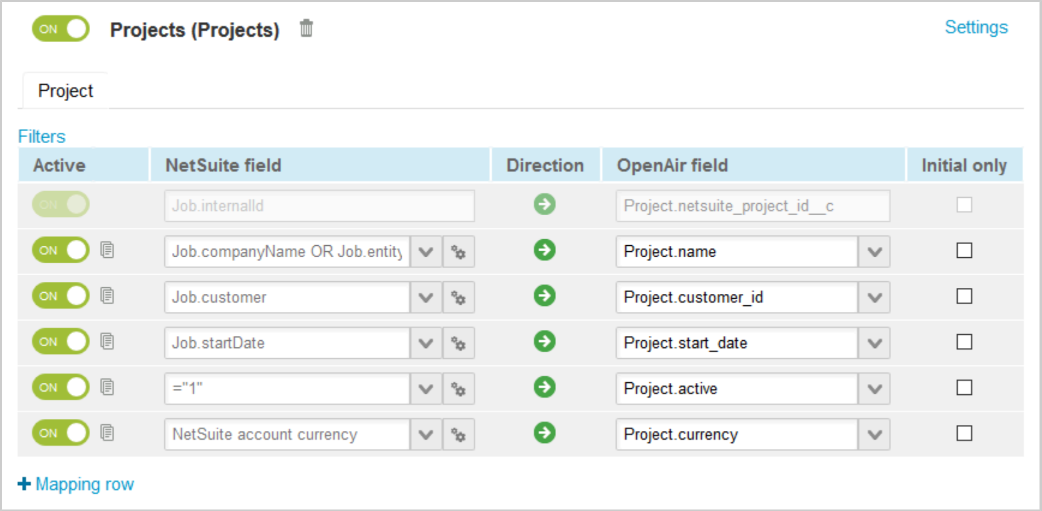
1. (Optional) To mark new project records for export to OpenAir by default:
    a. In NetSuite, do one of the following:
        - Go to Setup > Company > General Preferences (Custom Preferences), if you are using the SuiteScript 1 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
        - Go to Customization > Lists, Records and Fields > Records > [*View list forOpenAir: SRP Integration Settings*] > [*View OpenAir SRP Integration Settings Record*] , if you are using the SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.

    b. Check the **Export Projects by Default** box.

    After you enable this preference, the **Export to OpenAir** [`custentity_oa_export_to_openair`] box is checked by default when a user creates a new project record. Users with the relevant permissions can clear the box on individual records if required.

To set up the Projects (Projects) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector > Mapping.
2. Click **New workflow** then **Projects (Projects)**.
3. Click **Projects (Projects)** in the list of integration workflows.
4. Click **Workflow settings** to view and modify the settings specific to this workflow. For more information about the workflow settings, see Projects (Projects) Import Workflow Settings.
5. Modify the field mapping definition table for the Projects (Projects) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Projects (Projects) Import Field Mapping Definition.
6. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON **Projects (Projects)**

Settings

Project

Filters

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Job.internalId	→	Project.netsuite_project_id__c	□
ON	Job.companyName OR Job.entity	→	Project.name	□
ON	Job.customer	→	Project.customer_id	□
ON	Job.startDate	→	Project.start_date	□
ON	="1"	→	Project.active	□
ON	NetSuite account currency	→	Project.currency	□

+ Mapping row

7. Click **Save**.
8. (Optional) Add the Projects (Projects) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Projects (Projects) Import Workflow Settings

You can control Projects (Projects) import workflow settings from the NetSuite Connector UI. On the OpenAir NetSuite Connector Mapping screen, select the Projects (Projects) import workflow and click **Settings** to view the workflow settings form.

The following settings are available for the Projects (Projects) import workflow:

- Map NetSuite project class to project hierarchy
- Map NetSuite project department to project hierarchy
- Map NetSuite project location to project hierarchy
- Map NetSuite project subsidiary to project hierarchy
- Disable Export to OpenAir filtering for projects in NetSuite integration
- Create new projects based on the project template specified in NetSuite
- Enable expense policy duplication from template
- Enable project workspace creation
- Map the OpenAir project stage from the NetSuite project
- Map the OpenAir rate card from the NetSuite project
- Allow project start date to be updated

Map NetSuite project class to project hierarchy

Check this box to map the project **Class** in NetSuite to a project hierarchy in OpenAir.

- **NetSuite class** — The class is a type of classification used to identify, categorize and track records such as financials, transactions, and employees in NetSuite. Classes are useful for wider segments within your business. For more information about classes, see Departments and Classes Overview (external link to NetSuite SuiteAnswers | Answer ID: 9844).

> **Note:** After you enable classes for your account, you can create class records and use them to categorise transactions and employees. Classifications are not available for customer and project records as standard in NetSuite. However, you can create a List/Record custom entity field to use classes for customers and projects using a list/record field, if required. See Using Classifications for Customers and Projects in NetSuite.

- **OpenAir hierarchy** — You can use hierarchies are used to create multilevel classification trees and categorize records in your OpenAir account. Each hierarchy is associated to one of the following record types: customers, customers, or projects. Multiple hierarchies can apply to the same record type. For more information about hierarchies, see Administrator Guide.

After you check this box and you save the integration configuration, OpenAir NetSuite Connector creates the following records in OpenAir:

- The **Project Class** hierarchy. The Project properties form in OpenAir will show dropdown fields for the Project Class hierarchy if it contains any active nodes — Edit the hierarchy record or use form permissions to hide the Project Class hierarchy dropdown field on Project properties form, if required. You can also set the **Project Class** as required, and choose to display it as a column on list views.

  > **Note:** You can map the NetSuite class to an existing OpenAir hierarchy instead of using the hierarchy created by OpenAir NetSuite Connector. To do so, you need to delete the Project Class hierarchy created by OpenAir NetSuite Connector and edit the existing hierarchy, hierarchy levels, and nodes in OpenAir. This lets you retain the existing hierarchy you are already using in OpenAir.

- Custom fields for hierarchy and hierarchy nodes records. These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.
  - **NetSuite class hierarchy** `[netsuite_hierarchy_class]` — Check box used to identify if the hierarchy maps to the NetSuite class.
  - **NetSuite ID** `[netsuite_hierarchy_node_id]` — Text custom field used to map each level or node in the OpenAir hierarchy to the corresponding NetSuite class record.
  - **NetSuite class level or hierarchy node** `[netsuite_hierarchy_node_class]` — Check box used to identify if the OpenAir hierarchy level or node maps to a NetSuite class depth level or record.

For more information about importing or mapping NetSuite classifications to OpenAir hierarchies, see Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

Important: If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:

- Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
- Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Map NetSuite project department to project hierarchy

Check this box to map the project **Department** in NetSuite to a project hierarchy in OpenAir.

- **NetSuite department** — The department is a type of classification used to identify, categorize and track records such as financials, transactions, and employees in NetSuite. Departments are useful for wider segments within your business. For more information about departments, see Departments and Classes Overview (external link to NetSuite SuiteAnswers | Answer ID: 9844).

  Note: After you enable departments for your account, you can create department records and use them to categorise transactions and employees. Classifications are not available for customer and project records as standard in NetSuite. However, you can create a List/Record custom entity field to use departments for customers and projects using a list/record field, if required. See Using Classifications for Customers and Projects in NetSuite.

- **OpenAir hierarchy** — You can use hierarchies are used to create multilevel classification trees and categorize records in your OpenAir account. Each hierarchy is associated to one of the following record types: customers, customers, or projects. Multiple hierarchies can apply to the same record type. For more information about hierarchies, see Administrator Guide.

After you check this box and you save the integration configuration, OpenAir NetSuite Connector creates the following records in OpenAir:

- The **Project Department** hierarchy. The Project properties form in OpenAir will show dropdown fields for the Project Department hierarchy if it contains any active nodes — Edit the hierarchy record or use form permissions to hide the Project Department hierarchy dropdown field on Project properties form, if required. You can also set the **Project Department** as required, and choose to display it as a column on list views.

  Note: You can map the NetSuite department to an existing OpenAir hierarchy instead of using the hierarchy created by OpenAir NetSuite Connector. To do so, you need to delete the Project Department hierarchy created by OpenAir NetSuite Connector and edit the existing hierarchy, hierarchy levels, and nodes in OpenAir. This lets you retain the existing hierarchy you are already using in OpenAir.

- Custom fields for hierarchy and hierarchy nodes records. These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.

- **NetSuite department hierarchy** [netsuite_hierarchy_department] — Check box used to identify if the hierarchy maps to the NetSuite department.
- **NetSuite ID** [netsuite_hierarchy_node_id] — Text custom field used to map each level or node in the OpenAir hierarchy to the corresponding NetSuite department record.
- **NetSuite department level or hierarchy node** [netsuite_hierarchy_node_department] — Check box used to identify if the OpenAir hierarchy level or node maps to a NetSuite department depth level or record.

For more information about importing or mapping NetSuite classifications to OpenAir hierarchies, see Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

> ⚠ **Important:** If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:
>
> - Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
> - Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Map NetSuite project location to project hierarchy

Check this box to map the project **Location** in NetSuite to a project hierarchy in OpenAir.

- **NetSuite location** — The location is a type of classification used to identify, categorize and track records such as financials, transactions, and employees in NetSuite. Locations are useful for wider segments within your business. For more information about locations, see Locations Overview (external link to NetSuite SuiteAnswers | Answer ID: 9845).

  > ⓘ **Note:** After you enable locations for your account, you can create location records and use them to categorise transactions and employees. Classifications are not available for customer and project records as standard in NetSuite. However, you can create a List/Record custom entity field to use locations for customers and projects using a list/record field, if required. See Using Classifications for Customers and Projects in NetSuite.

- **OpenAir hierarchy** — You can use hierarchies are used to create multilevel classification trees and categorize records in your OpenAir account. Each hierarchy is associated to one of the following record types: customers, customers, or projects. Multiple hierarchies can apply to the same record type. For more information about hierarchies, see Administrator Guide.

After you check this box and you save the integration configuration, OpenAir NetSuite Connector creates the following records in OpenAir:

- The **Project Location** hierarchy. The Project properties form in OpenAir will show dropdown fields for the Project Location hierarchy if it contains any active nodes — Edit the hierarchy record or use form permissions to hide the Project Location hierarchy dropdown field on Project properties form, if required. You can also set the **Project Location** as required, and choose to display it as a column on list views.

> **Note:** You can map the NetSuite location to an existing OpenAir hierarchy instead of using the hierarchy created by OpenAir NetSuite Connector. To do so, you need to delete the Project Location hierarchy created by OpenAir NetSuite Connector and edit the existing hierarchy, hierarchy levels, and nodes in OpenAir. This lets you retain the existing hierarchy you are already using in OpenAir.

- Custom fields for hierarchy and hierarchy nodes records. These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.
  - **NetSuite location hierarchy** [`netsuite_hierarchy_location`] — Check box used to identify if the hierarchy maps to the NetSuite location.
  - **NetSuite ID** [`netsuite_hierarchy_node_id`] — Text custom field used to map each level or node in the OpenAir hierarchy to the corresponding NetSuite location record.
  - **NetSuite location level or hierarchy node** [`netsuite_hierarchy_node_location`] — Check box used to identify if the OpenAir hierarchy level or node maps to a NetSuite location depth level or record.

For more information about importing or mapping NetSuite classifications to OpenAir hierarchies, see Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

> **Important:** If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:
>
> - Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
> - Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Map NetSuite project subsidiary to project hierarchy

Check this box to map the project **Subsidiary** in NetSuite to a project hierarchy in OpenAir.

- **NetSuite subsidiary** — If you use NetSuite OneWorld, subsidiaries are the primary classification used to organize your NetSuite records, and additional classification records are associated to a subsidiary. Each subsidiary represents a separate company within your global organization. Subsidiaries can be international or domestic. For more information about subsidiaries, see Subsidiaries in OneWorld (external link to NetSuite SuiteAnswers | Answer ID: 9872).
- **OpenAir hierarchy** — You can use hierarchies are used to create multilevel classification trees and categorize records in your OpenAir account. Each hierarchy is associated to one of the following

record types: employees, customers, or projects. Multiple hierarchies can apply to the same record type. For more information about hierarchies, see Administrator Guide.

After you check this box and you save the integration configuration, OpenAir NetSuite Connector creates the following records in OpenAir:

- The **Project Subsidiary** hierarchy. The Project properties form in OpenAir will show dropdown fields for the Project Subsidiary hierarchy if it contains any active nodes — Edit the hierarchy record or use form permissions to hide the Project Subsidiary hierarchy dropdown field on Project properties form, if required. You can also set the **Project Subsidiary** as required, and choose to display it as a column on list views.

  > **Note:** You can map the NetSuite subsidiary to an existing OpenAir hierarchy instead of using the hierarchy created by OpenAir NetSuite Connector. To do so, you need to delete the Project Subsidiary hierarchy created by OpenAir NetSuite Connector and edit the existing hierarchy, hierarchy levels, and nodes in OpenAir. This lets you retain the existing hierarchy you are already using in OpenAir.

- Custom fields for hierarchy and hierarchy nodes records. These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.
  - **NetSuite subsidiary hierarchy** [`netsuite_hierarchy_subsidiary`] — Check box used to identify if the hierarchy maps to the NetSuite subsidiary .
  - **NetSuite ID** [`netsuite_hierarchy_node_id`] — Text custom field used to map each level or node in the OpenAir hierarchy to the corresponding NetSuite subsidiary record.
  - **NetSuite subsidiary level or hierarchy node** [`netsuite_hierarchy_node_subsidiary`] — Check box used to identify if the OpenAir hierarchy level or node maps to a NetSuite subsidiary depth level or record.

For more information about importing or mapping NetSuite classifications to OpenAir hierarchies, see Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

> **Important:** If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:
>
> - Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
> - Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Disable Export to OpenAir filtering for projects in NetSuite integration

Check this box to import all projects from NetSuite to OpenAir.

- When checked, OpenAir NetSuite Connector imports project records from NetSuite into OpenAir even if the **Export to OpenAir** [`custentity_oa_export_to_openair`] box is not checked on the NetSuite project record.
- Otherwise, OpenAir NetSuite Connector imports a project record only if the **Export to OpenAir** [`custentity_oa_export_to_openair`] box is checked.

Create new projects based on the project template specified in NetSuite

Check this box to add the ability to select an **OpenAir Project Template** when creating a new project record in NetSuite. When you import the project into OpenAir, the OpenAir project contains project-related information copied from the template.

Additional setup is required before you can use this feature:

- Create project templates in OpenAir. For more information about creating project templates in OpenAir, see Administrator Guide.

  ✓ **Tip:** Set the **Project stage** to `Templates` for ease of organizing and maintaining your project templates in OpenAir.

- Create OpenAir: Project Template records in NetSuite to match the OpenAir project templates you want to use.

  ⚠ **Important:** Review the following guidelines:

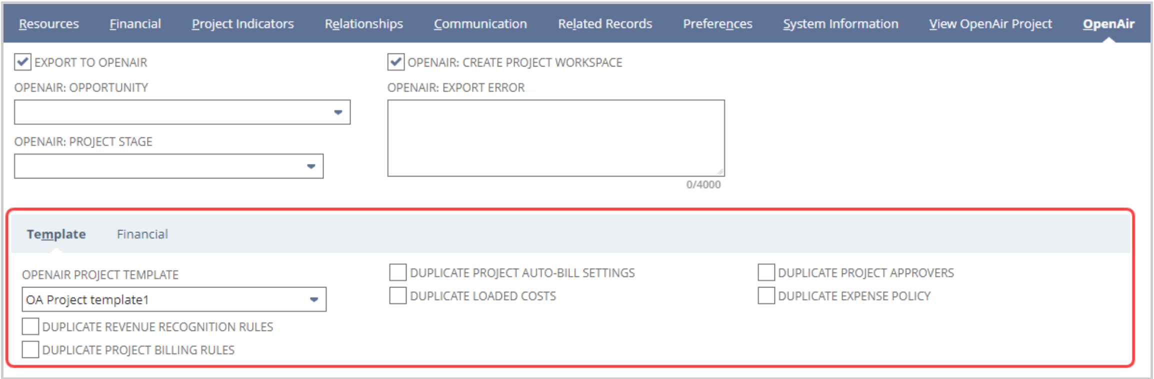
  - The OpenAir: Project Template **Name** must be exactly the same as the **Project name** on the properties form for the project template in OpenAir. If the OpenAir: Project Template **Name** does not match any project template in OpenAir, the Projects (Projects) import workflow creates a new project in OpenAir but does not copy any information from any project template.
  - Make sure project template names are unique. If there are multiple active project templates with the same name in OpenAir, the Projects (Projects) import workflow uses the first project template to create the new project in OpenAir.

After you set up project templates in both NetSuite in OpenAir, you can select an **OpenAir Project Template** on the project record in NetSuite (Template subtab within the OpenAir subtab) and check the **Duplicate <project-related record type>** boxes for the project-related information you want to copy from the project template.

- The project tasks associated with the project template are copied by default including task IDs, names, priority, phase, task type, planned hours, percent complete, dates, assignments, and other standard task information.
- The following information is copied based on the **Duplicate <project-related record type>** boxes checked on the project record in NetSuite: Project pricing, budget, approvers, cost types, loaded costs, expense policy, dashboard settings, notification settings, phase and task custom fields, bookings, issues, billing rules, auto-bill settings recognition rules, recognition auto-run settings and invoice layout settings.

Important: Review the following guidelines:

- To be able to copy the expense policy from the project template, make sure you check the **Enable expense policy duplication from template** box in the workflow setting for the Projects (Projects import workflow in OpenAir. See Enable expense policy duplication from template.
- The ability to create project templates complete with bookings and then copy all bookings from your templates across to the new projects you create from these templates was introduced in the April 2022 OpenAir release. For more information, see the help topic Copying Bookings with Projects.



☑ EXPORT TO OPENAIR

☑ OPENAIR: CREATE PROJECT WORKSPACE

OPENAIR: OPPORTUNITY

OPENAIR: EXPORT ERROR

OPENAIR: PROJECT STAGE

OPENAIR PROJECT TEMPLATE: OA Project template1

□ DUPLICATE REVENUE RECOGNITION RULES

□ DUPLICATE PROJECT BILLING RULES

□ DUPLICATE PROJECT AUTO-BILL SETTINGS

□ DUPLICATE LOADED COSTS

□ DUPLICATE PROJECT APPROVERS

□ DUPLICATE EXPENSE POLICY

The OpenAir: Project Templates [customrecord_oa_project_templates] custom record type and the following custom entity fields are created when you install the NetSuite OpenAir SRP Integration bundle — Bundle ID 369637 (previously 2851) 1.10 or later version.

- **OpenAir Project Template** [custentity_oa_project_template]
- **Duplicate project pricing** [custentity_oa_copy_project_pricing] — Hidden by default
- **Duplicate project budget** [custentity_oa_copy_project_budget] — Hidden by default
- **Duplicate project approvers** [custentity_oa_copy_approver]
- **Duplicate cost types** [custentity_oa_copy_cost_type] — Hidden by default
- **Duplicate loaded costs** [custentity_oa_copy_loaded_cost]
- **Duplicate expense policy** [custentity_oa_copy_expense_policy]
- **Duplicate dashboard settings** [custentity_oa_copy_dashboard_settings] — Hidden by default
- **Duplicate notification settings** [custentity_oa_copy_notification_settings] — Hidden by default
- **Duplicate phase and task custom fields** [custentity_oa_copy_custom_fields] — Hidden by default
- **Duplicate issues** [custentity_oa_copy_issues] — Hidden by default
- **Duplicate project billing rules** [custentity_oa_copy_prbill_rules]
- **Duplicate project auto-bill settings** [custentity_oa_copy_prbill_auto_settings]
- **Duplicate revenue recognition rules** [custentity_oa_copy_revrec_rules]
- **Duplicate recognition auto-run settings** [custentity_oa_revrec_auto_settings] — Hidden by default

- **Duplicate invoice layout settings** [custentity_oa_copy_invoice_layout] — Hidden by default

The following custom entity field is created when you install the NetSuite OpenAir SRP Integration bundle — Bundle ID 369637 (previously 2851) 3.0 or later version:

- **Duplicate bookings** [custentity_oa_copy_bookings]

Some of the fields on the Template subtab are hidden by default. NetSuite account administrators can edit each custom entity field to show or hide the field on the project record in NetSuite and to set a default value for the field, if required. See Editing a Custom Field (external link to NetSuite SuiteAnswers | Answer ID: 42612).

> ⚠ **Important:** If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:
>
> - Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
> - Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Enable expense policy duplication from template

Check this box to copy the expense policy from the OpenAir project template when you select an **OpenAir Project Template** and check the **Duplicate expense policy** box on the project record in NetSuite and then import the project from NetSuite into OpenAir.

> ⚠ **Important:** The **Create new projects based on the project template specified in NetSuite** box must also be checked. See Create new projects based on the project template specified in NetSuite.

Enable project workspace creation

Check this box to create a workspace automatically for project imported into OpenAir if the **OpenAir: Create Project Workspace** box is checked on the project record (OpenAir subtab) in NetSuite.

If the **Enable project workspace creation** box (workflow setting) and the **OpenAir: Create Project Workspace** box on the NetSuite project record are both checked, OpenAir NetSuite Connector creates a workspace associated with the project when importing the project from NetSuite to OpenAir.

Note: The **OpenAir: Create Project Workspace** [custentity_oa_create_project_workspace] custom field is created when you install the NetSuite OpenAir SRP Integration bundle — Bundle ID 369637 (previously 2851) 1.22 or later version.

Important: If this option is enabled, NetSuite Connector uses software logic associated with the Project form to save changes in OpenAir when importing project records from NetSuite to OpenAir. This has the following consequences:

- Form scripts associated with the Project form and triggered by an “On submit”, “Before save”, or “After save” event in OpenAir will run for all imported project records. This will impact the performance of your integration runs and may result in errors related to scripting governance limits. You may be able to reduce the performance impact by setting a dedicated integration user and making sure the script do not run if the dedicated integration user executes the script.
- Form validation in OpenAir will be enforced for all imported project records. A required field with no value or form permission may prevent imported records to be saved in OpenAir. NetSuite Connector will log an error for each imported record that fails to save in OpenAir.

Map the OpenAir project stage from the NetSuite project

Check this box to import project stage information when importing a project record from NetSuite into OpenAir project. This lets you set the project stage using the **OpenAir: Project Stage** custom field on the project record (OpenAir subtab) in NetSuite and import it into OpenAir along with the other project record information.

If you want to import the project stage information only the first time a project record is imported from NetSuite into OpenAir, use a field mapping definition row and check the box in the **initial only** column. The project stage information will be imported when the integration creates a record in OpenAir but not when it updates a record already imported.

Important: Review the following guidelines:

- Make sure you have the exact same project stage records in both NetSuite and OpenAir, before you start using this feature. OpenAir NetSuite Connector returns an integration error if project stages do not match. In NetSuite, create OpenAir: Project Stage records to match each active OpenAir project stages.

  The OpenAir: Project Stages [`customrecord_oa_project_stages`] custom record type and **OpenAir: Project Stage** [`custentity_oa_project_stage`] custom entity field are created when you install the NetSuite OpenAir SRP Integration bundle — Bundle ID 369637 (previously 2851) 1.21 or later version.
- To review project stages configured in OpenAir, go to Administration > Application Settings > Projects > Project Stages.
- Remember to create or update project stages in both applications when you make changes.
- If the **OpenAir: Project Stage** field is empty on the NetSuite project record, the integration does not update the project stage on the OpenAir project record. This is to prevent any existing project stage being removed accidentally from a project in OpenAir.

Map the OpenAir rate card from the NetSuite project

Check this box to import rate card information when importing a project record from NetSuite into OpenAir project. This lets you set the rate card in NetSuite using the **OpenAir: Project Rate Card** custom field on the project record (OpenAir subtab) in NetSuite and import it into OpenAir along with the other project record information.

Important: Review the following guidelines:

- Make sure you have the exact same rate card records in both NetSuite and OpenAir, before you start using this feature. OpenAir NetSuite Connector returns an integration error if rate cards do not match. In NetSuite, create OpenAir: Project Rate Card records to match each active OpenAir rate cards.

  The OpenAir: Project Rate Card [`customrecord_oa_project_rate_card`] custom record type and **OpenAir: Project Rate Card** [`custentity_oa_project_rate_card`] custom entity field are created when you install the NetSuite OpenAir SRP Integration bundle — Bundle ID 369637 (previously 2851) 1.28 or later version.
- To review rate cards configured in OpenAir, go to Administration > Global Settings > Jobs, Rates> Rate Cards.
- Remember to create or update rate cards in both applications when you make changes.

Allow project start date to be updated

Check this box to update the **Start date** for the OpenAir project when you update the **Start Date** on the project record in NetSuite then run the integration for the Projects (Projects) workflow.

Projects (Projects) Import Field Mapping Definition

The Projects (Projects) import workflow includes one mapping group:

- Project [Integration point: JOB_TO_PROJECT] — See Project Mapping Group Tab.

Project Mapping Group Tab

Use the Project mapping group to map information on project records in OpenAir with information on project records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes project name, associated customer, currency, and start date. Filters are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Project [Job] Schema: Job (2020.1)|Job (2023.1) XSD: JobSearchRowBasic platformCommon (2020.1)|platformCommon (2023.1)	Project [project] Data Dictionary: project	Project [JOB_TO_PROJECT]

Default and Custom Filters

By default, project records are imported from NetSuite into OpenAir if the following conditions are all met:

- The **Inactive** box is NOT checked on the project record in NetSuite.
- The **Export to OpenAir** [custentity_oa_export_to_openair] box is checked on the project record in NetSuite. You can modify this behavior — See Disable Export to OpenAir filtering for projects in NetSuite integration.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definitions for the Project mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Note: Features controlled by the following workflow settings add additional information associated with the project when creating or updating a project using the Projects (Projects) import workflow. For more information, see the settings descriptions.

- Map NetSuite project class to project hierarchy — Can be used to import project class information as a hierarchy node associated with the project.
- Map NetSuite project department to project hierarchy — Can be used to import project department information as a hierarchy node associated with the project.
- Map NetSuite project location to project hierarchy — Can be used to import project location information as a hierarchy node associated with the project.
- Map NetSuite project subsidiary to project hierarchy — Can be used to import project subsidiary information as a hierarchy node associated with the project.
- Create new projects based on the project template specified in NetSuite — Can be used to add project information from an OpenAir project template, including the work breakdown structure.
- Enable expense policy duplication from template — Can be used to add expense policy information from the OpenAir project template.
- Enable project workspace creation — Can be used to create a workspace associated with the project.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

See the OpenAir Help Center for detailed information about additional field mapping definitions for the Project mapping group tab that may be required or useful in some cases. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Services (Services Items) Import Workflow

The Services (Services Items) import workflow lets you:

- Create new service item records in NetSuite first and then import them into OpenAir as service records.
- Update service item records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding service record in OpenAir.

Note: You can also:

- Import items of all types (both service items and other types of items) from NetSuite into OpenAir as services records using the Services (Services Items) import workflow. If you use NetSuite CRM, you can import items of all subtypes from NetSuite into OpenAir. This can be useful to import other type of items that hold billing information. See Disable NetSuite service type filtering for services in NetSuite integration and Disable NetSuite item subtype filtering for services in NetSuite integration (for NetSuite CRM+ accounts only).
- Import purchase item records (items used on purchase orders) from NetSuite into OpenAir as product records using the Products (Purchase Items) import workflow. See Products (Purchase Items) Import Workflow.

Important: NetSuite is the primary application for item records. Changes made to service records in OpenAir after the initial import are not exported back to NetSuite.

For information about workflow specific settings, see Services (Service Items) Import Workflow Settings.

For information about field mapping definitions for this workflow, see Services (Service Items) Import Field Mapping Definition.

Requirements and Downstream Dependencies

Matching NetSuite service item and OpenAir service records must exist before you can export invoices and credit memos from OpenAir to NetSuite.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
✓	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir service record after you create or update the service item record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir service record when the service item form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir service record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Services (Service Items) Import Workflow

The Services (Service Items) import workflow is available as standard. Use the following steps to set up the workflow.

To set up the Services (Service Items) import workflow in NetSuite

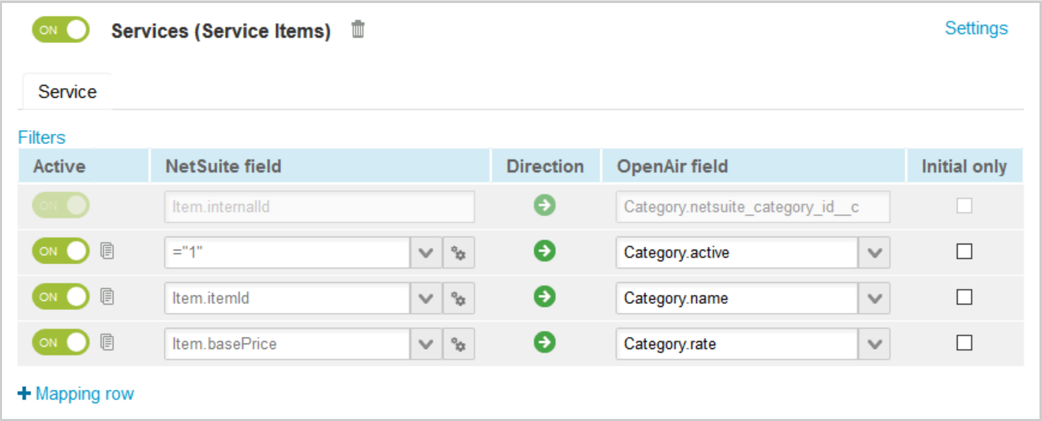
1. (Optional) To mark new service item records for export to OpenAir by default:
    a. In NetSuite, do one of the following:
        - Go to Setup > Company > General Preferences (Custom Preferences), if you are using the SuiteScript 1 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
        - Go to Customization > Lists, Records and Fields > Records > [*View list forOpenAir: SRP Integration Settings*] > [*View OpenAir SRP Integration Settings Record*] , if you are using the SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.

    b. Check the **Export Service Items by Default** box.

    After you enable this preference, the **Export to OpenAir** [`custitem_oa_export_to_openair`] box is checked by default when a user creates a new service item record. Users with the relevant permissions can clear the box on individual records if required.

To set up the Services (Service Items) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector > Mapping.
2. Click **New workflow** then **Services (Service Items)**.
3. Click **Services (Service Items)** in the list of integration workflows.
4. Click **Workflow settings** to view and modify the settings specific to this workflow. For more information about the workflow settings, see Services (Service Items) Import Workflow Settings.
5. Modify the field mapping definition table for the Services (Service Items) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Services (Service Items) Import Field Mapping Definition.
6. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON Services (Service Items) Settings

Service

Filters

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Item.internalId	→	Category.netsuite_category_id__c	□
ON	="1"	→	Category.active	□
ON	Item.itemId	→	Category.name	□
ON	Item.basePrice	→	Category.rate	□

+ Mapping row

7. Click **Save**.
8. (Optional) Add the Services (Service Items) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Services (Service Items) Import Workflow Settings

You can control Services (Service Items) import workflow settings from the NetSuite Connector UI. On the OpenAir NetSuite Connector Mapping screen, select the Services (Service Items) import workflow and click **Settings** to view the workflow settings form.

The following settings are available for the Services (Service Items) import workflow:

- Disable Export to OpenAir filtering for services in NetSuite integration
- Disable NetSuite service type filtering for services in NetSuite integration
- Disable NetSuite item subtype filtering for services in NetSuite integration (for NetSuite CRM+ accounts only)

Disable Export to OpenAir filtering for services in NetSuite integration

Check this box to import all service items from NetSuite into OpenAir.

- When checked, OpenAir NetSuite Connector imports item records from NetSuite into OpenAir as services even if the **Export to OpenAir** box is not checked on the NetSuite item record.
- Otherwise, OpenAir NetSuite Connector imports an item record only if the **Export to OpenAir** box is checked on the item record in NetSuite.

> **Note:** By default, item records are imported from NetSuite into OpenAir if the following conditions are also met:
>
> - The item **Type** is “Service”. You can modify this behavior to items of all types — See Disable NetSuite service type filtering for services in NetSuite integration.
> - The item **Subtype** is “For Sale” or “For Resale”. You can modify this behavior to disable this condition if you use NetSuite CRM — See Disable NetSuite item subtype filtering for services in NetSuite integration (for NetSuite CRM+ accounts only).

Disable NetSuite service type filtering for services in NetSuite integration

Check this box to import items of all types from NetSuite into OpenAir as services if the **Export to OpenAir** box is checked on the NetSuite item record.

- When checked, OpenAir NetSuite Connector imports items of all types from NetSuite into OpenAir as services.
- Otherwise, OpenAir NetSuite Connector imports only service items (items of type “Service”).

By default, item records are imported from NetSuite into OpenAir if the following conditions are also met:

- The item **Subtype** is “For Sale” or “For Resale”. You can modify this behavior to disable this condition if you use NetSuite CRM — See Disable NetSuite item subtype filtering for services in NetSuite integration (for NetSuite CRM+ accounts only).
- The **Export to OpenAir** [`custitem_oa_export_to_openair`] box is checked on the item record in NetSuite. You can modify this behavior to import items even if the **Export to OpenAir**

[custitem_oa_export_to_openair] box is not checked — See Disable Export to OpenAir filtering for services in NetSuite integration.

Disable NetSuite item subtype filtering for services in NetSuite integration (for NetSuite CRM+ accounts only)

If you use NetSuite CRM, check this box to import items of all subtypes from NetSuite into OpenAir as services.

- When checked, OpenAir NetSuite Connector imports items of all subtypes from NetSuite into OpenAir.
- Otherwise, OpenAir NetSuite Connector imports only items of subtype “For Sale” or ”For Resale”.

> **Note:** NetSuite SOAP web services support for the subtype property is limited if you use NetSuite CRM. Disabling the default subtype filter condition to ensure NetSuite items are imported from a NetSuite CRM account into OpenAir.

By default, item records are imported from NetSuite into OpenAir if the following conditions are also met:

- The item **Type** is “Service”. You can modify this behavior to items of all types — See Disable NetSuite service type filtering for services in NetSuite integration.
- The **Export to OpenAir** [custitem_oa_export_to_openair] box is checked on the item record in NetSuite. You can modify this behavior to import items even if the **Export to OpenAir** [custitem_oa_export_to_openair] box is not checked — See Disable Export to OpenAir filtering for services in NetSuite integration.

Services (Service Items) Import Field Mapping Definition

The Services (Service Items) import workflow includes one mapping group:

- Service [Integration point: ITEM_TO_CATEGORY] — See Service Mapping Group.

Service Mapping Group

Use the Service mapping group to map information about service records in OpenAir with information on service item records in NetSuite.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes service name, rate, and income and deferred revenue account. Filters are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Item [Item] Schema: <Type><Subtype>Item (2020.1)|<Type><Subtype>Item (2023.1) XSD: ItemSearchRowBasic — platformCommon (2020.1)|platformCommon (2023.1)	Service [category] Data Dictionary: category	Service [ITEM_TO_CATEGORY]

Note: The relevant SOAP objects as described in the NetSuite Schema Browser include:

- ServiceSaleItem or ServiceResaleItem (always)
- DiscountItem, and MarkupItem, and other <Type><Subtype>Item objects, where <Type> is the item **Type** and <Subtype> is the item **Subtype** (such as Sale, Resale, or Purchase) depending on the your integration configuration. See Default and Custom Filters

Default and Custom Filters

By default, Item records are imported from NetSuite into OpenAir if the following conditions are all met:

- The **Inactive** box is NOT checked on the item record in NetSuite.
- The **Export to OpenAir** [custitem_oa_export_to_openair] box is checked on the item record in NetSuite. You can modify this behavior — See Disable Export to OpenAir filtering for services in NetSuite integration.

  Note: Item for purchase and for resale records may also include the **Export to OpenAir Product** [custitem_oa_export_to_openair_product] box if the Products (Purchase Items) import workflow is also enabled. See Products (Purchase Items) Import Workflow.

  - Check the **Export to OpenAir** [custitem_oa_export_to_openair] to import the item into OpenAir as a service using the Services (Service Items) import workflow.
  - Check the **Export to OpenAir Product** [custitem_oa_export_to_openair_product] to import the item into OpenAir as a product using the Products (Purchase Items) import workflow.
  - Check both boxes to import the item into OpenAir as both a service and a product using the Services (Service Items) and Products (Purchase Items) import workflows.

- The item **Type** and **Subtype** is “Service For Sale” or “Service For Resale” in NetSuite, or the item **Type** . You can modify this behavior to:
  - Import items of type “Discount” or “Markup” if the **Create billing rules for NetSuite discount and markup lines on sales orders** box is checked on the Invoices (Invoices) export workflow settings. See Create billing rules for NetSuite discount and markup lines on sales orders.
  - Import items of other types and subtypes— See Disable NetSuite service type filtering for services in NetSuite integration and Disable NetSuite item subtype filtering for services in NetSuite integration (for NetSuite CRM+ accounts only).

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definitions for the Service mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

⚠ **Important:** Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

See the OpenAir Help Center for detailed information about additional field mapping definitions for the Service mapping group tab that may be required or useful in some cases. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Expense Items (Expense Categories) Import Workflow

The Expense Items (Expense Categories) import workflow lets you:

- Create new expense category records in NetSuite first and then import them into OpenAir as expense item records.
- Update expense category records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding expense item record in OpenAir.

⚠ **Important:** NetSuite is the primary application for item records. Changes made to expense item records in OpenAir after the initial import are not exported back to NetSuite.

For information about field mapping definitions for this workflow, see Expense Items (Expense Categories) Import Field Mapping Definition.

Requirements and Downstream Dependencies

Matching NetSuite expense category and OpenAir expense items records must exist before you can export expense reports from OpenAir to NetSuite.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir expense item record after you create or update the expense category record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir expense item record when the expense category form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir expense item record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

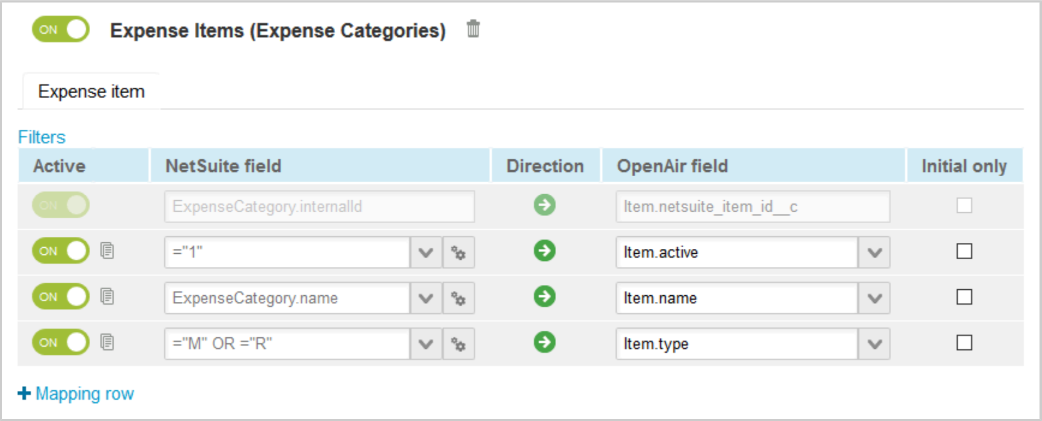
Setting up the Expense Items (Expense Categories) Import Workflow

The Services (Service Items) workflow is available as standard. Use the following steps to set up the workflow.

No additional setting is required in NetSuite.

To set up the Expense Items (Expense Categories) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector > Mapping.
2. Click **New workflow** then **Expense Items (Expense Categories)**.
3. Click **Expense Items (Expense Categories)** in the list of integration workflows.
4. Modify the field mapping definition table for the Expense Items (Expense Categories) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Expense Items (Expense Categories) Import Field Mapping Definition.
5. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON **Expense Items (Expense Categories)**

Expense item

Filters

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	ExpenseCategory.internalId	➔	Item.netsuite_item_id__c	□
ON	="1"	➔	Item.active	□
ON	ExpenseCategory.name	➔	Item.name	□
ON	="M" OR ="R"	➔	Item.type	□

+ Mapping row

6. Click **Save**.
7. (Optional) Add the Expense Items (Expense Categories) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Expense Items (Expense Categories) Import Field Mapping Definition

The Expense Items (Expense Categories) import workflow includes one mapping group:

- Expense Item [Integration point: `EXPENSE_CATEGORY_TO_ITEM`] — See Expense Item Mapping Group.

Expense Item Mapping Group

Use the Expense Item mapping group to map information on expense item records in OpenAir with information on expense category records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes expense item name and type. Filters are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [`SOAP object`]	OpenAir Record Type [`table`]	Mapping Group [`Integration point`]
Expense Category [`ExpenseCategory`] Schema: ExpenseCategory (2020.1)|ExpenseCategory (2023.1) XSD: `ItemSearchRowBasic` platformCommon (2020.1)|platformCommon (2023.1)	Expense Item [`item`] Data Dictionary: item	Expense item [`EXPENSE_CATEGORY_TO_ITEM`]

Default and Custom Filters

By default, expense category records are imported from NetSuite into OpenAir if the **Inactive** box is NOT checked on the expense category record in NetSuite. There is no **Export to OpenAir** box on the expense category record in NetSuite. All active expenses categories are imported from NetSuite into OpenAir when the Expense Items (Expense Categories) import workflow is active.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Expense Item mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

> ⚠ **Important:** Review the following guidelines
>
> - You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
> - For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
> - Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Contacts (Contacts) Import Workflow

The Contacts (Contacts) import workflow lets you:

- Create new contact records in NetSuite first and then import them into OpenAir.
- Update contact records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding contact record in OpenAir.

> ⚠ **Important:** NetSuite is the primary application for contact records. Changes made to contact records in OpenAir after the initial import are not exported back to NetSuite.

You can use additional integration settings on the OpenAir NetSuite Connector administration form to:

- Import all contacts associated with the customers you import from NetSuite into OpenAir — See Import all NetSuite contacts associated with NetSuite customers marked 'Export to NS OpenAir'
- Import all contacts associated with the projects you import from NetSuite into OpenAir — See Import all NetSuite contacts associated with NetSuite projects marked 'Export to NS OpenAir'.

For information about field mapping definitions for this workflow, see Contacts (Contacts) Import Field Mapping Definition.

Requirements and Downstream Dependencies

Matching customer records must exist in both NetSuite and OpenAir before you can import contacts associated with these customers or these customers’ projects from NetSuite into OpenAir.

Matching contact records must exist in both NetSuite and OpenAir before you can import projects associated with these contacts from NetSuite into OpenAir or export projects from OpenAir to NetSuite.

Supported Features

Workflow Settings	**Filters**	**Advanced Mapping**	**Real-Time**	**OpenAir Within NetSuite**	**Monitored Fields**
		✓		✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir contact record after you create or update the contact record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- View and edit the OpenAir contact record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

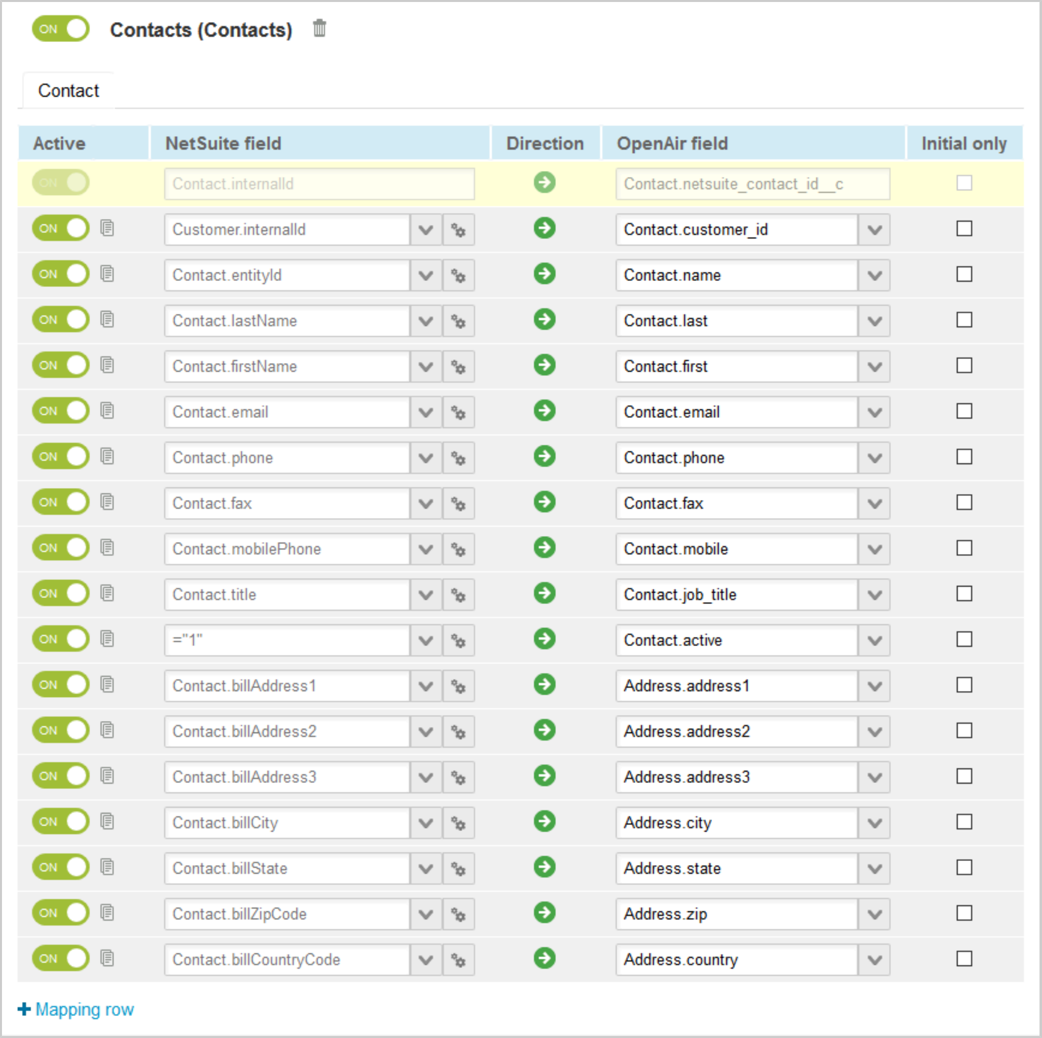
Setting up the Contacts (Contacts) Import Workflow

The Contacts (Contacts) workflow is available as standard. Use the following steps to set up the workflow.

No additional setting is required in NetSuite.

To set up the Contacts (Contacts) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector > Mapping.
2. Click **New workflow** then **Contacts (Contacts)**.
3. Click **Contacts (Contacts)** in the list of integration workflows.
4. Modify the field mapping definition table for the Contacts (Contacts) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Contacts (Contacts) Import Field Mapping Definition.
5. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Contact.internalId	→	Contact.netsuite_contact_id__c	☐
ON	Customer.internalId	→	Contact.customer_id	☐
ON	Contact.entityId	→	Contact.name	☐
ON	Contact.lastName	→	Contact.last	☐
ON	Contact.firstName	→	Contact.first	☐
ON	Contact.email	→	Contact.email	☐
ON	Contact.phone	→	Contact.phone	☐
ON	Contact.fax	→	Contact.fax	☐
ON	Contact.mobilePhone	→	Contact.mobile	☐
ON	Contact.title	→	Contact.job_title	☐
ON	="1"	→	Contact.active	☐
ON	Contact.billAddress1	→	Address.address1	☐
ON	Contact.billAddress2	→	Address.address2	☐
ON	Contact.billAddress3	→	Address.address3	☐
ON	Contact.billCity	→	Address.city	☐
ON	Contact.billState	→	Address.state	☐
ON	Contact.billZipCode	→	Address.zip	☐
ON	Contact.billCountryCode	→	Address.country	☐

6. Click **Save**.
7. (Optional) Add the Contacts (Contacts) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Contacts (Contacts) Import Field Mapping Definition

The Contacts (Contacts) import workflow includes one mapping group:

- Contact [Integration point: CONTACT_TO_CONTACT] — See Contact Mapping Group.

Contact Mapping Group

Use the Contact mapping group to map information on the OpenAir contact record with information on the NetSuite contact record. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.

- Default Field Mappings — Overview of default filters and how to modify the default behavior.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes customer internal ID, contact name, email, telephone, and address. Filters are supported.
- Additional Field Mapping — Best practice guidelines for custom field mapping definitions.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	**OpenAir Record Type [table]**	**Mapping Group [Integration point]**
Contact [Contact] Schema: Contact (2020.1)|Contact (2023.1) XSD: ContactSearchRowBasic platformCommon (2020.1)|platformCommon (2023.1)	Contact [contact and address] Data Dictionary: contact and address	Contact [CONTACT_TO_CONTACT]

Default and Custom Filters

By default, contact records are imported from NetSuite into OpenAir if the following conditions are all met:

- The **Inactive** box is NOT checked on the contact record in NetSuite.
- An OpenAir customer record exists matching the customer associated with the contact record in NetSuite, or the customer associated with the project associated with the contact record, if the **Import all NetSuite contacts associated with NetSuite projects marked 'Export to NS OpenAir'** box is checked on the OpenAir NetSuite Connector administration from .
- The **Export to OpenAir** [custentity_oa_export_to_openair] box is checked on the contact record in NetSuite. You can modify this behavior to:
    - Import all contacts associated with customers you import from NetSuite into OpenAir by checking the following box on the OpenAir NetSuite Connector administration form: Import all NetSuite contacts associated with NetSuite customers marked 'Export to NS OpenAir'. The **Export to OpenAir** [custentity_oa_export_to_openair] box is checked on the customer record associated with the contact in NetSuite.
    - Import all contacts associated with projects you import from NetSuite into OpenAir by checking the following box on the OpenAir NetSuite Connector administration form: Import all NetSuite contacts associated with NetSuite projects marked 'Export to NS OpenAir'. The **Export to OpenAir** [custentity_oa_export_to_openair] box is checked on the project record associated with the contact in NetSuite.

Custom filters are not supported for this mapping group.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Contact mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mapping

⚠ **Important:** Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

Review the following guidelines:

- The contact entity form in OpenAir includes the **Copy customer address** box. This is not a field but UI logic to populate the address fields with the current values from the customer record. As this is not a field, you cannot set it using field mapping definition when importing a record. However, you could use advanced field mapping definition to lookup the address of the customer associated with the contact in OpenAir and use the values of the customer address fields to set the contact address fields as part of the Contacts (Contacts) import workflow.

Employees (Vendors) Import Workflow

The Employees (Vendors) import workflow lets you:

- Create new vendor records in NetSuite first and then import them into OpenAir as employee records.
- Update vendor records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding employee record in OpenAir.

You can also use the Vendors (Vendors) import workflow to import NetSuite vendor records into OpenAir as vendor records. Both workflows can be used in conjunction. Additional setup is required to enable the Vendors (Vendors) import workflow, if the Employees (Vendors) import workflow is already enabled. See Setting up the Vendors (Vendors) Import Workflow.

⚠ **Important:** NetSuite is the primary application for vendor records. Changes made to employee records in OpenAir after the initial import are not exported back to NetSuite.

For information about workflow specific settings impacting this workflow, see Employees (Vendors) Import Workflow Settings.

For information about field mapping definitions for this workflow, see Employees (Vendors) Import Field Mapping Definition.

Requirements and Downstream Dependencies

Matching NetSuite vendor records and OpenAir employee records must exist before you can export expense reports from OpenAir to NetSuite as vendor bills.

Using the Vendors (Vendors) and Employees (Vendors) import workflows in conjunction enables you to import vendor bills or purchase orders from NetSuite into OpenAir as purchase orders, which requires matching vendor records in both applications and to export expense reports from OpenAir to NetSuite as vendor bills, which requires the associated employee record in OpenAir being matched with a vendor record in NetSuite. See POs (Vendor Bills) Import Workflow, POs (Purchase Orders) Import Workflow and Expense Reports (Expense Reports) Export Workflow.

When using the Vendors (Vendors) and Employees (Vendors) import workflows in conjunction, use the **OpenAir: User or Vendor** [`custentity_oa_user_or_vendor`] field on the vendor record in NetSuite to determine whether the vendor should be imported into OpenAir as a vendor or as an employee (user).

> **Note:** You can only set the **OpenAir: User or Vendor** [`custentity_oa_user_or_vendor`] on the vendor record in NetSuite if the following conditions are all met. The field is read-only otherwise.
>
> - The **Expense Report – Vendor Bill Integration** box is checked in the integration preferences.
> - Either the **Vendor Bill to OpenAir PO Integration** box or the **Enable OpenAir Purchase Integration** box is checked in the integration preferences.
> - The **Export to OpenAir** [`custentity_oa_export_to_openair`] box is checked on the vendor record.
>
> For more information about setting integration preferences, see Configuring Integration Preferences and Terminology Preferences in NetSuite.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir employee record after you create or update the vendor record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir employee record when the Vendor form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir employee record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Employees (Vendors) Import Workflow

Use the following steps to set up the workflow.

To set up the Employees (Vendors) import workflow in NetSuite

1. (Optional) To mark new vendor records for export to OpenAir by default:
   a. In NetSuite, do one of the following:

- Go to Setup > Company > General Preferences (Custom Preferences), if you are using the SuiteScript 1 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
- Go to Customization > Lists, Records and Fields > Records > [*View list forOpenAir: SRP Integration Settings*] > [*View OpenAir SRP Integration Settings Record*] , if you are using the SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.

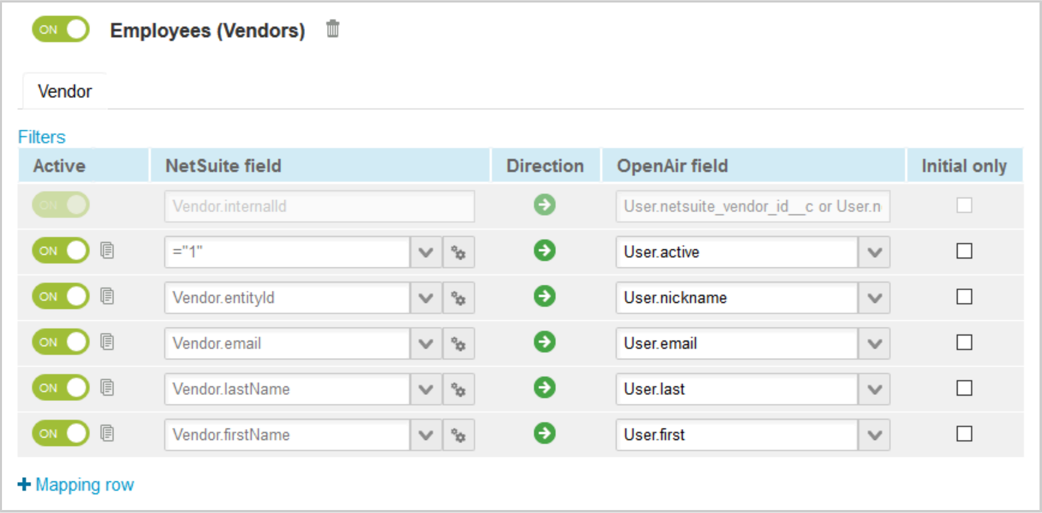
b. Check the **Export Vendors by Default** box.

After you enable this preference, the **Export to OpenAir** [`custentity_oa_export_to_openair`] box is checked by default when a user creates a new vendor record. Users with the relevant permissions can clear the box on individual records if required.

To set up the Employees (Vendors) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button then NetSuite Connector Administration.

   The OpenAir NetSuite Connector administration form appears.
3. Check the **Employees (Vendors)** box under the Imports section.
4. Click **Save** then **Close**.
5. Go to Administration > NetSuite Connector > Mapping.
6. Click **New workflow** then **Employees (Vendors)**.
7. Click **Employees (Vendors)** in the list of integration workflows.
8. Review the workflow settings impacting this workflow. For more information, see Employees (Vendors) Import Workflow Settings.
9. Modify the field mapping definition table for the Employees (Vendors) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Employees (Vendors) Import Field Mapping Definition.
10. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON Employees (Vendors)

Vendor

Filters

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Vendor.internalId	→	User.netsuite_vendor_id__c or User.n	□
ON	="1"	→	User.active	□
ON	Vendor.entityId	→	User.nickname	□
ON	Vendor.email	→	User.email	□
ON	Vendor.lastName	→	User.last	□
ON	Vendor.firstName	→	User.first	□

11. Click **Save**.
12. (Optional) Add the Employees (Vendors) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Employees (Vendors) Import Workflow Settings

The following setting is available for the Vendors (Vendors) import workflow and also impacts the Employees (Vendors) import workflow:

- Disable Export to OpenAir filtering for vendors in NetSuite integration

Employees (Vendors) Import Field Mapping Definition

The Employees (Vendors) import workflow includes one mapping group:

- Vendor [Integration point: VENDOR_TO_USER] — See Vendor Mapping Group Tab.

Vendor Mapping Group Tab

Use the Vendor mapping group to map information on employee records in OpenAir with information on vendor records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mapping — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes name, email, tax nexus, NetSuite vendor ID and parent vendor ID, and currency. Filters are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Vendor [Vendor] Schema: Vendor (2020.1)|Vendor (2023.1) XSD: VendorSearchRowBasic platformCommon (2020.1)|platformCommon (2023.1)	Employee [user] Data Dictionary: user	Vendor [VENDOR_TO_USER]

Default and Custom Filters

By default, vendor records are imported from NetSuite into OpenAir as employees if the following conditions are all met:

- The **Inactive** box is NOT checked on the vendor record in NetSuite.
- The **Export to OpenAir** [custentity_oa_export_to_openair] box is checked on the vendor record in NetSuite. You can modify this behavior — See Disable Export to OpenAir filtering for vendors in NetSuite integration.
- If the Vendors (Vendors) import workflow is also active, the **OpenAir: User or Vendor** [custentity_oa_user_or_vendor] field is set to User on the vendor record in NetSuite.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mapping

See the OpenAir Help Center for detailed information about default field mapping definition for the Vendor mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Vendors (Vendors) Import Workflow

The Vendors(Vendors) import workflow lets you:

- Create new vendor records in NetSuite first and then import them into OpenAir as vendor records.
- Update vendor records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding vendor record in OpenAir.

You can also use the Employees (Vendors) import workflow to import NetSuite vendor records into OpenAir as employee records. Both workflows can be used in conjunction. See Employees (Vendors) Import Workflow.

Important: NetSuite is the primary application for vendor records. Changes made to vendor records in OpenAir after the initial import are not exported back to NetSuite.

For information about workflow specific settings, see Vendors (Vendors) Import Workflow Settings.

For information about field mapping definitions for this workflow, see Vendors (Vendors) Import Field Mapping Definition.

Requirements and Downstream Dependencies

Matching vendor records must exist in both NetSuite and OpenAir before you can import purchase orders or vendor bills from NetSuite into OpenAir as purchase orders.

Using the Vendors (Vendors) and Employees (Vendors) import workflows in conjunction enables you to import vendor bills or purchase orders from NetSuite into OpenAir as purchase orders, which requires matching vendor records in both applications and to export expense reports from OpenAir to NetSuite as vendor bills, which requires the associated employee record in OpenAir being matched with a vendor record in NetSuite. See POs (Vendor Bills) Import Workflow, POs (Purchase Orders) Import Workflow and Expense Reports (Expense Reports) Export Workflow.

When using the Vendors (Vendors) and Employees (Vendors) import workflows in conjunction, use the **OpenAir: User or Vendor** [`custentity_oa_user_or_vendor`] field on the vendor record in NetSuite to determine whether the vendor should be imported into OpenAir as a vendor or as an employee (user).

> **Note:** You can only set the **OpenAir: User or Vendor** [`custentity_oa_user_or_vendor`] on the vendor record in NetSuite if the following conditions are all met. The field is read-only otherwise.
>
> - The **Expense Report – Vendor Bill Integration** box is checked in the integration preferences.
> - Either the **Vendor Bill to OpenAir PO Integration** box or the **Enable OpenAir Purchase Integration** box is checked in the integration preferences.
> - The **Export to OpenAir** [`custentity_oa_export_to_openair`] box is checked on the vendor record.
>
> For more information about setting integration preferences, see Configuring Integration Preferences and Terminology Preferences in NetSuite.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir vendor record after you create or update the vemdor record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir vendor record when the vendor form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir vendor record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Vendors (Vendors) Import Workflow

Use the following steps to set up the workflow.

To set up the Vendors (Vendors) import workflow in NetSuite

1. (Optional) To mark new vendor records for export to OpenAir by default:
   a. In NetSuite, do one of the following:
      - Go to Setup > Company > General Preferences (Custom Preferences), if you are using the SuiteScript 1 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
      - Go to Customization > Lists, Records and Fields > Records > [*View list forOpenAir: SRP Integration Settings*] > [*View OpenAir SRP Integration Settings Record*] , if you are using the SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.

   b. Check the **Export Vendors by Default** box.

   After you enable this preference, the **Export to OpenAir** [`custentity_oa_export_to_openair`] box is checked by default when a user creates a new vendor record. Users with the relevant permissions can clear the box on individual records if required.

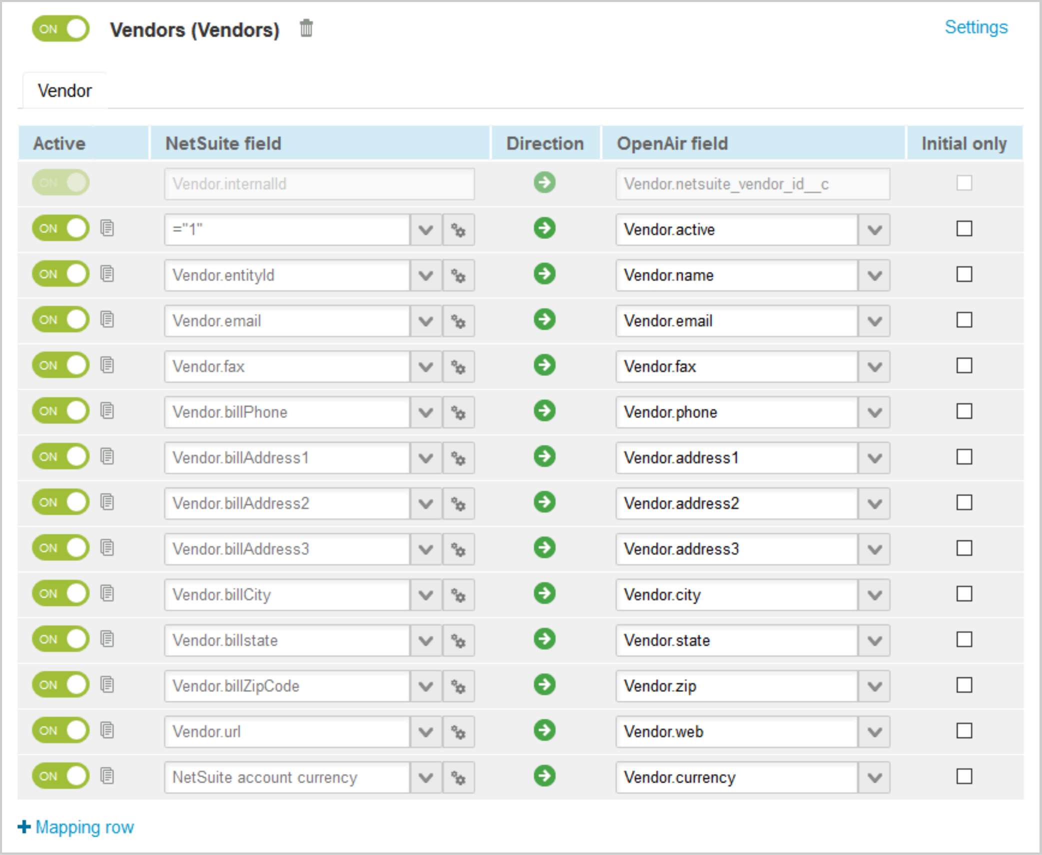
To set up the Vendors (Vendors) import workflow in OpenAir

1. If the Employees (Vendors) import workflow is already enabled, additional setup is required before you can enable the Vendors (Vendors) import workflow.
   a. In OpenAir, go to Administration > Global Settings > Custom Settings.
   b. Locate the custom field name `netsuite_vendor_id` associated with the Employee form.
   c. Click `netsuite_vendor_id`.
   d. Change the custom field **Name** to `netsuite_vendor2_id`.
   e. Click **Save**

   > ⚠ **Important:** If this custom field is used in any reports or integrations, make sure you make all necessary changes.

2. In OpenAir, go to Administration > NetSuite Connector.
3. Click the Tips button then NetSuite Connector Administration.

   The OpenAir NetSuite Connector administration form appears.
4. Check the **Vendors (Vendors)** box under the Imports section.
5. Click **Save** then **Close**.
6. Go to Administration > NetSuite Connector > Mapping.
7. Click **New workflow** then **Vendors (Vendors)**.
8. Click **Vendors (Vendors)** in the list of integration workflows.
9. Click **Workflow settings** to view and modify the settings specific to this workflow. For more information about the workflow settings, see Vendors (Vendors) Import Workflow Settings.
10. Modify the field mapping definition table for the Vendors (Vendors) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Vendors (Vendors) Import Field Mapping Definition.
11. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Vendor.internalId	→	Vendor.netsuite_vendor_id__c	☐
ON	="1"	→	Vendor.active	☐
ON	Vendor.entityId	→	Vendor.name	☐
ON	Vendor.email	→	Vendor.email	☐
ON	Vendor.fax	→	Vendor.fax	☐
ON	Vendor.billPhone	→	Vendor.phone	☐
ON	Vendor.billAddress1	→	Vendor.address1	☐
ON	Vendor.billAddress2	→	Vendor.address2	☐
ON	Vendor.billAddress3	→	Vendor.address3	☐
ON	Vendor.billCity	→	Vendor.city	☐
ON	Vendor.billstate	→	Vendor.state	☐
ON	Vendor.billZipCode	→	Vendor.zip	☐
ON	Vendor.url	→	Vendor.web	☐
ON	NetSuite account currency	→	Vendor.currency	☐

12. Click **Save**.
13. (Optional) Add the Vendors(Vendors) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Vendors (Vendors) Import Workflow Settings

You can control Vendors (Vendors) import workflow settings from the NetSuite Connector UI. On the NetSuite Connector Mapping screen, select the Vendors (Vendors) import workflow and click **Settings** the workflow settings form.

The following setting is available for the Vendors (Vendors) import workflow:

- Disable Export to OpenAir filtering for vendors in NetSuite integration

Disable Export to OpenAir filtering for vendors in NetSuite integration

Check this box to import all vendors from NetSuite to OpenAir.

- When checked, OpenAir NetSuite Connector imports vendor records from NetSuite into OpenAir even if the **Export to OpenAir** [`custentity_oa_export_to_openair`] box is not checked on the NetSuite vendor record.
- Otherwise, OpenAir NetSuite Connector imports a vendor record only if the **Export to OpenAir** [`custentity_oa_export_to_openair`] box is checked.

Vendors (Vendors) Import Field Mapping Definition

The Vendors (Vendors) import workflow includes one mapping group:

- Vendor [Integration point: VENDOR_TO_VENDOR] — See Vendor Mapping Group Tab.

Vendor Mapping Group Tab

Use the Vendor mapping group to map information on vendor records in OpenAir with information on vendor records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes name, email, address, website, and currency. Filters are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	**OpenAir Record Type [table]**	**Mapping Group [Integration point]**
Vendor [Vendor] Schema: Vendor (2020.1)|Vendor (2023.1) XSD: VendorSearchRowBasic platformCommon (2020.1)|platformCommon (2023.1)	Vendor [vendor] Data Dictionary: vendor	Vendor [VENDOR_TO_VENDOR]

Default and Custom Filters

By default, vendor records are imported from NetSuite into OpenAir as employees if the following conditions are all met:

- The **Inactive** box is NOT checked on the vendor record in NetSuite.
- The **Export to OpenAir** [custentity_oa_export_to_openair] box is checked on the vendor record in NetSuite.
- If the Employees (Vendors) import workflow is also active, the **OpenAir: User or Vendor** [custentity_oa_user_or_vendor] field is set to Vendor on the vendor record in NetSuite.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Vendor mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

> **Important:** Review the following guidelines
>
> - You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
> - For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
> - Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Products (Purchase Items) Import Workflow

The Products (Purchase Items) import workflow lets you:

- Create new inventory , non-inventory, service, kit/package, and other charge purchase item records in NetSuite first and then import them into OpenAir as product records.
- Update inventory , non-inventory, service, kit/package, and other charge purchase item records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding service record in OpenAir.

> **Note:** You can also import items of all types (both service items and other types of items) from NetSuite into OpenAir as services records using the Services (Services Items) import workflow. See Services (Services Items) Import Workflow.

> **Important:** NetSuite is the primary application for item records. Changes made to product records in OpenAir after the initial import are not exported back to NetSuite.

For information about field mapping definitions for this workflow, see Products (Purchase Items) Import Field Mapping Definition.

Requirements and Downstream Dependencies

Matching NetSuite purchase item and OpenAir product records must exist before you can import purchase orders or vendor bills from NetSuite into OpenAir as purchase orders.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir product record after you create or update the purchase item record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir product record when the purchase item form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir product record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Products (Purchase Items) Import Workflow

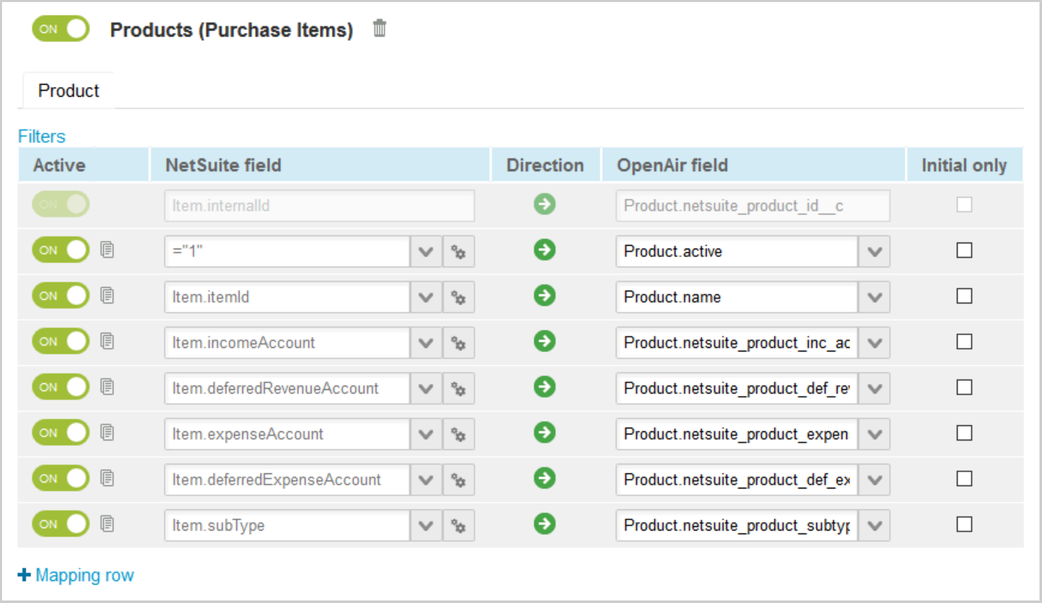
Use the following steps to set up the workflow.

No additional setup is required in NetSuite.

To set up the Products (Purchase Items) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button then NetSuite Connector Administration.

   The OpenAir NetSuite Connector administration form appears.
3. Check the **Products (Purchase Items)** box under the Imports section.
4. Click **Save** then **Close**.
5. Go to Administration > NetSuite Connector > Mapping.
6. Click **New workflow** then **Products (Purchase Items)**.
7. Click **Products (Purchase Items)** in the list of integration workflows.
8. Modify the field mapping definition table for the Products (Purchase Items) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Products (Purchase Items) Import Field Mapping Definition.
9. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON Products (Purchase Items)

Product

Filters

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Item.internalId	→	Product.netsuite_product_id__c	□
ON	="1"	→	Product.active	□
ON	Item.itemId	→	Product.name	□
ON	Item.incomeAccount	→	Product.netsuite_product_inc_ac	□
ON	Item.deferredRevenueAccount	→	Product.netsuite_product_def_re	□
ON	Item.expenseAccount	→	Product.netsuite_product_expen	□
ON	Item.deferredExpenseAccount	→	Product.netsuite_product_def_ex	□
ON	Item.subType	→	Product.netsuite_product_subtyp	□

10. Click **Save**.

11. (Optional) Add the Products (Purchase Items) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Products (Purchase Items) Import Field Mapping Definition

The Products (Purchase Items) import workflow includes one mapping group:

- Product [Integration point: ITEM_TO_PRODUCT] — See Product Mapping Group.

Product Mapping Group

Use the Service mapping group to map information about service records in OpenAir with information on service item records in NetSuite.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes service name, rate, and income and deferred revenue account. Filters are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Item [Item] Schema: <Type><Subtype>Item (2020.1)|<Type><Subtype>Item (2023.1) XSD: ItemSearchRowBasic — platformCommon (2020.1)|platformCommon (2023.1)	Product [product] Data Dictionary: product	Service [ITEM_TO_PRODUCT]

Note: The relevant SOAP objects as described in the NetSuite Schema Browser include:

- <Type><Subtype>Item objects, where <Type> is the item **Type** (Inventory, NonInventory, Service, Kit, or OtherCharge) and <Subtype> is the item **Subtype** (Resale, or Purchase).

Default and Custom Filters

By default, Item records are imported from NetSuite into OpenAir if the following conditions are all met:

- The **Inactive** box is NOT checked on the item record in NetSuite.
- The **Export to OpenAir Product** [custitem_oa_export_to_openair_product] box is checked on the item record in NetSuite. You can modify this behavior — See Disable Export to OpenAir filtering for services in NetSuite integration.

> **Note:** Item for purchase and for resale records may include both the **Export to OpenAir** [custitem_oa_export_to_openair] and **Export to OpenAir Product** [custitem_oa_export_to_openair_product] boxes.
>
> - Check the **Export to OpenAir** [custitem_oa_export_to_openair] to import the item into OpenAir as a service.
> - Check the **Export to OpenAir Product** [custitem_oa_export_to_openair_product] to import the item into OpenAir as a product.
> - Check both boxes to import the item into OpenAir as both a service and a product.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Product mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

> **Important:** Review the following guidelines
>
> - You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
> - For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
> - Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

See the OpenAir Help Center for detailed information about additional field mapping definitions for the Product mapping group tab that may be required or useful in some cases. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Customer POs (Sales Orders) Import Workflow

Note: This help topic is a placeholder only. Content will be added gradually to these pages as the content and structure of this guide are reviewed. For more information about the Customer POs (Sales Orders) integration workflows, please refer to the following chapters:

- Configuring Optional Features > Configuring the Sales Order Integration
- Using the Integration
- Data Mapping in the Integration

Customer POs (Sales Orders) Import Workflow Settings

You can control Customer POs (Sales Orders) import workflow settings from the NetSuite Connector UI. On the NetSuite Connector Mapping screen, select the Customer POs (Sales Orders) import workflow and click **Settings** in the top right corner to view the workflow settings form.

The following settings are available for the Customer POs (Sales Orders) import workflow:

- Create time billing rules in sales order integration with a service filter. Map sales order line description to billing rule Charge description
- Enable recognition rule creation in sales order integration
- Enable use of NetSuite time units of measure
- Only update last modified date if no error
- Set OpenAir Customer PO total value to NetSuite sales order net amount

Create time billing rules in sales order integration with a service filter. Map sales order line description to billing rule Charge description

Create time billing rules for activities in sales orders and map a line description to a billing rule charge description.

Enable recognition rule creation in sales order integration

Create recognition rules along with billing rules on NetSuite sales orders and transfer both the billing rules and recognition rules to OpenAir.

Enable use of NetSuite time units of measure

This setting within the CustomerPOs (Sales Orders) workflow page forces OpenAir to integrate Sales Orders using supported time units of measure you have set up in NetSuite. Supported units of measure are **Second**, **Minute**, **Hour**, and **Day**. When enabled, new mapping lines to two new OpenAir custom fields are added to the workflow:

- Transaction.quantityUOM > netsuite_sales_order_line_quantity_uom

- Transaction.unit > netsuite_sales_order_line_unit

These mapping lines also map to two new custom fields, respectively:

- transaction.quantityUOM
- transaction.unit

This setting requires that you set up the units of time measure in NetSuite before enabling it.

Only update last modified date if no error

When checked, OpenAir NetSuite Connector does not update the last import timestamp for any of the sales order records included integration run if at least one sales order record fails to import from NetSuite into OpenAir due to an integration error. The next time the Customer POs (Sales Orders) import workflow runs, it will attempt to import all the sales order records included in the previous run again in addition to any other newly modified sales order records.

Otherwise, when the **Only update last modified date if no error** box is not checked, the last import timestamp is updated for each sales order record included in the integration run whether the import is successful or not.

Set OpenAir Customer PO total value to NetSuite sales order net amount

⚠ **Important:** Check this box only if the SuiteTax feature is enabled for your NetSuite account.

By default, OpenAir NetSuite Connector processes the gross amount for the sales order line item in NetSuite to calculate the total value of the corresponding customer PO in OpenAir.

This may introduce discrepancies if the SuiteTax feature is enabled for your NetSuite account. Check this box to avoid these discrepancies.

When checked, OpenAir NetSuite Connector sets the Customer PO total value [`customerpo.total`] in OpenAir to the net amount of the sales order line item [`Transactions.netAmount`] in NetSuite.

Reimbursements (Expense Report Reimbursements) Import Workflow

The Reimbursements (Expense Report Reimbursements) import workflow lets you:

- Record bill payments against expense reports or vendor bills previously exported from OpenAir expense reports to NetSuite, and then import the bill payment records into OpenAir as reimbursements.
- Update bill payments in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding reimbursements in OpenAir.

⚠ **Important:** Changes made to the reimbursement records in OpenAir after the initial import are not exported back to NetSuite.

For information about field mapping definitions for this workflow, see Reimbursements (Expense Report Reimbursements) Import Field Mapping Definition.

Requirements and Downstream Dependencies

Matching expense report transaction records must exist in both NetSuite and OpenAir before you can import bill payments applied to expense reports from NetSuite into OpenAir as reimbursements.

Matching vendor bill transaction records in NetSuite and expense report records in OpenAir must exist before you can import bill payments applied to vendor bills from NetSuite into OpenAir as reimbursements.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir reimbursement records after you create or update the bill payment records in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir reimbursement records when the bill payment form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir reimbursement record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Reimbursements (Expense Report Reimbursements) Import Workflow

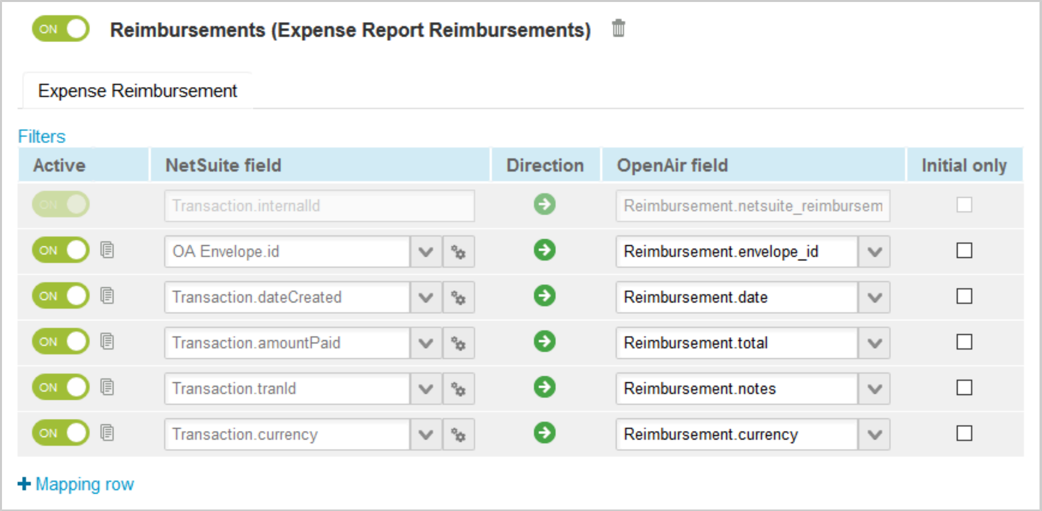
Use the following steps to set up the Reimbursements (Expense Report Reimbursements) import workflow.

No additional setting is required in NetSuite.

To set up the Reimbursements (Expense Report Reimbursements) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button then NetSuite Connector Administration.

   The OpenAir NetSuite Connector administration form appears.
3. Check the **Reimbursements (Expense Report Reimbursements)** box under the Imports section.
4. Click **Save** then **Close**.
5. Go to Administration > NetSuite Connector > Mapping.
6. Click **New workflow** then **Reimbursements (Expense Report Reimbursements)**.
7. Click **Reimbursements (Expense Report Reimbursements)** in the list of integration workflows.
8. Modify the field mapping definition table for the Reimbursements (Expense Report Reimbursements) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Reimbursements (Expense Report Reimbursements) Import Field Mapping Definition.
9. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON Reimbursements (Expense Report Reimbursements)

Expense Reimbursement

Filters

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Transaction.internalId	→	Reimbursement.netsuite_reimbursem	☐
ON	OA Envelope.id	→	Reimbursement.envelope_id	☐
ON	Transaction.dateCreated	→	Reimbursement.date	☐
ON	Transaction.amountPaid	→	Reimbursement.total	☐
ON	Transaction.tranId	→	Reimbursement.notes	☐
ON	Transaction.currency	→	Reimbursement.currency	☐

+ Mapping row

10. Click **Save**.
11. (Optional) Add the Reimbursements (Expense Report Reimbursements) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Reimbursements (Expense Report Reimbursements) Import Field Mapping Definition

The Reimbursements (Expense Report Reimbursements) import workflow includes one mapping group:

- Expense Reimbursement [Integration point: `BILLPAYMENT_TO_REIMBURSEMENT`] — See Expense Reimbursement Mapping Group.

Expense Reimbursement Mapping Group

Use the Expense Reimbursement mapping group to map information about reimbursement records in OpenAir with information about boll payments applied to expense report transaction records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes project name, associated customer, and start date. Filters are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Bill Payment [VendorPayment] Schema: VendorPayment (2020.1)|VendorPayment (2023.1) XSD: TransactionSearchRowBasic — platformCommon (2020.1)|platformCommon (2023.1)	Reimbursement [reimbursement] Data Dictionary: reimbursement	Expense Reimbursement [BILLPAYMENT_TO_REIMBURSEMENT]

Default and Custom Filters

By default, bill payment records applied are imported from NetSuite into OpenAir as reimbursements if there is an expense report record in OpenAir matching the expense report or vendor bill record to which the bill payment was applied in NetSuite, and if the expense report was not already reimbursed in OpenAir (balance is greater than 0).

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Expense Reimbursement mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

OpenAir NetSuite Connector updates automatically the OpenAir expense report record corresponding to the NetSuite expense report transaction to which the bill payment is applied. The following information is updated:

- Envelope.balance — outstanding balance on the expense report.

Additional Field Mappings

> **Important:** Review the following guidelines
>
> - You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
> - For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
> - Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Payments (Invoice Payments) Import Workflow

The Payments (Invoice Payments) import workflow lets you:

- Record payments against invoices previously exported from OpenAir to NetSuite, and then import the invoice payment records into OpenAir.
- Update payments in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding invoice payments in OpenAir.

> **Important:** Changes made to the payment records in OpenAir after the initial import are not exported back to NetSuite.

For more information about recording or correcting a payment in NetSuite, see Applying a Payment on an Invoice (External link to NetSuite SuiteAnswers | Answer ID: 9149) and Correcting Payments to Invoices (External link to NetSuite SuiteAnswers | Answer ID: 9115).

For information about field mapping definitions for this workflow, see Payments (Invoice Payments) Import Field Mapping Definition.

Requirements and Downstream Dependencies

Matching invoice transaction records must exist in both NetSuite and OpenAir before you can import payments applied to invoices from NetSuite into OpenAir.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir payment records after you create or update the payment record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir payment records when the Payment form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir payment record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Payments (Invoice Payments) Import Workflow

Use the following steps to set up the Payments (Invoice Payments) import workflow.

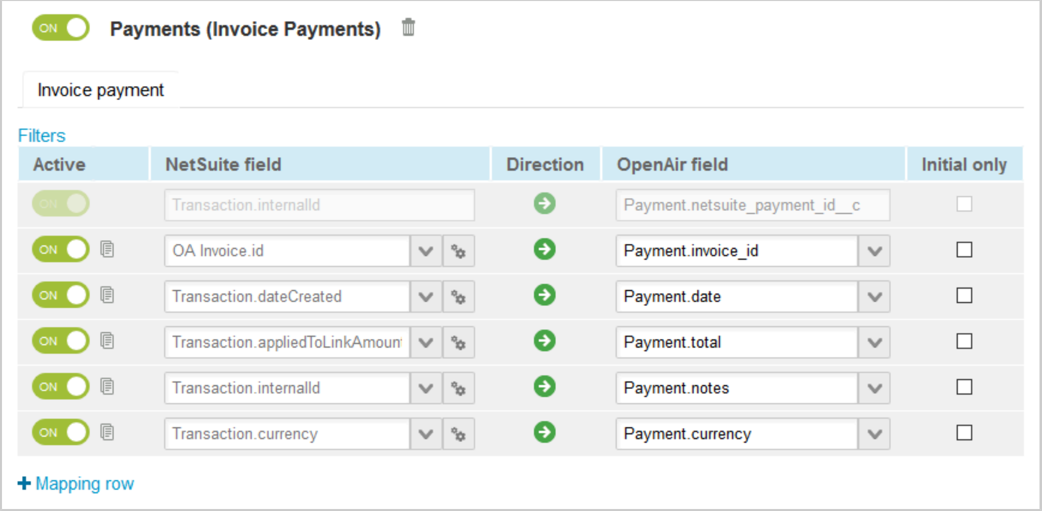
No additional setting is required in NetSuite.

To set up the Payments (Invoice Payments) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button then NetSuite Connector Administration.

   The OpenAir NetSuite Connector administration form appears.
3. Check the **Payments (Invoice Payments)** box under the Imports section.
4. Click **Save** then **Close**.
5. Go to Administration > NetSuite Connector > Mapping.

6. Click **New workflow** then **Payments (Invoice Payments)**.
7. Click **Payments (Invoice Payments)** in the list of integration workflows.
8. Modify the field mapping definition table for the Payments (Invoice Payments) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Payments (Invoice Payments) Import Field Mapping Definition.
9. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON **Payments (Invoice Payments)**

Invoice payment

Filters

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Transaction.internalId	→	Payment.netsuite_payment_id__c	☐
ON	OA Invoice.id	→	Payment.invoice_id	☐
ON	Transaction.dateCreated	→	Payment.date	☐
ON	Transaction.appliedToLinkAmoun	→	Payment.total	☐
ON	Transaction.internalId	→	Payment.notes	☐
ON	Transaction.currency	→	Payment.currency	☐

+ Mapping row

10. Click **Save**.
11. (Optional) Add the Payments (Invoice Payments) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Payments (Invoice Payments) Import Field Mapping Definition

The Payments (Invoice Payments) import workflow includes one mapping group:

- Invoice Payment [Integration point: PAYMENT_TO_PAYMENT] — See Invoice Payment Mapping Group.

Invoice Payment Mapping Group

Use the Invoice Payment mapping group to map information about payment records in OpenAir with information about payments applied to invoice transaction records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes project name, associated customer, and start date. Filters are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Customer Payment [CustomerPayment] Schema: CustomerPayment (2020.1) | CustomerPayment (2023.1) XSD: TransactionSearchRowBasic — platformCommon (2020.1) | platformCommon (2023.1)	Payment [payment] Data Dictionary: payment	Invoice payment [PAYMENT_TO_PAYMENT]

Default and Custom Filters

By default, applied customer payment records are imported from NetSuite into OpenAir if there is an invoice record in OpenAir matching the invoice record to which the customer payment was applied in NetSuite.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Invoice Payment mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

OpenAir NetSuite Connector updates automatically the OpenAir invoice record corresponding to the NetSuite invoice transaction to which the customer payment is applied. The following information is updated:

- Invoice.balance — outstanding balance on the invoice.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Tasks (Project Tasks) Import Workflow

The Tasks (Project Tasks) import workflow lets you:

- Create new project task records in NetSuite first and then import them into OpenAir.
- Update project task records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding task record in OpenAir.

> **Note:** You can also:
>
> - Create and update support case records in NetSuite and then import them into OpenAir as tasks. See Tasks (Support Cases) Import Workflow.
> - Update tasks records in OpenAir and run the integration to update the project task records in NetSuite using the Update NetSuite Tasks with OpenAir Task Data export workflow. See Update NetSuite Tasks with OpenAir Task Data.

For information about field mapping definitions for this workflow, see Tasks (Project Tasks) Import Field Mapping Definition.

Requirements and Downstream Dependencies

Matching project records must exist in both NetSuite and OpenAir before you can import project tasks from NetSuite into OpenAir.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
✓	✓	✓	✓	✓	✓

- Run the integration manually, use a scheduled integration run to create or update the OpenAir task record after you create or update the project task record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir task record when the Project task form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir task record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.
- When you install NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) 2.04 or later version on your NetSuite account, NetSuite task records are imported into OpenAir only if one of the monitored field values have changed. See Monitored Fields.

Setting up the Tasks (Project Tasks) Import Workflow

Use the following steps to set up the workflow.

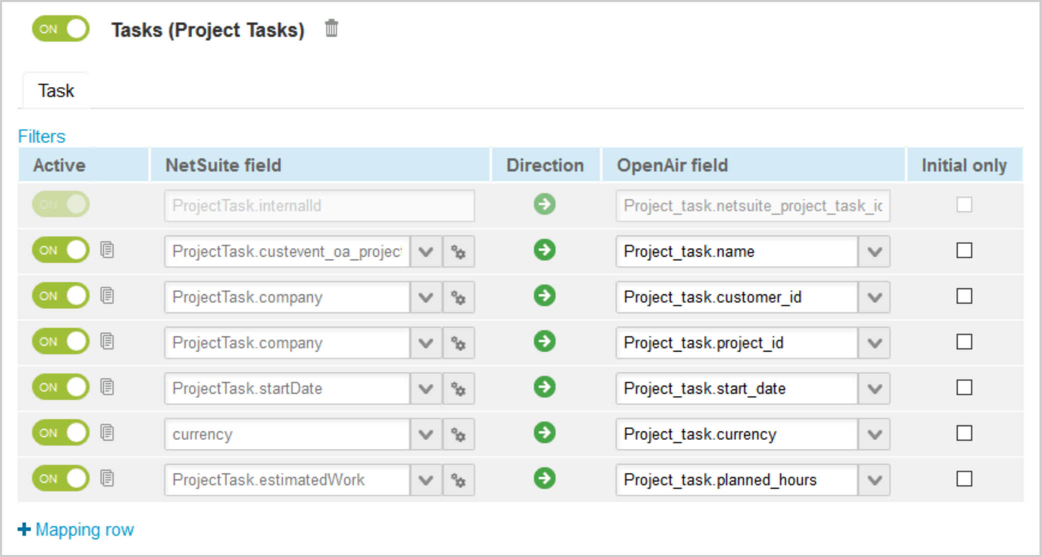
To set up the Tasks (Project Tasks) import workflow in NetSuite

1. In NetSuite, do one of the following:
   - Go to Setup > Company > General Preferences (Custom Preferences), if you are using the SuiteScript 1 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
   - Go to Customization > Lists, Records and Fields > Records > [*View list forOpenAir: SRP Integration Settings*] > [*View OpenAir SRP Integration Settings Record*] , if you are using the SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
2. Check the **Enable OpenAir Task Integration** box.
3. Click **Save**.

To set up the Tasks (Project Tasks) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button then NetSuite Connector Administration.

   The OpenAir NetSuite Connector administration form appears.
3. Check the **Tasks (Project Tasks)** box under the Imports section.
4. Click **Save** then **Close**.
5. Go to Administration > NetSuite Connector > Mapping.
6. Click **New workflow** then **Tasks (Project Tasks)**.
7. Click **Tasks (Project Tasks)** in the list of integration workflows.
8. Modify the field mapping definition table for the Tasks (Project Tasks) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Tasks (Project Tasks) Import Field Mapping Definition.
9. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON **Tasks (Project Tasks)**

Task

Filters

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	ProjectTask.internalId	→	Project_task.netsuite_project_task_ic	□
ON	ProjectTask.custevent_oa_projec	→	Project_task.name	□
ON	ProjectTask.company	→	Project_task.customer_id	□
ON	ProjectTask.company	→	Project_task.project_id	□
ON	ProjectTask.startDate	→	Project_task.start_date	□
ON	currency	→	Project_task.currency	□
ON	ProjectTask.estimatedWork	→	Project_task.planned_hours	□

+ Mapping row

10. Click **Save**.
11. (Optional) Add the Tasks (Project Tasks) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Tasks (Project Tasks) Import Field Mapping Definition

The Tasks (Project Tasks) import workflow includes one mapping group:

- Task [Integration point: NS_PR_TASK_TO_OA_PR_TASK] — See Task Mapping Group.

Task Mapping Group

Use the task mapping group to map information on task records in OpenAir with information on project task records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes task name, associated customer and project, start date, currency, and planned hours.
- Additional Field Mappings — Best practice guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Project Task [ProjectTask] Schema: ProjectTask (2020.1) | ProjectTask (2023.1) XSD: ProjectTaskSearchRowBasic — platformCommon (2020.1) | platformCommon (2023.1)	Task [project_task] Data Dictionary: project_task	Task [NS_PR_TASK_TO_OA_PR_TASK]

Default and Custom Filters

By default, project task records are imported from NetSuite into OpenAir if the following conditions are met:

- Matching project records exists in OpenAir and NetSuite for the project associated with the project task in NetSuite.
- The **Export to OpenAir** [custevent_oa_export_to_openair] box is checked on the project task record in NetSuite.

Existing task records in OpenAir are updated only if the **Closed task** [project_task.closed] box is not checked in OpenAir. You can change this behavior by adding a custom field mapping definition to change the **Closed task** field depending on the **Status** of the project task in NetSuite. See Additional Field Mappings.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Task mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

OpenAir NetSuite Connector runs the project calculation automatically when a task is added or updated using the Tasks (Project Tasks) import workflow

Additional Field Mappings

⚠ **Important:** Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

See the OpenAir Help Center for detailed information about additional field mapping definitions for the Task mapping group tab that may be required or useful in some cases. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

POs (Purchase Orders) Import Workflow

The POs (Purchase Orders) import workflow lets you:

- Create new purchase order records in NetSuite first and then import them into OpenAir.
- Update purchase order records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding purchase order record in OpenAir.

When importing a purchase order into OpenAir, the following records are created in OpenAir:

- Purchase order — A single purchase order is created or updated in OpenAir to match the NetSuite purchase order.
- Purchase items — Each purchase order line item marked for export on the NetSuite purchase order imports into a purchase item.
- Purchase requests —A purchase request record is created for each purchase order line item marked for export on the NetSuite purchase order. This is because in OpenAir, purchase request and request item in OpenAir must be associated to a single project, whereas in NetSuite, purchase orders can reference multiple projects.
- Request items — Each purchase request includes one request item.
- Fulfillments — A fulfillment record is created for each purchase item.

Note: You can also create purchase requests in OpenAir and export them to NetSuite as purchase requisitions, create purchase orders in NetSuite and import them into OpenAir to create purchase orders associated with the purchase requests previously exported to NetSuite. See Purchase Requests (Purchase Requisitions) Export Workflow.

You can also import vendor credits from NetSuite into OpenAir as purchase orders with negative purchase items. See POs (Vendor Credits) Import Workflow.

You can import vendor bills instead of purchase orders from NetSuite into OpenAir as purchase orders using the POs (Vendor Bills) import workflows. The POs (Vendor Bills) and POs (Purchase Orders) import workflows are mutually exclusive. See POs (Vendor Bills) Import Workflow.

Important: NetSuite is the primary application for purchase order records. Changes made to purchase order records in OpenAir after the initial import are not exported back to NetSuite.

For information about purchase orders in NetSuite, see Purchase Order Management (External link to NetSuite SuiteAnswers | Answer ID 11133.

For information about field mapping definitions for this workflow, see POs (Purchase Orders) Import Field Mapping Definition.

Requirements and Downstream Dependencies

This workflow is mutually exclusive with the POs (Vendor Bills) import workflow.

The following workflows must also be enabled:

- Products (Purchase Items) Import Workflow
- Vendors (Vendors) Import Workflow

Matching project records must exist in both NetSuite and OpenAir before you can import purchase orders from NetSuite into OpenAir.

Matching vendor records must exist in both NetSuite and OpenAir before you can import purchase orders from NetSuite into OpenAir.

Note: You can use the POs (Purchase Orders) import workflow and Expense Reports (Expense Reports) export workflow in conjunction to import purchase orders from NetSuite into OpenAir as purchase orders and to export expense reports from OpenAir to NetSuite as vendor bills. See Expense Reports (Expense Reports) Export Workflow.

Exporting expense reports from OpenAir to NetSuite as vendor bills requires the associated user record in OpenAir being matched with a vendor record in NetSuite. To use both workflows in conjunction, enable the Employees (Vendors) import workflow and use the **OpenAir: User or Vendor** [`custentity_oa_user_or_vendor`] field on the vendor record in NetSuite to determine whether the vendor should be imported into OpenAir as a vendor or as a user. See Employees (Vendors) Import Workflow.

After you import purchase orders from NetSuite into OpenAir, you can use project billing rules to create product charges based on purchase items and fulfillments, add these charges to your invoices in OpenAir, and export the invoices to NetSuite.

Important: Items with a subtype “For Purchase” cannot be billed to customers in NetSuite — they should not be included in purchase billing rules.

When exporting invoices with product charges for purchase items imported into OpenAir using the POs (Purchase Orders) import workflow, these charges export as invoice line items on the NetSuite invoice Items subtab, and not as billable line items on the Billable Items subtab.

The following Invoices (Invoices) import workflow mapping groups are used to export billable purchase items imported from a NetSuite vendor bill:

- Use the Invoice Line mapping group to export information on the product charges created from purchase items. See Invoice Line Mapping Group.
- Use the Purchase Line mapping group to export information on the purchase items. See Purchase Line Mapping Group.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir purchase order record after you create or update the purchase order record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir purchase order record when the purchase order form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir purchase order record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the POs (Purchase Orders) Import Workflow

Use the following steps to set up the workflow.

To set up the POs (Purchase Orders) import workflow in NetSuite

1. In NetSuite, do one of the following:
   - Go to Setup > Company > General Preferences (Custom Preferences), if you are using the SuiteScript 1 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.

- Go to Customization > Lists, Records and Fields > Records > [*View list forOpenAir: SRP Integration Settings*] > [*View OpenAir SRP Integration Settings Record*] , if you are using the SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.

2. Check the **Enable OpenAir Purchase Integration** box.
3. Click **Save**.
4. In NetSuite, go to Setup > Company > Enable Features, and click the Transactions subtab.
5. Check the **Advanced Receiving** box under the Shipping & Receiving section. The Advanced Receiving feature is required to create purchase item receipts in NetSuite.
6. Click **Save**.
7. To let employees view projects and items and check the **Billable** and **Export to OpenAir** boxes for purchase order line items and purchase order records, go to Home > Set Preferences and check the **Validate OpenAir PO/Vendor Bill** box on the Custom Preferences tab.

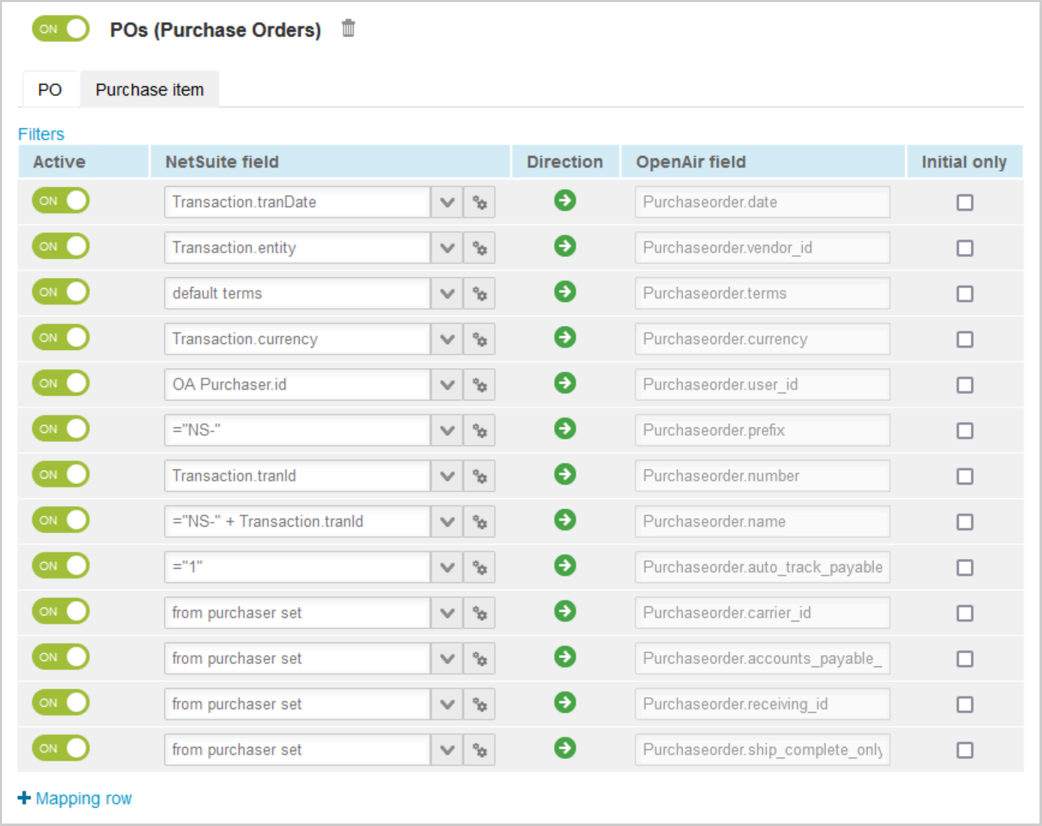
To set up the POs (Purchase Orders) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button then NetSuite Connector Administration.

   The OpenAir NetSuite Connector administration form appears.
3. Check the **POs (Purchase Orders)** box under the Imports section.
4. Click **Save** then **Close**.
5. If the purchaser integration is not enabled, designate a purchaser to use when importing purchase orders into OpenAir. To do so:
   a. Go to Administration > Application Settings > Purchases > Purchasers > [*Select a purchaser*].
   b. Check the **NetSuite Purchase Order Integration purchaser** [`netsuite_purchaser`] box.
   c. Click **Save**.

> **Note:** To create a purchaser, you must create at least one record for each of the following record types in Administration > Application Settings > Purchases > Accounts payable locations, Receiving locations, Carriers, Shipping terms, and F.O.B. locations.
>
> If the purchaser integration is enabled, matching employee records must exist in OpenAir and NetSuite for employees creating purchase orders in NetSuite, and purchaser records marked as NetSuite purchaser must exist for these employees in OpenAir. For more information about the purchaser integration, see Enable purchaser integration

6. Go to Administration > NetSuite Connector > Mapping.
7. Click **New workflow** then **POs (Purchase Orders)**.
8. Click **POs (Purchase Orders)** in the list of integration workflows.
9. Modify the field mapping definition table for the Tasks (Project Tasks) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and POs (Purchase Orders) Import Field Mapping Definition.
10. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Transaction.tranDate	→	Purchaseorder.date	□
ON	Transaction.entity	→	Purchaseorder.vendor_id	□
ON	default terms	→	Purchaseorder.terms	□
ON	Transaction.currency	→	Purchaseorder.currency	□
ON	OA Purchaser.id	→	Purchaseorder.user_id	□
ON	="NS-"	→	Purchaseorder.prefix	□
ON	Transaction.tranId	→	Purchaseorder.number	□
ON	="NS-" + Transaction.tranId	→	Purchaseorder.name	□
ON	="1"	→	Purchaseorder.auto_track_payable	□
ON	from purchaser set	→	Purchaseorder.carrier_id	□
ON	from purchaser set	→	Purchaseorder.accounts_payable_	□
ON	from purchaser set	→	Purchaseorder.receiving_id	□
ON	from purchaser set	→	Purchaseorder.ship_complete_only	□

11. Click **Save**.
12. (Optional) Add the POs (Purchase Orders) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

POs (Purchase Orders) Import Field Mapping Definition

The POs (Purchase Orders) import workflow includes the following mapping groups:

- PO [Integration point: PURCHASEORDER_TO_PURCHASEORDER] — See PO Mapping Group.
- Purchase Item [Integration point: PURCHASE_ITEM] — See Purchase Item Mapping Group.

Note: The POs (Purchase Orders) import workflow also creates purchase request, request item and fulfillment records. Mappings are not available for these on the OpenAir NetSuite Connector UI and cannot be modified. See Other Records Created or Updated by the POs (Purchase Orders) Import Workflow.

- Purchase request and request item records are created because they are required in OpenAir — A purchase item must be associated with a request item and purchase request.
- Fulfillment records are created based on whether the item can be fulfilled.

PO Mapping Group

Use the PO mapping group to map information on purchase order records in OpenAir with information on purchase order records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes task name, associated customer and project, start date, currency, and planned hours.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Purchase Order [PurchaseOrder] Schema: PurchaseOrder (2020.1)|PurchaseOrder (2023.1) XSD: TransactionSearchRowBasic — platformCommon (2020.1)|platformCommon (2023.1)	PO [Purchaseorder] Data Dictionary: purchaseorder	PO [PURCHASEORDER_TO_PURCHASEORDER]

Default and Custom Filters

By default, purchase order records are imported from NetSuite into OpenAir if the following conditions are met:

- The **Status** of the purchase order in NetSuite must be one of the following: Fully Billed, Partially Received, Pending Bill, Pending Billing/Partially Received, or Pending Receipt.
- The **Export to OpenAir** [custbody_oa_export_to_openair] box is checked on the project task record in NetSuite.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the PO mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Purchase Item Mapping Group

Use the Purchase Item mapping group to map information on purchase item records in OpenAir with information on purchase order line items in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes task name, associated customer and project, start date, currency, and planned hours.
- Additional Field Mappings — Best practice guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Purchase Order Item [PurchaseOrderItem] Schema: PurchaseOrderItem (2020.1) | PurchaseOrderItem (2023.1) XSD: TransactionSearchRowBasic — platformCommon (2020.1) | platformCommon (2023.1)	Purchase Item [Purchase_item] Data Dictionary: purchase_item	Purchase Item [PURCHASE_ITEM]

Default and Custom Filters

By default, purchase order item records are imported from NetSuite into OpenAir if the following conditions are met:

- The purchase order is imported from NetSuite into OpenAir.
- The purchase order must reference a vendor.
- The purchase order line item must reference an item with a matching product record in OpenAir.
- The **Export to OpenAir** [`custcol_oa_export_to_openair`] box is checked on the purchase order line item in NetSuite.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

NetSuite fields declared and used in filters in the Purchase Item mapping group should also be declared in the PO filters.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Purchase Item mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

> ⚠ **Important:** Review the following guidelines
>
> - You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
> - For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
> - Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

Review the following guidelines:

- NetSuite fields declared and used in advanced mappings in the Purchase Item mapping group should also be declared in the PO mapping group.

Other Records Created or Updated by the POs (Purchase Orders) Import Workflow

The POs (Purchase Orders) import workflow also creates purchase request, request item and fulfillment records. Mappings are not available for these on the OpenAir NetSuite Connector UI and cannot be modified.

Purchase Requests

See the OpenAir Help Center for detailed information about information added to the purchase request records created for each purchase order line item imported into OpenAir. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Request Items

See the OpenAir Help Center for detailed information about information added to the request item records created for each purchase order line item imported into OpenAir. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Fulfillments

See the OpenAir Help Center for detailed information about information added or updated to the fulfillment records created for each purchase order line item imported into OpenAir if not fulfilled. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

POs (Vendor Bills) Import Workflow

The POs (Vendor Bills) import workflow lets you:

- Create new vendor bill records in NetSuite first and then import them into OpenAir as purchase order.
- Update vendor bill records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding purchase order record in OpenAir.

When importing a vendor bill into OpenAir as purchase order, the following records are created in OpenAir:

- Purchase order — A single purchase order is created or updated in OpenAir to match the NetSuite vendor bill.
- Purchase items — Each vendor bill line item marked for export on the NetSuite vendor bill imports into a purchase item.
- Purchase requests —A purchase request record is created for each vendor bill line item marked for export on the NetSuite vendor bill. This is because in OpenAir, purchase request and request item in OpenAir must be associated to a single project, whereas in NetSuite, vendor bills can reference multiple projects.
- Request items — Each purchase request includes one request item.
- Fulfillments — A fulfillment record is created for each purchase item.

> **Note:** You can also import vendor credits from NetSuite into OpenAir as purchase orders with negative purchase items. See POs (Vendor Credits) Import Workflow.
>
> You can import purchase orders instead of vendor bills from NetSuite into OpenAir as purchase orders using the POs (Purchase Orders) import workflows. The POs (Vendor Bills) and POs (Purchase Orders) import workflows are mutually exclusive. See POs (Purchase Orders) Import Workflow.

> **Important:** Changes made to purchase order records in OpenAir after the initial import are not exported back to NetSuite.

For information about vendor bills in NetSuite, see Vendor Bills (External link to NetSuite SuiteAnswers | Answer ID 11156).

For information about field mapping definitions for this workflow, see POs (Vendor Bills) Import Field Mapping Definition.

Requirements and Downstream Dependencies

This workflow is mutually exclusive with the POs (Purchase Orders) import workflow.

The following workflows must also be enabled:

- Products (Purchase Items) Import Workflow
- Vendors (Vendors) Import Workflow

Matching project records must exist in both NetSuite and OpenAir before you can import vendor bills from NetSuite into OpenAir as purchase orders.

Matching vendor records must exist in both NetSuite and OpenAir before you can import vendor bills from NetSuite into OpenAir as purchase orders.

> **Note:** You can use the POs (Vendor Bills) import workflow and Expense Reports (Expense Reports) export workflow in conjunction to import vemdor bills from NetSuite into OpenAir as purchase orders and to export expense reports from OpenAir to NetSuite as vendor bills. See Expense Reports (Expense Reports) Export Workflow.
>
> Exporting expense reports from OpenAir to NetSuite as vendor bills requires the associated user record in OpenAir being matched with a vendor record in NetSuite. To use both workflows in conjunction, enable the Employees (Vendors) import workflow and use the **OpenAir: User or Vendor** [custentity_oa_user_or_vendor] field on the vendor record in NetSuite to determine whether the vendor should be imported into OpenAir as a vendor or as an employee (user). See Employees (Vendors) Import Workflow.

After you import vendor bills from NetSuite into OpenAir as purchase orders, you can use project billing rules to create product charges based on purchase items and fulfillments, add these charges to your invoices in OpenAir, and export the invoices to NetSuite. Markups can be used.

- The **NetSuite vendor bill billable** [netsuite_vb_billable] box is checked on the purchase item in OpenAir if the vendor bill line item is billable in NetSuite. When exporting invoices with product charges for purchase items imported into OpenAir using the POs (Vendor Bills) import workflow, these charges export as billable line items on the NetSuite invoice Billable Items subtab.
  - Use the Bill Line mapping group to export information on the product charges created from purchase items. See Bill Line Mapping Group.
  - Use the Vendor Bill Line mapping group to export information on the purchase items. See Vendor Bill Line Mapping Group.
- If the **NetSuite vendor bill billable** [netsuite_vb_billable] box is NOT checked on the purchase item in OpenAir or if you are using Invoices (Invoices) workflow settings modifying the default behavior, charges export as invoice line items on the NetSuite invoice Items subtab, and not as billable line items on the Billable Items subtab.

  The following Invoices (Invoices) workflow settings modify the default behavior:
  - You can check the **Disable vendor bill linkage on invoices in NetSuite integration** to remove the vendor bill reference on the invoice line item. See Disable vendor bill linkage on invoices in NetSuite integration.

- You can also check the **Use Service item instead of Product item during invoices export for purchase charges** to reference a service item instead of a product item on the invoice line item. See Use Service item instead of Product item during invoices export for purchase charges

The following Invoices (Invoices) import workflow mapping groups are used to export billable purchase items imported from a NetSuite vendor bill:

- Use the Invoice Line mapping group to export information on the product charges created from purchase items. See Invoice Line Mapping Group.
- Use the Purchase Line mapping group to export information on the purchase items. See Purchase Line Mapping Group.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir purchase order record after you create or update the vendor bill record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir purchase order record when the vendor bill form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir purchase order record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the POs (Vendor Bills) Import Workflow

Use the following steps to set up the workflow.

To set up the POs (Vendor Bills) import workflow in NetSuite

1. In NetSuite, do one of the following:
   - Go to Setup > Company > General Preferences (Custom Preferences), if you are using the SuiteScript 1 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
   - Go to Customization > Lists, Records and Fields > Records > [*View list forOpenAir: SRP Integration Settings*] > [*View OpenAir SRP Integration Settings Record*] , if you are using the SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
2. Check the **Vendor Bill to OpenAir PO Integration** box.
3. Click **Save**.
4. Go to Setup > Company > Enable Features, and click the Transactions subtab.
5. Check the **Advanced Receiving** box under the Shipping & Receiving section. The Advanced Receiving feature is required to create purchase item receipts in NetSuite.
6. Click **Save**.

7. Go to Setup > Accounting > Invoicing Preferences and clear the **Combine Billable Items on Invoices** box. this feature is not compatible with the POs (vendor Bills) import workflow.
8. To let employees view projects and items and check the **Billable** and **Export to OpenAir** boxes for vendor bill line items and vendor bill records, go to Home > Set Preferences and check the **Validate OpenAir PO/Vendor Bill** box on the Custom Preferences tab.

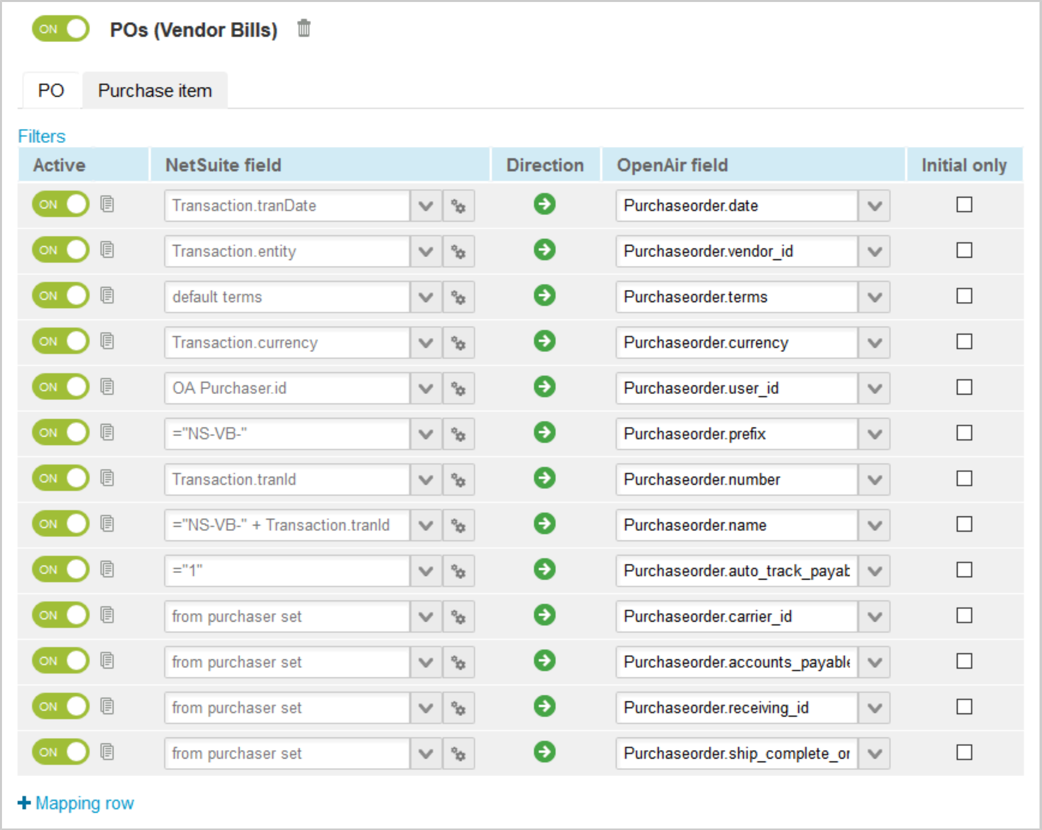
To set up the POs (Vendor Bills) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button then NetSuite Connector Administration.

   The OpenAir NetSuite Connector administration form appears.
3. Check the **POs (Vendor Bills)** box under the Imports section.
4. Click **Save** then **Close**.
5. If the purchaser integration is not enabled, designate a purchaser to use when importing vendor bills into OpenAir as purchase orders. To do so:
   a. Go to Administration > Application Settings > Purchases > Purchasers > [*Select a purchaser*].
   b. Check the **NetSuite Purchase Order Integration purchaser** [`netsuite_purchaser`] box.
   c. Click **Save**.

> **Note:** To create a purchaser, you must create at least one record for each of the following record types in Administration > Application Settings > Purchases > Accounts payable locations, Receiving locations, Carriers, Shipping terms, and F.O.B. locations.
>
> If the purchaser integration is enabled, matching employee records must exist in OpenAir and NetSuite for employees creating purchase orders in NetSuite, and purchaser records marked as NetSuite purchaser must exist for these employees in OpenAir. For more information about the purchaser integration, see Enable purchaser integration

6. Go to Administration > NetSuite Connector > Mapping.
7. Click **New workflow** then **POs (Vendor Bills)**.
8. Click **POs (Vendor Bills)** in the list of integration workflows.
9. Modify the field mapping definition table for the Tasks (Project Tasks) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and POs (Vendor Bills) Import Field Mapping Definition.
10. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Transaction.tranDate	➔	Purchaseorder.date	☐
ON	Transaction.entity	➔	Purchaseorder.vendor_id	☐
ON	default terms	➔	Purchaseorder.terms	☐
ON	Transaction.currency	➔	Purchaseorder.currency	☐
ON	OA Purchaser.id	➔	Purchaseorder.user_id	☐
ON	="NS-VB-"	➔	Purchaseorder.prefix	☐
ON	Transaction.tranId	➔	Purchaseorder.number	☐
ON	="NS-VB-" + Transaction.tranId	➔	Purchaseorder.name	☐
ON	="1"	➔	Purchaseorder.auto_track_payat	☐
ON	from purchaser set	➔	Purchaseorder.carrier_id	☐
ON	from purchaser set	➔	Purchaseorder.accounts_payabl	☐
ON	from purchaser set	➔	Purchaseorder.receiving_id	☐
ON	from purchaser set	➔	Purchaseorder.ship_complete_or	☐

11. Click **Save**.
12. (Optional) Add the POs (Vendor Bills) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

POs (Vendor Bills) Import Field Mapping Definition

The POs (Vendor Bills) import workflow includes the following mapping groups:

- PO [Integration point: VENDORBILL_TO_PURCHASEORDER] — See PO Mapping Group.
- Purchase Item [Integration point: VENDORBILL_TO_PURCHASE_ITEM] — See Purchase Item Mapping Group.

Note: The POs (Vendor Bills) import workflow also creates purchase request, request item and fulfillment records. Mappings are not available for these on the OpenAir NetSuite Connector UI and cannot be modified. See Other Records Created or Updated by the POs (Vendor Bills) Import Workflow.

- Purchase request and request item records are created because they are required in OpenAir — A purchase item must be associated with a request item and purchase request.
- Fulfillment records are created based on the **Quantity** on the vendor bill line item.

PO Mapping Group

Use the PO mapping group to map information on purchase order records in OpenAir with information on vendor bill records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes task name, associated customer and project, start date, currency, and planned hours.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Vendor Bill [VendorBill] Schema: VendorBill (2020.1) | VendorBill (2023.1) XSD: TransactionSearchRowBasic — platformCommon (2020.1) | platformCommon (2023.1)	PO [Purchaseorder] Data Dictionary: purchaseorder	PO [VENDORBILL_TO_PURCHASEORDER]

Default and Custom Filters

By default, vendor bill records are imported from NetSuite into OpenAir as purchase orders if the following conditions are met:

- The **Status** of the vendor bill in NetSuite must be one of the following: Open or Paid in Full. Vendor bills that are not approved do not import to OpenAir.
- The **Export to OpenAir** [custbody_oa_export_to_openair] box is checked on the project task record in NetSuite.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the PO mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Purchase Item Mapping Group

Use the Purchase Item mapping group to map information on purchase item records in OpenAir with information on vendor bill line items in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes task name, associated customer and project, start date, currency, and planned hours.
- Additional Field Mappings — Best practice guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Purchase Order Item [PurchaseOrderItem] Schema: PurchaseOrderItem (2020.1) | PurchaseOrderItem (2023.1) XSD: TransactionSearchRowBasic — platformCommon (2020.1) | platformCommon (2023.1)	Purchase Item [Purchase_item] Data Dictionary: purchase_item	Purchase Item [VENDORBILL_TO_PURCHASE_ITEM]

Default and Custom Filters

By default, vendor bill line item records are imported from NetSuite into OpenAir as purchase items if the following conditions are met:

- The vendor bill is imported from NetSuite into OpenAir as a purchase order.
- The vendor bill must reference a vendor.
- The vendor bill line item must reference an item with a matching product record in OpenAir.
- The **Export to OpenAir** [`custcol_oa_export_to_openair`] box is checked on the vendor bill line item in NetSuite.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

NetSuite fields declared and used in filters in the Purchase Item mapping group should also be declared in the PO filters.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Purchase Item mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

> ⚠ **Important:** Review the following guidelines
>
> - You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
> - For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
> - Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

Review the following guidelines:

- NetSuite fields declared and used in advanced mappings in the Purchase Item mapping group should also be declared in the PO mapping group.

Other Records Created or Updated by the POs (Vendor Bills) Import Workflow

The POs (Vendor Bills) import workflow also creates purchase request, request item and fulfillment records. Mappings are not available for these on the OpenAir NetSuite Connector UI and cannot be modified.

Purchase Requests

See the OpenAir Help Center for detailed information about information added to the purchase request records created for each vendor bill line item imported into OpenAir. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Request Items

See the OpenAir Help Center for detailed information about information added to the request item records created for each vendor bill line item imported into OpenAir. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Fulfillments

See the OpenAir Help Center for detailed information about information added or updated to the fulfillment records created for each vendor bill line item imported into OpenAir if not fulfilled. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Retainers (Applied Customer Deposits) Import Workflow

The Retainers (Applied Customer Deposits) import workflow lets you:

- Record customer deposits and apply them against invoices previously exported from OpenAir to NetSuite, and then import the applied customer deposits into OpenAir as retainers. OpenAir NetSuite Connector creates retainer records for the amount of the customer deposit that was applied against a particular invoice in NetSuite and not for the full amount of the customer deposit. OpenAir NetSuite Connector also adds a draw against retainer line to the corresponding invoice in OpenAir.
- Update applied customer deposits in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding retainers in OpenAir.

Important: Changes made to the retainer records in OpenAir after the initial import are not exported back to NetSuite.

For more information about recording or applying a customer deposit in NetSuite, see help topics under Customer Deposits (External link to NetSuite SuiteAnswers | Answer ID: 9121).

For information about field mapping definitions for this workflow, see Retainers (Applied Customer Deposits) Field Mapping Definition.

Requirements and Downstream Dependencies

Matching invoice transaction records must exist in both NetSuite and OpenAir before you can import deposits applied to invoices from NetSuite into OpenAir.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir retainer records after you create or update the applied customer deposit record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir retainer records when the customer deposit form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir retainer record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Retainers (Applied Customer Deposits) Import Workflow

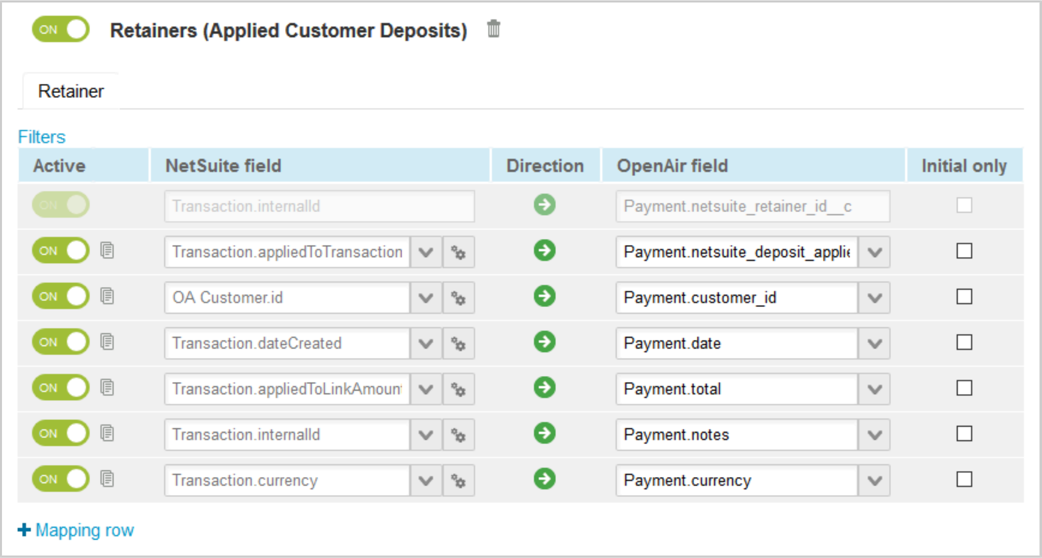
Use the following steps to set up the Retainers (Applied Customer Deposits) import workflow.

No additional setting is required in NetSuite.

To set up the Retainers (Applied Customer Deposits) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button then NetSuite Connector Administration.

   The OpenAir NetSuite Connector administration form appears.
3. Check the **Retainers (Applied Customer Deposits)** box under the Imports section.
4. Click **Save** then **Close**.
5. Go to Administration > NetSuite Connector > Mapping.
6. Click **New workflow** then **Retainers (Applied Customer Deposits)**.
7. Click **Retainers (Applied Customer Deposits)** in the list of integration workflows.
8. Modify the field mapping definition table for the Retainers (Applied Customer Deposits) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Retainers (Applied Customer Deposits) Field Mapping Definition.
9. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON Retainers (Applied Customer Deposits)

Retainer

Filters

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Transaction.internalId	→	Payment.netsuite_retainer_id__c	□
ON	Transaction.appliedToTransaction	→	Payment.netsuite_deposit_applie	□
ON	OA Customer.id	→	Payment.customer_id	□
ON	Transaction.dateCreated	→	Payment.date	□
ON	Transaction.appliedToLinkAmoun	→	Payment.total	□
ON	Transaction.internalId	→	Payment.notes	□
ON	Transaction.currency	→	Payment.currency	□

+ Mapping row

10. Click **Save**.
11. (Optional) Add the Retainers (Applied Customer Deposits) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Retainers (Applied Customer Deposits) Field Mapping Definition

The Retainers (Applied Customer Deposits) import workflow includes one mapping group:

- Retainer [Integration point: DEPOSIT_TO_RETAINER] — See Retainer Mapping Group.

Retainer Mapping Group

Use the Retainer mapping group to map information on retainer records in OpenAir with information about customer deposits applied to invoice transaction records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes NetSuite invoice ID, customer internal ID in OpenAir, date, amount, notes, and currency. Filters are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Customer Deposit [CustomerDeposit] Schema: CustomerDeposit (2020.1)|CustomerDeposit (2023.1) XSD: TransactionSearchRowBasic — platformCommon (2020.1)|platformCommon (2023.1)	Payment [payment] Data Dictionary: payment	Retainer [DEPOSIT_TO_RETAINER]

Default and Custom Filters

By default, applied customer deposit records are imported from NetSuite into OpenAir if there is an invoice record in OpenAir matching the invoice record to which the customer deposit was applied in NetSuite.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Retainer mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

OpenAir NetSuite Connector updates automatically the OpenAir invoice record corresponding to the NetSuite invoice transaction to which the customer deposit is applied. The following information is updated:

- `Invoice.draw` — total amount drawn against the retainer for the invoice.
- `Invoice.draw_date` — the date of the draw.
- `Invoice.balance` — outstanding balance on the invoice.

Additional Field Mappings

⚠ **Important:** Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Credits (Applied Credit Memos) Import Workflow

The Credits (Applied Credit Memos) import workflow lets you:

- Create credit memo records and apply to invoices previously exported from OpenAir to NetSuite, and then import the credit memos into OpenAir as credit invoices with negative charges.

  ⓘ **Note:** You can also create credit invoices in OpenAir and export them to NetSuite as credit memos. See Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Export Workflow.

- Update credit memo records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding invoices and negative charges in OpenAir.

⚠ **Important:** Changes made to the invoice and charge records in OpenAir after the initial import are not exported back to NetSuite.

When you import a new credit memo record from NetSuite into OpenAir:

- OpenAir NetSuite Connector creates a credit invoice in OpenAir matching the credit memo in NetSuite.
- OpenAir NetSuite Connector adds a credit line on the original invoice in OpenAir corresponding to the invoice against which you applied the credit memo in NetSuite.

When the **Import taxes applied to exported invoices** box is checked on the settings form for the Invoices (Invoices) export workflow, the tax for items included on the credit memo is also imported onto the credit invoice in OpenAir. See Import taxes applied to exported invoices.

For more information about applying a credit memo in NetSuite, see Applying a Customer Credit Memo (External link to NetSuite SuiteAnswers | Answer ID: 9108).

For information about workflow specific settings, see Credits (Applied Credit Memos) Import Workflow Settings.

For information about field mapping definitions for this workflow, see Credits (Applied Credit Memos) Import Field Mapping Definition.

Requirements and Downstream Dependencies

Matching customer, project, service item and invoice transaction records must exist in both NetSuite and OpenAir before you can import applied credit memos from NetSuite into OpenAir.

Credit memos can only be imported into OpenAir if they were applied to invoices that were exported from OpenAir to NetSuite.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
✓	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir credit invoice and charge records after you create or update the credit memo record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir invoice and charge records when the Credit Memo form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir invoice record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Credits (Applied Credit Memos) Import Workflow

Use the following steps to set up the Credits (Applied Credit Memos) import workflow.

No additional setting is required in NetSuite.

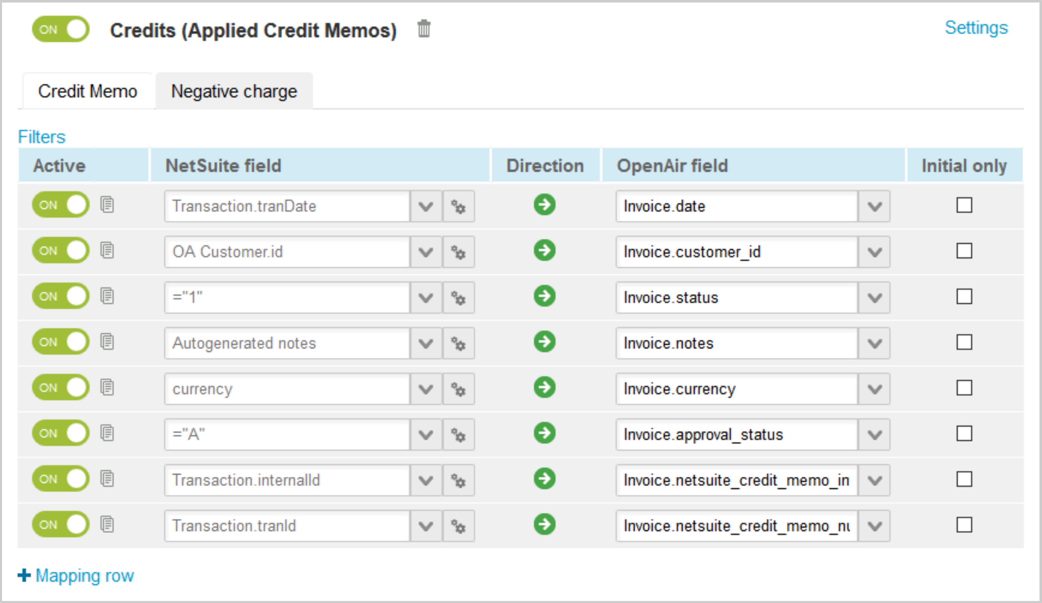
> **Note:** Before you set up the workflow make sure the Credit and Rebill Invoices optional feature is enabled in OpenAir. For more information, see Optional Features.
>
> To enable the Credit and Rebill Invoices feature, contact OpenAir Customer Support.

To set up the Credits (Applied Credit Memos) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button then NetSuite Connector Administration.

   The OpenAir NetSuite Connector administration form appears.

3. Check the **Credits (Applied Credit Memos)** box under the Imports section.
4. Click **Save** then **Close**.
5. Go to Administration > NetSuite Connector > Mapping.
6. Click **New workflow** then **Credits (Applied Credit Memos)**.
7. Click **Credits (Applied Credit Memos)** in the list of integration workflows.
8. Click **Workflow settings** to view and modify the settings specific to this workflow. For more information about the workflow settings, see Credits (Applied Credit Memos) Import Workflow Settings.
9. Modify the field mapping definition table for the Credits (Applied Credit Memos) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Credits (Applied Credit Memos) Import Field Mapping Definition.
10. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON **Credits (Applied Credit Memos)** Settings

Credit Memo | Negative charge

Filters

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Transaction.tranDate	→	Invoice.date	□
ON	OA Customer.id	→	Invoice.customer_id	□
ON	="1"	→	Invoice.status	□
ON	Autogenerated notes	→	Invoice.notes	□
ON	currency	→	Invoice.currency	□
ON	="A"	→	Invoice.approval_status	□
ON	Transaction.internalId	→	Invoice.netsuite_credit_memo_in	□
ON	Transaction.tranId	→	Invoice.netsuite_credit_memo_nu	□

+ Mapping row

11. Click **Save**.
12. (Optional) Add the Credits (Applied Credit Memos) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Credits (Applied Credit Memos) Import Workflow Settings

You can control Credits (Applied Credit Memos) import workflow settings from the NetSuite Connector UI. On the NetSuite Connector Mapping screen, select the Credits (Applied Credit Memos) import workflow and click **Settings** to view the workflow settings form.

The following settings are available for the Credits (Applied Credit Memos) import workflow:

- Map the charge type from the NetSuite credit memo

Map the charge type from the NetSuite credit memo

Check this box to store the OpenAir **Charge type** for each item line on the invoices exported from OpenAir to NetSuite and use the same **Charge type** in the negative charge created in OpenAir when applying a credit memo in NetSuite and importing it into OpenAir as a negative invoice.

If you apply a credit memo against an invoice and the **OpenAir: Charge Type** is not set on the invoice line item in NetSuite, the **OpenAir: Charge Type** shows a null value on the credit memo line item. You can select the **OpenAir: Charge Type** on the credit memo line item to determine the **Charge type** of the corresponding negative charge in OpenAir. If left blank, the Credits (Applied Credit Memos) import workflow determines the charge type based on the **Quantity** on the credit memo line item.

> **Note:** The **OpenAir: Charge Type** [custcol_oa_credit_charge_type] transaction line custom field as well as the customrecord_oa_credit_charge_type custom record and customsearch_oa_credit_charge_type custom record are created when you install NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) 1.22 or later version.

Credits (Applied Credit Memos) Import Field Mapping Definition

The Credits (Applied Credit Memos) import workflow includes the following mapping groups:

- Credit Memo [Integration point: CREDIT_MEMO_TO_CREDIT] — See Credit Memo Mapping Group.
- Negative Charge [Integration point: CREDIT_LINE] — See Negative Charge Mapping Group.

Credit Memo Mapping Group

Use the Credit Memo mapping group to map information on credit (negative invoice) records in OpenAir with information about credit memos applied to invoice transaction records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes credit memo number, date, customer internal ID in OpenAir, notes, and currency. Filters are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Credit Memo [CreditMemo] Schema: CreditMemo (2020.1) | CreditMemo (2023.1) XSD: TransactionSearchRowBasic — platformCommon (2020.1) | platformCommon (2023.1)	Invoice [invoice] Data Dictionary: invoice	Credit Memo [CREDIT_MEMO_TO_CREDIT]

Default and Custom Filters

By default, applied credit memo records are imported from NetSuite into OpenAir if the invoice records to which the credit memo was applied in NetSuite were imported from OpenAir.

> **Important:** The Credits (Applied Credit Memos) import returns an error if one or more of the invoices the credit memo was applied to in NetSuite did not originate from an OpenAir invoice.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Credit Memo mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

OpenAir NetSuite Connector updates automatically the OpenAir invoice records corresponding to the NetSuite invoice transactions to which the credit memo is applied. The following information is updated:

- `Invoice.credit` — total amount credited against the invoice.
- `Invoice.credit_reason` — appended with text `'NetSuite credit memo' + Transaction.tranId`.
- `Invoice.balance` — outstanding balance on the invoice.

Additional Field Mappings

> **Important:** Review the following guidelines
>
> - You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
> - For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
> - Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Negative Charge Mapping Group

Use the Negative Charge mapping group to map information about charges on credit (negative invoice) records in OpenAir with information about credit memos applied to invoice transaction records in NetSuite.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes credit memo number,

date, amount, internal ID of the associated customer, service, product and project in OpenAir, charge type, time and rate or quantity and cost, and currency.

- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Credit Memo [CreditMemoItem] Schema: CreditMemoItem (2020.1) | CreditMemoItem (2023.1) XSD: CreditMemo — tranCust (2020.1) | tranCust (2023.1)	Charge [slip] Data Dictionary: slip	Negative Charge [CREDIT_LINE]

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Negative Charge mapping group tab. Note that you need to log in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Tasks (Support Cases) Import Workflow

The Tasks (Support Cases) import workflow lets you:

- Create new support case records in NetSuite first and then import them into OpenAir as tasks.
- Update support case records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding task record in OpenAir.

Note: You can also:

- Create and update project task records in NetSuite and then import them into OpenAir as tasks. See Tasks (Project Tasks) Import Workflow.
- Update tasks records in OpenAir and run the integration to update the project task records in NetSuite using the Update NetSuite Tasks with OpenAir Task Data export workflow. See Update NetSuite Tasks with OpenAir Task Data.

Important: Changes made to the task records in OpenAir after the initial import are not exported back to support case records in NetSuite.

For information about field mapping definitions for this workflow, see Tasks (Support Cases) Import Field Mapping Definition.

Requirements and Downstream Dependencies

Matching project records must exist in both NetSuite and OpenAir before you can import project tasks from NetSuite into OpenAir.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
✓	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir task record after you create or update the support case record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir task record when the Support Case form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir task record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Tasks (Project Tasks) Import Workflow

Use the following steps to set up the workflow

To set up the Tasks (Support Cases) import workflow in NetSuite

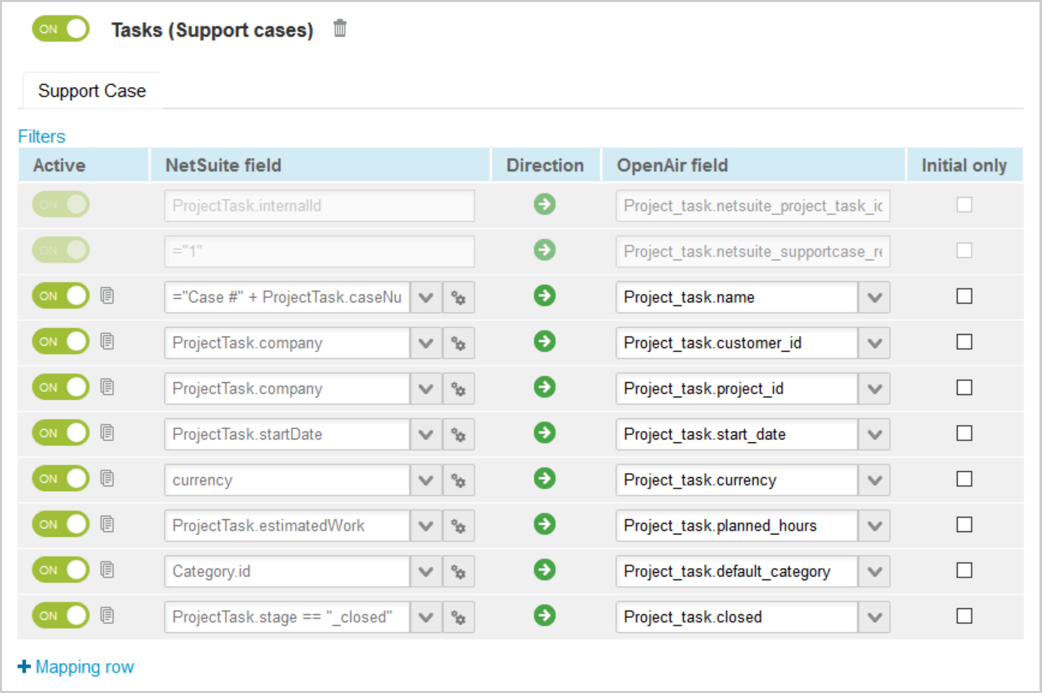
1. In NetSuite, do one of the following:
   - Go to Setup > Company > General Preferences (Custom Preferences), if you are using the SuiteScript 1 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
   - Go to Customization > Lists, Records and Fields > Records > [*View list forOpenAir: SRP Integration Settings*] > [*View OpenAir SRP Integration Settings Record*] , if you are using the SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.

2. Check the **Enable Support Case Integration** box.
3. Click **Save**.

To set up the Tasks (Support Cases) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button then NetSuite Connector Administration.

   The OpenAir NetSuite Connector administration form appears.
3. Check the **Tasks (Support Cases)** box under the Imports section.
4. Click **Save** then **Close**.
5. Go to Administration > NetSuite Connector > Mapping.
6. Click **New workflow** then **Tasks (Support Cases)**.
7. Click **Tasks (Support Cases)** in the list of integration workflows.
8. Modify the field mapping definition table for the Tasks (Support Cases) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Tasks (Support Cases) Import Field Mapping Definition.
9. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON Tasks (Support cases)

Support Case

Filters

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	ProjectTask.internalId	→	Project_task.netsuite_project_task_ic	□
ON	="1"	→	Project_task.netsuite_supportcase_re	□
ON	="Case #" + ProjectTask.caseNu	→	Project_task.name	□
ON	ProjectTask.company	→	Project_task.customer_id	□
ON	ProjectTask.company	→	Project_task.project_id	□
ON	ProjectTask.startDate	→	Project_task.start_date	□
ON	currency	→	Project_task.currency	□
ON	ProjectTask.estimatedWork	→	Project_task.planned_hours	□
ON	Category.id	→	Project_task.default_category	□
ON	ProjectTask.stage == "_closed"	→	Project_task.closed	□

10. Click **Save**.
11. (Optional) Add the Tasks (Project Tasks) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Tasks (Support Cases) Import Field Mapping Definition

The Tasks (Support Cases) import workflow includes one mapping group:

- Support Case [Integration point: SUPPORTCASE_TO_PR_TASK] — See Support Case Mapping Group.

Support Case Mapping Group

Use the task mapping group to map information on task records in OpenAir with information on project task records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes task name, associated customer and project, start date, currency, and planned hours.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Support case[SupportCase] Schema: SupportCase (2020.1)|SupportCase (2023.1) XSD: SupportCaseSearchRowBasic — platformCommon (2020.1)|platformCommon (2023.1)	Task [project_task] Data Dictionary: project_task	Support Case [SUPPORTCASE_TO_PR_TASK]

Default and Custom Filters

By default, project task records are imported from NetSuite into OpenAir if the following conditions are met:

- Matching project records exists in OpenAir and NetSuite for the project associated with the project task in NetSuite.
- The **Export to OpenAir** [custevent_oa_export_to_openair] box is checked on the support case record in NetSuite.

> **Note:** If the **Export to OpenAir** [custevent_oa_export_to_openair] box is checked on the customer record associated with the support case, the support case is marked automatically for export to OpenAir.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Support Case mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

OpenAir NetSuite Connector creates or updates a project task assignment record in OpenAir when a task is added or updated using the Tasks (Support Cases) import workflow and the NetSuite support case is assigned to an employee matching an employee record in OpenAir. The project task assignment record includes the following information:

- `Project_task_assign.user_id` —*Looked up OpenAir field value*: `id` of OpenAir employee record with `netsuite_user_id` equal to the NetSuite field `SupportCase.userId`
- `Project_task_assign.project_task_id` — `id` of the imported project task record.

OpenAir NetSuite Connector runs the project calculation automatically when a task is added or updated using the Tasks (Support Cases) import workflow.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines.

POs (Vendor Credits) Import Workflow

The POs (Vendor Credits) import workflow lets you:

- Create new vendor credit records in NetSuite first and then import them into OpenAir as purchase order. NetSuite vendor credit details are imported as negative purchase items.
- Update vendor credit records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding purchase order record in OpenAir.

When importing a vendor credit into OpenAir as purchase order, the following records are created in OpenAir:

- Purchase order — A single purchase order is created or updated in OpenAir to match the NetSuite vendor credit.
- Purchase items — Each vendor credit line item marked for export on the NetSuite vendor credit imports into a purchase item with a negative **Price** and negative **Total**.

- Purchase requests —A purchase request record is created for each vendor credit line item marked for export on the NetSuite vendor credit. This is because in OpenAir, purchase request and request item in OpenAir must be associated to a single project, whereas in NetSuite, vendor credits can reference multiple projects.
- Request items — Each purchase request includes one request item.
- Fulfillments — A fulfillment record is created for each purchase item.

Note: You can import purchase orders or vendor bills from NetSuite into OpenAir as purchase orders. The POs (Vendor Bills) and POs (Purchase Orders) import workflows are mutually exclusive. See POs (Purchase Orders) Import Workflow and POs (Vendor Bills) Import Workflow.

Important: Changes made to purchase order records in OpenAir after the initial import are not exported back to NetSuite.

For information about vendor credits in NetSuite, see Vendor Credits (External link to NetSuite SuiteAnswers | Answer ID 11165).

For information about field mapping definitions for this workflow, see POs (Vendor Credits) Import Field Mapping Definition.

Requirements and Downstream Dependencies

The following workflows must also be enabled:

- Products (Purchase Items) Import Workflow
- Vendors (Vendors) Import Workflow

Matching project records must exist in both NetSuite and OpenAir before you can import vendor credits from NetSuite into OpenAir as purchase orders.

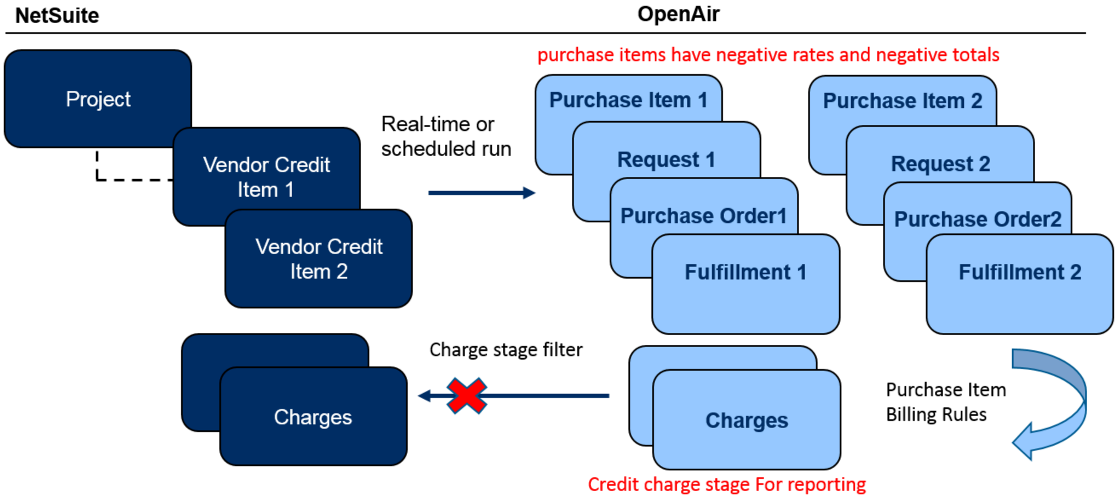
Matching vendor records must exist in both NetSuite and OpenAir before you can import vendor credits from NetSuite into OpenAir as purchase orders.

After you import vendor credits from NetSuite into OpenAir as purchase orders, you can use project billing rules to create negative product charges based on purchase items and fulfillments, add these charges to your invoices in OpenAir, and export the invoices to NetSuite, in the same way as with vendor bills imported as purchase orders. See POs (Vendor Bills) Import Workflow.

Note: When importing vendor credits associated with a specific project, you can create a project billing rule to exclude negative charges associated with purchase items imported from vendor credits from invoicing. This also stops negative charges from being exported to NetSuite as part of the Invoices (Invoices) or Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos export workflows. See Excluding Charges Associated with Imported Vendor Credits from Invoicing.

Excluding Charges Associated with Imported Vendor Credits from Invoicing

You can create a project billing rule to exclude negative charges associated with purchase items imported from vendor credits from invoicing. This also stops negative charges from being exported to NetSuite as part of the Invoices (Invoices) or Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos export workflows. This can be useful if you do not want to include the negative charges as a discount on invoices or on a credit memo to customers, but want to report on negative charges from vendor credits.



To exclude charges associated with imported vendor credits from invoicing

1. Go to Projects > Projects > [select a project associated with vendor credits]
2. Click the **Create** button then **Purchase item billing rule** under Billing rule.

   The Purchase item billing rule form appears
3. Click **Create** next to **POs From NetSuite vendor credit** under the Billing rule filters form section.

   The from NetSuite vendor credit filter form appears.
4. Select **Yes**.
5. Click **OK** to save the filter and return to the Purchase item billing rule form.
6. Select a **Charge stage** under the Other form section. Make sure the **Exclude charge from invoicing** box is checked on the charge stage record you select.

   > **Note:** To review the charge stages, go to Administration > Application Settings > Invoices > Charge Stages.
7. Set the **Billing rule position** to [First billing rule].

   > **Important:** Placing the billing rule before other purchase item billing rule ensures that the purchase items are not processed by other billing rules and that the associated purchase charges are excluded from invoicing. Billing rules run in the order specified by the **Billing rule position**. Purchase item billing rules can only process purchase items that have not already been processed by other billing rules ahead in the running order.
8. Click **Save**.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir purchase order record after you create or update the vendor credit record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir purchase order record when the vendor credit form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir purchase order record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the POs (Vendor Credits) Import Workflow

Use the following steps to set up the workflow.

To set up the POs (Vendor Credits) import workflow in NetSuite

1. In NetSuite, do one of the following:
   - Go to Setup > Company > General Preferences (Custom Preferences), if you are using the SuiteScript 1 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
   - Go to Customization > Lists, Records and Fields > Records > [*View list forOpenAir: SRP Integration Settings*] > [*View OpenAir SRP Integration Settings Record*] , if you are using the SuiteScript 2 versions of NetSuite OpenAir SRP Integration (Bundle ID 369637) bundle scripts.
2. Check the **Vendor Credit to OpenAir PO Integration** box.
3. Click **Save**.
4. In NetSuite, go to Setup > Company > Enable Features, and click the Transactions subtab.
5. Check the **Advanced Receiving** box under the Shipping & Receiving section. The Advanced Receiving feature is required to create purchase item receipts in NetSuite.
6. Click **Save**.
7. To let employees view projects and items and check the **Billable** and **Export to OpenAir** boxes for vendor credit line items and vendor credit records, go to Home > Set Preferences and check the **Validate OpenAir PO/Vendor Bill** box on the Custom Preferences tab.

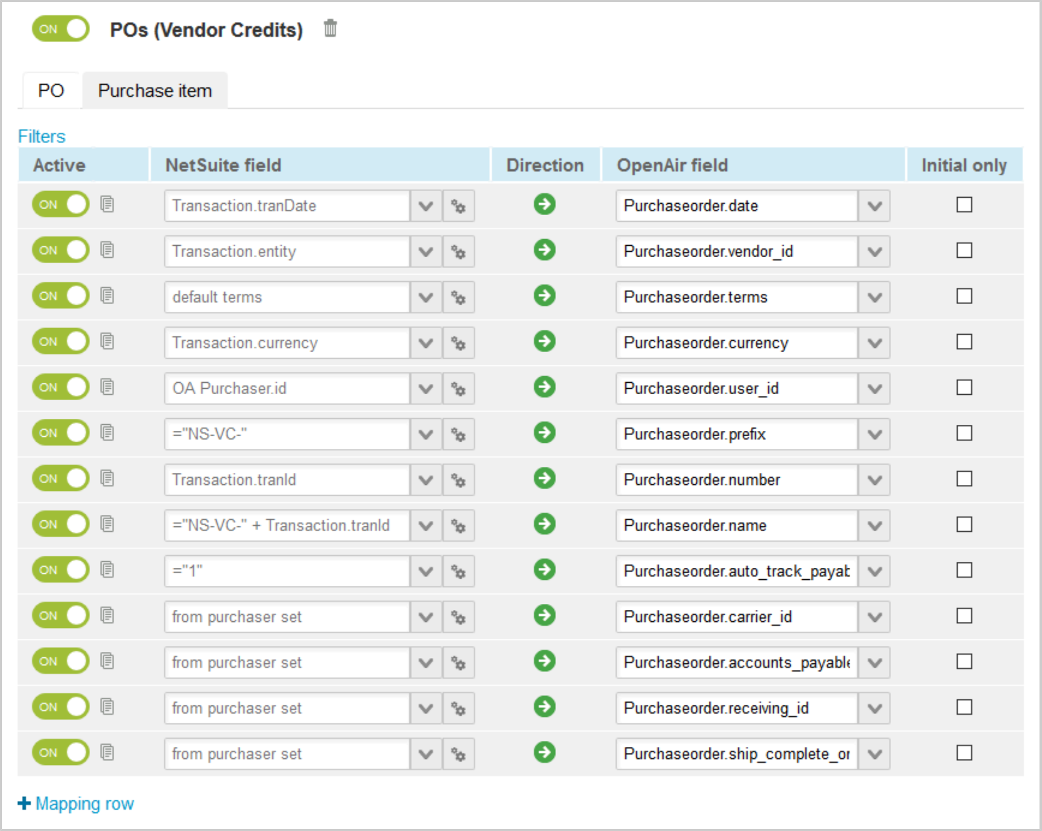
To set up the POs (Vendor Credits) import workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector.
2. Click the Tips button then NetSuite Connector Administration.

   The OpenAir NetSuite Connector administration form appears.
3. Check the **POs (Vendor Credits)** box under the Imports section.
4. Click **Save** then **Close**.
5. If the purchaser integration is not enabled, designate a purchaser to use when importing vendor credits into OpenAir as purchase orders. To do so:
   a. Go to Administration > Application Settings > Purchases > Purchasers > [*Select a purchaser*].
   b. Check the **NetSuite Purchase Order Integration purchaser** [`netsuite_purchaser`] box.
   c. Click **Save**.

> **Note:** To create a purchaser, you must create at least one record for each of the following record types in Administration > Application Settings > Purchases > Accounts payable locations, Receiving locations, Carriers, Shipping terms, and F.O.B. locations.
>
> If the purchaser integration is enabled, matching employee records must exist in OpenAir and NetSuite for employees creating purchase orders in NetSuite, and purchaser records marked as NetSuite purchaser must exist for these employees in OpenAir. For more information about the purchaser integration, see Enable purchaser integration

6. Go to Administration > NetSuite Connector > Mapping.
7. Click **New workflow** then **POs (Vendor Credits)**.
8. Click **POs (Vendor Credits)** in the list of integration workflows.
9. Modify the field mapping definition table for the Tasks (Project Tasks) import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and POs (Vendor Credits) Import Field Mapping Definition.
10. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



Active	NetSuite field	Direction	OpenAir field	Initial only
ON	Transaction.tranDate	→	Purchaseorder.date	□
ON	Transaction.entity	→	Purchaseorder.vendor_id	□
ON	default terms	→	Purchaseorder.terms	□
ON	Transaction.currency	→	Purchaseorder.currency	□
ON	OA Purchaser.id	→	Purchaseorder.user_id	□
ON	="NS-VC-"	→	Purchaseorder.prefix	□
ON	Transaction.tranId	→	Purchaseorder.number	□
ON	="NS-VC-" + Transaction.tranId	→	Purchaseorder.name	□
ON	="1"	→	Purchaseorder.auto_track_payat	□
ON	from purchaser set	→	Purchaseorder.carrier_id	□
ON	from purchaser set	→	Purchaseorder.accounts_payabl	□
ON	from purchaser set	→	Purchaseorder.receiving_id	□
ON	from purchaser set	→	Purchaseorder.ship_complete_or	□

11. Click **Save**.
12. (Optional) Add the POs (Vendor Credits) import workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

POs (Vendor Credits) Import Field Mapping Definition

The POs (Vendor Credits) import workflow includes the following mapping groups:

- PO [Integration point: VENDORBILL_TO_PURCHASEORDER] — See PO Mapping Group.
- Purchase Item [Integration point: VENDORBILL_TO_PURCHASE_ITEM] — See Purchase Item Mapping Group.

Note: The POs (Vendor Credits) import workflow also creates purchase request, request item and fulfillment records. Mappings are not available for these on the OpenAir NetSuite Connector UI and cannot be modified. See Other Records Created or Updated by the POs (Vendor Credits) Import Workflow.

- Purchase request and request item records are created because they are required in OpenAir — A purchase item must be associated with a request item and purchase request.
- Fulfillment records are created based on the **Quantity** on the vendor credit line item.

PO Mapping Group

Use the PO mapping group to map information on purchase order records in OpenAir with information on vendor credit records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes task name, associated customer and project, start date, currency, and planned hours.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Vendor Credit [VendorCredit] Schema: VendorCredit (2020.1)|VendorCredit (2023.1) XSD: TransactionSearchRowBasic — platformCommon (2020.1)|platformCommon (2023.1)	PO [Purchaseorder] Data Dictionary: purchaseorder	PO [VENDORCREDIT_TO_PURCHASEORDER]

Default and Custom Filters

By default, vendor credit records are imported from NetSuite into OpenAir as purchase orders if the following conditions are met:

- The **Status** of the vendor credit in NetSuite must be one of the following: Fully Billed, Partially Received, Pending Bill, Pending Billing/Partially Received, or Pending Receipt.
- The **Export to OpenAir** [custbody_oa_export_to_openair] box is checked on the project task record in NetSuite.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the for detailed information about default field mapping definition for the PO mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Purchase Item Mapping Group

Use the Purchase Item mapping group to map information on purchase item records in OpenAir with information on vendor credit line items in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes task name, associated customer and project, start date, currency, and planned hours.
- Additional Field Mappings — Best practice guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Purchase Order Item [PurchaseOrderItem] Schema: PurchaseOrderItem (2020.1) | PurchaseOrderItem (2023.1) XSD: TransactionSearchRowBasic — platformCommon (2020.1) | platformCommon (2023.1)	Purchase Item [Purchase_item] Data Dictionary: purchase_item	Purchase Item [VENDORCREDIT_TO_PURCHASE_ITEM]

Default and Custom Filters

By default, vendor credit line item records are imported from NetSuite into OpenAir as purchase items if the following conditions are met:

- The vendor credit is imported from NetSuite into OpenAir as a purchase order.
- The vendor credit must reference a vendor.
- The vendor credit line item must reference an item with a matching product record in OpenAir.
- The **Export to OpenAir** [custcol_oa_export_to_openair] box is checked on the vendor credit line item in NetSuite.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

NetSuite fields declared and used in filters in the Purchase Item mapping group should also be declared in the PO filters.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Purchase Item mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

Review the following guidelines:

- NetSuite fields declared and used in advanced mappings in the Purchase Item mapping group should also be declared in the PO mapping group.

Other Records Created or Updated by the POs (Vendor Credits) Import Workflow

The POs (Vendor Credits) import workflow also creates purchase request, request item and fulfillment records. Mappings are not available for these on the OpenAir NetSuite Connector UI and cannot be modified.

Purchase Requests

See the OpenAir Help Center for detailed information about information added to the purchase request records created for each vendor credit line item imported into OpenAir. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Request Items

See the OpenAir Help Center for detailed information about information added to the request item records created for each vendor credit line item imported into OpenAir. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Fulfillments

See the OpenAir Help Center for detailed information about information added or updated to the fulfillment records created for each vendor credit line item imported into OpenAir if not fulfilled. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Custom Import Workflows

> **Note:** This help topic describes a new approach to custom import workflows introduced in the April 15, 2023 OpenAir release.
>
> The new approach is more flexible than custom imports defined on the OpenAir NetSuite Connector administration form and does not rely on NetSuite saved searches. See also Custom Imports.

You can add custom workflows to import data from NetSuite service items or custom record types into a supported record type in OpenAir using your own field mapping definitions. See Importing Records from NetSuite into OpenAir Using Custom Workflows.

Custom import workflows can be set to:

- Create new records in NetSuite first and then import them into OpenAir.
- Update records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding record in OpenAir.

Custom import workflow field mapping supports filters and advanced mappings. See Advanced Mappings for Custom Import Workflows.

You can add custom import workflows to workflow groups and include them in your scheduled integration runs, or when running the integration manually. For more information about workflow groups, see Creating and Managing Workflow Groups.

By default, a maximum of 4 custom export or import workflows can be active at any one time. To review your requirements, and increase the maximum number of active custom export or import workflows accordingly, contact OpenAir Customer Support.

You can review the setting details you selected or entered when creating each custom import workflow at any time. See Viewing Custom import Workflows Settings.

You can manage custom import workflows and delete those you no longer need. See Deleting Custom Import Workflows.

Enabling Custom Import Workflows

Custom import workflows require the Enable Direct TBA NetSuite Connector administration settings to be enabled for your account.

To enable custom import workflows:

1. Go to Administration > NetSuite Connector.
2. Click the Tips button, then **NetSuite Connector administration**.
3. Check the **Custom Import** box under the Imports section. See Custom Import.

Guidelines and Limitations

Review the following guidelines:

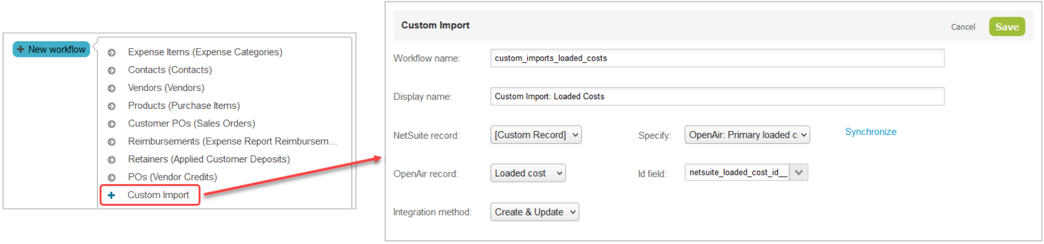
- Custom import workflows currently support the following OpenAir record types: Cost (Loaded cost), Cost center, Job code, Payroll type, Rate card, Service 1-5 line, Time type.
- Custom import workflows currently support the following NetSuite record types: Custom record types, Service item. For more information about creating custom record types in NetSuite, see Creating Custom Record Types (External link to NetSuite SuiteAnswers | Answer ID: 10140).
- Custom import workflows importing NetSuite custom record types OpenAir: Cost Center as Cost center records in OpenAir and OpenAir: Project Rate Card as Rate card records in OpenAir support the Real-Time Record Import feature. See Real-Time Record Import and Configuring Real-Time Record Import from NetSuite into OpenAir.
- Custom import workflow field mapping supports standard fields available in the NetSuite endpoint used for the integration and custom fields. Custom segments are not currently supported.
- A custom import workflow imports all records of the selected type from NetSuite each time the custom import workflow even if the NetSuite records were imported previously and were not modified since the last import. To limit the number of records imported, use filters. See Using Filters.

Importing Records from NetSuite into OpenAir Using Custom Workflows

You can create custom workflows to import data from a supported NetSuite record type, including custom record types, into a record type in OpenAir using your own field mapping definitions.

To import records from NetSuite into OpenAir using a custom import workflow:

1. Go to Administration > NetSuite Connector > Mappings.
2. Click **New workflow** then **Custom Import**. The Custom workflow popup window appears.


+ New workflow
Expense Items (Expense Categories)
Contacts (Contacts)
Vendors (Vendors)
Products (Purchase Items)
Customer POs (Sales Orders)
Reimbursements (Expense Report Reimbursem...
Retainers (Applied Customer Deposits)
POs (Vendor Credits)
+ Custom Import
Cancel
Save
Synchronize

Custom Import

Workflow name: custom_imports_loaded_costs

Display name: Custom Import: Loaded Costs

NetSuite record: [Custom Record]

Specify: OpenAir: Primary loaded c

OpenAir record: Loaded cost

Id field: netsuite_loaded_cost_id__

Integration method: Create & Update

3. Enter a **Workflow name** — the value must be up to 40 characters long and must contain only alphanumeric or underscore characters.
4. Enter a **Display name** up to 80 characters long.
5. Select the **NetSuite record** holding the data you want to import into the selected OpenAir record type. To import data from a custom record type in NetSuite, select "[Custom Record]", and select the NetSuite custom record type from the **Specify** dropdown.

   For more information about creating custom record types in NetSuite, see Creating Custom Record Types (External link to NetSuite SuiteAnswers | Answer ID: 10140).

   ✓ **Tip:** Click **Synchronize** to update the list of NetSuite custom record types without leaving the custom import workflow configuration form.

6. Select the **OpenAir record** type you want to import NetSuite data into.
7. Select the **ID field** for the OpenAir record — this is the custom field that will hold the internal ID of the matching NetSuite record. The **ID field** is required and cannot be changed after you save the custom import workflow configuration.

   Note: In most cases, you must create the custom field before you create the custom import workflow.

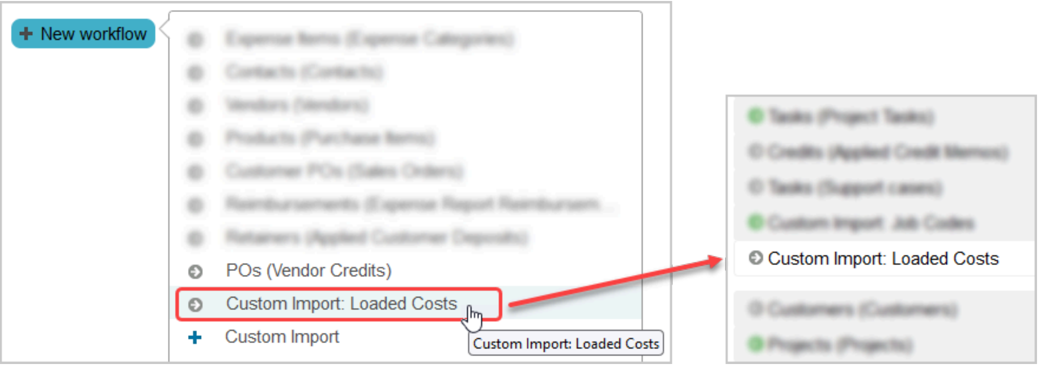
   It is not possible to create custom fields for loaded cost records in OpenAir. To enable custom import workflows for loaded cost, contact OpenAir Customer Support. Enabling the Loaded Cost Custom Import optional feature, creates the custom field `netsuite_loaded_cost_id` that will hold the internal ID of the NetSuite record. This custom field is not visible on the OpenAir UI.

   - To view the NetSuite internal ID for an imported loaded cost record, export the `loaded_cost` table using the Import/Export functionality in OpenAir. For more information about the Import/Export functionality, see Administrator Guide.
   - To set the NetSuite internal ID for an existing loaded cost record, you need to use the OpenAir XML API, OpenAir SOAP API, or OpenAir Integrations Manager. As with all import workflows, if you have existing loaded cost records in OpenAir, you should ensure that importing loaded cost information from NetSuite does not create duplicate entries in OpenAir. Setting the NetSuite internal ID for existing records is one way of doing so.

8. The only **Integration method** available for custom import workflows is **Create and Update**— The workflow lets you:

- Create new records in OpenAir first and then import them to NetSuite.
- Update records in OpenAir and use the NetSuite <> OpenAir integration to update the corresponding record in NetSuite.

9. Click **Save**.
10. Click **New workflow** then click the display name of the custom import workflow you created. The custom import workflow is added to the list.

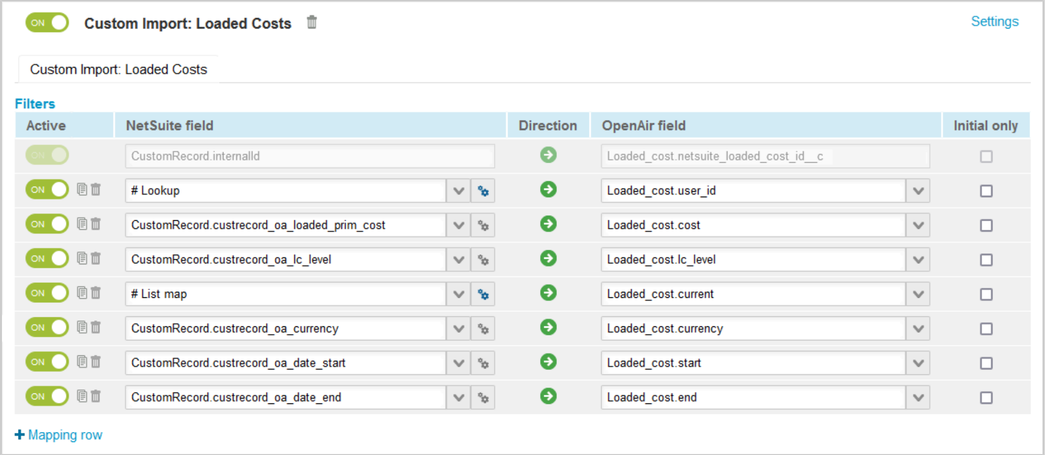

+ New workflow
POs (Vendor Credits)
Custom Import: Loaded Costs
+ Custom Import
Custom Import: Loaded Costs
Custom Import: Loaded Costs

11. Select the custom import workflow from the list of workflows to show the field mappings for this workflow. Custom import workflows are listed after built-in workflows.
12. Edit the field mappings as per your integration requirements.

    One row mapping the internal ID of the OpenAir record to the relevant custom field on the NetSuite record is shown by default. This required mapping row is protected and cannot be modified.

    You must specify at least one additional mapping row before you can save the workflow.

> **Note:** Review the following guideline:
>
> - Before you create a custom import workflow, make sure you add a custom field to the OpenAir record type to store the NetSuite internal IDs of the records imported using the custom workflow.
> - Consult NetSuite SOAP schema browser for reference when defining field mapping for custom import workflows. It includes information about any specific requirements for importing NetSuite data into OpenAir.



ON **Custom Import: Loaded Costs** Settings

Custom Import: Loaded Costs

Filters

Active	NetSuite field	Direction	OpenAir field	Initial only
ON	CustomRecord.internalId	➔	Loaded_cost.netsuite_loaded_cost_id__c	□
ON	# Lookup	➔	Loaded_cost.user_id	□
ON	CustomRecord.custrecord_oa_loaded_prim_cost	➔	Loaded_cost.cost	□
ON	CustomRecord.custrecord_oa_lc_level	➔	Loaded_cost.lc_level	□
ON	# List map	➔	Loaded_cost.current	□
ON	CustomRecord.custrecord_oa_currency	➔	Loaded_cost.currency	□
ON	CustomRecord.custrecord_oa_date_start	➔	Loaded_cost.start	□
ON	CustomRecord.custrecord_oa_date_end	➔	Loaded_cost.end	□

+ Mapping row

13. (Optional) Add a filter to limit the number of records imported when the workflow runs. See Using Filters.

    By default, the custom import workflow imports all records of the selected type from NetSuite each time the custom import workflow even if the NetSuite records were imported previously and were not modified since the last import.

14. Click **Save**.

Advanced Mappings for Custom Import Workflows

You can set up advanced mappings for custom import workflows. For more information about advanced mappings, see Using the Advanced Tag Notation and Advanced Mapping and Filter Notation Reference.

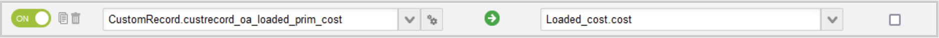
Note the following differences between advanced mappings for custom import workflows and for standard integration workflows:

- The following field declarations are not required: OA_FIELDS (OpenAir fields), OA_CUSTOM_FIELDS (OpenAir custom fields), NS_FIELDS (NetSuite fields), and NS_CUSTOM_FIELDS (NetSuite custom fields). You can use them but they will be ignored.

  For example:

  - **Value to value (from schema) mapping** — Advanced mapping is not required as you should be able to select the NetSuite field from the dropdown options on the Mapping tab. However, you could use the equivalent advanced mapping.

```
cost   custrecord_oa_loaded_prim_cost
```



CustomRecord.custrecord_oa_loaded_prim_cost

Loaded_cost.cost

- □ **Value to value (lookup) mapping** — You can specify the lookup in the same way as for standard workflows but the field declaration is not required.

  ```
  user_id lookup=custrecord_oa_user_id:lookup_table=user:lookup_by=netsuite_user_id:lookup_return=id
  ```

- □ **Value to value (list map)** — You can use the mapping table in the same way as for standard workflows but the field declaration is not required. In the following example, if the `custrecord_oa_current` value is `true` on the NetSuite record, `current` is set to 1 in OpenAir.

  ```
  <current custrecord_oa_current>
     'true' '1'
     'false' '0'
  </current>
  ```

- □ **Value to value (IF condition)** — You can use the IF condition in the same way as for standard workflows but the field declaration is not required. In the following example, if the `custrecord_oa_current` value is `true` on the NetSuite record, `current` is set to 1 in OpenAir.

  ```
  IF <<END
  IF (custrecord_oa_current='true') THEN current '1'
  IF (custrecord_oa_current='false') THEN current '0'
  END
  ```

- The following field declarations are not supported currently: `OA_FIELDS_INITIAL_ONLY` (OpenAir fields initial-only), `OA_CUSTOM_FIELDS_INITIAL_ONLY` (OpenAir custom fields initial-only), `OA_FIELDS_GROUP_BY` (OpenAir fields group by), and `OA_FIELDS_SORT_BY` (OpenAir fields group by).

Viewing Custom import Workflows Settings

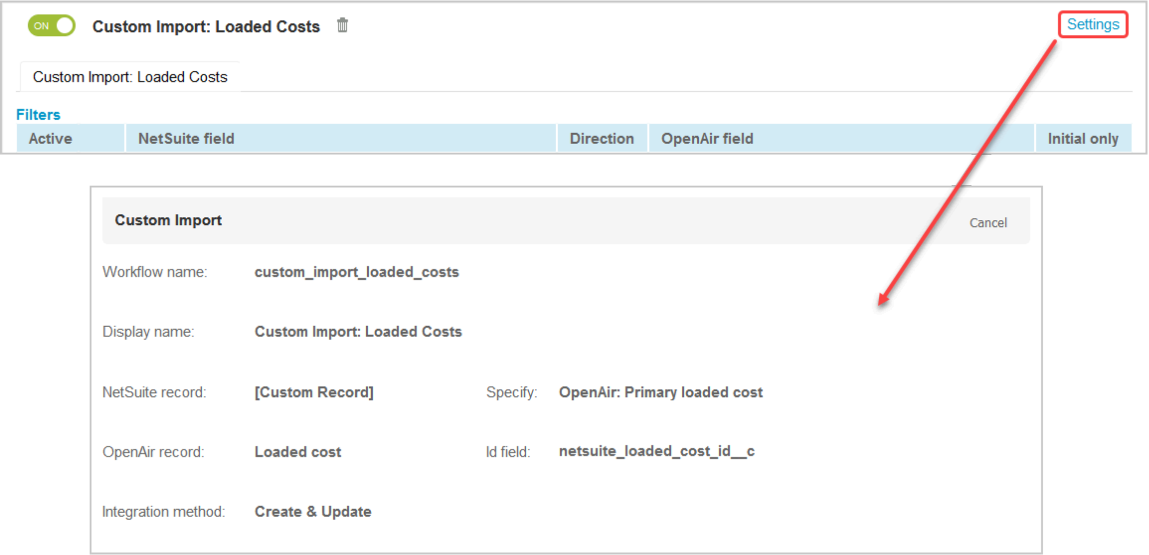
You can view the settings for your existing custom import workflows and get a reminder of the following details:

- Workflow name.
- NetSuite record type being imported.
- OpenAir record type holding the imported data.
- Integration method — The integration method is **Create & Update** for all custom import workflows.

The settings are read-only and cannot be edited.

To view the settings for an existing custom import workflow:

1. In OpenAir, go to Administration > NetSuite Connector > Mappings.
2. Select the custom import workflow from the list of workflows. The mapping definition table for this workflow appears in the right pane.
3. Click **Settings**. The custom workflow popup window appears showing the setting details for the workflow as read-only information.



Workflow name: custom_import_loaded_costs

Display name: Custom Import: Loaded Costs

NetSuite record: [Custom Record] Specify: OpenAir: Primary loaded cost

OpenAir record: Loaded cost Id field: netsuite_loaded_cost_id__c

Integration method: Create & Update

Deleting Custom Import Workflows

You can manage custom import workflows and delete those you no longer need.

To delete a custom workflow:

1. Go to Administration > NetSuite Connector > Mappings.
2. Select the custom workflow you want to delete.
3. Click the delete icon 🗑 . A confirmation dialog appears.
4. (Optional) To delete the custom workflow from and remove it permanently from the **New workflow** dropdown, check the **Permanently delete** box.
5. Click **Delete** to confirm. The custom workflow and associated field mapping are deleted.

Importing or Mapping NetSuite Classifications to OpenAir Hierarchies

You can configure the NetSuite <> OpenAir integration to import NetSuite classifications and create a hierarchy in OpenAir for each classification in NetSuite and for each record type. Imported hierarchies have the same levels and nodes as the NetSuite classifications.

- **NetSuite classifications** — Departments, classes, and locations are multilevel classifications that you can use to identify and categorize records in your NetSuite account. If you use NetSuite OneWorld, subsidiaries are the primary classification used to organize your NetSuite records, and departments, classes, locations, are additional classifications associated to a subsidiary. For more information about classifications, see Classifications Overview (external link to NetSuite SuiteAnswers | Answer ID: 9843).
    - For more information about NetSuite classifications and creating classification records, see Departments and Classes Overview (external link to NetSuite SuiteAnswers | Answer ID: 9844), Locations Overview (external link to NetSuite SuiteAnswers | Answer ID: 9845), and related topics.

- For more information about setting up subsidiaries in a NetSuite OneWorld account, see Subsidiary Setup (external link to NetSuite SuiteAnswers | Answer ID: 9866).

> **Note:** After you enable the relevant classifications for your account, you can create department, class, or location records and use them to categorise transactions and employees. Classifications are not available for customer and project records as standard in NetSuite. However, you can create custom entity fields to use classifications for customers and projects using a list/record field, if required. See Using Classifications for Customers and Projects in NetSuite.

- **OpenAir hierarchies** — You can use hierarchies are used to create multilevel classification trees and categorize records in your OpenAir account. Each hierarchy is associated to one of the following record types: employees, customers, or projects. Multiple hierarchies can apply to the same record type. For more information about hierarchies, see Administrator Guide.

You can also map NetSuite classifications to existing OpenAir hierarchies. You can edit existing hierarchy nodes in OpenAir and enter the internal ID of the corresponding classification record in NetSuite. This lets you retain any existing hierarchies you are already using in OpenAir.

> ✓ **Tip:** Update the integration bundles installed on your NetSuite account to the latest available version to take advantage of new OpenAir NetSuite Connector features that are introduced from time to time. A custom timestamp is added on department, class, location and subsidiary records when you install NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) 2.05 or later version. OpenAir NetSuite Connector uses the custom timestamp to import only the classifications created or updated since the last integration run instead of importing all classifications with each integration run. This significantly reduces the run time of any integration workflows importing large numbers of classifications from NetSuite.
>
> If you have an earlier version of the bundle installed, to take advantage of this performance optimization:
>
> 1. Update the NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) to the latest version.
> 2. In OpenAir, go to Administration > NetSuite Connector and click the Tips button then **Synchronize NetSuite custom fields**.
> 3. Run the integration for all integration workflows using map NetSuite classification to OpenAir hierarchy options. The first integration run will import all classifications from NetSuite. See Running the Bulk Import and Export Integration on Demand.

To import or map NetSuite classifications to OpenAir hierarchies for a record type:

1. In OpenAir, go to Administration > NetSuite Connector > Mapping.
2. Click the relevant import workflow in the list of integration workflows.
   - Click **Employees (Employees)** if you want to import NetSuite classifications applied to employees.
   - Click **Customers (Clients)** if you want to import NetSuite classifications applied to customers.
   - Click **Projects (Projects)** if you want to import NetSuite classifications applied to projects.

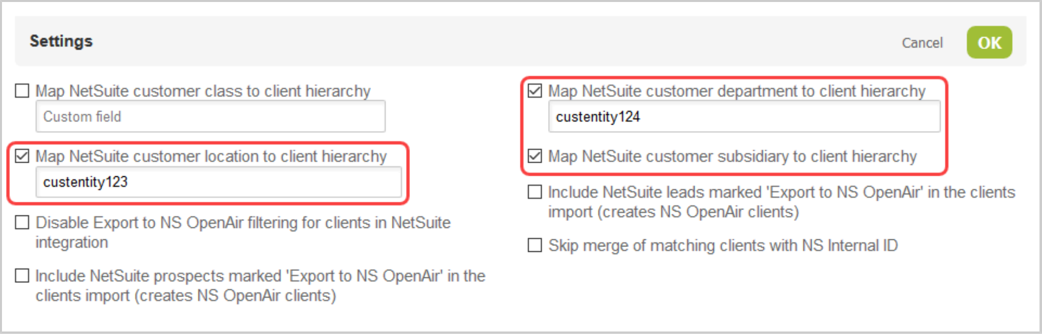
The field mapping definition table for the selected workflow appears.

3. Click **Workflow settings**.

   The Workflow settings popup form appears.

4. Check the relevant boxes for the classifications you want to import. The label for each settings shows the classification, NetSuite record type and the OpenAir record type it refers to: **Map NetSuite <NetSuite record type> <classification> to <OpenAir record type> hierarchy**.

   If you want to import the NetSuite departments, classes and locations for customer and project records, enter the **ID** of the custom entity field you created to hold the relevant classification information. If the ID was generated automatically by NetSuite, it will typically be formatted as follows: `custentity`*<number>*. See Using Classifications for Customers and Projects in NetSuite.



Settings

☐ Map NetSuite customer class to client hierarchy

Custom field

☑ Map NetSuite customer location to client hierarchy

custentity123

☐ Disable Export to NS OpenAir filtering for clients in NetSuite integration

☐ Include NetSuite prospects marked 'Export to NS OpenAir' in the clients import (creates NS OpenAir clients)

☑ Map NetSuite customer department to client hierarchy

custentity124

☑ Map NetSuite customer subsidiary to client hierarchy

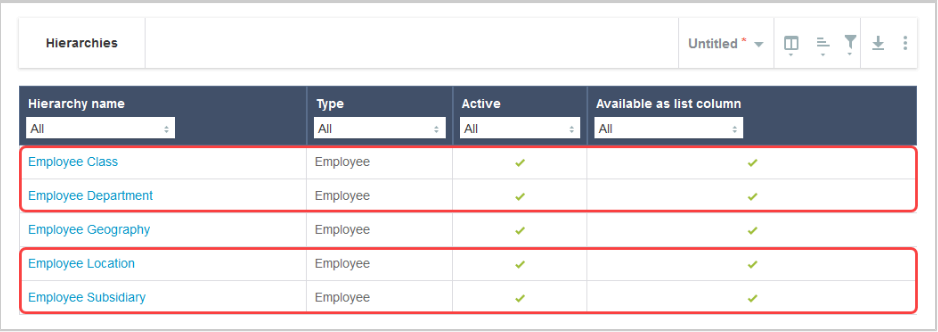
☐ Include NetSuite leads marked 'Export to NS OpenAir' in the clients import (creates NS OpenAir clients)

☐ Skip merge of matching clients with NS Internal ID

5. Click **OK** then **Save**.

   OpenAir NetSuite Connector creates the following records in OpenAir when you save the integration configuration:

   - Hierarchies corresponding to every NetSuite classifications to be imported. The default name for each hierarchy includes the OpenAir record type and the NetSuite classification: **<OpenAir record type> <classification>**. The entity form for the OpenAir record type will show dropdown fields for each hierarchy with nodes — Edit the hierarchy records or use form permissions to hide the NetSuite classification hierarchy dropdown fields on entity forms, if required. You can also set these fields as required, and choose to display them as columns on list views.



Hierarchies

Hierarchy name	Type	Active	Available as list column
All	All	All	All
Employee Class	Employee	✓	✓
Employee Department	Employee	✓	✓
Employee Geography	Employee	✓	✓
Employee Location	Employee	✓	✓
Employee Subsidiary	Employee	✓	✓

   - Custom fields for hierarchy records. These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.
     - **NetSuite class hierarchy** [`netsuite_hierarchy_class`] — Check box used to identify if the hierarchy maps to the NetSuite class.
     - **NetSuite department hierarchy** [`netsuite_hierarchy_department`] — Check box used to identify if the hierarchy maps to the NetSuite department.
     - **NetSuite location hierarchy** [`netsuite_hierarchy_location`] — Check box used to identify if the hierarchy maps to the NetSuite location.

 - **NetSuite subsidiary hierarchy** `[netsuite_hierarchy_subsidiary]` — Check box used to identify if the hierarchy maps to the NetSuite subsidiary.

- Custom fields for hierarchy level and node records . These check box custom fields are hidden by default — Edit the custom field records to show them on the hierarchy form, if required.
  - **NetSuite ID** `[netsuite_hierarchy_node_id]` — Text custom field used to map the level or node to the corresponding NetSuite classification or subsidiary depth level or record.
  - **NetSuite class level or hierarchy node** `[netsuite_hierarchy_node_class]` — Check box used to identify if the level or node maps to a NetSuite class depth level or record.
  - **NetSuite department level or hierarchy node** `[netsuite_hierarchy_node_department]` — Check box used to identify if the level or node maps to a NetSuite department depth level or record.
  - **NetSuite location level or hierarchy node** `[netsuite_hierarchy_node_location]` — Check box used to identify if the level or node maps to a NetSuite location depth level or record.
  - **NetSuite subsidiary level or hierarchy node** `[netsuite_hierarchy_node_subsidiary]` — Check box used to identify if the level or node maps to a NetSuite subsidiary depth level or record.

6. (Optional) Map NetSuite classifications to existing OpenAir hierarchies.

   > **Note:** Repeat steps c-e for each NetSuite classifications you want to map to an existing OpenAir hierarchy.

   a. Edit each of the custom fields listed in the previous step, clear the **Hide on data entry forms** box, and click **Save**.

   b. Delete the hierarchy created by OpenAir NetSuite Connector corresponding to the NetSuite classification.

   c. Edit the existing OpenAir hierarchy you want to map to the NetSuite classification. Check the relevant **NetSuite <classification> hierarchy** box.

   d. Edit each hierarchy level. Do the following:

      i. Check the relevant **NetSuite <classification> level or hierarchy node** box.

      ii. Enter the hierarchy level depth in the **NetSuite ID** text box. For example, if this is the top ancestor level, enter 1, or if it is the level immediately below the top ancestor, enter 2.

   e. Edit each hierarchy node. Do the following:

      i. Check the relevant **NetSuite <classification> level or hierarchy node** box.

      ii. Enter the **NetSuite ID** for the corresponding classification record.

   > **Important:** Review the following guidelines:
   >
   > - Make sure that all hierarchy nodes follow the same parent and child relationships as the NetSuite classification records.
   >
   >   To import child classification nodes to OpenAir, the parent and all ancestor classification nodes must be active in NetSuite. If the parent node or any of the ancestor nodes are inactive, the child nodes will not be imported to OpenAir.

7. Run the integration for the relevant import workflow. The integration run will import classification records from NetSuite into OpenAir as hierarchy nodes. See Running the Bulk Import and Export Integration on Demand

Using Classifications for Customers and Projects in NetSuite

Department, Class and Location classifications are not available for customer and project records as standard in NetSuite. However, you can create custom entity fields to use these classifications for customers and projects using a list/record field, if required.

> **Note:** If you use NetSuite OneWorld, subsidiary classification is available for employees, customers, and projects by default. You must assign a subsidiary to each entity record before you can enter transactions for the entity.

To use classifications for customers and projects in NetSuite:

1. In NetSuite, go to Setup > Customization > Lists, Records, & Fields > Entity Fields.
2. Enter a **Label**.
3. Select `List/Record` from the **Type** dropdown list.
4. Select the **List/Record**.
   - To replicate an classification and apply it to customers or projects, select `Class`, `Department`, or `Location`.
   - To create a unique list of values, select `New`. The Custom List window appears. See Creating a Custom List (external link to NetSuite SuiteAnswers | Answer ID: 10069).
5. Check the **Store value** and **Show in list** boxes, to make a custom list available for other custom entity fields, or to show the field automatically in the list of records.
6. On the Applies to subtab, check the boxes for the record forms to which you want to add the custom entity field, then click **Apply to Forms**.
7. Click **Save**.
8. Make a note of the entity field **ID**. You will need it to configure OpenAir NetSuite Connector to import the NetSuite classification to an OpenAir hierarchy integration in OpenAir. See Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

Exporting Data From OpenAir to NetSuite

Note: Some of the help topics referenced below are placeholders. Content will be added gradually to these pages as the content and structure of this guide are reviewed. For more information about a specific integration workflow, please refer to the following chapters:

- Configuring Optional Features
- Using the Integration
- Data Mapping in the Integration

This section provides reference information for each integration workflows exporting information from a specific OpenAir record type into a specific NetSuite record type.

- Customers (Customers) Export Workflow
- Projects (Projects) Export Workflow
- Purchase Requests (Purchase Requisitions) Export Workflow
- Expense Reports (Expense Reports) Export Workflow
- Invoices (Invoices) Export Workflow
- Time Entries (Time) Export Workflow
- Recognition Transactions (Revenue Recognition Transactions) Export Workflow
- Loaded Cost (Custom Record) Export Workflow
- Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Export Workflow
- Recognition Rules (Custom Record: OpenAir: Revenue Rules for ARM) Export Workflow
- Recognition Transactions (Custom Record: OpenAir: Revenue Transactions for ARM) Export Workflow
- Update NetSuite Projects with OpenAir Project Data Export Workflow
- Update NetSuite Tasks with OpenAir Task Data
- WBS (Opportunity) Export Workflow
- Custom Export Workflows

Customers (Customers) Export Workflow

The Customers (Customers) export workflow lets you create new customer records in OpenAir first and then export them to NetSuite.

NetSuite is the primary application for customer records. Changes made to customer records in OpenAir after the initial export are not exported to NetSuite. The Customers (Customers) export workflow can be used to create new customer records in NetSuite but not to update already integrated customer records.

Note: You can use the Customers (Customers) import workflow to:

- Create new customer records in NetSuite first and then import them into OpenAir. See Employees (Employees) Import Workflow.
- Update customer records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding customer records in OpenAir.

For more information, see Customers (Customers) Import Workflow.

If you use the deprecated 3–way Salesforce, OpenAir and NetSuite integration to export customer records from OpenAir to NetSuite, you should use the Customers (Customers) export workflow instead. See Replacing the Deprecated 3-Way Salesforce, OpenAir and NetSuite Integration With the Customers (Customers) or Projects (Projects) Export Workflows.

For information about workflow specific settings, see Customers (Customers) Export Workflow Settings.

For information about field mapping definitions for this workflow, see Customers (Customers) Export Field Mapping Definition.

Requirements and Downstream Dependencies

Matching customer records must exist in both NetSuite and OpenAir before you can:

- Import projects associated with these customers from NetSuite into OpenAir.
- Export projects from OpenAir to NetSuite.
- Import contacts associated with these customers or these customers' projects from NetSuite into OpenAir.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
✓	✓	✓	✓	✓	

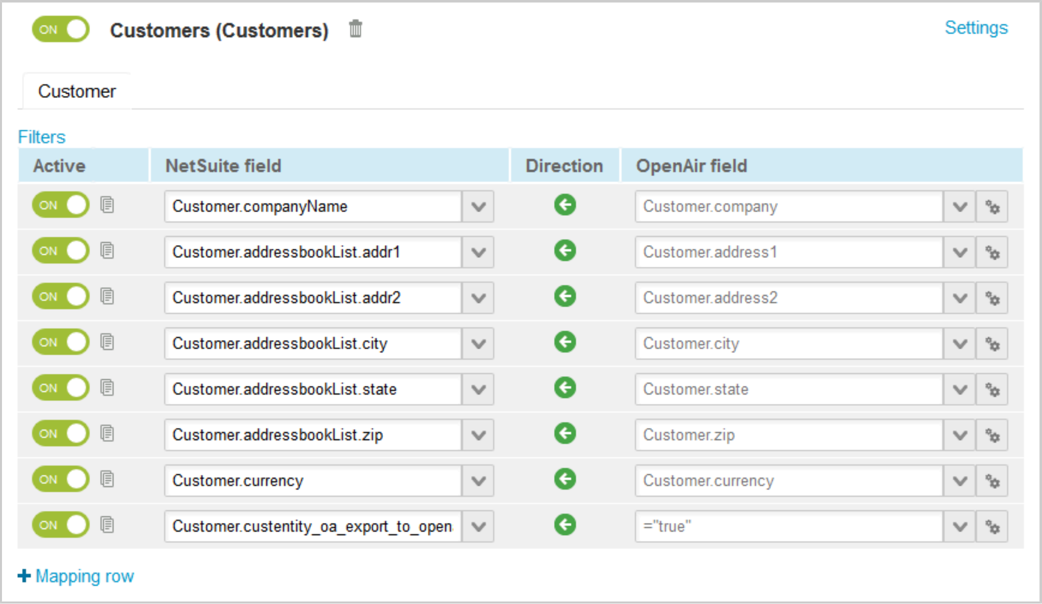
- Run the integration manually, use a scheduled integration run to create or update the OpenAir customer record after you create or update the customer record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create or update the OpenAir customer record when the Customer form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir customer record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Customers (Customers) Export Workflow

Use the following steps to set up the Customers (Customers) export workflow.

To set up the Customers (Customers) export workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector, click the Tips menu button, then NetSuite Connector administration form.
2. Check the **Customers (Customers)** box under the Exports form section.
3. Go to Administration > NetSuite Connector > Mapping.
4. Click **New workflow** then **Customers (Customers)**.
5. Click **Customers (Customers)** in the list of integration workflows.
6. Click **Workflow settings** to view and modify the settings specific to this workflow. For more information about the workflow settings, see Customers (Customers) Export Workflow Settings.
7. Modify the field mapping definition table for the Customers (Customers) export workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Customers (Customers) Export Field Mapping Definition.
8. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON **Customers (Customers)** Settings

Customer

Filters

Active	NetSuite field	Direction	OpenAir field
ON	Customer.companyName	←	Customer.company
ON	Customer.addressbookList.addr1	←	Customer.address1
ON	Customer.addressbookList.addr2	←	Customer.address2
ON	Customer.addressbookList.city	←	Customer.city
ON	Customer.addressbookList.state	←	Customer.state
ON	Customer.addressbookList.zip	←	Customer.zip
ON	Customer.currency	←	Customer.currency
ON	Customer.custentity_oa_export_to_open	←	="true"

+ Mapping row

9. Click **Save**.
10. (Optional) Add the Customers (Customers) export workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Customers (Customers) Export Workflow Settings

You can control Customers (Customers) export workflow settings from the NetSuite Connector UI. On the NetSuite Connector Mapping screen, select the Customers(Customers) export workflow and click **Settings** to view the workflow settings form.

The following settings are available for the Customers(Customers) export workflow:

- Map country when exporting customers to NetSuite
- Map country when exporting customers to NetSuite in 3-way Salesforce.com, OpenAir and NetSuite integration — DEPRECATED

Map country when exporting customers to NetSuite

When checked, the **Country** field on the customer record in OpenAir exports to the **Country** field on the customer record in NetSuite. The field mapping is disabled otherwise and it is not possible to map the **Country** field manually.

Map country when exporting customers to NetSuite in 3-way Salesforce.com, OpenAir and NetSuite integration — DEPRECATED

When checked, the **Country** field on the customer record in OpenAir exports to the **Country** field on the customer record in NetSuite if you use the deprecated 3-way Salesforce.com, OpenAir and NetSuite integration. The field mapping is disabled otherwise and it is not possible to map the **Country** field manually.

> ⚠ **Important:** The 3-way Salesforce, OpenAir and NetSuite integration for customers and projects is deprecated and will be removed at a future date to be announced.
>
> You should use the Projects (Projects) and Customers (Customers) export workflows instead. See Replacing the Deprecated 3-Way Salesforce, OpenAir and NetSuite Integration With the Customers (Customers) or Projects (Projects) Export Workflows.
>
> The **Enable 3-way Salesforce.com, OpenAir and NetSuite integration** setting has been moved to the Legacy compatibility section of the NetSuite Connector administration form. You may continue using the 3-way Salesforce, OpenAir and NetSuite integration if it is already enabled for your account. However, it can no longer be enabled.

Customers (Customers) Export Field Mapping Definition

The Customers (Customers) export workflow includes one mapping group:

- Customer [Integration point: `OA_CUSTOMER_TO_NS_CUSTOMER`] — See Customer Mapping Group.

Customer Mapping Group

Use the Customer mapping group to map information on information on customer records in OpenAir with customer records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information exported by default includes company name, address and currency. Filters and pullbacks are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group. The mapping group does not support custom pullback field mappings, retrieving information from the record created in NetSuite back into OpenAir.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Customer [Customer] Schema: Customer (2020.1)|Customer (2023.1) XSD: listRel (2020.1)|listRel (2023.1)	Customer [customer] Data Dictionary: customer	Customer [OA_CUSTOMER_TO_NS_CUSTOMER]

Note: The Customer SOAP object is used for customer, lead, and prospect records. This is determined by the property Customer.entityStatus with the value customer, lead, or prospect.

Default and Custom Filters

By default, customer records are exported from OpenAir to NetSuite if the following conditions are met:

- The **Active** box is checked on the customer record in OpenAir.
- The customer record was not already exported — Customer.exported is empty.
- The **NetSuite Customer ID** [netsuite_customer_id] custom field is empty OR the **Allow objects with a NetSuite ID to be exported during integration run** box is checked on the OpenAir NetSuite Connector administration form. See Allow objects with a NetSuite ID to be exported during integration run.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Customer mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Projects (Projects) Export Workflow

The Projects (Projects) export workflow lets you create new project records in OpenAir first and then export them to NetSuite.

You can use the Update NetSuite Projects with OpenAir Project Data export workflow to update project records in OpenAir and run the integration to update the project records in NetSuite. See Update NetSuite Projects with OpenAir Project Data Export Workflow.

> **Note:** You can also use the Projects (Projects) import workflow to:
>
> - Create new project records in NetSuite first and then import them into OpenAir. See Projects (Projects) Export Workflow.
> - Update project records in NetSuite and use the NetSuite <> OpenAir integration to update the corresponding project record in OpenAir.
> - Create project templates in OpenAir, select an **OpenAir Project Template** when creating a new project in NetSuite. When you import the project into OpenAir, the OpenAir project contains project-related information copied from the template. The information copied from the project template can include tasks, assignments, billing rules, loaded costs, revenue recognition rules, auto-bill settings, project approvers, and more.
>
> For more information, see Projects (Projects) Import Workflow

For information about field mapping definitions for this workflow, see Projects (Projects) Export Field Mapping Definition.

Requirements and Downstream Dependencies

Matching customer records must exist in both NetSuite and OpenAir before you can export projects from OpenAir to NetSuite.

Matching project records must exist in both NetSuite and OpenAir before you can export expense reports and timesheets associated with these projects, use purchase orders, or use time and expenses in your billing rules and generate invoices.

A project must exist in NetSuite before a contact or sales order associated with the project can be imported into OpenAir.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
✓	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create or update the OpenAir project record after you create or update the project record in NetSuite. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.

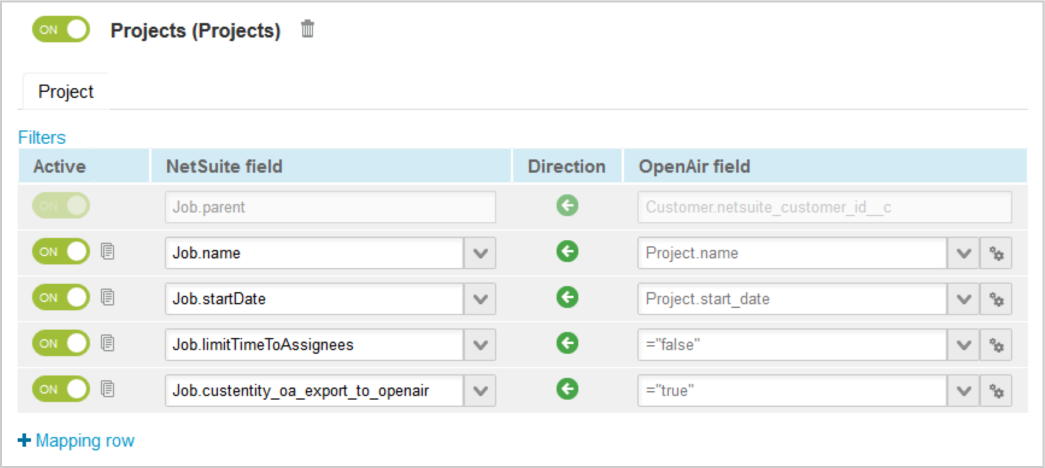
- Setup the real-time integration to create or update the OpenAir project record when the Project form is saved in NetSuite. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir project record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Projects (Projects) Export Workflow

Use the following steps to set up the Projects (Projects) export workflow.

To set up the Projects (Projects) export workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector, click the Tips menu button, then NetSuite Connector administration form.
2. Check the **Projects (Projects)** box under the Exports form section.
3. Go to Administration > NetSuite Connector > Mapping.
4. Click **New workflow** then **Projects (Projects)**.
5. Click **Projects (Projects)** in the list of integration workflows.
6. Modify the field mapping definition table for the Projects (Projects) export workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Projects (Projects) Export Field Mapping Definition.
7. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON **Projects (Projects)**

Project

Filters

Active	NetSuite field	Direction	OpenAir field
ON	Job.parent	←	Customer.netsuite_customer_id__c
ON	Job.name	←	Project.name
ON	Job.startDate	←	Project.start_date
ON	Job.limitTimeToAssignees	←	="false"
ON	Job.custentity_oa_export_to_openair	←	="true"

8. Click **Save**.
9. (Optional) Add the Projects (Projects) export workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Projects (Projects) Export Field Mapping Definition

The Projects (Projects) export workflow includes one mapping group:

- Project [Integration point: PROJECT_TO_JOB] — See Project Mapping Group Tab.

Project Mapping Group Tab

Use the Project mapping group to map information on project records in OpenAir with information on project records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information exported by default includes project name, associated customer, and start date. Filters and pullbacks are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group. The mapping group does not support custom pullback field mappings, retrieving information from the record created in NetSuite back into OpenAir.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Project [Job] Schema: Job (2020.1)|Job (2023.1) XSD: listRel (2020.1)|listRel (2023.1)	Project [project] Data Dictionary: project	Project [PROJECT_TO_JOB]

Default and Custom Filters

By default, project records are exported from OpenAir to NetSuite if the following conditions are met:

- The **Active** box is checked on the project record in OpenAir.
- The project record was not already exported — `Project.exported` is empty.
- The **NetSuite Project ID** [`netsuite_project_id`] custom field is empty OR the **Allow objects with a NetSuite ID to be exported during integration run** box is checked on the OpenAir NetSuite Connector administration form. See Allow objects with a NetSuite ID to be exported during integration run.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Project mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

⚠ **Important:** Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Purchase Requests (Purchase Requisitions) Export Workflow

ⓘ **Note:** This help topic is a placeholder only. Content will be added gradually to these pages as the content and structure of this guide are reviewed. For more information about the Purchase Requests (Purchase Requisitions) integration workflows, please refer to the following chapters:

- Using the Integration
- Data Mapping in the Integration

Expense Reports (Expense Reports) Export Workflow

The Expense Reports (Expense Reports) export workflow lets you create new expense reports and follow the approval in OpenAir, and then export approved expense reports to NetSuite. OpenAir expense reports are exported to NetSuite as:

- Expense report transaction records (Transactions > Employees > Enter Expense Reports > List), if the employee record associated with the expense report in OpenAir was imported using the Employees (Employees) import workflow and if there is matching employee record in NetSuite. See Employees (Employees) Import Workflow.
- Vendor bill transaction records (Transactions > Payables > Enter Bills > List), if the employee record associated with the expense report in OpenAir was imported using the Employees (Vendors) import workflow and if there is matching vendor record in NetSuite. The vendor bill can be associated with an individual , a parent vendor or a subcontracting company. See Employees (Vendors) Import Workflow.

All receipts in the OpenAir expense report are exported to NetSuite as expense lines on the expense report (Expenses subtab) or vendor bill (Expenses and Items subtab). Expense line items cannot be grouped when exporting expense reports from OpenAir to NetSuite.

You can include tax breakouts on the expense reports and vendor bills exported from OpenAir to NetSuite. The tax breakout is based on the tax locations specified for each receipt or expense item. See Enable tax export on expense reports.

⚠ **Important:** The Expense Reports (Expense Reports) export workflow is not fully compatible with the SuiteTax feature:

- Tax amounts are calculated correctly when exporting OpenAir expense reports to NetSuite as vendor bills.
- When exporting OpenAir expense reports to NetSuite as expense reports, however, tax details need to be entered manually in NetSuite when using the SuiteTax engine. See Entering Expense Reports with Tax Impact in SuiteTax (external link to NetSuite SuiteAnswers | Answer ID: 65664)

Note that SuiteTax compatibility testing is currently in progress. You should test the NetSuite <> OpenAir integration with the SuiteTax feature on sandbox accounts before you enable the feature in your NetSuite production account. See Known Limitations of SuiteTax (external link to NetSuite SuiteAnswers | Answer ID: 98937). See also Legacy Tax or SuiteTax.

⚠ **Important:** Changes made to expense reports and receipts in OpenAir after the initial export are not exported to NetSuite. The Expense Reports (Expense Reports) export workflow can be used to create new expense reports or vendor bills in NetSuite but not to update already integrated expense report or vendor bills.

For information about workflow specific settings, see Expense Reports (Expense Reports) Export Workflow Settings.

For information about field mapping definitions for this workflow, see Expense Reports (Expense Reports) Export Field Mapping Definition.

Requirements and Downstream Dependencies

Matching employee, project and expense item (category) records must exist in both NetSuite and OpenAir before you can export expense reports from OpenAir to NetSuite as expense reports.

To export expense reports from OpenAir to NetSuite as vendor bills, you must import NetSuite vendors into OpenAir as employees first. See Employees (Vendors) Import Workflow.

Note: You can use the Expense Reports (Expense Reports) export workflow and either the POs (Vendor Bills) or POs (Purchase Orders) import workflow in conjunction to export expense reports from OpenAir to NetSuite as vendor bills and to import vendor bills or purchase orders from NetSuite into OpenAir as purchase orders. See POs (Purchase Orders) Import Workflow and POs (Vendor Bills) Import Workflow.

Importing purchase orders or vendor bills from NetSuite into OpenAir as purchase orders requires matching vendor records in both application. To use both workflows in conjunction, enable the Vendors (Vendors) import workflow and use the **OpenAir: User or Vendor** [custentity_oa_user_or_vendor] field on the vendor record in NetSuite to determine whether the vendor should be imported into OpenAir as a vendor or as an employee (user). See Vendors (Vendors) Import Workflow.

Matching OpenAir employee records and NetSuite vendor records as well as matching project and expense item (category) records must exist before you can export expense reports from OpenAir to NetSuite as vendor bills.

Important: If you use NetSuite OneWorld, the vendor must have the same NetSuite subsidiary as the project.

After you export expense reports from OpenAir to NetSuite as expense reports or vendor bills, you can:

- Reimburse expenses to employees and pay vendor bills, then use the Reimbursements (Expense Reports Reimbursements) import workflow to import reimbursement information into OpenAir. See Reimbursements (Expense Report Reimbursements) Import Workflow.
- Bill customers for expenses in your invoices. When you export invoices from OpenAir to NetSuite, billable expenses from OpenAir expense reports exported to NetSuite as expense reports or vendor bills are listed in the Billable Expenses subtab on the invoice transaction record in NetSuite. See Invoices (Invoices) Export Workflow.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
✓	✓ (Expense Report and Vendor Bill mapping groups)	✓	✓ (On Approval)	✓	

- Run the integration manually, use a scheduled integration run to create the NetSuite expense report or vendor bill transaction record after the expense report is approved in OpenAir. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create the expense report or vendor bill transaction record when the Expense Report is approved in NetSuite. See Configuring Real-Time Record Export from OpenAir to NetSuite.
- View and edit the OpenAir expense report record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Expense Reports (Expense Reports) Export Workflow

The Expense Reports (Expense Reports) workflow is available as standard.

To export of expense reports

To set up the Expense Reports (Expense Reports) export workflow in NetSuite

1. In NetSuite, go to Setup > Accounting > Accounting Preferences, and click the Time & Expenses subtab.
2. Clear the **Combine Detail Items on Expense Reports** box under the Expenses section.

   > ⚠ **Important:** Expense line items cannot be grouped when exporting expense reports from OpenAir to NetSuite. When the **Combine Detail Items on Expense Reports** is checked, the Expense Reports (Expense Reports) export workflow results in an integration error.

3. (Optional) Enable the export of OpenAir expense reports to NetSuite as vendor bills. To do so:
   a. In NetSuite, go to Setup > Company > General Preferences, and click the Custom Preferences subtab.
   b. Check the **Expense Report — Vendor Bill Integration** box under the NetSuite OpenAir SRP Integration section.
   c. Click **Save**.

   > ⚠ **Important:** Make sure the export of OpenAir expense reports to NetSuite as vendor bills is also enabled in OpenAir

4. (Optional) Configure NetSuite to use the OpenAir **Tracking number** instead of the auto-generated record number for expense reports in NetSuite. To do so:
   a. In NetSuite, go to Setup > Company >Auto-Generated Numbers, and click the Document Numbers subtab.
   b. Check the box under the **Allow Override** column for Expense Report. When checked, exported expense reports have the same number in both NetSuite and OpenAir.
   c. Click **Save**.

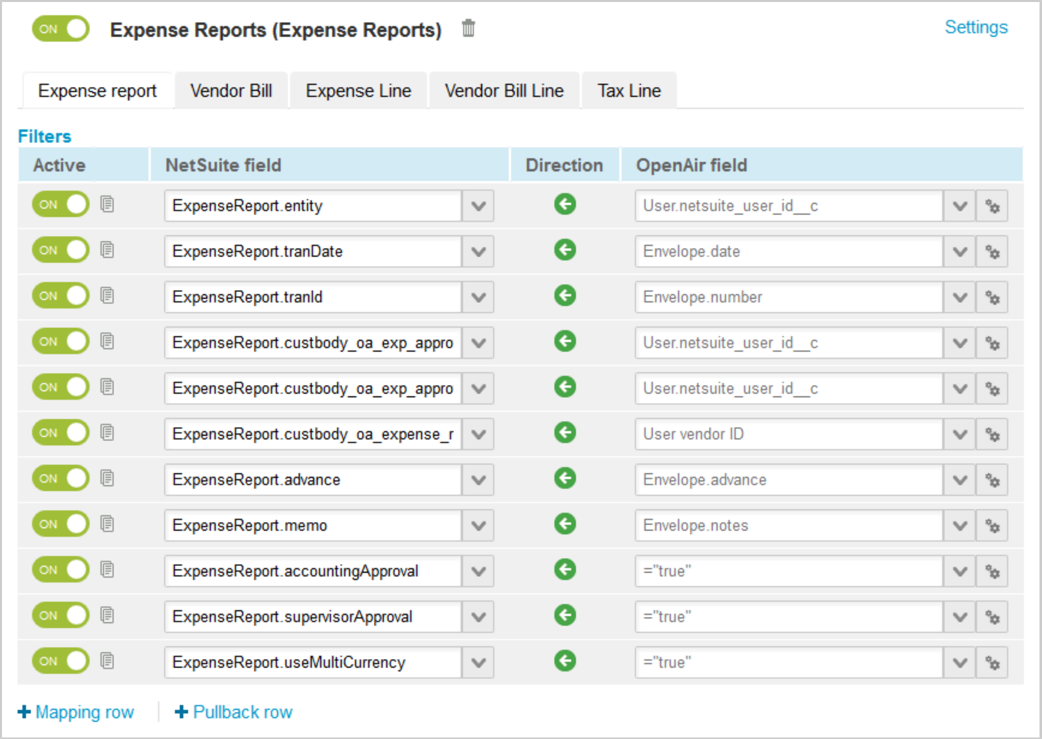
To set up the Expense Reports (Expense Reports) export workflow in OpenAir

1. (Optional) Enable the export of OpenAir expense reports to NetSuite as vendor bills. To do so:
   a. In OpenAir, go to the OpenAir NetSuite Connector administration form.
   b. Check the Employees (Vendors) box under the Imports section of the form. See Employees (Vendors).
   c. Check the **Enable OpenAir expense reports to NetSuite vendor bill integration** box.

   > ⚠ **Important:** Make sure the export of OpenAir expense reports to NetSuite as vendor bills is also enabled in NetSuite

2. In OpenAir, go to Administration > NetSuite Connector > Mapping.

3. Click **New workflow** then **Expense Reports (Expense Reports)**.
4. Click **Expense Reports (Expense Reports)** in the list of integration workflows.
5. Click **Workflow settings** to view and modify the settings specific to this workflow. For more information about the workflow settings, see Expense Reports (Expense Reports) Export Workflow Settings.
6. Modify the field mapping definition tables for the Expense Reports (Expense Reports) export workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Expense Reports (Expense Reports) Export Field Mapping Definition.
7. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON Expense Reports (Expense Reports) — Settings

Filters

Active	NetSuite field	Direction	OpenAir field
ON	ExpenseReport.entity	←	User.netsuite_user_id__c
ON	ExpenseReport.tranDate	←	Envelope.date
ON	ExpenseReport.tranId	←	Envelope.number
ON	ExpenseReport.custbody_oa_exp_appro	←	User.netsuite_user_id__c
ON	ExpenseReport.custbody_oa_exp_appro	←	User.netsuite_user_id__c
ON	ExpenseReport.custbody_oa_expense_r	←	User vendor ID
ON	ExpenseReport.advance	←	Envelope.advance
ON	ExpenseReport.memo	←	Envelope.notes
ON	ExpenseReport.accountingApproval	←	="true"
ON	ExpenseReport.supervisorApproval	←	="true"
ON	ExpenseReport.useMultiCurrency	←	="true"

8. Click **Save**.
9. (Optional) Add the Expense Reports (Expense Reports) export workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Expense Reports (Expense Reports) Export Workflow Settings

You can control Expense Reports (Expense Reports) export workflow settings from the NetSuite Connector UI. On the OpenAir NetSuite Connector Mapping screen, select the Expense Reports (Expense Reports) export workflow and click **Settings** to view the workflow settings form.

The following settings are available for the Expense Reports (Expense Reports) export workflow:

- Enable tax export on expense reports
- Allow manual override of Canadian taxes on Expense report export
- Automatically create negative Expense report lines in NetSuite for non-reimbursable expenses

- Do not set the foreign currency in NetSuite for foreign currency receipts when exporting Expense reports
- Keep customer reference on negative Expense report lines in NetSuite for non-reimbursable expenses
- Show last NetSuite integration error on expense reports

Enable tax export on expense reports

When checked, OpenAir expense reports exported to NetSuite as expense reports or vendor bills include tax breakouts based on the tax locations specified for each receipt or expense item, or (in some cases) on the tax amount values entered manually on the receipt. If your OpenAir account is configured to use tax locations on receipts or expense items, OpenAir calculates and displays the tax breakout for each receipt based on the tax rates for the specified tax location. OpenAir also lets you enter tax amounts manually to override the calculated values. If tax location records in OpenAir are matched with tax code or tax group records in NetSuite, the tax breakout for each expense or vendor bill line in NetSuite is calculated based on the NetSuite tax code or tax group corresponding to the tax location specified on the receipt in OpenAir.

Best Practice Guidelines

Review the following guidelines:

- Additional setup is required before you can export expense report tax information to NetSuite when exporting expense reports. See Setting Up Expense Report Tax Export.
- The tax location used on OpenAir receipts must have a corresponding tax code or tax group in NetSuite. The **NetSuite Tax Code ID** [`netsuite_tax_rate_id`] on the tax location entity form holds the internal ID of the corresponding tax code or tax group record in NetSuite.
- In OpenAir, employees enter the gross amount on receipt - the full amount including tax of the expense to be reimbursed. When the expense report is exported to NetSuite and the expense report tax export is enabled, the tax is subtracted from the expense line and NetSuite recalculates the tax based on the applicable tax code or tax group.

  > **Note:** This may occasionally result in discrepancies due to rounding precision differences in OpenAir and NetSuite. In OpenAir, tax amounts are rounded to 4 decimal places. In NetSuite, tax amounts are rounded to the currency precision (2 decimal placed for Canadian Dollars, for example).
  >
  > You can avoid discrepancy by entering tax amount manually to the currency precision in OpenAir. In the case of Canadian tax amounts, you need to enable this separately. Note See Allow manual override of Canadian taxes on Expense report export.

- In NetSuite, the information shown on the expense line depends on the applicable tax nexus type:
  - United States nexuses — The tax breakout is not supported on the expense line in NetSuite. The expense line **Amount** should include tax.
  - Canadian nexuses — The tax breakout is included on the expense line in NetSuite. The **Amount** is net of tax. You can select a tax code and NetSuite applies the tax rates from the tax code or tax group automatically. By default, you cannot enter a tax amount manually or set a field mapping definition to override the calculated values in NetSuite. However you can configure the integration to export Canadian tax amounts entered manually on the receipt in OpenAir. See Allow manual override of Canadian taxes on Expense report export.
  - Other nexuses — The tax breakout is included on the expense line in NetSuite. The **Amount** is net of tax. You can select a tax code and NetSuite applies the tax rates from the tax code or tax group

automatically. You can enter a tax amount manually or set a field mapping definition to override the calculated value, by default.

Setting Up Expense Report Tax Export

Additional setup is required before you can export tax location information to NetSuite when exporting expense reports.

To set up expense report tax export

1. If you use the Legacy Tax features in NetSuite, enable the Advanced Taxes feature for your NetSuite account. See Enabling Advanced Taxes (External link to NetSuite SuiteAnswers | Answer Id: 8020).

> ⚠ **Important:** The Expense Reports (Expense Reports) export workflow is not fully compatible with the SuiteTax feature:
>
> - Tax amounts are calculated correctly when exporting OpenAir expense reports to NetSuite as vendor bills.
> - When exporting OpenAir expense reports to NetSuite as expense reports, however, tax details need to be entered manually in NetSuite when using the SuiteTax engine. See Entering Expense Reports with Tax Impact in SuiteTax (external link to NetSuite SuiteAnswers | Answer ID: 65664)
>
> Note that SuiteTax compatibility testing is currently in progress. You should test the NetSuite <> OpenAir integration with the SuiteTax feature on sandbox accounts before you enable the feature in your NetSuite production account. See Known Limitations of SuiteTax (external link to NetSuite SuiteAnswers | Answer ID: 98937). See also Legacy Tax or SuiteTax.

2. (Optional) If you export expense report from OpenAir to NetSuite as vendor bills:

   a. In NetSuite, go to Setup > Company > General Preferences

   b. On the Custom Preferences subtab, under the NetSuite OpenAir SRP Integration section, check the **Exp Report to Vendor Bill Taxes** box.

   c. Click **Save**.

   In NetSuite, go to Setup > Accounting > Accounting Preferences, and click the Time & Expenses subtab.

3. Configure OpenAir to use tax locations on receipts or expense items to calculate tax.

   a. In OpenAir, go to Administration > Global Settings > Organization > Tax Options.

   b. Set the **Tax calculation method** to `Tax locations`.

   c. Check the **Enable tax locations on receipts** box.

   > ⓘ **Note:** Skip this step if your OpenAir account is configured to determine the receipt tax location based on the tax location specified for the expense item.

   d. Click **Save**.

4. Create tax code or tax group records in NetSuite and matching tax location records in OpenAir.
   a. In NetSuite, set up tax nexus, tax codes and tax groups as required. See topics under Advanced Taxes (External link to NetSuite SuiteAnswers | Answer Id: 8775) and Managing Tax Codes (External link to NetSuite SuiteAnswers | Answer Id: 8031).

      > **Note:** NetSuite uses the term nexus to describe a tax jurisdiction or geographic area where you pay taxes and where specific tax rates and regulations apply.
      >
      > For employee and vendor records, The tax nexus associated with an employee or vendor record is determined based on either of the following.
      >
      > - If you use NetSuite OneWorld, the country of the subsidiary associated with the employee or vendor. Note that vendors can be associated with multiple subsidiaries.
      > - Country specified in the shipping address for an employee, or in the billing or shipping address (for a vendor).

   b. In OpenAir, create or modify tax location records so that each tax location corresponds to a tax code or tax group in NetSuite. Enter the **NetSuite Tax Code ID** and set the **Tax rates**.
      - The **NetSuite Tax Code ID** [netsuite_tax_rate_id] on the tax location entity form holds the internal ID of the corresponding tax code or tax group record in NetSuite.
      - Make sure the information under the Tax rates section matches the rate on the corresponding tax code record in NetSuite, or the rates of each tax codes in the corresponding tax group record in NetSuite.

      > **Important:** Remember to create or update tax locations in OpenAir when you make changes to tax codes and groups in NetSuite.

5. If you export expense report from OpenAir to NetSuite as vendor bills, add tax export information to vendor records in NetSuite, on the OpenAir subtab:
   - **OpenAir: User Tax Nexus Type** [custentity_oa_vendor_tax_nexus] — the type of tax nexus associated with the vendor or the parent vendor. Select one of the following options:
     - United States — Federal Tax and State Tax
     - Canada — Goods and Services Tax (GST) or Harmonized Sales Tax (HST) and Provincial Sales Taxes (PST)
     - Other — Value Added Tax (VAT)

     > **Note:** The tax nexus associated with the vendor or parent vendor is determined based on either of the following.

   - **OpenAir: Map Exp. Rep. to Parent Vendor** [custentity_oa_map_to_parent_vendor] — Check this box to map OpenAir expense reports associated with the corresponding employee record to a parent vendor bill.
   - **OpenAir: Map Exp. Rep. to Vendor Bill**[custentity_oa_expense_to_vendorbill] — Check this box to map OpenAir expense reports associated with the corresponding employee record to a vendor bill.

   For more information, see Employees (Vendors) Import Workflow.

6. Add a shipping address to integrated employee records in NetSuite. The shipping address is used to determine the **NetSuite Tax Nexus Type** for the employee.

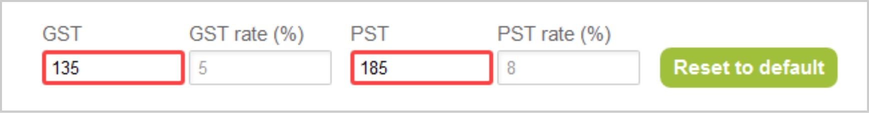
7. Run the integration for the Employees (Employees) import workflow, and the Employees (Vendors) import workflow (if in use) to update the employee records in OpenAir with **NetSuite Tax Nexus Type** information.

   > **Note:** Verify that the demographic form for employees in OpenAir shows the correct **NetSuite Tax Nexus Type**. If empty, select a **NetSuite Tax Nexus Type** from the dropdown options. This field is required to export expense report tax information from OpenAir to NetSuite.

8. In OpenAir, go to Administration > Application Settings > Expenses > Expense report layout and make sure the expense report layout includes the tax information you want to export.
9. In NetSuite, customize the expense report and vendor bill transaction forms to include the relevant tax information.

Allow manual override of Canadian taxes on Expense report export

Check this box to export the tax amounts entered manually on receipts on OpenAir instead of using the tax location to determine tax amount when exporting Canadian expense report tax from OpenAir to NetSuite. In NetSuite, the exported expense report includes custom fields showing the tax amounts entered manually both at the expense line level and at the expense report level.



GST	GST rate (%)	PST	PST rate (%)
135	5	185	8

Additional setup is required before you can use this feature:

1. In NetSuite, go to Setup > Company > General Preferences.
2. On the Custom Preferences subtab, under the NetSuite OpenAir SRP Integration section, check the **Enable Canadian Expense Tax Override** box.

Note: The following custom transaction body fields and transaction line fields are created when you install the NetSuite OpenAir SRP Integration bundle - Bundle ID 369637 (previously 2851) 1.23 or later version. These fields are used to display information – the field values are not used in any calculations.

- **OpenAir: GST/HST Amount** [`custbody_oa_gst_override`] — Transaction body field on expense reports and vendor bills holding the calculated sum of Goods and Services Tax (GST) or Harmonized Sales Tax (HST) across all expense lines.
- **OpenAir: PST Amount** [`custbody_oa_pst_override`] — Transaction body field on expense reports and vendor bills holding the calculated sum of Provincial Sales Tax (PST) across all expense lines.
- **OpenAir: GST/HST** [`custcol_oa_gst_override`] — Transaction line field on expense reports and vendor bills holding the Goods and Services Tax (GST) or Harmonized Sales Tax (HST) entered manually on the OpenAir receipt.
- **OpenAir: PST** [`custcol_oa_pst_override`] — Transaction line field on expense reports and vendor bills holding the calculated sum of Provincial Sales Tax (PST) entered manually on the OpenAir receipt.

Best Practice Guidelines

Review the following guidelines:

- Before you enable this feature, first enable and set up expense report tax export. See Enable tax export on expense reports.
- The Canadian tax location used on OpenAir receipts must have a corresponding tax code or tax group in NetSuite. The **NetSuite Tax Code ID** [`netsuite_tax_rate_id`] on the tax location entity form holds the internal ID of the corresponding tax code or tax group record in NetSuite.
- The employee must be associated with a Canadian tax nexus in NetSuite. The subsidiary country or the shipping address determines the applicable tax nexus for employees. To verify the nexus associated with an employee in NetSuite, go to Transactions > Employees > Enter Expense Reports, select the **Employee** and click the expenses subtab. The field displayed at the expense line level are different depending on the applicable nexus.
- Expense lines in NetSuite Tax amount on each expense line is not displayed. This is a system limitation.
- For tax reporting, the tax on each expense line may not match the tax amounts entered manually on the OpenAir receipts, especially when multiple tax codes are involved.
- If there are rounding discrepancies in the total tax amount for the expense report in OpenAir and NetSuite, the total GST amount is adjusted to keep the total amount to be reimbursed to the employee consistent across the two applications.

Automatically create negative Expense report lines in NetSuite for non-reimbursable expenses

Check this box to create a negative expense line when exporting a non-reimbursable receipt from OpenAir to NetSuite. This effectively creates two expense lines for non-reimbursable receipts, a positive expense line which can be marked as billable if you need to bill the customer for this expense, and a negative expense line for the same amount to ensure that the employee is not reimbursed for this amount. See also Keep customer reference on negative Expense report lines in NetSuite for non-reimbursable expenses.

⚠ **Important:** This workflow setting only applies when you export expense report from OpenAir to NetSuite as expense reports. It is not currently possible to create negative expense lines in vendor bills using the Expense Reports (Expense Reports export workflow.

ⓘ **Note:** You can use advanced mapping to set the value of the **Non-reimbursable** [`expenseReportExpense.isNonReimbursable`] field on the NetSuite expense line in function of the receipt **Status** [`ticket.status`] in OpenAir.

Do not set the foreign currency in NetSuite for foreign currency receipts when exporting Expense reports

Exported expense reports can include foreign currency receipts. By default, the NetSuite expense line corresponding to the OpenAir foreign currency receipt shows the receipt amount in the foreign currency specified on the receipt in OpenAir.

Check this box to show the receipt amount in the currency for the employee or vendor instead of the foreign currency specified on the receipt.

Keep customer reference on negative Expense report lines in NetSuite for non-reimbursable expenses

Check this box to include the customer and project reference on negative expense lines created when exporting non-reimbursable receipts from OpenAir to NetSuite. See also Automatically create negative Expense report lines in NetSuite for non-reimbursable expenses.

Show last NetSuite integration error on expense reports

Check this box to show the integration error message on the expense report in OpenAir if an error occurred during the last attempt to export this expense report record to NetSuite.

When checked, OpenAir stores the integration error message for that record in the **Last NetSuite integration error** custom field.

This feature lets users with the relevant role permission identify common issues preventing the export of expense reports to NetSuite and take any corrective actions within their control.

- Only the error message from the last integration run is stored, if any. OpenAir clears the **Last NetSuite integration error** field if the expense report exports successfully to NetSuite.
- The **Last NetSuite integration error** field shows:
  - As a read only field on the properties form for Expense reports.
  - As a column on Expense reports list views.

- When **Show last NetSuite integration error on expense reports** is enabled, the following integration role permission becomes available in Administration > Global Settings > Users > Roles: **Allow employee to view last NetSuite error on expense reports**.
- Only administrators and employees with the **Allow employee to view last NetSuite error on expense reports** role permission can see the **Last NetSuite integration error**.
- The error message displayed on the properties form or list view is the same as:
  - The message displayed on screen when running the integration manually.
  - The message recorded on the integration run log accessible from the Status screen. See Reviewing the Integration Run History and Status.

Note: Review the following guidelines:

- If you enable **Show last NetSuite integration error on expense reports** for the first time, OpenAir creates a custom field associated with expense reports to store the last NetSuite integration error message.
- The **Last NetSuite integration error** field is not cleared if you disable **Show last NetSuite integration error on expense reports**. Employees with the relevant role permission will continue to see the last integration error recorded before you disabled the workflow setting. To hide the last integration error recorded, you must disable the role permission before disabling the workflow setting.

Expense Reports (Expense Reports) Export Field Mapping Definition

The Expense Reports (Expense Reports) export workflow includes the following mapping groups:

- Expense Report [Integration point: `EXPENSE_REP_TO_EXPENSE_REP`] — See Expense Report Mapping Group.
- Vendor Bill [Integration point: `EXPENSE_REP_TO_VENDORBILL`] — See Vendor Bill Mapping Group.
- Expense Line [Integration point: `EXPENSE_REP_LINE`] — See Expense Line Mapping Group.
- Vendor Bill Line [Integration point: `EXPENSE_REP_LINE_TO_VENDORBILL_LINE`] — See Vendor Bill Line Mapping Group.
- Tax Line [Integration point: `EXPENSE_REP_LINE_TAX`] — See Tax Line Mapping Group.

Expense Report Mapping Group

Use the Expense Report mapping group to map expense report header information on the OpenAir expense report record with expense report header information on the NetSuite expense report transaction record. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information exported by default includes NetSuite employee ID

for the user who created the expense report, expense report date, expense report number, advance, notes, accounting and supervisor approval (with NetSuite user IDs of approvers), and multicurrency flag. Filters and pullbacks are supported.

- Additional Field Mappings — There are no specific best practice guidelines for custom field mapping definitions. The mapping group supports custom pullback field mappings, retrieving information from the record created in NetSuite back into OpenAir.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	**OpenAir Record Type [table]**	**Mapping Group [Integration point]**
Expense Report [ExpenseReport] Schema: ExpenseReport (2020.1) | ExpenseReport (2023.1) XSD: tranEmp (2020.1) | tranEmp (2023.1) XSD (pullback): TransactionSearchRowBasic — platformCommon (2020.1) | platformCommon (2023.1)	Expense Report [envelope] Data Dictionary: envelope	Expense Report ■ [EXPENSE_REP_TO_EXPENSE_REP] ■ Pullback row [NS_EXPENSE_REP_TO_OA_EXPENSE_REP]

Default and Custom Filters

By default, expense report records are exported from OpenAir to NetSuite if the following conditions are met:

- The expense report is approved in OpenAir.
- The expense report was not already exported — Envelope.exported is empty.
- The **NetSuite ID** [netsuite_envelope_id] custom field is empty OR the **Allow objects with a NetSuite ID to be exported during integration run** box is checked on the OpenAir NetSuite Connector administration form. See Allow objects with a NetSuite ID to be exported during integration run.
- The expense report must contain at least one receipt.
- If the **Enable OpenAir expense reports to NetSuite vendor bill integration** box is checked on the OpenAir NetSuite Connector administration form, the **Export folders from this employee as NetSuite vendor bills** [User.netsuite_export_envelope_to_vendorbill__c] box is not checked on the employee record associated with the expense report in OpenAir. When both are checked, the Vendor Bill mapping group is used instead. See Vendor Bill Mapping Group.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

> **Note:** Custom filters for the Expense Report and Vendor bill mapping group are the same. Use the Expense Report mapping group tab to set custom filters for both mapping groups.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Expense Report mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

⚠ **Important:** Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

See the OpenAir Help Center for detailed information about additional field mapping definitions for the Expense Report mapping group tab that may be required or useful in some cases. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Vendor Bill Mapping Group

Use the Vendor Bill mapping group to map expense report header information on the OpenAir expense report record with vendor bill header information on the NetSuite vendor bill transaction record. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information exported by default includes NetSuite employee ID for the user who created the expense report, NetSuite vendor ID or [parent vendor ID, expense report date, expense report number, notes, NetSuite user IDs of accounting and supervisor approvers, and vendor currency. Filters and pullbacks are supported.
- Additional Field Mappings — There are no specific best practice guidelines for custom field mapping definitions. The mapping group does not support custom pullback field mappings, retrieving information from the record created in NetSuite back into OpenAir.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [`SOAP object`]	OpenAir Record Type [`table`]	Mapping Group [`Integration point`]
Vendor Bill [`VendorBill`, `TransactionSearch`] Schema: VendorBill (2020.1) | VendorBill (2023.1) XSD: tranPurch (2020.1) | tranPurch (2023.1)	Expense Report [`envelope`]	Vendor Bill - [`EXPENSE_REP_TO_VENDORBILL`]

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
XSD (pullback): TransactionSearchRowBasic — platformCommon (2020.1) | platformCommon (2023.1)	Data Dictionary: envelope	- Pullback row [NS_EXPENSE_REP_TO_OA_EXPENSE_REP]

Default and Custom Filters

By default, expense report records are exported from OpenAir to NetSuite if the following conditions are met:

- The expense report is approved in OpenAir.
- The expense report was not already exported — Envelope.exported is empty.
- The **NetSuite ID** [netsuite_envelope_id] custom field is empty OR the **Allow objects with a NetSuite ID to be exported during integration run** box is checked on the OpenAir NetSuite Connector administration form. See Allow objects with a NetSuite ID to be exported during integration run.
- The expense report must contain at least one receipt.
- The **Enable OpenAir expense reports to NetSuite vendor bill integration** box is checked on the OpenAir NetSuite Connector administration form and the **Export folders from this employee as NetSuite vendor bills** [User.netsuite_export_envelope_to_vendorbill__c] box is checked on the employee record associated with the expense report in OpenAir. Otherwise, the Expense Report mapping group is used instead. See Expense Report Mapping Group.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Note: Custom filters for the Expense Report and Vendor bill mapping group are the same. Use the Expense Report mapping group tab to set custom filters for both mapping groups.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Vendor Bill mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Expense Line Mapping Group

Use the Expense Line mapping group to map information from receipt records in the OpenAir expense report with information on the expense line items on the NetSuite expense report transaction record.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information exported by default includes .
- Additional Field Mappings — There are no specific best practice guidelines for custom field mapping definitions. The mapping group does not support custom pullback field mappings, retrieving information from the line item created in NetSuite back into OpenAir.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type `[SOAP object]`	**OpenAir Record Type** `[table]`	**Mapping Group** `[Integration point]`
Expense Report Expense Line Item `[ExpenseReportExpense]` Schema: ExpenseReportExpense (2020.1)|ExpenseReportExpense (2023.1) XSD: tranEmp (2020.1)|tranEmp (2023.1)	Receipt `[ticket, item]` Data Dictionary: ticket, item	Expense Line `[EXPENSE_REP_LINE]`

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Expense Line mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

Review the following guidelines:

- Reimbursable or Non-reimbursable **Status** — Exported expense reports can include both reimbursable and non-reimbursable receipts. OpenAir receipts have a Reimbursable (R) or Non-reimbursable (N) **Status**. To make sure you can bill customers for non-reimbursable expenses without reimbursing the employee, configure OpenAir NetSuite Connector to create an additional negative line for non-reimbursable receipts. See Automatically create negative Expense report lines in NetSuite for non-reimbursable expenses.

> ⚠ **Important:** You should not use advanced mapping to set the value of the **Non-reimbursable** [expenseReportExpense.isNonReimbursable] field on the NetSuite expense line in function of the receipt **Status** [ticket.status] in OpenAir. Support for the **Non-reimbursable** field in NetSuite is limited and impacts the ability to rebill expense line items.

Vendor Bill Line Mapping Group

Use the Vendor Bill Line mapping group to map information from receipt records in the OpenAir expense report with information on the line items on the NetSuite vendor bill transaction record.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information exported by default includes .
- Additional Field Mappings — There are no specific best practice guidelines for custom field mapping definitions. The mapping group does not support custom pullback field mappings, retrieving information from the line item created in NetSuite back into OpenAir.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Vendor Bill Expense Line Item [VendorBillExpense] Schema: VendorBillExpense (2020.1)|VendorBillExpense (2023.1) XSD: tranPurch (2020.1)|tranPurch (2023.1)	Receipt [ticket, item] Data Dictionary: ticket, item	Expense Report [EXPENSE_REP_LINE_TO_VENDORBILL_LINE]

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Vendor Bill Line mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

Review the following guidelines:

- **Currency** and **Foreign Amount** — When exporting expense reports from OpenAir to NetSuite as vendor bills, the receipts **Currency** should be the same as the **Currency** on the vendor or parent vendor record in NetSuite. Currency and foreign amount information is not exported by default as part of the Vendor Bill Line mapping group.
- The following fields are not available as standard on the vendor bill expense line in NetSuite, and not exported by default as part of the Vendor Bill Line mapping group: **Date**, **Receipt**.

Tax Line Mapping Group

Use the Tax Line mapping group to map information from the tax location record associated with the tax rate record associated with the receipt in the OpenAir expense report with information on the line items on the NetSuite expense report transaction record. The Tax Line mapping group is used only when the following conditions are all met:

- The **Enable tax export on Expense reports** box is checked on the Expense Reports (Expense Reports) export workflow settings form.
- Your OpenAir account is configured to use tax locations on receipts or expense items to calculate tax.
- The **NetSuite Tax Nexus Type** [`netsuite_tax_nexus`] custom field on the demographic form for the employee associated with the expense report in OpenAir is set to `Canada` or `VAT`. For United States nexuses, tax breakout information on expense lines is not supported in NetSuite — the expense line **Amount** should include tax.

For more information about setting up the export of tax information with expense reports, see Enable tax export on expense reports.

⚠ **Important:** Review the following guidelines:

- This mapping group cannot be used to map information from the tax location record with information on the line items on the NetSuite vendor bill transaction record. It can only be used to map receipt tax information when exporting expense reports from OpenAir to NetSuite as expense reports.
- If you add custom mapping definitions in the Tax Line mapping group tab for the NetSuite fields that are also mapped in the Expense Line mapping group tab, the field mapping definition in the Expense Line mapping group tab takes precedence, any value set for that NetSuite field using the Expense Line field mapping definition overrides the value set for that NetSuite field using the Tax Line field mapping definition.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information exported by default includes .
- Additional Field Mappings — There are no specific best practice guidelines for custom field mapping definitions. The mapping group does not support custom pullback field mappings, retrieving information from the line item in NetSuite back into OpenAir.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Expense Report Expense Line Item [ExpenseReportExpense] Schema: ExpenseReportExpense (2020.1)|ExpenseReportExpense (2023.1) XSD: tranEmp (2020.1)|tranEmp (2023.1)	Tax location [tax_location] Data Dictionary: tax_location	Tax Line [EXPENSE_REP_LINE_TAX]

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Tax Line mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Invoices (Invoices) Export Workflow

The Invoices (Invoices) export workflow lets you create new invoices and follow the approval in OpenAir, and then export approved invoices to NetSuite.

You can export both invoice and project header information to the invoice transaction record. This allows you to export project-specific information stored in standard or custom OpenAir fields, and then to display this information on the invoice header in NetSuite and use this information in reports.

All charges in the OpenAir invoice export to NetSuite as line items on the Items, Billable Items, or Billable Expenses subtabs.

You can import the tax amounts calculated for the exported invoice in NetSuite back into the OpenAir invoice, including GST and PST taxes. Set up the Advanced Taxes accounting feature in NetSuite and enable Tax Options in OpenAir. Set up nexuses in NetSuite for the countries or locations from which taxes will be paid and create or identify the tax codes or tax groups for each nexus. When you export an invoice to NetSuite, taxes on the total amount are calculated in NetSuite and import back to the invoice in OpenAir. See Import taxes applied to exported invoices.

Important: If the SuiteTax is enabled for your NetSuite account, NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) to version 2.11 or later is required in order for the tax amount calculated in NetSuite to be imported into OpenAir. With earlier versions, the tax amount calculated in NetSuite with SuiteTax cannot be imported back into OpenAir.

Note that SuiteTax compatibility testing is currently in progress. You should test the NetSuite <> OpenAir integration with the SuiteTax feature on sandbox accounts before you enable the feature in your NetSuite production account. See Known Limitations of SuiteTax (external link to NetSuite SuiteAnswers). See also Legacy Tax or SuiteTax.

If you use the Customer POs (Sales Orders) import workflow to import sales order from NetSuite into OpenAir as Customer POs, you can use a workflow setting to retrieve invoice information sourced from the sales orders in NetSuite and include this information when exporting invoices from OpenAir to NetSuite. See Initialize invoices from corresponding sales orders when exporting invoices to NetSuite.

For information about workflow specific settings, see Invoices (Invoices) Export Workflow Settings.

For information about field mapping definitions for this workflow, see Invoices (Invoices) Export Field Mapping Definition.

Important: Changes made to invoices and charges in OpenAir after the initial export are not exported to NetSuite. The Invoices (Invoices) export workflow can be used to create new invoices in NetSuite but not to update already integrated invoices.

Requirements and Downstream Dependencies

The Expense Reports (Expense Reports) export workflow must be active — You must export expense reports to export invoices. See Expense Reports (Expense Reports) Export Workflow.

Matching NetSuite service item and OpenAir service records must exist before you can export invoices and credit memos from OpenAir to NetSuite.

Invoice approvals must be enabled to export invoices from OpenAir to NetSuite. The Invoices (Invoices) export workflow is not available otherwise.

After you export an invoice from OpenAir to NetSuite, you can apply credit memos, customer deposits and customer payments against the invoice, and import the information back into OpenAir.

- Credit memos applied in NetSuite are imported into OpenAir as negative invoices. The original OpenAir invoice also shows the credited amount. See Credits (Applied Credit Memos) Import Workflow.

  Note: You can also create negative invoices in OpenAir and export them to NetSuite as credit memos. See Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Export Workflow.

- Deposits applied in NetSuite are imported into OpenAir as retainers. See Retainers (Applied Customer Deposits) Import Workflow.
- Payments applied in NetSuite are imported into OpenAir as payments. See Payments (Invoice Payments) Import Workflow.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
✓	✓ (Invoice mapping group)	✓	✓ (On Approval)	✓	

- Run the integration manually, use a scheduled integration run to create the NetSuite invoice transaction record after the invoice is approved in OpenAir. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.
- Setup the real-time integration to create the invoice transaction record when the invoice is approved in NetSuite. See Configuring Real-Time Record Export from OpenAir to NetSuite.
- View and edit the OpenAir invoice record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Invoices (Invoices) Export Workflow

The Invoices (Invoices) workflow is available as standard.

To set up the Invoices (Invoices) export workflow in NetSuite

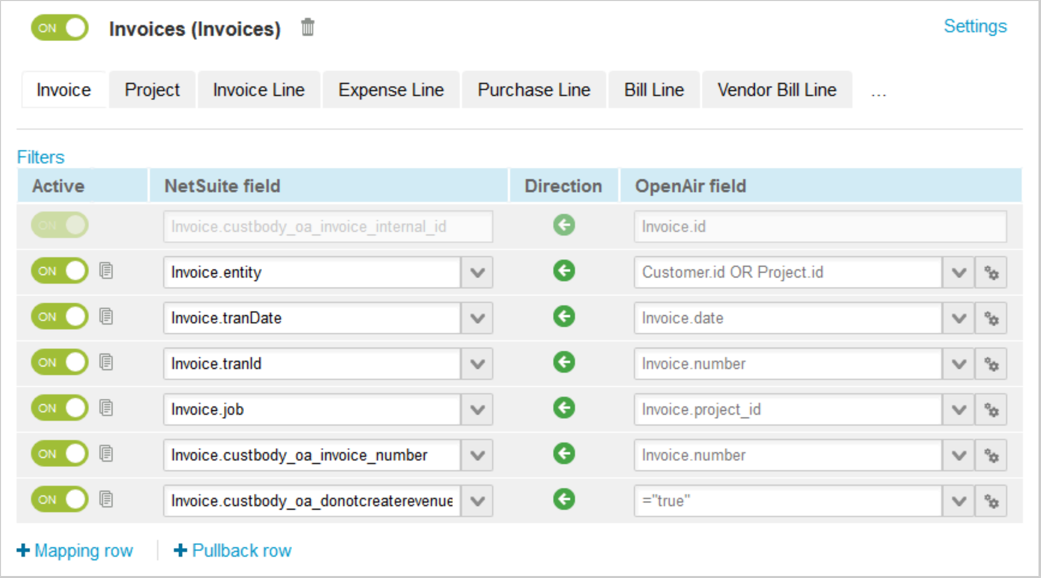
1. (Optional) If you want to export OpenAir project header information to the NetSuite invoice, do the following:
   a. In NetSuite, go to Setup > Accounting > Accounting Preferences.
   b. Clear the **Consolidate Projects on Sales Transactions** box on the Items/Transactions subtab, under Sales & Pricing.
   c. Click **Save**.
2. (Optional) Configure NetSuite to use the OpenAir **Invoice number** instead of the auto-generated record number for invoices in NetSuite. To do so:
   a. In NetSuite, go to Setup > Company >Auto-Generated Numbers, and click the Document Numbers subtab.
   b. Check the box under the **Allow Override** column for Invoice. When checked, exported invoices have the same number in both NetSuite and OpenAir.
   c. Click **Save**.

To set up the Invoices (Invoices) export workflow in OpenAir

1. Enable invoice approvals in OpenAir. To do so:
   a. In OpenAir, go to Administration > Global settings > Organization > Approval Options.
   b. Check the **Enable approvals for invoices** box.
   c. Click **Save**.
2. Set up the Expense Reports (Expense Reports) export workflow. See Expense Reports (Expense Reports) Export Workflow.
3. (Optional) If you want to export OpenAir project header information to the NetSuite invoice, do the following:
   a. In OpenAir, go to Administration > Application Settings > Invoices > Other Settings.
   b. Check the **One project per invoice** box.
   c. Click **Save**.
4. (Optional) Set up related workflows and integration features:
   - To import Credit memos applied in NetSuite into OpenAir as negative invoices, see Credits (Applied Credit Memos) Import Workflow.
   - To create negative invoices in OpenAir and export them to NetSuite as credit memos, see Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Export Workflow.
   - To import deposits applied in NetSuite into OpenAir as retainers, see Retainers (Applied Customer Deposits) Import Workflow.
   - To import payments applied in NetSuite into OpenAir as payments, see Payments (Invoice Payments) Import Workflow.
   - To export expense reports from OpenAir to NetSuite as vendor bills and include billable items from vendor bills when exporting invoices from OpenAir to NetSuite, see Expense Reports (Expense Reports) Export Workflow and Employees (Vendors) Import Workflow.

- To bill activities of all types, make sure you import both service items and other types of items from NetSuite into OpenAir as services records using the Services (Services Items) import workflow. See Disable NetSuite service type filtering for services in NetSuite integration.

5. In OpenAir, go to Administration > NetSuite Connector > Mapping.
6. Click **New workflow** then **Invoices (Invoices)**.
7. Click **Invoices (Invoices)** in the list of integration workflows.
8. Click **Workflow settings** to view and modify the settings specific to this workflow. For more information about the workflow settings, see Invoices (Invoices) Export Workflow Settings.
9. Modify the field mapping definition tables for the Invoices (Invoices) export workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Invoices (Invoices) Export Field Mapping Definition.
10. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



Active	NetSuite field	Direction	OpenAir field
ON	Invoice.custbody_oa_invoice_internal_id	←	Invoice.id
ON	Invoice.entity	←	Customer.id OR Project.id
ON	Invoice.tranDate	←	Invoice.date
ON	Invoice.tranId	←	Invoice.number
ON	Invoice.job	←	Invoice.project_id
ON	Invoice.custbody_oa_invoice_number	←	Invoice.number
ON	Invoice.custbody_oa_donotcreaterevenue	←	="true"

11. Click **Save**.
12. (Optional) Add the Invoices (Invoices) export workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Invoices (Invoices) Export Workflow Settings

You can control Invoices (Invoices) export workflow settings from the NetSuite Connector UI. On the OpenAir NetSuite Connector Mapping screen, select the Invoices (Invoices) export workflow and click **Settings** to view the workflow settings form.

The following settings are available for the Invoices (Invoices) export workflow:

- Import taxes applied to exported invoices
- Adjust quantity based on % invoiced for time charges related to NetSuite sales order lines with quantities that do not reflect hours
- Always respect the NetSuite sales order customer and project when exporting invoices
- Combine charges by service on invoices
- Create billing rules for NetSuite discount and markup lines on sales orders

- Disable expense report references on invoices in NetSuite integration
- Disable vendor bill linkage on invoices in NetSuite integration
- Enable charges to map to NetSuite description items
- Enable expense charge linkage with sales order lines
- Enable the OpenAir invoice link mapping to NetSuite
- Initialize invoices from corresponding sales orders when exporting invoices to NetSuite
- Show last NetSuite integration error on invoices
- Use Service item instead of Product item during invoices export for purchase charges
- When using the sales order integration, use invoice date to determine NetSuite due date.

Import taxes applied to exported invoices

Check this box to import the total tax calculated in NetSuite for the invoice back into OpenAir. When you export invoices from OpenAir to NetSuite, NetSuite calculates the tax and OpenAir NetSuite Connector imports the calculated total tax back into OpenAir and the OpenAir invoice includes the total tax calculated in NetSuite.

Setting Up Invoice Tax Import

Additional setup is required before you can import tax information from NetSuite back into OpenAir when exporting invoices.

To set up invoice tax import

1. If you use the Legacy Tax features in NetSuite, enable the Advanced Taxes feature for your NetSuite account. See Enabling Advanced Taxes (External link to NetSuite SuiteAnswers | Answer Id: 8020).

   > ⚠ **Important:** If the SuiteTax is enabled for your NetSuite account, NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) 2.11 or later version is required in order for the tax amount calculated in NetSuite to be imported into OpenAir. With earlier versions, the tax amount calculated in NetSuite with SuiteTax cannot be imported back into OpenAir.
   >
   > Note that SuiteTax compatibility testing is currently in progress. You should test the NetSuite <> OpenAir integration with the SuiteTax feature on sandbox accounts before you enable the feature in your NetSuite production account. See Known Limitations of SuiteTax (external link to NetSuite SuiteAnswers). See also Legacy Tax or SuiteTax.

2. Create tax code or tax group records and enter the relevant tax information on customer records in NetSuite.
   a. In NetSuite, set up tax nexus, tax codes and tax groups as required. See topics under Advanced Taxes (External link to NetSuite SuiteAnswers | Answer Id: 8775) and Managing Tax Codes (External link to NetSuite SuiteAnswers | Answer Id: 8031).

      > ⓘ **Note:** NetSuite uses the term nexus to describe a tax jurisdiction or geographic area where you pay taxes and where specific tax rates and regulations apply.

   b. Go to each customer records and select the standard tax code or tax group associated with each customer.

Adjust quantity based on % invoiced for time charges related to NetSuite sales order lines with quantities that do not reflect hours

⚠ **Important:** This feature is available when the Invoices (Invoices) export workflow is used in conjunction with the Customer POs (Sales Orders) import workflow. See Customer POs (Sales Orders) Import Workflow.

Check this box to bill the customer for a percentage of the amount specified on the sales order line in NetSuite instead of billing for the number of hours in the time charge.

Always respect the NetSuite sales order customer and project when exporting invoices

Check this box to include customer and project information from the NetSuite sales order including sales team information on invoices you export from OpenAir to NetSuite. See also Customer POs (Sales Orders) Import Workflow.

Combine charges by service on invoices

Check this box to roll up non-expense charges by service items on invoices you export from OpenAir to NetSuite. When checked, OpenAir NetSuite Connector groups charges for the same project first, and then for the same service onto one line item on the NetSuite invoice.

When you use the Customer POs (Sales Order) import workflow, and time and fixed-fee charges reference a customer PO in OpenAir, OpenAir NetSuite Connector rolls up these charges by sales order line on the NetSuite invoice.

Review the following guidelines:

- The display order of service items for each project does no respect the order specified on the invoice layout.
- This setting only impacts non-expense charges, and not billable expenses. To roll up billable expenses in NetSuite, go to Setup > Accounting > Invoicing Preferences and check the **Combine Expense Items on Invoices** box.

Create billing rules for NetSuite discount and markup lines on sales orders

⚠ **Important:** This feature is available when the Invoices (Invoices) export workflow is used in conjunction with the Customer POs (Sales Orders) import workflow. See Customer POs (Sales Orders) Import Workflow.

Check this box to create a billing rule for each discount item or markup item on the sales order in NetSuite. When you add a discount item or markup item to the sales order in NetSuite, and run the Customer POs (Sales Order) import workflow, OpenAir NetSuite Connector creates customer POs and billing rules for NetSuite discount and markup lines on sales orders. See Customer POs (Sales Orders) Import Workflow.

When you run the billing rules, OpenAir creates charges corresponding to the discounts and markups on the NetSuite sales order. You can edit the billing rules to apply the discounts or markups as required before you run the billing rules.

> **Note:** You can also create negative or positive charges manually corresponding to the discounts and markups you want to apply.

Disable expense report references on invoices in NetSuite integration

Check this box to disable expense report references on invoices exported to NetSuite.

When checked, expense charges export as line items listed on the Items subtab, instead of billable items listed on the Billable Items subtab.

> **Important:** When checked, expenses exported to NetSuite are not billable. Unless they are created manually, billable items on the invoice in NetSuite must reference an expense report and an expense line on the expense report.
>
> You should discuss this setting with OpenAir Professional Services before changing it.

Disable vendor bill linkage on invoices in NetSuite integration

> **Important:** This feature is available when the Invoices (Invoices) export workflow is used in conjunction with the POs (Vendor Bills) import workflow. See POs (Vendor Bills) Import Workflow.

Check this box to export all purchase charges on the OpenAir invoice to line items listed on the Items subtab on the NetSuite invoice.

When checked, all purchase charges on the OpenAir invoice export to line items listed on the Items subtab, even if the original vendor bill in NetSuite is billable.

- The line items are not associated with the vendor bill. Consider all accounting implications in NetSuite before you change this setting.
- It is best to clear the **Billable** box if the **Export to OpenAir** is checked on the vendor bill line item. The vendor bill line item will not be listed on the Billable Items subtab on the invoice when billed to customers.
- The POs (Vendor Bills) import workflow requirement to disable the **Combine Billable Items on Invoices** invoicing preference in NetSuite no longer applies (Setup > Accounting > Invoicing Preferences).

Otherwise and by default, when you import a vendor bill from NetSuite as a purchase order, bill the vendor bill lines imported into OpenAir as purchase items, and export the invoice from OpenAir to NetSuite:

- Purchase charges export as billable line items listed on the Billable items subtab on the NetSuite invoice if the **NetSuite vendor bill line billable** box is checked on the purchase item record in OpenAir.
- Purchase charges export as line items listed on the Items subtab on the NetSuite invoice otherwise. The line items on the NetSuite invoice are not linked to the NetSuite vendor bill.

Enable charges to map to NetSuite description items

⚠ **Important:** This feature is available when the Invoices (Invoices) export workflow is used in conjunction with the Customer POs (Sales Orders) import workflow. See Customer POs (Sales Orders) Import Workflow.

Check this box to export information from time ("hourly" or "other" rate) and fixed fee ("flat rate") charges on the OpenAir invoice as description line items on the NetSuite invoice.

When you are using the Customer POs (Sales Orders) import workflow and this box is checked you can export detailed information from OpenAir time (hourly rate), time (other) or flat rate charges to NetSuite as description line items. To do so, add custom mapping definitions to the mapping groups corresponding to each of the charge types you want to export to description line items.

- Group by Time — Custom field mapping definitions for hourly rate time charges (charge type T) information exported to description line items.
- Group by Other — Custom field mapping definitions for other time charges (charge type 0) information exported to description line items.
- Group by Fee — Custom field mapping definitions for flat rate charges (charge type F) information exported to description line items.

For more information and examples of custom field mapping definitions for these mapping groups, see Group by Time, Group by Other, and Group by Fee Mapping Groups.

⚠ **Important:** Information from OpenAir charges only export to description line items in NetSuite when all the following conditions are met:

- The Customer POs (Sales Orders) import workflow is enabled to sales orders from NetSuite into OpenAir as customer POs.
- Some charges on the invoice (but not necessarily all charges) are associated with a customer PO imported from a NetSuite sales order line. Information from all OpenAir time (hourly rate), time (other) or flat rate charges, including charges that do not reference a customer PO imported from a NetSuite sales order line, exports to description line items based on the charge type.
- The charge type is "hourly", "other", or "flat rate"
- Valid custom mapping definitions exist for the charge type.

Enable expense charge linkage with sales order lines

⚠ **Important:** This feature is available when the Invoices (Invoices) export workflow is used in conjunction with the Customer POs (Sales Orders) import workflow. See Customer POs (Sales Orders) Import Workflow.

Check this box to enable sales order line item references on expense charges exported to NetSuite. The **Disable Expense report references on invoices in NetSuite integration** box must also be checked — see Disable expense report references on invoices in NetSuite integration.

When checked, all expense charges on the OpenAir invoice export as line items listed on the Items subtab instead of billable line items listed on the Billable Items subtab.

- Expense charges on the NetSuite invoice that reference sales order line item roll up by sales order line item on the NetSuite invoice the same way time and fixed-fee charges are rolled up when the **Enable charges to map to NetSuite description items** box is checked. See also Enable charges to map to NetSuite description items.
- Expenses exported to NetSuite are not billable. The standard billable expense functionality in NetSuite cannot be used. You should contact OpenAir Professional Services to discuss your requirements before changing this setting.
- Expense billing rules must be created manually in OpenAir. Expense billing rules are not created automatically as part of this feature.

  > **Note:** Your OpenAir account configuration must let you set a service on expense charges. To enable this feature, contact OpenAir Customer Support.

- The **Adjust quantity based on % invoiced for time charges related to NetSuite sales order lines with quantities that do not reflect hours** setting is supported. This is useful when the **Quantity** is set to 1 on sales order line items used for expenses. The **Calculate Quantity From % Invoiced** box must be also checked on the Customer PO record in OpenAir.

Example 1. Billing Project Expenses Full Amount

The sales order includes a line item for expenses with **Rate** set to 1 and the **Quantity** set to the full amount for your project expenses. You can create a customer PO in OpenAir to match this sales order line item and an expense billing rule associated with both this customer PO and with a service. When running the billing rule, expense charge reference the customer PO. When you export the invoice to NetSuite, OpenAir NetSuite Connector rolls up the expense charges by sales order line item and associate it to the sales order line item in NetSuite.

> **Note:** You can associate the sales order line item with a supported billing rule type to create the customer PO automatically when importing the Sales Order into OpenAir. When you do so, delete the billing that was created automatically by the integration before you create the expense billing rule.

Example 2. Billing Percentage of Project Expenses

For this example, the **Adjust quantity based on % invoiced for time charges related to NetSuite sales order lines with quantities that do not reflect hours** must be checked on the Invoices (Invoices) workflow setting form.

The sales order includes a line item for expenses with **Rate** set to the full amount for your project expenses and the **Quantity** set to 1. You can create a customer PO in OpenAir to match this sales order line item and check the **Calculate Quantity From % Invoiced** box. You can then create a time billing rule associated with both the customer PO and with a service. When running the billing rule, expense charge reference the customer PO. When you export the invoice to NetSuite, OpenAir NetSuite Connector rolls up the expense charges by sales order line item and associate it to the sales order line item in NetSuite. The quantity will be calculated based on the percentage of the sales order line item full amount you are invoicing.

> **Note:** You can associate the sales order line item with a supported billing rule type to create the customer PO automatically when importing the Sales Order into OpenAir. When you do so, delete the billing that was created automatically by the integration before you create the expense billing rule.

Enable the OpenAir invoice link mapping to NetSuite

Check this box to export a link to the OpenAir **Online invoice** to a custom field on the NetSuite invoice record.

Note: The **OpenAir: Invoice Link** [`custbody_oa_invoice_link`] custom transaction body field is created when you install the NetSuite OpenAir SRP Integration bundle - Bundle ID 369637 (previously 2851) 1.20 or later version.

Initialize invoices from corresponding sales orders when exporting invoices to NetSuite

Important: This feature is available when the Invoices (Invoices) export workflow is used in conjunction with the Customer POs (Sales Orders) import workflow. See Customer POs (Sales Orders) Import Workflow.

Check this box to retrieve invoice information sourced from the sales orders in NetSuite and include this information when exporting invoices from OpenAir to NetSuite.

When checked, the Invoices (Invoices) export workflow initializes the invoice to be exported by retrieving information from NetSuite.

In NetSuite, when you click **Next Bill** on a sales order record, an invoice form appears and NetSuite sources information from the sales order to populate fields automatically on the invoice form. You can then review and modify the information as required before you click **Save** to create the invoice.

OpenAir NetSuite Connector follows a similar process to initialize the invoice. It retrieves information from NetSuite similar to the information populated when creating an invoice from the sales order. This information includes fields and values that would not be available otherwise. OpenAir NetSuite Connector then adds, removes or modifies line items to match the charges on the OpenAir invoice before exporting the invoice to NetSuite.

Note: Review the following guidelines:

- The extra step to initialize the invoice before export by retrieving information from NetSuite increases the time required to export an invoice from OpenAir to NetSuite.
- This feature reduces the need to use field mapping definitions. It also makes it possible to include header information from the sales order on the exported invoice header.
- You can use field mapping definitions instead of this feature to save information from the NetSuite sales order on the customer PO record in OpenAir, and include this information on line items when exporting Invoices from OpenAir to NetSuite. This lets you include exactly the information you need on the exported invoices.

Example 3. Tax Code Override on Sales Order Line Items

NetSuite populates the tax information automatically for sales order line items but lets users override the line item tax by selecting a different tax code. When this feature is enabled, OpenAir NetSuite Connector retrieves the tax information for each sales order line item from NetSuite and uses the tax code overrides to create the invoice in NetSuite. Otherwise, you need to set field mapping definitions to import the tax code from the sales order line items into the OpenAir customer POs, and to source the tax code on the invoice line items from the customer POs when exporting the invoice to NetSuite.

Information from transaction body and transaction line **custom fields** on the NetSuite sales order is included on the exported invoice in NetSuite.

- By default, NetSuite sources custom field values from the default values on the sales order. “Default values” here refers to the values that are automatically populated when creating the sales order — a transaction line custom field value sourced from an item pricing, for example. that sources its value unless you use field mapping definitions to set custom field values on the NetSuite invoice. This is adequate unless users can override the custom field default value on the NetSuite sales order.

- If users can override the custom field default value on the NetSuite sales order, you can include the override value using advanced mapping and field declaration. See Field Declarations (NS_CUSTOM_FIELDS_FROM_SO_INVOICE_HEADER and NS_CUSTOM_FIELDS_FROM_SO_INVOICE_LINE_ITEM).

> **Important:** Review the following guidelines:
>
> - When you use the NS_CUSTOM_FIELDS_FROM_SO_INVOICE_HEADER and NS_CUSTOM_FIELDS_FROM_SO_INVOICE_LINE_ITEM field declaration, OpenAir NetSuite Connector attempts to set the declared custom field explicitly to the value on the sales order. An integration error occurs if the value is no longer active (if users can no longer set the custom field to this default or override value on the sales order). Field declaration should only be used if users can override the custom field on the sales order.
> - If the declared field is used in another field mapping definition, the value will be set according to that other field mapping definition instead of the value from the sales order.

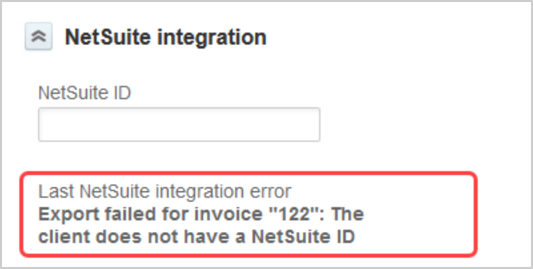
Show last NetSuite integration error on invoices

Check this box to show the integration error message on the invoice in OpenAir if an error occurred during the last attempt to export this invoice record to NetSuite.

When checked, OpenAir stores the integration error message for that record in the **Last NetSuite integration error** custom field.

This feature lets users with the relevant role permission identify common issues preventing the export of invoices to NetSuite and take any corrective actions within their control.

- Only the error message from the last integration run is stored, if any. OpenAir clears the **Last NetSuite integration error** field if the invoice exports successfully to NetSuite.
- The **Last NetSuite integration error** field shows:
  - As a read only field on the properties form for Invoices.

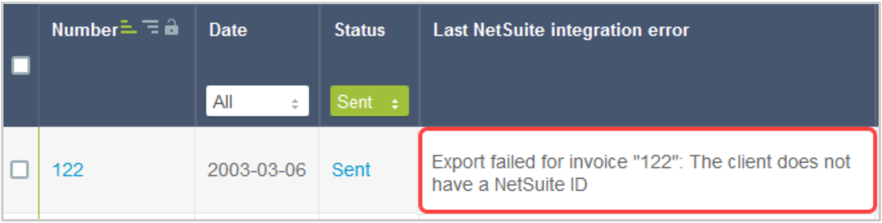


NetSuite integration

NetSuite ID

Last NetSuite integration error
Export failed for invoice "122": The client does not have a NetSuite ID

  - As a column on Invoices list views.



□	Number	Date	Status	Last NetSuite integration error
		All	Sent	
□	122	2003-03-06	Sent	Export failed for invoice "122": The client does not have a NetSuite ID

- When **Show last NetSuite integration error on invoices** is enabled, the following integration role permission becomes available in Administration > Global Settings > Users > Roles: **Allow employee to view last NetSuite error on invoices**.
- Only administrators and employees with the **Allow employee to view last NetSuite error on invoices** role permission can see the **Last NetSuite integration error**.
- The error message displayed on the properties form or list view is the same as:
  - The message displayed on screen when running the integration manually.
  - The message recorded on the integration run log accessible from the Status screen. See Reviewing the Integration Run History and Status.

Note: Review the following guidelines:

- If you enable **Show last NetSuite integration error on invoices** for the first time, OpenAir creates a custom field associated with invoices to store the last NetSuite integration error message.
- The **Last NetSuite integration error** field is not cleared if you disable **Show last NetSuite integration error on invoices**. Employees with the relevant role permission will continue to see the last integration error recorded before you disabled the workflow setting. To hide the last integration error recorded, you must disable the role permission before disabling the workflow setting.

Use Service item instead of Product item during invoices export for purchase charges

Check this box to export purchase items you bill to customers in OpenAir as service items instead of product items on the NetSuite invoice.

Important: The **Disable vendor bill linkage on invoices in NetSuite integration** box must also be checked on the workflow settings form.

When you import vendor bills from NetSuite into OpenAir as purchase orders, bill the purchase items in OpenAir, and export the invoice to NetSuite, purchase charges are listed as billable line items on the Billable Items subtab in NetSuite, by default.

When the **Disable vendor bill linkage on invoices in NetSuite integration** box is checked, purchase charges are listed as line items on the Items subtab instead.By default, the item on the NetSuite invoice appears as the purchase item in OpenAir.

When the **Use Service item instead of Product item during invoices export for purchase charges** box is checked and the purchase charge is associated with a service in OpenAir, the line item shows as a service item instead of a purchase item.

When using the sales order integration, use invoice date to determine NetSuite due date.

⚠ **Important:** This feature is available when the Invoices (Invoices) export workflow is used in conjunction with the Customer POs (Sales Orders) import workflow. See Customer POs (Sales Orders) Import Workflow.

When checked, the date of the OpenAir invoice is used to determine the due date on the NetSuite invoice instead of the date on the invoice information retrieved from NetSuite.

Invoices (Invoices) Export Field Mapping Definition

The Invoices (Invoices) export workflow includes the following mapping groups:

- Invoice [Integration point: `INVOICE_TO_INVOICE`] — See Invoice Mapping Group.
- Project [Integration point: `INVOICE_PROJECT`] — See Project Mapping Group.
- Invoice Line [Integration point: `INVOICE_LINE`] — See Invoice Line Mapping Group.
- Expense Line [Integration point: `INVOICE_EXPENSE_LINE`] — See Expense Line Mapping Group.
- Purchase Line [Integration point: `INVOICE_PURCHASE_LINE`] — See Purchase Line Mapping Group.
- Bill Line [Integration point: `INVOICE_BILLABLE_ITEM_LINE`] — See Bill Line Mapping Group.
- Vendor Bill Line [Integration point: `INVOICE_PURCHASE_VB_LINE`] — See Vendor Bill Line Mapping Group.
- Group by Time [Integration point: `INVOICE_LINE_TIME_GROUP_BY`] — See Group by Time, Group by Other, and Group by Fee Mapping Groups.
- Group by Other [Integration point: `INVOICE_LINE_OTHER_GROUP_BY`] — See Group by Time, Group by Other, and Group by Fee Mapping Groups.
- Group by Fee [Integration point: `INVOICE_LINE_FEE_GROUP_BY`] — See Group by Time, Group by Other, and Group by Fee Mapping Groups.

Invoice Mapping Group

Use the Invoice mapping group to map invoice header information on the OpenAir invoice record with invoice header information on the NetSuite invoice transaction record. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information exported by default includes customer and project ID, invoice date, invoice number, service, rate, and amount information. Filters and pullbacks are supported.
- Additional Field Mappings — Best practice guidelines for custom field mapping definitions. The workflow supports pullback field mappings, retrieving information from the record created in NetSuite back into OpenAir.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	**OpenAir Record Type [table]**	**Mapping Group [Integration point]**
Invoice [Invoice, TransactionSearch (pullback)] Schema: Invoice (2020.1) | Invoice (2023.1) XSD: tranSales (2020.1) | tranSales (2023.1) XSD (pullback): TransactionSearchRowBasic — platformCommon (2020.1) | platformCommon (2023.1)	Invoice [invoice] Data Dictionary: invoice	Invoice ▪ [INVOICE_TO_INVOICE] ▪ Pullback row [NS_INVOICE_TO_OA_INVOICE]

Default and Custom Filters

By default, invoice records are exported from OpenAir to NetSuite if the following conditions are met:

- The invoice is approved in OpenAir. Invoice approvals must be enabled for your OpenAir account.
- The invoice was not already exported — Invoice.exported is empty.
- The **NetSuite Invoice ID** [netsuite_invoice_id] custom field is empty OR the **Allow objects with a NetSuite ID to be exported during integration run** box is checked on the OpenAir NetSuite Connector administration form. See Allow objects with a NetSuite ID to be exported during integration run.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Invoice mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

⚠ **Important:** Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

See the OpenAir Help Center for detailed information about additional field mapping definitions for the Invoice mapping group tab that may be required or useful in some cases. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Project Mapping Group

Use the Project mapping group to include project-specific information from the project record in OpenAir in the invoice header information on the NetSuite invoice transaction record.

⚠ **Important:** Review the following guidelines:

- To include project-specific information in the exported invoice header information, each invoice in OpenAir must relate to one project only. To restrict each invoice to one project only in OpenAir, check the **One project per invoice** box in Administration > Application Settings > Invoices > Other settings.
- If you add custom mapping definitions in the Project mapping group tab for the NetSuite fields that are also mapped in the Invoice mapping group tab, the field mapping definition in the Project mapping group tab takes precedence, any value set for that NetSuite field using the Project field mapping definition overrides the value set for that NetSuite field using the Invoice field mapping definition.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — There are no default field mapping definitions for this mapping group.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Invoice [Invoice] Schema: Invoice (2020.1) | Invoice (2023.1) XSD: tranSales (2020.1) | tranSales (2023.1) XSD (pullback): TransactionSearchRowBasic — platformCommon (2020.1) | platformCommon (2023.1)	Project [project] Data Dictionary: project	Project [INVOICE_PROJECT]

Default Field Mappings

None

Additional Field Mappings

> **Important:** Review the following guidelines
>
> - You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
> - For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
> - Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

Review the following guidelines:

- Field mapping definitions on this tabs are ignored unless the **One project per invoice** box is checked in the Invoices application settings in OpenAir (Administration > Application Settings > Invoices > Other settings).

Invoice Line Mapping Group

Use the Invoice Line mapping group to export information from charges on the OpenAir invoice other than expense charges or product charges associated with a billable purchase item to the corresponding NetSuite invoice line items listed on the Items subtab.

The Invoice Line mapping group is NOT used in the following cases:

- The charge is an expense charge (type E) that was created automatically from a receipt in OpenAir unless he **Disable expense report references on invoices in NetSuite integration** box is checked on the Invoices (Invoices) export workflow settings form. See Disable expense report references on invoices in NetSuite integration. The Expense Line mapping group is used in that case — see Expense Line Mapping Group.
- The charge is an expense charge (type E) that was created manually in OpenAir and does not reference a **Service**.

> **Important:** Expense charges created manually in OpenAir also export as line items listed on the Items subtab using the Invoice Line mapping group if a **Service** is selected for those expense charges. Expense charges created manually are not exported otherwise.

- The charge is a product charge (type P) that was created from a purchase item imported from a NetSuite vendor bill into OpenAir using the POs (Vendor Bills) import workflow and all the following conditions are met:
  - The **Use Service item instead of Product item during invoices export for purchase charges** box is NOT checked on the Invoices (Invoices) import workflow settings form. See Use Service item instead of Product item during invoices export for purchase charges
  - The **Billable** box is checked on the vendor bill line item in NetSuite and the **NetSuite vendor bill line billable** [`netsuite_vb_billable`] box is checked on the corresponding purchase item in OpenAir.
  - The **Disable vendor bill linkage on invoices in NetSuite integration** box is NOT checked on the Invoices (Invoices) import workflow settings form. See Disable vendor bill linkage on invoices in NetSuite integration.

  The Bill Line mapping group is used in that case — see Bill Line Mapping Group.

> **Tip:** You can use the Purchase Line mapping group to include information from the original purchase order line item or vendor bill line item in NetSuite that was imported into OpenAir as purchase item on the invoice line item in NetSuite. See Purchase Line Mapping Group

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information exported by default includes the internal ID of the NetSuite item associated with the service or product associated with the charge, quantity, rate, total amount, and internal ID of the NetSuite project associated with the charge.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [`SOAP object`]	OpenAir Record Type [`table`]	Mapping Group [`Integration point`]
Line Item [`InvoiceItem`] Schema: InvoiceItem (2020.1)|InvoiceItem (2023.1) XSD: tranSales (2020.1)|tranSales (2023.1)	Charge [`slip`] Data Dictionary: slip	*Not visible on mapping definition table* Invoice Line [`INVOICE_LINE`]

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Invoice Line mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

> **Note:** Workflow settings and other integration workflows impact the way OpenAir charges export to line items on the invoice Items subtab in NetSuite. See Additional Guidelines About Settings Impacting Field Mapping.

Additional Guidelines About Settings Impacting Field Mapping

Review the following guidelines:

- If the **Combine charges by services on invoices** box is checked on the Invoices (Invoices) export workflow settings form, charges are combined by Invoice line items listed on the Items subtab are rolled up by project [`InvoiceItem.job`], if there are line items associated with more than one project in the invoice, and by service item [`InvoiceItem.item`] under each project. Each line item on the NetSuite invoice Items sublist combine multiple charges on the OpenAir invoice with the following summary information:
    - `InvoiceItem.quantity` — Total quantity across all charges for the same service item in the same project.
    - `InvoiceItem.amount` — Total amount across all charges for the same service item in the same project.
    - `InvoiceItem.rate` — Combined rate calculated based on the total amount and total quantity across all charges for the same service item in the same project.
    - `InvoiceItem.description` — Descriptions from all charges for the same service item in the same project, separated by a line break.
- If the Customer POs (Sales Orders) import workflow is enabled:
    - Charges referencing a customer PO are combined by sales order line item and not by services. This includes charges of type `T`, `F`, `O`. I also includes charges of type `E` if the **Enable expense charge linkage with sales order lines** box is checked on the Invoices (Invoices) export workflow settings form.
    - If the **Create billing rules for NetSuite discount and markup lines on sales orders** box is checked on the Invoices (Invoices) export workflow settings form:
        - If the charge is a header discount (the **NetSuite Sales Order Header Discount Customer PO** [`Customerpo.netsuite_sales_order_header_discount_po`] box is checked on the customer PO record), the charge information is used to set the **Discount Item** [`Invoice.discountItem`] and **Discount Rate** [`Invoice.discountRate`] on the invoice header.
        - If the charge is a line discount (the **NetSuite Sales Order Line Discount Customer PO** [`Customerpo.netsuite_sales_order_line_discount_po`] box is checked on the customer PO record), the **Quantity** `InvoiceItem.quantity` is removed.
    - If the **Create billing rules for NetSuite discount and markup lines on sales orders** box is checked on the Invoices (Invoices) export workflow settings form, additional discount or markup lines are created then rolled up together with charges referencing a customer PO by sales order line item.
    - The **Quantity** and **Rate** on the invoice line item are adjusted to ensure that the referenced sales order line can be closed. This takes into account any charges associated with the customer PO that were already exported.
    - If the **Adjust quantity based on % invoiced for time charges related to NetSuite sales order lines with quantities that do not reflect hours** box is checked on the Invoices (Invoices) export workflow settings form and the **Calculate Quantity From % Invoiced** [`Customerpo.netsuite_sales_order_quantity_is_not_hours`] box is checked on the Customer PO record, the **Quantity** and **Rate** on the invoice line item are calculated as a percentage of the amount on the sales order line. See Adjust quantity based on % invoiced for time charges related to NetSuite sales order lines with quantities that do not reflect hours.

- If the **Always respect the NetSuite sales order customer and project when exporting invoices** box is checked on the Invoices (Invoices) export workflow settings form, the `InvoiceItem.job` mapping is not active (the information is copied from the sales order instead). See Always respect the NetSuite sales order customer and project when exporting invoices.
- Invoice line items in NetSuite reference the sales order line item

NetSuite Field	Direction	OpenAir Field	Notes
`InvoiceItem.orderLine`	⬅	*Looked up OpenAir field value —* `Customerpo.netsuite_sales_order_line_id__c` for the customer PO with `Customerpo.id` equal to `Slip.customerpo_id`	*Not visible on mapping definition table* Active only if the Customer POs (Sales Orders) import workflow is enabled. The internal ID of the internal ID of the sales order line in NetSuite

- If the **Initialize invoices from corresponding sales orders when exporting invoices to NetSuite** box is checked on the Invoices (Invoices) export workflow settings form, OpenAir NetSuite Connector retrieves invoice information sourced from the sales orders in NetSuite and include this information when exporting invoices from OpenAir to NetSuite. This is similar to the information sourced from the sales order when you click Next Bill to create an invoice from a sales order in the NetSuite UI. See Initialize invoices from corresponding sales orders when exporting invoices to NetSuite.
- If the sales order line item references an item group instead of an item, the invoice item line also references this item group in `InvoiceItem.item`.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Expense Line Mapping Group

Use the Expense Line mapping group to export information on the expense charges created automatically from approved billable expense receipts and included on the invoice in OpenAir to the corresponding expense line items listed on the NetSuite invoice Billable Expenses subtab.

This mapping group is used to export each expense charge as a billable expense line item on the invoice transaction record in NetSuite when the following conditions are met:

- The expense report containing the billable receipt was exported from OpenAir to NetSuite as an expense reports or vendor bill using the Expenses Reports (Expenses Reports) export workflow. See Expense Reports (Expense Reports) Export Workflow.
- The line item references the NetSuite expense report or vendor bill. Expense charges export as line items listed on the Items subtab on the NetSuite invoice and the Invoice Line mapping group is used otherwise.

⚠ **Important:** The field mappings on the Expense Line mapping group tab are not used for expense charges created manually in OpenAir also export. The Invoice Line mapping group is used instead if a **Service** is selected on the expense charge. See Invoice Line Mapping Group.

Expense charges created manually in OpenAir are not exported to NetSuite otherwise.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information exported by default includes the internal ID of the NetSuite expense report corresponding to the expense report containing the receipt associated with the expense charge in OpenAir, the NetSuite expense line ID, and the amount.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Expense Line Item [InvoiceExpCost] Schema: InvoiceExpCost(2020.1)|InvoiceExpCost(2023.1) XSD: tranSales (2020.1)|tranSales (2023.1)	Charge [slip] of type E (Expense item) or M (Mileage expense item) Data Dictionary: slip	Expense Line [INVOICE_EXPENSE_LINE]

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Expense Line mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Purchase Line Mapping Group

Use the Purchase Line mapping group to export information on the purchase items imported from NetSuite purchase order line item and associated with the product charges (type P) included on the invoice in OpenAir to corresponding NetSuite invoice line items listed on the Items subtab.

The Purchase Line mapping group is NOT used if the product charge was created from a purchase item imported from a NetSuite vendor bill into OpenAir using the POs (Vendor Bills) import workflow and all the following conditions are met:

- The **Use Service item instead of Product item during invoices export for purchase charges** box is NOT checked on the Invoices (Invoices) import workflow settings form. See Use Service item instead of Product item during invoices export for purchase charges
- The **Billable** box is checked on the vendor bill line item in NetSuite and the **NetSuite vendor bill line billable** [netsuite_vb_billable] box is checked on the corresponding purchase item in OpenAir.
- The **Disable vendor bill linkage on invoices in NetSuite integration** box is NOT checked on the Invoices (Invoices) import workflow settings form. See Disable vendor bill linkage on invoices in NetSuite integration.

The Vendor Bill Line mapping group is used in that case — See Vendor Bill Line Mapping Group.

Otherwise the Purchase Line mapping group is used when the charge is a product charge (type P) that was created from a non-billable purchase item imported from a NetSuite purchase order or vendor bill into OpenAir using the POs (Purchase Orders) or POs (Vendor Bills) import workflows.

Tip: You can use this mapping group to include information from the original purchase order line item or vendor bill line item in NetSuite that was imported into OpenAir as purchase item on the invoice line item in NetSuite.

Important: If you add custom mapping definitions in the Purchase Line mapping group tab for the NetSuite fields that are also mapped in the Invoice Line mapping group tab, the field mapping definition in the Purchase Line mapping group tab takes precedence, any value set for that NetSuite field using the Purchase Line field mapping definition overrides the value set for that NetSuite field using the Invoice Line field mapping definition.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — There are no default field mapping definitions for this mapping group.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Line Item [InvoiceItem] Schema: InvoiceItem (2020.1)|InvoiceItem (2023.1) XSD: tranSales (2020.1)|tranSales (2023.1)	Purchase item [purchase_item] Data Dictionary: purchase_item	Purchase Line [INVOICE_PURCHASE_LINE]

Default Field Mappings

None

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Bill Line Mapping Group

Use the Bill Line mapping group to export information on the product charges created automatically from purchase items and included on the invoice in OpenAir to the corresponding billable line items listed on the Billable Items subtab on the NetSuite invoice.

This mapping group is used when the product charges are created from purchase items imported from NetSuite vendor bills into OpenAir using the POs (Vendor Bills) import workflow and all the following condition are met.

- The charge is a product charge (type P) that was created from a purchase item imported from a NetSuite vendor bill into OpenAir using POs (Vendor Bills) import workflows. See POs (Vendor Bills) Import Workflow.
- The **Use Service item instead of Product item during invoices export for purchase charges** box is NOT checked on the Invoices (Invoices) import workflow settings form. See Use Service item instead of Product item during invoices export for purchase charges
- The **Billable** box is checked on the vendor bill line item in NetSuite and the **NetSuite vendor bill line billable** [netsuite_vb_billable] box is checked on the corresponding purchase item in OpenAir.
- The **Disable vendor bill linkage on invoices in NetSuite integration** box is NOT checked on the Invoices (Invoices) import workflow settings form. See Disable vendor bill linkage on invoices in NetSuite integration.

Otherwise, the Invoice Line mapping group is used to export information from product charges in OpenAir to the corresponding line items listed on the Items subtab on the NetSuite invoice. See Invoice Line Mapping Group.

✓ **Tip:** You can use the Vendor Bill Line mapping group to include information from the original vendor bill line item in NetSuite that was imported into OpenAir as purchase item on the invoice billable line item in NetSuite. See Vendor Bill Line Mapping Group

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information exported by default includes
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	**OpenAir Record Type [table]**	**Mapping Group [Integration point]**
Billable Line Item [InvoiceItemCost] Schema: InvoiceItemCost(2020.1)|InvoiceItemCost(2023.1) XSD: tranSales (2020.1)|tranSales (2023.1)	Charge [slip] Data Dictionary: slip	Bill Line [INVOICE_BILLABLE_ITEM_LINE]

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Bill Line mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

> **Important:** Review the following guidelines
>
> - You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
> - For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
> - Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Vendor Bill Line Mapping Group

Use the Vendor Bill Line mapping group to export information on the purchase items imported from NetSuite vendor bill line item and associated with the product charges (type P) included on the invoice in OpenAir to corresponding billable line items listed on the Billable Items subtab on the NetSuite invoice.

This mapping group is used when the product charges are created from purchase items imported from NetSuite vendor bills into OpenAir using the POs (Vendor Bills) import workflow and all the following condition are met.

- The charge is a product charge (type P) that was created from a purchase item imported from a NetSuite vendor bill into OpenAir using POs (Vendor Bills) import workflows. See POs (Vendor Bills) Import Workflow.
- The **Use Service item instead of Product item during invoices export for purchase charges** box is NOT checked on the Invoices (Invoices) import workflow settings form. See Use Service item instead of Product item during invoices export for purchase charges
- The **Billable** box is checked on the vendor bill line item in NetSuite and the **NetSuite vendor bill line billable** `[netsuite_vb_billable]` box is checked on the corresponding purchase item in OpenAir.
- The **Disable vendor bill linkage on invoices in NetSuite integration** box is NOT checked on the Invoices (Invoices) import workflow settings form. See Disable vendor bill linkage on invoices in NetSuite integration.

Otherwise, the Purchase Line mapping group is used to export purchase item information associated with product charges to the corresponding line items listed on the Items subtab on the NetSuite invoice. See Purchase Line Mapping Group.

> **Tip:** You can use this mapping group to include information from the original vendor bill line item in NetSuite that was imported into OpenAir as purchase item on the invoice billable line item in NetSuite.

> **Important:** If you add custom mapping definitions in the Vendor Bill Line mapping group tab for the NetSuite fields that are also mapped in the Bill Line mapping group tab, the field mapping definition in the Vendor Bill Line mapping group tab takes precedence, any value set for that NetSuite field using the Vendor Bill Line field mapping definition overrides the value set for that NetSuite field using the Bill Line field mapping definition.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — There are no default field mapping definitions for this mapping group.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	**OpenAir Record Type [table]**	**Mapping Group [Integration point]**
Billable Line Item [InvoiceItemCost] Schema: InvoiceItemCost(2020.1)|InvoiceItemCost(2023.1) XSD: tranSales (2020.1)|tranSales (2023.1)	Purchase item [purchase_item] Data Dictionary: purchase_item	Bill Line [INVOICE_PURCHASE_VB_LINE]

Default Field Mappings

None

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Group by Time, Group by Other, and Group by Fee Mapping Groups

Use the Group by Time, Group by Other, and Group by Fee mapping groups to map information from time (“hourly” or “other” rate) and fixed fee (“flat rate”) charges on the OpenAir invoice as description line items on the NetSuite invoice.

To do so, add custom mapping definitions to the mapping groups corresponding to each of the charge types you want to export to description line items.

- Group by Time — Custom field mapping definitions for hourly rate time charges (charge type T) information exported to description line items.
- Group by Other — Custom field mapping definitions for other time charges (charge type 0) information exported to description line items.
- Group by Fee — Custom field mapping definitions for flat rate charges (charge type F) information exported to description line items.

Information from each charge may be mapped to one description line or non-expense charges are rolled up by service items if the **Combine charges by service on invoices** box is checked on the Invoices (Invoices) workflow settings form. If the **Combine charges by service on invoices** box is checked, charges that are not linked to a sales order are also included in the above description item mappings based on the slip type.

> **Important:** The Group by Time, Group by Other, and Group by Fee mapping groups are used only when al the following conditions are met:
>
> - The Customer POs (Sales Orders) import workflow is enabled and in use. See Customer POs (Sales Orders) Import Workflow.
> - Some charges on the invoice (but not necessarily all charges) are associated with a customer PO imported from a NetSuite sales order line. Information from all OpenAir time (hourly rate), time (other) or flat rate charges, including charges that do not reference a customer PO imported from a NetSuite sales order line, exports to description line items based on the charge type.
> - The **Enable charges to map to NetSuite description items** box is checked on the Invoices (Invoices) export workflow settings form. See Enable charges to map to NetSuite description items

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — There are no default field mapping definitions for this mapping group.
- Additional Field Mappings — Best practice guidelines for custom field mapping definitions.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Description Line Item [InvoiceItem]	Charge [slip] of type:	- Group by Time [INVOICE_LINE_TIME_GROUP_BY]

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Schema: InvoiceItem (2020.1)|InvoiceItem (2023.1) XSD: tranSales (2020.1)|tranSales (2023.1)	■ T — Hourly rate time bill (Group by Time) ■ 0 — Other rate (day rate, for example) time bill (Group by Other) ■ F — Flat rate time bill (Group by Fee) Data Dictionary: slip	■ Group by Other [INVOICE_LINE_OTHER_GROUP_BY] ■ Group by Time [INVOICE_LINE_FEE_GROUP_BY]

Default Field Mappings

None

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

You can use advanced mappings to export detailed information from OpenAir time (hourly rate), time (other) or flat rate charges to NetSuite as description line items. See Advanced Mapping and Filter Notation Reference.

You can group or sort charges when exporting information from OpenAir charges to description line items on NetSuite invoices using field declaration. See Field Declarations (OA_FIELDS_GROUP_BY and OA_FIELDS_SORT_BY). If charges are grouped, the description line shows summary information for all the charges in each group such as calculated total monetary amounts and number of hours, for example.

Use \\n to insert a line break in the description.

Important: Conditional branching is not supported for advanced mappings exporting information to description line items (Group by Time, Group by Other, and Group by Fee mapping groups).

Refer to the following examples of advanced mapping to export charge information to description line item for each supported charge type.

Example 4. Group by Time Advanced Mapping

In this example, a description line item in NetSuite is created for each “hourly” time charge in OpenAir generated by a time billing rule. Note that a description lines may be created for each sales order line, and combine all charges associated with that sales order line, if **Combine charges by services on invoices** box is checked on the workflow settings form. Description line items are sorted by the name of the employee and include information from the OpenAir charges including the name of the employee, date, hours, rate, notes, and total amount.

```
OA_FIELDS               user_id, notes, hour, date, total, rate, id
NS_FIELDS              description
OA_FIELDS_SORT_BY      user.name

description             'Name:', lookup=user_id:lookup_table=user:lookup_by=id:lookup_return=name,
                       '\\nDate:', date, '\\nHours:', hour, '\\nRate:', rate,'\\nNotes:', notes, '\\nAmount:',
                       total, '\\n\\n\\n'
```

Example 5. Group by Other Advanced Mapping

In this example, information about “other” time charges on the OpenAir invoice (for example, time charge using a day rate instead of an hourly rate) is grouped by project task. A description line item is created for each task on the NetSuite invoice with summary information combining all charges associated with the task. Summary information includes the service name, quantity, rate, total amount across all charges for the task, and project task name.

```
OA_FIELDS               notes, cost, id, total, quantity, project_task_id
NS_FIELDS               description
OA_FIELDS_GROUP_BY      project_task_id
OA_FIELDS_SORT_BY      project_task.name

description            'Service: ', lookup=category_id:lookup_table=category:lookup_by=id:lookup_return=name, '\\nQty: ',
                     quantity, '\\nRate: ', cost, '\\nAmount: ', total, '\\nProject Task Name: ',
                           lookup=project_task_id:lookup_table=project_task:lookup_by=id:lookup_return=name, '\\n\\n\\n'
```

Example 6. Group by Fee Advanced Mapping

In this example, information about “flat rate” charges on the OpenAir invoice is grouped by project task A description line item is created for each task on the NetSuite invoice with summary information combining all charges associated with the task. Summary information includes the name of the project task (milestone), the user name, and the total for that milestone.

```
OA_FIELD                  project_task_id, total, user_id
NS_FIELDS               description
OA_FIELDS_GROUP_BY      project_task_id
OA_FIELDS_SORT_BY      user.name
description            'Milestone: ', lookup=project_task_id:lookup_table=project_task:lookup_by=id:lookup_return=name,
                           'Name:', lookup=user_id:lookup_table=user:lookup_by=id:lookup_return=name, '\\nAmount:', total, '\\n\\n
\\n'
```

Time Entries (Time) Export Workflow

Note: This help topic is a placeholder only. Content will be added gradually to these pages as the content and structure of this guide are reviewed. For more information about the Time Entries (Time) integration workflows, please refer to the following chapters:

- Configuring Optional Features > Configuring the Time Entries Export Integration
- Using the Integration
- Data Mapping in the Integration

Time Entries (Time) Export Workflow Settings

You can control Time Entries (Time) export workflow settings from the NetSuite Connector UI. On the NetSuite Connector Mapping screen, select the Time Entries (Time) export workflow and click **Settings** in the top right corner to view the workflow settings form.

The following settings are available for the Time Entries (Time) export workflow:

- Always attempt to send all viable time entries
- Do not require a service on time entry export to NetSuite
- Enable loaded cost export with the timesheet export
- Map cost center on loaded cost export with the timesheet export
- Show last NetSuite integration error on timesheets

Always attempt to send all viable time entries

This option allows you to integrate all viable time entries to NetSuite when a timesheet contains time entries which should not be exported.

When this feature is enabled, all time entries in a timesheet are checked for export to NetSuite, and if they are supposed to be integrated to NetSuite, the integration tries to send them.

When disabled, the timesheet integration ends after the first time entry which should not go to NetSuite is found (for example, if a NS project ID or category ID are missing).

Do not require a service on time entry export to NetSuite

If you use the time entry integration, time entries export with or without an activity or service item.

Enable loaded cost export with the timesheet export

Enable the export of loaded costs on each time entry that is exported to NetSuite.

This feature uses the standard timesheet export. Please refer to the documentation for the timesheet export to understand that feature.

This feature does not account for primary loaded cost normalization feature. If the primary loaded cost is normalized after the export of time entries and costs to NetSuite, the normalized costs will not be reflected in NetSuite.

Up to three loaded cost levels are supported. The integration will export as many as have been enabled in the account.

Map cost center on loaded cost export with the timesheet export

Map OpenAir cost center to the custom field “OpenAir: Cost Centers” on the NetSuite time entry. This setting requires the NetSuite custom record type, “OpenAir: Cost Centers” to be installed into the NetSuite account as part of the standard integration bundle — NetSuite OpenAir SRP Integration bundle 369637 (previously 2851).

Cost centers associated with the time entry can be used to source additional information needed in NetSuite to create journal entries.

Show last NetSuite integration error on timesheets

This feature lets users with the relevant role permission identify common issues preventing the export of timesheets to NetSuite and take any corrective actions within their control.

If this setting is enabled and if an error occurs when OpenAir attempts to export a timesheet to NetSuite, OpenAir stores the integration error message for that record in the **Last NetSuite integration error** custom field.

- Only the error message from the last integration run is stored, if any. OpenAir clears the **Last NetSuite integration error** field if the timesheet exports successfully to NetSuite.
- The **Last NetSuite integration error** field shows:
    - As a read only field on the properties form for Timesheets.
    - As a column on Timesheets list views.
- When **Show last NetSuite integration error on timesheets** is enabled, the following integration role permission becomes available in Administration > Global Settings > Users > Roles: **Allow employee to view last NetSuite error on timesheets**.
- Only administrators and employees with the **Allow employee to view last NetSuite error on timesheets** role permission can see the **Last NetSuite integration error**.
- The error message displayed on the properties form or list view is the same as:
    - The message displayed on screen when running the integration manually.
    - The message recorded on the integration run log accessible from the Status screen. See Reviewing the Integration Run History and Status.

Note: Review the following guidelines:

- If you enable **Show last NetSuite integration error on timesheets** for the first time, OpenAir creates a custom field associated with timesheets to store the last NetSuite integration error message.
- The **Last NetSuite integration error** field is not cleared if you disable **Show last NetSuite integration error on timesheets**. Employees with the relevant role permission will continue to see the last integration error recorded before you disabled the workflow setting. To hide the last integration error recorded, you must disable the role permission before disabling the workflow setting.

Recognition Transactions (Revenue Recognition Transactions) Export Workflow

Note: This help topic is a placeholder only. Content will be added gradually to these pages as the content and structure of this guide are reviewed. For more information about the Recognition Transactions (Revenue Recognition Transactions) integration workflows, please refer to the following chapters:

- Configuring Optional Features > Configuring Revenue Recognition
- Using the Integration
- Data Mapping in the Integration

Recognition Transactions (Revenue Recognition Transactions) Export Workflow Settings

You can control Recognition Transactions (Revenue Recognition Transactions) export workflow settings from the NetSuite Connector UI. On the NetSuite Connector Mapping screen, select the Recognition Transactions (Revenue Recognition Transactions) export workflow and click **Settings** in the top right corner to view the workflow settings form.

The following settings are available for the Recognition Transactions (Revenue Recognition Transactions) export workflow:

- Disable NetSuite revenue recognition feature check for recognition transaction integration
- NetSuite recognition transaction batch size (maximum of 100)

Disable NetSuite revenue recognition feature check for recognition transaction integration

Check the **Disable NetSuite revenue recognition feature check for recognition transaction integration** box to integrate revenue recognition transactions to NetSuite without verifying whether the Revenue Recognition feature is enabled in NetSuite.

By default, when integrating Revenue Recognition transactions, OpenAir NetSuite Connector first verifies that the Revenue Recognition feature is enabled in NetSuite before sending recognition transactions to NetSuite. However, when using the Advanced Revenue Management (ARM) feature in NetSuite, the Revenue Recognition feature must be disabled.

NetSuite recognition transaction batch size (maximum of 100)

This feature lets you integrate OpenAir revenue recognition transactions to NetSuite recognition transaction in batches instead of one by one. Set a batch size up to a maximum of 100 records. The feature will significantly reduce the run time of the Recognition Transactions (Revenue Recognition Transactions) workflow if you export large numbers of journal entries to NetSuite.

See also Exporting Revenue Recognition Transactions to NetSuite in Batches.

⚠ **Important:** If you have scripts running when records are created in your NetSuite account, reduce the batch size to avoid your scripts timing out. Test this feature in your sandbox account to determine the best batch size for your NetSuite customizations.

ⓘ **Note:** Exporting revenue recognition transaction in batches may delay the first web request. This is expected behavior. The subsequent web requests will follow immediately after the first one.

Loaded Cost (Custom Record) Export Workflow

ⓘ **Note:** This help topic is a placeholder only. Content will be added gradually to these pages as the content and structure of this guide are reviewed. For more information about the Loaded Cost (Custom Record) integration workflows, please refer to the following chapters:

- Configuring Optional Features
- Using the Integration
- Data Mapping in the Integration

Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Export Workflow

The Send OpenAir Credit (negative invoice) as NetSuite Credit Memo export workflow lets you create credit invoices in OpenAir, and then export the credit invoices to NetSuite as credit memos.

In OpenAir, you can:

- Create credit invoices against an approved invoice using the Credit and Rebill Invoices optional feature. If the invoice was previously exported to NetSuite, when exporting the credit invoice to NetSuite, the resulting credit memo is applied to the corresponding invoice in NetSuite. For more information about the Credit and Rebill Invoices optional feature, see Optional Features.

- Create self-contained credit invoices — credit invoices that are not linked to an invoice in OpenAir. When exporting the credit invoice to NetSuite, the resulting credit memo is not associated with any invoice in NetSuite.

> **Note:** You can also create credit memo records in NetSuite and apply to invoices previously exported from OpenAir to NetSuite, and then import the credit memos into OpenAir as credit invoices with negative charges. See Credits (Applied Credit Memos) Import Workflow.

You can export both credit invoice and project header information to the credit memo transaction record. This allows you to export project-specific information stored in standard or custom OpenAir fields, and then to display this information on the credit memo header in NetSuite and use this information in reports.

You can import the tax amounts calculated for the credit memo in NetSuite back onto the OpenAir credit invoice, including GST and PST taxes. Set up the Advanced Taxes accounting feature in NetSuite and enable Tax Options in OpenAir. Set up nexuses in NetSuite for the countries or locations from which taxes will be paid and create or identify the tax codes or tax groups for each nexus. When you export a credit invoice to NetSuite, taxes on the total amount are calculated in NetSuite and import back to the credit invoice in OpenAir.

> **Important:** If the SuiteTax is enabled for your NetSuite account, NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) to version 2.11 or later is required in order for the tax amount calculated in NetSuite to be imported into OpenAir. With earlier versions, the tax amount calculated in NetSuite with SuiteTax cannot be imported back into OpenAir.
>
> Note that SuiteTax compatibility testing is currently in progress. You should test the NetSuite <> OpenAir integration with the SuiteTax feature on sandbox accounts before you enable the feature in your NetSuite production account. See Known Limitations of SuiteTax (external link to NetSuite SuiteAnswers). See also Legacy Tax or SuiteTax.

For information about workflow specific settings, see Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Workflow Settings.

For information about field mapping definitions for this workflow, see Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Field Mapping Definitions.

> **Important:** Changes made to the credit records in OpenAir after the initial import are not exported to NetSuite. The Send OpenAir Credit (negative invoice) as NetSuite Credit Memo export workflow an be used to create new credit memos in NetSuite but not to update already integrated credit memos.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
✓	✓	✓	✓	✓	

- Run the integration manually, use a scheduled integration run to create a credit memo record and apply the credit memo to an invoice in NetSuite after you create a credit invoice in OpenAir. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.

- Setup the real-time integration to create the NetSuite credit memo invoice and apply it it to the invoice in NetSuite when the credit invoice is approved in OpenAir. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir credit invoice record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Requirements and Downstream Dependencies

Matching customer, project and service (service item) records must exist in NetSuite and OpenAir before you can export credit invoices from OpenAir to NetSuite as credit memos.

Invoice approvals must be enabled to export credit invoices from OpenAir to NetSuite. The Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos export workflow is not available otherwise.

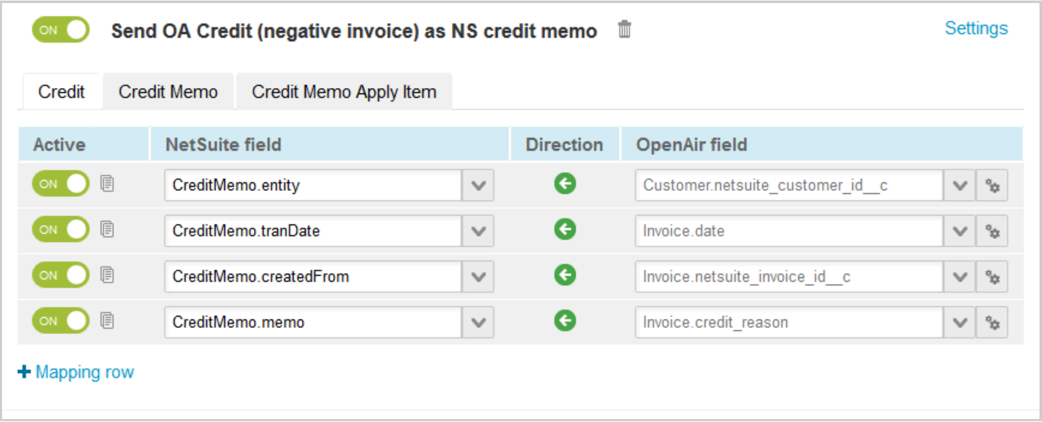
If you create credit invoice records using the Credit and Rebill Invoices optional feature in OpenAir, matching records for the original invoices that were credited (`Invoice.original_invoice_id`) must exist in both NetSuite and OpenAir before you can export the credit invoice. For more information about the Credit and Rebill Invoices optional feature, see Optional Features.

Setting Up the Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Export Workflow

Use the following steps to set up the Send OpenAir Credit (negative invoice) as NetSuite Credit Memo export workflow.

To set up the Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos export workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector, click the Tips menu button, then NetSuite Connector administration form.
2. Check the **Enable NetSuite credit memo integration for invoices** box under the Additional integrations form section.
3. Click **Save**.
4. (Optional) If you want to export OpenAir project header information to the NetSuite credit memo, do the following:
   a. In OpenAir, go to Administration > Application Settings > Invoices > Other Settings.
   b. Check the **One project per invoice** box.
   c. Click **Save**.
5. Go to Administration > NetSuite Connector > Mapping
6. Click **New workflow** then **Send OA Credit (negative invoice) as NS Credit Memo**.
7. Click **Send OA Credit (negative invoice) as NS Credit Memo** in the list of integration workflows.
8. Click **Workflow settings** to view and modify the settings specific to this workflow. For more information about the workflow settings, see Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Workflow Settings.
9. Modify the field mapping definition table for the Send OpenAir Credit (negative invoice) as NetSuite Credit Memo import workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Field Mapping Definitions.
10. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON **Send OA Credit (negative invoice) as NS credit memo** Settings

Credit | Credit Memo | Credit Memo Apply Item

Active	NetSuite field	Direction	OpenAir field
ON	CreditMemo.entity	←	Customer.netsuite_customer_id__c
ON	CreditMemo.tranDate	←	Invoice.date
ON	CreditMemo.createdFrom	←	Invoice.netsuite_invoice_id__c
ON	CreditMemo.memo	←	Invoice.credit_reason

+ Mapping row

11. Click **Save**.
12. (Optional) Add the Send OpenAir Credit (negative invoice) as NetSuite Credit Memo export workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Workflow Settings

You can control Send OpenAir Credit (negative invoice) as NetSuite Credit Memo export workflow settings from the NetSuite Connector UI. On the NetSuite Connector Mapping screen, select the Send OpenAir Credit (negative invoice) as NetSuite Credit Memo export workflow and click **Settings** to view the workflow settings form.

The following settings are available for the Send OA Credit (negative invoice) export workflow:

- Import taxes applied to exported OA Credit (negative invoice)

Import taxes applied to exported OA Credit (negative invoice)

Check this box to import the total tax calculated in NetSuite for the credit memo back into OpenAir. When you export credit invoices from OpenAir to NetSuite as credit memos, NetSuite calculates the tax, OpenAir NetSuite Connector imports the calculated total tax back into OpenAir, and the OpenAir credit invoice includes the total tax calculated in NetSuite.

> **Important:** The **Import taxes applied to exported invoices** box must also checked on the settings form for the Invoices (Invoices) export workflow and invoice tax import must be set up. See Import taxes applied to exported invoices.

Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos Field Mapping Definitions

The Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos export workflow includes the following mapping groups:

- Credit [Integration point: `CREDIT_TO_CREDIT_MEMO`] — See Credit Mapping Group.
- Credit Memo [Integration point: `CREDIT_MEMO_ITEM`] — See Credit Memo Mapping Group.

- Credit Memo Apply Item [Integration point: CREDIT_MEMO_APPLY_ITEM] — See Credit Memo Apply Item Mapping Group.
- Credit [Integration point: CREDIT_PROJECT] — See Project Mapping Group.

Credit Mapping Group

Use the Credit mapping group to export information on credit (negative invoice) records in OpenAir to credit memo transaction records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes the internal IDs of the customer and the invoice being credited in NetSuite, credit reason, and date. Filters are supported.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Credit Memo [CreditMemo] Schema: CreditMemo (2020.1) | CreditMemo (2023.1) XSD: tranCust (2020.1) | tranCust (2023.1) XSD (pullback): TransactionSearchRowBasic — platformCommon (2020.1) | platformCommon (2023.1)	Invoice [invoice] with negative total Data Dictionary: invoice	Credit Memo [CREDIT_TO_CREDIT_MEMO]

Default and Custom Filters

By default, credit (negative invoice) records are exported from OpenAir to NetSuite if the following conditions are met:

- The credit (negative invoice) is approved in OpenAir. Invoice approvals must be enabled for your OpenAir account.
- The invoice must be a credit invoice — Invoice.total is negative.
- The credit (negative invoice) was not already exported — Invoice.exported is empty.
- The **NetSuite Credit Memo ID** [netsuite_credit_memo_invoice_id] custom field is empty OR the **Allow objects with a NetSuite ID to be exported during integration run** box is checked on the OpenAir NetSuite Connector administration form. See Allow objects with a NetSuite ID to be exported during integration run.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Credit mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

⚠ **Important:** Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Credit Memo Mapping Group

Use the Credit Memo mapping group to export information about charges on credit (negative invoice) records in OpenAir to credit memo line items in NetSuite.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes the internal IDs of the project and purchase or service item in NetSuite, time and rate or quantity and cost, and total.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Credit Memo [CreditMemoItem] Schema: CreditMemoItem (2020.1)|CreditMemoItem (2023.1) XSD: tranCust (2020.1)|tranCust (2023.1)	Charge [slip] Data Dictionary: slip	Negative Charge [CREDIT_LINE]

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Credit Memo mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

⚠ **Important:** Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Credit Memo Apply Item Mapping Group

Use the Credit Memo Apply mapping group to export information about credit (negative invoice) records in OpenAir to a credit memo apply sublist line in NetSuite. This mapping group is used when the credit invoice record was created using the Credit and Rebill Invoices optional feature in OpenAir. In this case, the credit invoice references the original invoice that was credited (`Invoice.original_invoice_id`). If the credit is applied to an invoice that was previously exported to NetSuite, when exporting the credit invoice to NetSuite, the resulting credit memo is applied to the NetSuite invoice corresponding to the original OpenAir invoice that was credited. For more information about the Credit and Rebill Invoices optional feature, see Optional Features.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — Overview of default field mapping definitions including notes and best practice guidelines where relevant. Information imported by default includes the internal IDs of the project and purchase or service item in NetSuite, time and rate or quantity and cost, and total.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Credit Memo [CreditMemoApply] Schema: CreditMemoApply (2020.1)|CreditMemoApply (2023.1) XSD: tranCust (2020.1)|tranCust (2023.1)	Invoice [invoice] Data Dictionary: invoice	Credit Memo Apply [CREDIT_MEMO_APPLY]

Default Field Mappings

See the OpenAir Help Center for detailed information about default field mapping definition for the Credit Memo Apply Item mapping group tab. Note that you need to sign in to OpenAir before you can access the OpenAir Help Center.

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

No specific guidelines

Project Mapping Group

Use the Project mapping group to include project-specific information from the project record in OpenAir in the invoice header information on the NetSuite invoice transaction record.

Important: Review the following guidelines:

- To include project-specific information in the exported header information, each invoice and credit record in OpenAir must relate to one project only. To restrict each invoice and credit to one project only in OpenAir, check the **One project per invoice** box in Administration > Application Settings > Invoices > Other settings.
- If you add custom mapping definitions in the Project mapping group tab for the NetSuite fields that are also mapped in the Credit mapping group tab, the field mapping definition in the Project mapping group tab takes precedence, any value set for that NetSuite field using the Project field mapping definition overrides the value set for that NetSuite field using the Credit field mapping definition.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default Field Mappings — There are no default field mapping definitions for this mapping group.
- Additional Field Mappings — There are no specific guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Credit Memo [CreditMemo] Schema: CreditMemo (2020.1) | CreditMemo (2023.1) XSD: tranCust (2020.1) | tranCust (2023.1) XSD (pullback): TransactionSearchRowBasic — platformCommon (2020.1) | platformCommon (2023.1)	Project [project] Data Dictionary: project	Project [INVOICE_PROJECT]

Default Field Mappings

None

Additional Field Mappings

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

Review the following guidelines:

- Field mapping definitions on this tabs are ignored unless the **One project per invoice** box is checked in the Invoices application settings in OpenAir (Administration > Application Settings > Invoices > Other settings).

Recognition Rules (Custom Record: OpenAir: Revenue Rules for ARM) Export Workflow

Note: This help topic is a placeholder only. Content will be added gradually to these pages as the content and structure of this guide are reviewed. For more information about the Recognition Rules (Custom Record: OpenAir: Revenue Rules for ARM) integration workflows, please refer to the following chapters:

- Configuring Optional Features > OpenAir Revenue Recognition with NetSuite Advanced Revenue Management
- Using the Integration
- Data Mapping in the Integration

Recognition Transactions (Custom Record: OpenAir: Revenue Transactions for ARM) Export Workflow

Note: This help topic is a placeholder only. Content will be added gradually to these pages as the content and structure of this guide are reviewed. For more information about the Recognition Transactions (Custom Record: OpenAir: Revenue Transactions for ARM) integration workflows, please refer to the following chapters:

- Configuring Optional Features > OpenAir Revenue Recognition with NetSuite Advanced Revenue Management
- Using the Integration
- Data Mapping in the Integration

Update NetSuite Projects with OpenAir Project Data Export Workflow

The Update NetSuite Projects with OpenAir Project Data export workflow lets you update project records in OpenAir and use the NetSuite <> OpenAir integration to update the corresponding project record in NetSuite.

Note: You can use the Projects (Projects) import workflow to create new project records in NetSuite first and then import them into OpenAir. See Projects (Projects) Import Workflow.

You can also use the Projects (Projects) export workflow to create new project records in OpenAir first and then export them to NetSuite. See Projects (Projects) Export Workflow.

For information about workflow-specific settings, see Update NetSuite projects with OpenAir project data Export Workflow Settings.

For information about field mapping definitions for this workflow, see Update NetSuite Projects with OpenAir Project Data Field Mapping Definition.

Requirements and Downstream Dependencies

Matching project records must exist in both NetSuite and OpenAir before you can update projects in OpenAir and use the NetSuite <> OpenAir integration to update the corresponding project record in NetSuite.

To export updated information for projects with a given **Project stage** from OpenAir to NetSuite, the project stage record in OpenAir must be matched with a project status record in NetSuite.

Note: When you enable the Update NetSuite Projects with OpenAir Project Data export workflow, OpenAir NetSuite Connector adds the following custom fields associated with **Project stage** records in OpenAir:

- **NetSuite Status ID** [`netsuite_project_stage_id`] — Text field that holds the internal ID of the status record in NetSuite corresponding to the OpenAir project stage. You can match more than one project stage records in OpenAir with the same project status record in NetSuite.
- **Allow projects in this project stage to be updated in NetSuite** [`netsuite_push_project_stage_filter`] — Check the box to export updated information for projects with this project stage from OpenAir to NetSuite. Note that the workflow supports custom filters and that you should use a custom filter condition to update only projects at specific project stages. However the **Allow projects in this project stage to be updated in NetSuite** box method is still supported.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
✓	✓	✓ * Settings dependent	✓	✓	

- Run the integration manually, use a scheduled integration run to update the NetSuite records after you update the project records in OpenAir. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.

- Setup the real-time integration to update the NetSuite project records when the Project Properties form is saved in OpenAir. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir project record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Update NetSuite Projects with OpenAir Project Data Export Workflow

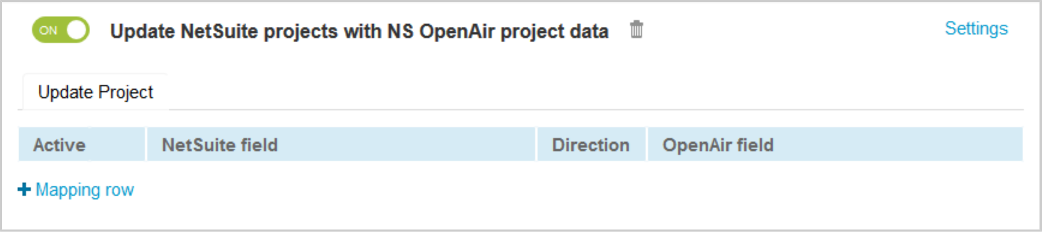
Use the following steps to set up the Update NetSuite Projects with OpenAir Project Data export workflow.

To set up the Update NetSuite Projects with OpenAir Project Data export workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector, click the Tips menu button, then NetSuite Connector administration form.
2. Check the **Update NetSuite Projects with OpenAir Project Data** box under the Exports form section.
3. (Optional) Check the **Exclude reporting values from Update NetSuite projects with OpenAir project data** box under the Performance features form section. You should check this box unless you need a wider range of OpenAir fields available for mapping and performance is not a concern for your integration. See Exclude reporting values from Update NetSuite projects with OpenAir project data.
4. Click **Save**.
5. Go to Administration > Application Settings > Projects > Project stages.
6. For each project stage, if you want updated information to be exported to NetSuite for projects with this project stage, do the following:
   a. Enter the **NetSuite Status ID** — the **Internal ID** of the corresponding project status record in NetSuite.

      > **Note:** You can use the same **NetSuite Status ID** for one or more project stages. A 1:1 relationship is not required between NetSuite statuses and OpenAir project stages.
      >
      > If you need to create new statuses in NetSuite, go to Setup > Accounting > Accounting Lists > New > Project Status.

   b. Check the **Allow projects in this project stage to be updated in NetSuite** box.
   c. Click **Save**.
7. Go to Administration > NetSuite Connector > Mapping.
8. Click **New workflow** then **Update NetSuite Projects with OpenAir Project Data**.
9. Click **Update NetSuite Projects with OpenAir Project Data** in the list of integration workflows.
10. Click **Workflow settings** to view and modify the settings specific to this workflow. For more information about the workflow settings, see Update NetSuite projects with OpenAir project data Export Workflow Settings.
11. Modify the field mapping definition table for the Update NetSuite Projects with OpenAir Project Data export workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Update NetSuite Projects with OpenAir Project Data Field Mapping Definition.
12. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



13. Click **Save**.
14. (Optional) Add the Update NetSuite Projects with OpenAir Project Data export workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Update NetSuite projects with OpenAir project data Export Workflow Settings

You can control Update NetSuite projects with OpenAir project data export workflow settings from the NetSuite Connector UI. On the NetSuite Connector Mapping screen, select the Update NetSuite projects with OpenAir project data export workflow and click **Settings** to view the workflow settings form.

The following settings are available for the Update NetSuite projects with OpenAir project data export workflow:

- NetSuite project update batch size (maximum of 20)

NetSuite project update batch size (maximum of 20)

Enter the number of projects to update in one batch. The value must be an integer between 1 and 20. OpenAir NetSuite Connector can send multiple project records in a single NetSuite SOAP web services request when exporting project information to update project records in NetSuite. Using a batch size of 20 or under can speed up the integration in some cases.

Update NetSuite Projects with OpenAir Project Data Field Mapping Definition

The Update NetSuite Projects with OpenAir Project Data export workflow includes one mapping group:

- Update Project [Integration point: `UPDATE_PROJECT_TO_JOB`] — See Update Project Mapping Group.

Update Project Mapping Group

Use the Update Project mapping group to map information on the project record in OpenAir with information on the project record in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.

- Default Field Mapping — There are no default field mapping definitions for this mapping group.
- Additional Field Mapping — There are no specific best practice guidelines for custom field mapping definitions. The mapping group does not support custom pullback field mappings, retrieving information from the record created in NetSuite back into OpenAir.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type `[SOAP object]`	**OpenAir Record Type** `[table]`	**Mapping Group** `[Integration point]`
Project [Job] Schema: Job (2020.1) | Job (2023.1) XSD: listRel (2020.1) | listRel (2023.1)	■ If the **Exclude reporting values from Update NetSuite projects with NS OpenAir project data** box is enabled on the OpenAir NetSuite Connector administration form: □ Record Type: Project [`project`] □ Data Dictionary: project ■ Otherwise, the OpenAir information is sourced from a detailed report. Refer to the UI for the list of report columns available for mapping. See also Exclude reporting values from Update NetSuite projects with OpenAir project data.	Update Project [UPDATE_PROJECT_TO_JOB]

Default and Custom Filters

By default, updated project information is exported from OpenAir to NetSuite if the following conditions are met:

- Matching project records exists in OpenAir and NetSuite.
- The **Allow projects in this project stage to be updated in NetSuite** `[netsuite_push_project_stage_filter]` box is checked on the project stage record associated with the OpenAir project.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Note: Only standard and custom fields for the Project [`project`] record type can be used in custom filters even if other fields (report values) are available for mapping.

If you want to update only projects at specific project stages, you should now use a custom filter condition instead of checking the **Allow projects in this project stage to be updated in NetSuite** [netsuite_push_project_stage_filter] box on the project stage records. However, the **Allow projects in this project stage to be updated in NetSuite** [netsuite_push_project_stage_filter] box method is still supported.

Default Field Mapping

None

Additional Field Mapping

Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

Advanced mappings are supported only if the **Exclude reporting values from Update NetSuite projects with OpenAir project data** box is checked under the Performance features section of the OpenAir NetSuite Connector administration form. See Exclude reporting values from Update NetSuite projects with OpenAir project data.

Update NetSuite Tasks with OpenAir Task Data

The Update NetSuite Tasks with OpenAir Task Data export workflow lets you update task records in OpenAir and use the NetSuite <> OpenAir integration to update the corresponding task record in NetSuite.

Note: You can use the Tasks (Project Tasks) import workflow to create and update project task records in NetSuite first and then import them into OpenAir or update the already imported task records in OpenAir. See Tasks (Project Tasks) Import Workflow.

For information about field mapping definitions for this workflow, see Update NetSuite Tasks with OpenAir Task Data Field Mapping Definitions.

Requirements and Downstream Dependencies

Matching task records must exist in both NetSuite and OpenAir before you can update project tasks in OpenAir and use the NetSuite <> OpenAir integration to update the corresponding project record in NetSuite.

Supported Features

Workflow Settings	Filters	Advanced Mapping	Real-Time	OpenAir Within NetSuite	Monitored Fields
	✓	✓ * Settings dependent	✓	✓	

- Run the integration manually, use a scheduled integration run to update the NetSuite records after you update the task records in OpenAir. See Running the Bulk Import and Export Integration on Demand and Scheduling the Integration to Run Automatically.

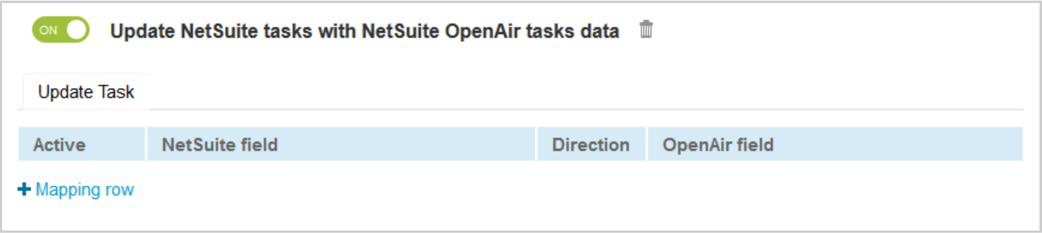
- Setup the real-time integration to update the NetSuite task records when the Task Properties form is saved in OpenAir. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- View and edit the OpenAir task record from the NetSuite UI using the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. See Enabling Access to OpenAir From Within NetSuite.

Setting up the Update NetSuite Tasks with OpenAir Task Data Export Workflow

Use the following steps to set up the Update NetSuite Tasks with OpenAir Task Data export workflow.

To set up the Update NetSuite Tasks with OpenAir Task Data export workflow in OpenAir

1. In OpenAir, go to Administration > NetSuite Connector, click the Tips menu button, then NetSuite Connector administration form.
2. Check the **Enable NetSuite task import and update** box under the Additional Integrations form section.
3. (Optional) Check the **Exclude reporting values from Update NetSuite tasks with OpenAir task data** box under the Performance features form section. You should check this box unless you need a wider range of OpenAir fields available for mapping and performance is not a concern for your integration. See Exclude reporting values from Update NetSuite tasks with OpenAir task data.
4. Click **Save**.
5. Go to Administration > NetSuite Connector > Mapping.
6. Click **New workflow** then **Update NetSuite Tasks with OpenAir Task Data**.
7. Click **Update NetSuite Tasks with OpenAir Task Data** in the list of integration workflows.
8. Modify the field mapping definition table for the Update NetSuite Tasks with OpenAir Task Data export workflow as required. See Mapping OpenAir Fields and NetSuite Fields and Update NetSuite Tasks with OpenAir Task Data Field Mapping Definitions.
9. Click the switch next to the workflow name to activate ON or to deactivate OFF the workflow.



ON Update NetSuite tasks with NetSuite OpenAir tasks data

Update Task

Active	NetSuite field	Direction	OpenAir field

+ Mapping row

10. Click **Save**.
11. (Optional) Add the Update NetSuite Tasks with OpenAir Task Data export workflow to a workflow group and schedule it to run automatically. See Scheduling the Integration to Run Automatically.

Update NetSuite Tasks with OpenAir Task Data Field Mapping Definitions

The Update NetSuite Tasks with OpenAir Task Data export workflow includes one mapping group:

- Update Task [Integration point: `UPDATE_PROJECT_TASK_TO_PROJECT_TASK`] — See Update Task Mapping Group.

Update Task Mapping Group

Use the Update Project mapping group to map information on task records in OpenAir with information on project task records in NetSuite. The mapping group supports filters.

For reference information and best practice guidelines, review the following topics:

- References — Overview of record types in OpenAir and NetSuite with links to the relevant reference material for information about fields available for mapping.
- Default and Custom Filters — Overview of default filters and how to modify the default behavior, and custom filters. Custom filters are supported for this mapping group.
- Default Field Mapping — There are no default field mapping definitions for this mapping group.
- Additional Field Mapping — Review best practice guidelines for custom field mapping definitions in this mapping group.

References

The following table gives reference information about the records and fields available for custom field mapping definition with links to the relevant reference material. See also Schema Reference Documentation.

NetSuite Record type [SOAP object]	OpenAir Record Type [table]	Mapping Group [Integration point]
Project Task [ProjectTask] Schema: ProjectTask (2020.1)|ProjectTask (2023.1) XSD: ProjectTask — actSched (2020.1)|actSched (2023.1)	- If the **Exclude reporting values from Update NetSuite projects with NS OpenAir project data** box is enabled on the OpenAir NetSuite Connector administration form: - Record Type: Project Task [project_task] - Data Dictionary: project_task - Otherwise, the OpenAir information is sourced from a detailed report. Refer to the UI for the list of report columns available for mapping. See also Exclude reporting values from Update NetSuite tasks with OpenAir task data.	Update Task [UPDATE_PROJECT_TASK_TO_PROJECT_TASK]

Default and Custom Filters

By default, updated task information is exported from OpenAir to NetSuite if the following conditions are met:

- Matching task records exists in OpenAir and NetSuite.
- The task was not closed with **% complete** set to 100% in OpenAir before the last export. The task is excluded automatically from future exports updating NetSuite project task records with OpenAir task data otherwise. To include the task record in future update exports, clear the **Do not send this task data to NetSuite** [netsuite_project_task_export_finished] box manually on the OpenAir task record. This may be useful if the task is subsequently reopened, for example. Note that the **Do not send this task data to NetSuite** [netsuite_project_task_export_finished] custom field is hidden by default.

Custom filters are supported for this mapping group. For more information about adding filter conditions, see Using Filters and Filter Definition.

Note: Only standard and custom fields for the Project Task [`project_task`] record type can be used in custom filters even if other fields (report values) are available for mapping.

Default Field Mapping

None

Additional Field Mapping

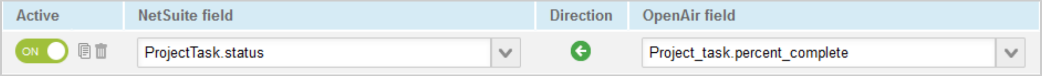
Important: Review the following guidelines

- You should contact your OpenAir Professional Services representative if you need to make changes to your integration configuration and field mapping definitions.
- For advanced mapping usage and best practice guidelines, see Advanced Field Mapping Definition.
- Familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.

As of the April 2022 OpenAir release, the Update NetSuite Tasks with OpenAir Task Data export workflow supports advanced mapping. Advanced mappings are supported only if the **Exclude reporting values from Update NetSuite tasks with OpenAir task data** box is checked under the Performance features section of the OpenAir NetSuite Connector administration form. See Exclude reporting values from Update NetSuite tasks with OpenAir task data.

Note that OpenAir NetSuite Connector also applies advanced mapping logic automatically in the following cases.

- You can add a field mapping definition to set the **Status** field on the NetSuite project task record [`ProjectTask.status`] depending on the **% Complete** field value on the OpenAir task record [`project_task.percent_complete`] and the **Status** field on the NetSuite project task record [`ProjectTask.status`]. To do so, select the OpenAir field `project_task.closed` and the NetSuite field `ProjectTask.custevent_oa_export_to_openair`.

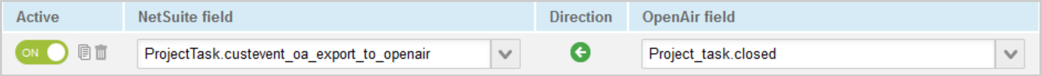
  

Active	NetSuite field	Direction	OpenAir field
ON	ProjectTask.status	←	Project_task.percent_complete

  When this field mapping definition is active, OpenAir NetSuite Connector sets the **Status** according to the following table.

% Complete [`project_task.percent_complete`] (OpenAir Task)	**Status [`ProjectTask.status`] (NetSuite Project Task)**
0	Not Started [`_notStarted`]
Greater than 0 and lower than 100	In Progress [`_inProgress`]
100	Completed [`_completed`]

- You can add a field mapping definition to clear the **Export to OpenAir** [`custevent_oa_export_to_openair`] box on the NetSuite project task record when the **Closed task** [project_task.closed] box is checked on the OpenAir task record. To do so, select the OpenAir field project_task.closed and the NetSuite field ProjectTask.custevent_oa_export_to_openair.



Active	NetSuite field	Direction	OpenAir field
ON	ProjectTask.custevent_oa_export_to_openair	←	Project_task.closed

WBS (Opportunity) Export Workflow

Note: This help topic is a placeholder only. Content will be added gradually to these pages as the content and structure of this guide are reviewed. For more information about the WBS (Opportunity) integration workflows, please refer to the following chapters:

- Configuring Optional Features
- Using the Integration
- Data Mapping in the Integration

WBS (Opportunity) Export Workflow Settings

You can control WBS (Opportunity) export workflow settings from the NetSuite Connector UI. On the NetSuite Connector Mapping screen, select the WBS (Opportunity) export workflow and click **Settings** in the top right corner to view the workflow settings form.

The following settings are available for the WBS (Opportunity) export workflow:

- Always export all task assignments when exporting project work breakdown structure to NetSuite opportunity
- Do not use task planned hours for manual task projected billings when exporting project work breakdown structure to NetSuite opportunity
- Exclude manual task projected billings from rollup when exporting project work breakdown structure to NetSuite opportunity
- Map task service to NetSuite item instead of task assignment job codes for manual task projected billings when exporting project work breakdown structure to NetSuite opportunity

Always export all task assignments when exporting project work breakdown structure to NetSuite opportunity

This will force this integration to always export each task assignment to NetSuite as an opportunity line item.

Do not use task planned hours for manual task projected billings when exporting project work breakdown structure to NetSuite opportunity

This option will not use the total planned hours from the task assignments for the quantity on the opportunity line item. Instead, it will default to 1.

This option can only be used with Map task service to NetSuite item instead of task assignment job codes for manual task projected billings when exporting project work breakdown structure to NetSuite opportunity.

Exclude manual task projected billings from rollup when exporting project work breakdown structure to NetSuite opportunity

This will force the integration to never include manually overridden projected billings to be combined with task assignments.

Map task service to NetSuite item instead of task assignment job codes for manual task projected billings when exporting project work breakdown structure to NetSuite opportunity

This will map the category (default_category field) on the project task instead of job code as the source of the NetSuite item.

None of the task assignments will be exported for project tasks with manual projected billings when using this option.

Custom Export Workflows

You can add custom workflows to export data from a supported OpenAir record type to a record type in NetSuite, including custom record types, using your own field mapping definitions. See Exporting Records from OpenAir to NetSuite Using Custom Workflows.

Custom export workflows can be set to:

- Create new records in OpenAir first and then export them to NetSuite.
- Update records in OpenAir and use the NetSuite <> OpenAir integration to update the corresponding record in NetSuite.

If the Business Intelligence Connector feature is enabled for your OpenAir account, you can add custom workflows to export OpenAir published report data to a NetSuite custom record type. You can then use NetSuite features to combine your OpenAir report data and NetSuite data or view them side to side in your custom dashboards and KPIs. See Exporting OpenAir Published Report Data To NetSuite Custom Record Types.

Custom export workflow field mapping supports filters and advanced mappings. Custom export workflows exporting OpenAir records to NetSuite currently support only one pullback row for the NetSuite record ID. See Advanced Mappings for Custom Export Workflows.

Custom export workflows have a built-in autorecovery feature in case the export process is interrupted. The autorecovery feature prevents the creation of duplicated records by identifying records included for export in the interrupted integration run, and verifying if the record was already exported to NetSuite. If the record exists in NetSuite, OpenAir NetSuite Connector updates the exported record in OpenAir. Otherwise, the record will be exported again.

You can add custom export workflows to workflow groups and include them in your scheduled integration runs, or when running the integration manually. For more information about workflow groups, see Creating and Managing Workflow Groups.

By default, a maximum of 4 custom export or import workflows can be active at any one time. To review your requirements, and increase the maximum number of active custom export or import workflows accordingly, contact OpenAir Customer Support.

You can review the setting details you selected or entered when creating each custom export workflow at any time. See Viewing Custom Export Workflows Settings.

You can manage custom export workflows and delete those you no longer need. See Deleting Custom Export Workflows.

Note: Review the following guidelines:

- When exporting OpenAir published report data to a custom record type, you can configure the custom export workflow either:
  - To create new records based on the OpenAir published report data. "Create Only" custom export workflows exporting OpenAir published report data to a custom record type can only run one time.
  - To create or replace records based on the OpenAir published report data. "Replace All" custom export workflows delete NetSuite custom records created previously by the same workflow and creates new records based on the current OpenAir published report data.
- Custom export workflows currently support the following OpenAir record types: Booking, Customer, Contact, Cost center, Customer PO, Employee, Expense item, Expense report, Invoice, Job code, PO, Payment, Payroll type, Product, Project, Purchase request, Rate card, Recognition transaction, Reimbursement, Service, Service 1 line, Service 2 line, Service 3 line, Service 4 line, Service 5 line, Task, Time entry, Time type, Timesheet, Vendor.
- Custom export workflows currently support the following NetSuite record types: Contact, Credit memo, Customer, Employee, Expense category, Expense report, Invoice, Journal entry, Opportunity, Project, Project task, Resource allocation, Time entry, Vendor, and Custom record types.
- Custom export workflow field mapping supports standard fields available in the NetSuite endpoint used for the integration, custom fields and custom segments.
- The built-in autorecovery feature is only available for custom export workflows exporting OpenAir records to NetSuite. It is not available for custom export workflows exporting published report data to NetSuite.

Enabling Custom Export Workflows

Custom export workflows require the following NetSuite Connector administration setting to be enabled: Enable Direct TBA.

To enable custom export workflows:

1. Go to Administration > NetSuite Connector.
2. Click the Tips button, then **NetSuite Connector administration**.

3. Check the **Custom export** box under the Exports section. See Custom Export.

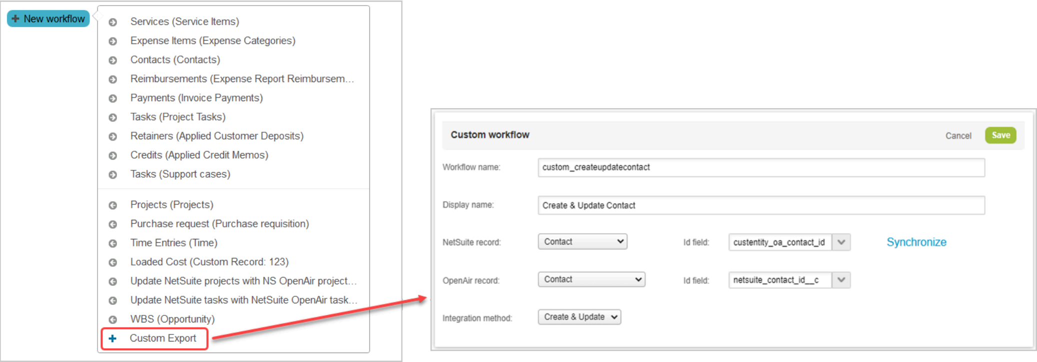
Exporting Records from OpenAir to NetSuite Using Custom Workflows

You can create custom workflows to export data from a supported OpenAir record type to a record type in NetSuite, including custom record types, using your own field mapping definitions.

Review the best practice guidelines before you follow these steps. See Best Practice Guidelines.

To export records from OpenAir to NetSuite using a custom export workflow:

1. Go to Administration > NetSuite Connector > Mappings.
2. Click **New workflow** then **Custom Export**. The Custom workflow popup window appears.


+ New workflow
Services (Service Items)
Expense Items (Expense Categories)
Contacts (Contacts)
Reimbursements (Expense Report Reimbursem…
Payments (Invoice Payments)
Tasks (Project Tasks)
Retainers (Applied Customer Deposits)
Credits (Applied Credit Memos)
Tasks (Support cases)
Projects (Projects)
Purchase request (Purchase requisition)
Time Entries (Time)
Loaded Cost (Custom Record: 123)
Update NetSuite projects with NS OpenAir project…
Update NetSuite tasks with NetSuite OpenAir task…
WBS (Opportunity)
+ Custom Export
Custom workflow
Cancel
Save
Synchronize

Workflow name: custom_createupdatecontact

Display name: Create & Update Contact

NetSuite record: Contact | Id field: custentity_oa_contact_id

OpenAir record: Contact | Id field: netsuite_contact_id__c

Integration method: Create & Update

3. Enter a **Workflow name** — the value must be up to 40 characters long and must contain only alphanumeric or underscore characters.
4. Enter a **Display name** up to 80 characters long.
5. Select the **OpenAir record** holding the data you want to export to the selected NetSuite record type.
6. Select the **ID field** for the OpenAir record — this is the field that will hold the internal ID of the matching NetSuite record (typically a custom field). The **ID field** is required and cannot be changed after you save the custom export workflow configuration.
7. Select the **NetSuite record** type you want to export OpenAir data to. To export OpenAir data to a custom record type in NetSuite, select “[Custom Record]”, and select the NetSuite custom record type from the **Specify** dropdown.
8. Select the **ID field** for the NetSuite record — this is the field that will hold the internal ID of the matching OpenAir record (typically a custom field). This **ID field** is required and cannot be changed after you save the custom export workflow configuration.

   > ✓ **Tip:** Click **Synchronize** to update the NetSuite custom fields available for field mapping without leaving the custom export workflow configuration form. This allows you to see the latest list of NetSuite custom field and select a suitable ID field from this list.

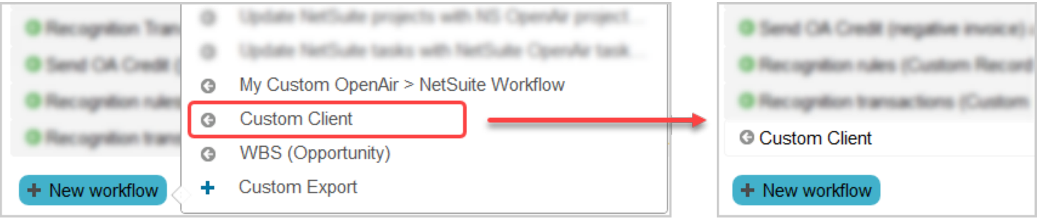
9. Choose the **Integration method**:
   - **Create Only** — The workflow lets you create new records in OpenAir first and then export them to NetSuite. However, the workflow does not update the NetSuite record after changes are made to the OpenAir record.
   - **Create and Update**— The workflow lets you:

- Create new records in OpenAir first and then export them to NetSuite.
- Update records in OpenAir and use the NetSuite <> OpenAir integration to update the corresponding record in NetSuite.

- **Replace All**.

> **Important:** The **Replace All** integration method cannot be used when exporting records from OpenAir to NetSuite using a custom export workflow. It can only be used when exporting OpenAir published report data to NetSuite.

10. Click **Save**.
11. Click **New workflow** then click the display name of the custom export workflow you created. The custom export workflow is added to the list.

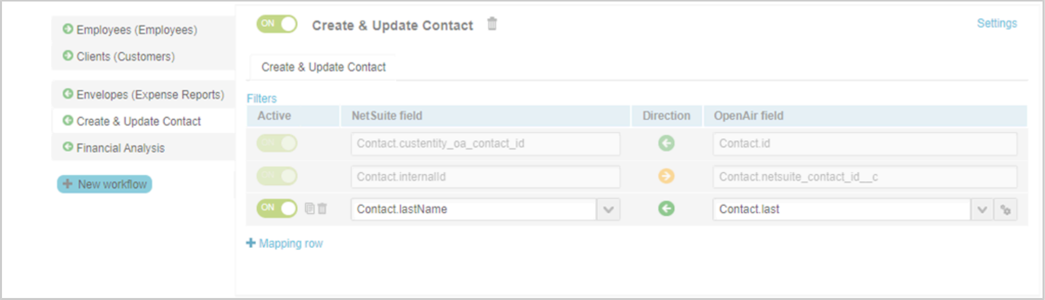


12. Select the custom export workflow from the list of workflows to show the field mappings for this workflow. Custom export workflows are listed after built-in workflows.
13. Edit the field mappings as per your integration requirements.

    Two mapping rows are shown by default. These required mapping rows are protected and cannot be modified:

    - One row mapping the internal ID of the OpenAir record to the relevant custom field on the NetSuite record.
    - One pullback row retrieving the internal ID of the NetSuite record and storing it in to the relevant custom field on the OpenAir record.

    You must specify at least one additional mapping row before you can save the workflow.



Create & Update Contact

Active	NetSuite field	Direction	OpenAir field
ON	Contact.custentity_oa_contact_id	←	Contact.id
ON	Contact.internalId	→	Contact.netsuite_contact_id__c
ON	Contact.lastName	←	Contact.last

Best Practice Guidelines

Review the following guideline:

- Before you create a custom export workflow, make sure you add custom fields to the NetSuite record type and to the OpenAir record type to store the internal IDs of the records integrated using the custom workflow.
- Before you create a custom export workflow exporting OpenAir records to NetSuite as resource allocation records, you must create a custom CRM field for resource allocation records in NetSuite to

hold the internal ID of the matching OpenAir record and specify this **ID field** for the NetSuite record when creating the custom export workflow.

Use the following details when creating the custom CRM field in NetSuite:

- ID: `_oa_resource_allocation_id`
- Type: `Free-form text`
- Applied to: `Resource Allocation`

> **Note:** By default, only the standard fields for NetSuite resource allocation records and the custom field described above are available for mapping. To make custom CRM field for resource allocation records other than `_oa_resource_allocation_id` available for mapping, you must list them on the OpenAir NetSuite Connector administration form. See NetSuite custom fields for resource allocation records in field mapping for custom export.
>
> When exporting bookings from OpenAir to NetSuite as resource allocations, bookings for both named resources and generic resources are included by default. You can add a filter to exclude bookings for generic resources, if you want to export only bookings for named resources.

For more information about the Resource Allocations feature in NetSuite, see Resource Allocations (external link to NetSuite SuiteAnswers | Answer ID: 31868).

For more information about creating a custom CRM field, see Creating Custom CRM Fields (external link to NetSuite SuiteAnswers | Answer ID: 10082).

- Update the NetSuite custom fields available for field mapping when you create a custom export workflow by clicking **Synchronize** from the custom export workflow configuration form.
- Consult NetSuite SOAP schema browser for reference when defining field mapping for custom export workflows. It includes information about any specific requirements for exporting OpenAir data to NetSuite.

  For example, if you are exporting project task records from OpenAir to NetSuite and you want to set the project task start date in NetSuite to match the start date in OpenAir, you need to add the following mapping lines. NetSuite calculates the earliest possible start date based on existing predecessors unless you set `constraintType` to the value `_fixedStart`.

  - NetSuite Field: `ProjectTask.startDate` — OpenAir Field: `Project_task.start_date`
  - NetSuite Field: `ProjectTask.constraintType` — OpenAir Field: `="_fixedStart"`

Exporting OpenAir Published Report Data To NetSuite Custom Record Types

> **Note:** This help topic describes features available if the following features are enabled for your OpenAir account:
>
> - Custom Export Workflows — see Enabling Custom Export Workflows.
> - Business Intelligence Connector — To enable this feature, contact your OpenAir account manager.

If the Business Intelligence Connector feature is enabled for your OpenAir account, you can add custom workflows to export OpenAir published report data to a NetSuite custom record type. You can then use

NetSuite features to combine your OpenAir report data and NetSuite data or view them side to side in your custom dashboards and KPIs.

The Business Intelligence Connector feature lets you publish OpenAir reports to the OpenAir OData service for the purpose of exporting your OpenAir report data to NetSuite. You can configure OpenAir to refresh your published report data according to a defined schedule. For more information about publishing reports, see Business Intelligence Connector.

After you publish a report to your OpenAir OData feed for access by OpenAir NetSuite Connector, you can create a custom export workflow to export data from your OpenAir published report to a custom record type in NetSuite.

You can add this custom export workflow to workflow groups and include them in your scheduled integration runs, or when running the integration manually.

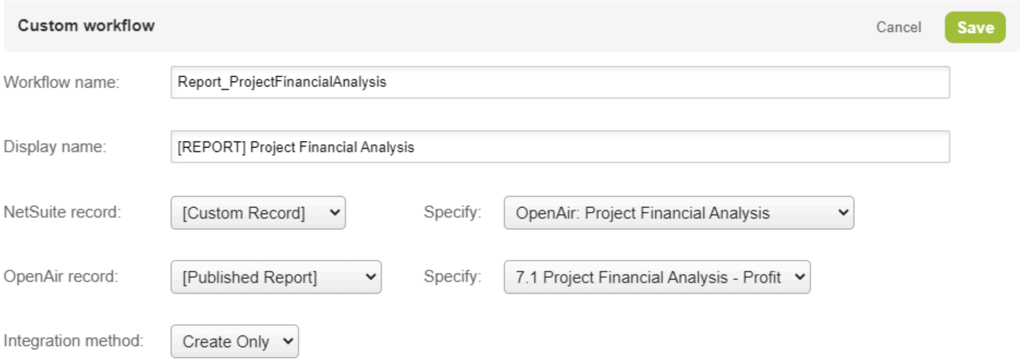
To export OpenAir published report data to NetSuite:

1. In NetSuite, create a custom record type and custom fields to store the published report data exported from OpenAir following the best practice tips.

   > **Note:** Use a naming convention for custom record types that will hold OpenAir published report data for ease of identification and use. For example:
   >
   > - Use the prefix `OpenAir:` followed by the name of your OpenAir report for the custom record type **Name**, such as `OpenAir: Project Financial Analysis`.
   > - Use the prefix `customrecord_oa_` followed by the name of your OpenAir report for the custom record type **ID**, such as `customrecord_oa_project_financial_analysis`.

   You can view best practice suggestions for setting up the custom record type in NetSuite. See Viewing Tips for Custom Export of OpenAir Reports to NetSuite.

2. In OpenAir, go to Administration > NetSuite Connector > Mappings.
3. Click the Tips button, then **Synchronize NetSuite custom fields** to ensure all the NetSuite custom records and custom fields are available for mapping.
4. Click **New workflow** then **Custom Export**. The Custom workflow popup window appears.



Custom workflow

Workflow name: Report_ProjectFinancialAnalysis

Display name: [REPORT] Project Financial Analysis

NetSuite record: [Custom Record] Specify: OpenAir: Project Financial Analysis

OpenAir record: [Published Report] Specify: 7.1 Project Financial Analysis - Profit

Integration method: Create Only

5. Enter a **Workflow name** — the value must be up to 40 characters long and must contain only alphanumeric or underscore characters.

6. Enter a **Display name** up to 80 characters long.
7. Select “[Published Report]” from the **OpenAir record** dropdown list, and select the OpenAir published report from the **Specify** dropdown list.
8. Select “[Custom Record]” from the **NetSuite record** dropdown list, and select the NetSuite custom record type you created from the **Specify** dropdown list. You cannot export OpenAir published report data to a standard record type in NetSuite.

   > ⚠ **Important:** It is not possible to have two active custom workflows exporting to the same custom record type in NetSuite. If you select a NetSuite custom record type that is already used for an active custom export workflows, an error appears and the new custom export workflow is not saved.

9. Choose the **Integration method**:
   - **Create Only** — The workflow creates but does not update NetSuite custom records with the OpenAir published report data. The workflow can run only one time. If you create another custom export workflow exporting OpenAir published report data to the same NetSuite custom record type using the Create Only **Integration method** and run this workflow, this new custom export workflow will create new records in NetSuite.
   - **Create and Update**.

     > ⚠ **Important:** The **Create and Update** integration method cannot be used when exporting published report data to NetSuite custom records. It can only be used with custom export workflows exporting OpenAir records to NetSuite.

   - **Replace All** — The workflow deletes all records exported previously by this workflow, then creates new records with the latest OpenAir published report data. If you have another custom export workflow exporting OpenAir published report data to the same NetSuite custom record type, the records created by that other workflow are not deleted.
10. Click **Save**.
11. Click **New workflow** then click the display name of the custom export workflow you created. The custom export workflow is added to the list.
12. Select the custom export workflow from the list of workflows to show the field mappings for this workflow. Custom export workflows are listed after built-in workflows.
13. Edit the field mappings as per your integration requirements.
    - OpenAir published report column names are listed in the OpenAir field column in the mapping definition table.

      > ⚠ **Important:** Review the following guidelines:
      >
      > - Special characters in the OpenAir published report column names are escaped in the name used to reference the column. When creating reports, ensure the column names will not result in duplicate field names for mapping purposes. If there are any duplicate field names, OpenAir NetSuite Connector appends a number to the field names (‘_1’, ‘_2’, ‘_3’) to avoid errors when running the integration workflow.
      > - Advanced mapping does not currently support OpenAir published report columns. You cannot reference OpenAir published report columns in your advanced mappings.

    - OpenAir NetSuite Connector fetches only the published report columns referenced in field mapping definitions.

- You can use the filter builder to define filter conditions and retrieve just the information you need from the published report. See Filter Builder.

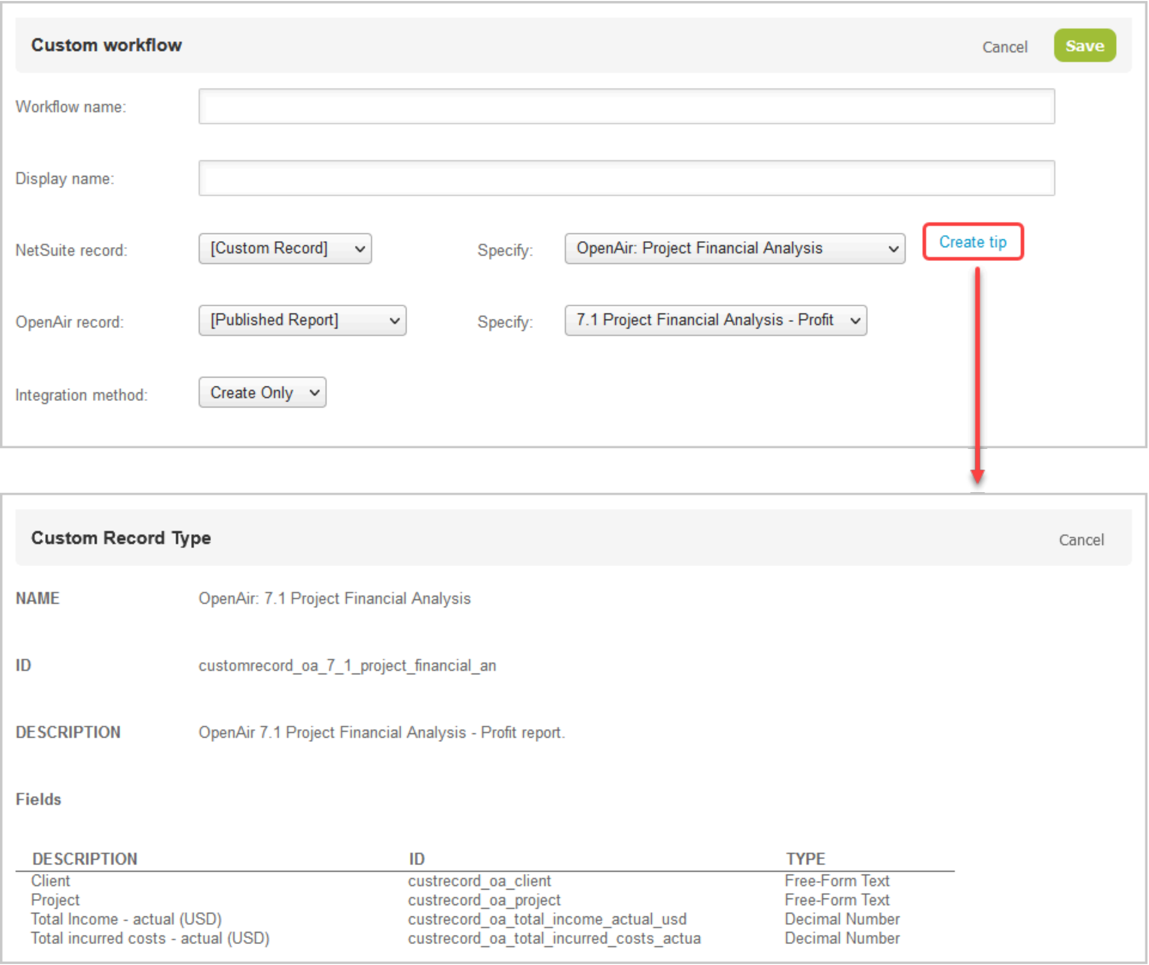
Viewing Tips for Custom Export of OpenAir Reports to NetSuite

You can view best practice suggestions for setting up the custom record type in NetSuite. This includes suggestions for:

- Custom record type **Name**, **ID**, and **Description**.
- Custom fields to include with suggested **Description**, **ID** and **Type** for each custom field.

To view best practice suggestions for setting up the custom record type in NetSuite:

1. In OpenAir, go to Administration > NetSuite Connector > Mappings.
2. Click **New workflow** then **Custom Export**. The Custom workflow popup window appears.
3. Select "[Published Report]" from the **OpenAir record** dropdown list, and select the OpenAir published report from the **Specify** dropdown list.
4. Select "[Custom Record]" from the **NetSuite record** dropdown list.
5. Click **Create Tip**.



Workflow name:

Display name:

NetSuite record: [Custom Record] Specify: OpenAir: Project Financial Analysis

OpenAir record: [Published Report] Specify: 7.1 Project Financial Analysis - Profit

Integration method: Create Only

NAME OpenAir: 7.1 Project Financial Analysis

ID customrecord_oa_7_1_project_financial_an

DESCRIPTION OpenAir 7.1 Project Financial Analysis - Profit report.

Fields

DESCRIPTION	ID	TYPE
Client	custrecord_oa_client	Free-Form Text
Project	custrecord_oa_project	Free-Form Text
Total Income - actual (USD)	custrecord_oa_total_income_actual_usd	Decimal Number
Total incurred costs - actual (USD)	custrecord_oa_total_incurred_costs_actua	Decimal Number

Advanced Mappings for Custom Export Workflows

You can set up advanced mappings for custom export workflows. For more information about advanced mappings, see Using the Advanced Tag Notation and Advanced Mapping and Filter Notation Reference.

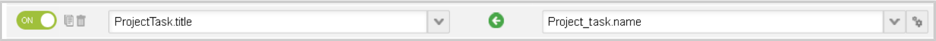
Note the following differences between advanced mappings for custom export workflows and for other integration workflows:

- The following field declarations are not required: `OA_FIELDS` (OpenAir fields), `OA_CUSTOM_FIELDS` (OpenAir custom fields), `NS_FIELDS` (NetSuite fields), and `NS_CUSTOM_FIELDS` (NetSuite custom fields). You can use them but they will be ignored.

  For example:

  - **Value to value (from schema) mapping** — Advanced mapping is not required as you should be able to select the NetSuite field from the dropdown options on the Mapping tab. However, you could use the equivalent advanced mapping.

    ```
    title   name
    ```

    ProjectTask.title → Project_task.name

  - **Value to value (lookup) mapping** — You can specify the lookup in the same way as for standard workflows but the field declaration is not required.

    ```
    company lookup=project_id:lookup_table=project:lookup_by=id:lookup_return=netsuite_project_id
    ```

  - **Value to value (list map)** — You can use the mapping table in the same way as for standard workflows but the field declaration is not required. In the following example, if the `project_id` value is 126 on the OpenAir record, `company` is set to 4971 in NetSuite.

    ```
    <company project_id>
       '126' '4971'
       '127' '4970'
    </company>
    ```

  - **Value to value (IF condition)** — You can use the IF condition in the same way as for standard workflows but the field declaration is not required. In the following example, if the `project_id` value is 126 on the OpenAir record, `company` is set to 4971 in NetSuite.

    ```
    IF <<END
    IF (project_id='126') THEN company '4971'
    IF (project_id='127') THEN company '4970' ELSE company '4972'
    END
    ```

- The following field declarations are not supported currently: `OA_FIELDS_INITIAL_ONLY` (OpenAir fields initial-only), `OA_CUSTOM_FIELDS_INITIAL_ONLY` (OpenAir custom fields initial-only), `OA_FIELDS_GROUP_BY` (OpenAir fields group by), and `OA_FIELDS_SORT_BY` (OpenAir fields group by).

Custom Export Best Practice Guidelines

Review the best practice guidelines for the following cases:

- Time Entries to Time Entries Custom Export Best Practice

Time Entries to Time Entries Custom Export Best Practice

If you are creating a custom export workflow exporting time entries from OpenAir to NetSuite, use the guidelines in the following table.

If you are using the custom export workflow in conjunction with the standard workflow for time export, you should use a filter to exclude all OpenAir time entries that already have a NetSuite ID from the Time Entries (Time) export workflow to avoid integration errors. For more information about the standard workflow for time export, see Time Entries (Time) Export Workflow.

NetSuite Field	Direction	OpenAir Field	Notes
`TimeBill.custcol_oa_time_entry_id`	←	`Task.id`	*Read-only* Select the NetSuite record **ID field** `custcol_oa_time_entry_id` when you create the custom export workflow. Using the ID fields specified in this table prevents duplicated time entries from being created in NetSuite by the custom export workflow. See Exporting Records from OpenAir to NetSuite Using Custom Workflows.
`TimeBill.internalid`	→	`Task.netsuite_timebill_id__c`	*Read-only* Select the OpenAir record **ID field** `netsuite_timebill_id__c` when you create the custom export workflow. Using the ID fields specified in this table prevents duplicated time entries from being created in NetSuite by the custom export workflow. See Exporting Records from OpenAir to NetSuite Using Custom Workflows.

`TimeBill.employee`	←	`#Employee lookup`
		`employee lookup=user_id:lookup_table=user:lookup_by=id:lookup_return=netsuite_user_id`
			An advanced mapping must be used to look up the OpenAir value to be exported to the **Employee** field in NetSuite. See Advanced Mapping and Filter Notation Reference and Advanced Mappings for Custom Export Workflows.

`TimeBill.item`	←	`#Item lookup`
		`item lookup=category_id:lookup_table=category:lookup_by=id:lookup_return=netsuite_category_id`
			An advanced mapping must be used to look up the OpenAir value to be exported to the **Item** field in NetSuite. See Advanced Mapping and Filter Notation Reference and Advanced Mappings for Custom Export Workflows.

`TimeBill.customer`	←	`#Customer-Project lookup`
		`customer lookup=project_id:lookup_table=project:lookup_by=id:lookup_return=netsuite_project_id`
			An advanced mapping must be used to look up the OpenAir value to be exported to the **Customer** field in NetSuite. See Advanced Mapping and Filter Notation Reference and Advanced Mappings for Custom Export Workflows.

`TimeBill.hours`	←	`Task.hours_and_minutes__c`	OpenAir uses two separate fields for time entry (hours and minutes), while NetSuite uses a single field (hours). It is not possible to map the time value directly in the OpenAir NetSuite Connector UI unless using full hours only. Instead, you should use a form script to combine the hours and minutes fields into a decimal value and store this value in a custom field in OpenAir, and map this decimal time value custom field to the time field in NetSuite. For information about form scripts, see User Scripting.

NetSuite Field	Direction	OpenAir Field	Notes

`TimeBill.isBillable`	←	`#Always billable unless marked as 'Non-Billable'`
		`IF <<END`
		`IF (netsuite_billable_time = 'Non-Billable') THEN isBillable 'false' ELSE isBillable 'true'`
		`END`
			If required, the OpenAir custom field **NetSuite Billable Time** [`netsuite_billable_time`] and an advanced mapping can be used to mark the time entries as billable in NetSuite. See Advanced Mapping and Filter Notation Reference and Advanced Mappings for Custom Export Workflows.
			Note: The **NetSuite Billable Time** [`netsuite_billable_time`] is created automatically when you activate the Time Entries (Time) standard export workflow. To show the custom field as columns on the Timesheet grid (Web application UI) or time card properties (Mobile app UI), contact OpenAir Customer Support. You will need to provide the internal ID for the NetSuite Billable Time custom field, which you can obtain from the custom field list (Administration > Global Settings > Custom Field), and clear the **Hide on data entry forms box** in the custom field properties.

`TimeBill.tranDate`	←	`Task.date`	—

Viewing Custom Export Workflows Settings

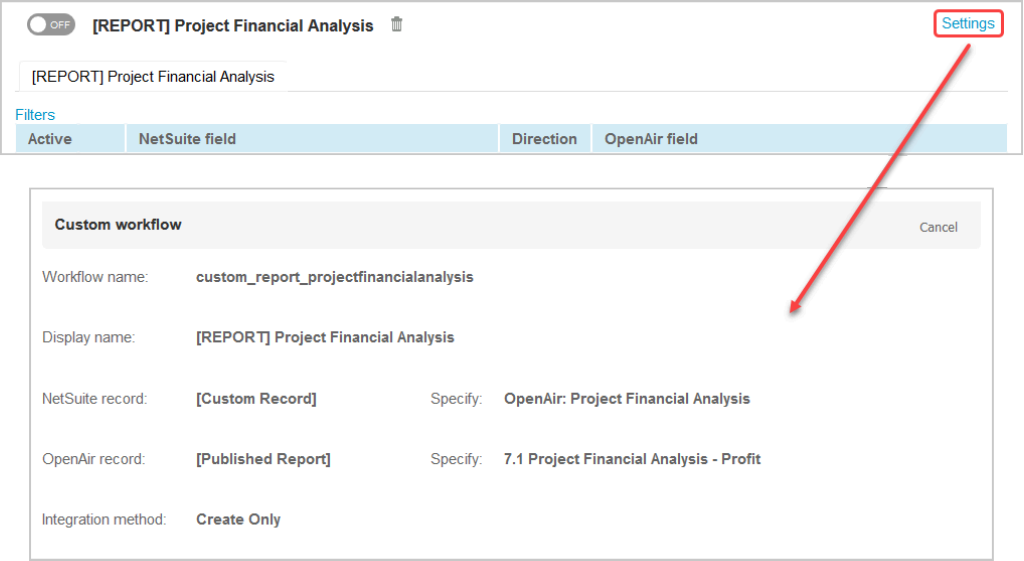
You can view the settings for your existing custom export workflows and get a reminder of the following details:

- Workflow name.
- OpenAir record type or published report being exported.
- NetSuite record type or custom record type holding the exported data.
- Integration method:
  - **Create Only** - for all workflows.
  - **Create & Update** - not for workflows exporting OpenAir published report data.
  - **Replace All** integration method - only for workflows exporting OpenAir published report data.

The settings are read-only and cannot be edited.

To view the settings for an existing custom export workflow:

1. In OpenAir, go to Administration > NetSuite Connector > Mappings.
2. Select the custom export workflow from the list of workflows. The mapping definition table for this workflow appears in the right pane.
3. Click Settings. The custom workflow popup window appears showing the setting details for the workflow as read-only information.



[REPORT] Project Financial Analysis

Active	NetSuite field	Direction	OpenAir field

Custom workflow

Workflow name:	custom_report_projectfinancialanalysis		
Display name:	[REPORT] Project Financial Analysis		
NetSuite record:	[Custom Record]	Specify:	OpenAir: Project Financial Analysis
OpenAir record:	[Published Report]	Specify:	7.1 Project Financial Analysis - Profit
Integration method:	Create Only		

Deleting Custom Export Workflows

You can manage custom export workflows and delete those you no longer need.

To delete a custom workflow:

1. Go to Administration > NetSuite Connector > Mappings.
2. Select the custom workflow you want to delete.
3. Click the delete icon 🗑 . A confirmation dialog appears.
4. (Optional) To delete the custom workflow from and remove it permanently from the **New workflow** dropdown, check the **Permanently delete** box.
5. Click **Delete** to confirm. The custom workflow and associated field mapping are deleted.

Using the Integration

After the initial data transfer between OpenAir and NetSuite, you view and modify that data in one or both applications and integrate the data back and forth between the two applications.

The following sections describe how data can and should be used in each application:

- Working with NetSuite Record Data in OpenAir
- Working with OpenAir Data in NetSuite

In addition, you may wish to review some Frequently Asked Questions (FAQs) about how the integration works between the two applications.

For specific help on using the NetSuite and OpenAir applications, see Getting Help.

For details on data mapping between the two applications, see Data Mapping in the Integration.

For specific details on advanced settings, which can be used to customize the integration and specify what is transferred and how it displays, see Advanced Mapping and Filter Notation Reference.

Working with NetSuite Record Data in OpenAir

After the integration is run, NetSuite record data transfers to OpenAir. Each record's data can then be viewed and modified in OpenAir. Note that the terminology may differ for the record when in OpenAir. See Terminology Mapping for the Integration for more information.

Review the following topics for details on how to work with each NetSuite record type in OpenAir:

- Working with Employee Records in OpenAir
- Working with Customer Records in OpenAir
- Working with Contact Records in OpenAir
- Working with Project Records in OpenAir
- Working with Task Records in OpenAir
- Working with Item Records in OpenAir
- Working with Expense Categories in OpenAir
- Working with Opportunities as Projects in OpenAir
- Working with Vendors in OpenAir
- Working with Purchases/Vendor Bills in OpenAir
- Working with Sales Orders in OpenAir
- Working with Expense Report Reimbursements in OpenAir
- Working with Invoice Payments in OpenAir
- Working with Customer Deposits Applied to Invoices in OpenAir
- Working with Credit Memos Applied to Invoices in OpenAir

Before you can view these records in OpenAir, you must first export the data. See Exporting NetSuite Records to OpenAir for more information.

Working with Employee Records in OpenAir

You can transfer employee records to OpenAir as Users.

After NetSuite Employee record data is transferred to OpenAir, each OpenAir employee or user is assigned a NetSuite User ID and some basic user information is transferred. For details, see NetSuite to OpenAir Data Mapping.

If you are using the NetSuite Classifications to OpenAir Hierarchies integration, you can transfer department, class, location, and subsidiary information. See Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

If you would like to limit the subsequent transfer of NetSuite employee data to OpenAir, you can use advanced settings to designate fields as initial-only export. See the instructions for Field Declarations.

To make an employee/user inactive, select the check box on the NetSuite Employee form for Employee is Inactive and clear the Active user check box on the OpenAir user demographic information form.

To add information to the User record in OpenAir:

1. Go to Administration > Global Settings > Users.
2. Click the User ID.

   Demographic information displays along with a number of links for entering additional user information.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Customer Records in OpenAir

Customer records transfer to OpenAir as Customers. NetSuite Leads and Prospects can also transfer to OpenAir as Customers. To enable the ability to transfer Leads and Prospects, check the following boxes on the Customers (Customers) workflow settings form:

- Include NetSuite leads marked 'Export to OpenAir' in the customers import
- Include NetSuite prospects marked 'Export to OpenAir' in the customers import

Once in OpenAir, each OpenAir customer receives a NetSuite Customer ID and some basic customer information transfers. For details, see NetSuite to OpenAir Data Mapping.

If you are using the NetSuite Classifications to OpenAir Hierarchies integration, you can transfer department, class, location, and subsidiary information. See Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

If you would like to limit the subsequent transfer of NetSuite customer data to OpenAir, you can use advanced settings to designate fields as initial-only export. See the instructions for Field Declarations.

To make a customer, lead or prospect inactive, select the check box on the appropriate NetSuite form for Customer is Inactive and clear the Active check box on the OpenAir Edit demographic information form for customers, leads, and prospects.

To add additional information to the Customer record in OpenAir:

1. Go to Administration > Global Settings > Customers > Customers.
2. Click the Nickname.

   Demographic information displays along with a number of links for entering additional customer information.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Contact Records in OpenAir

Contact records transfer to OpenAir as Contacts. Once in OpenAir, each OpenAir contact receives a NetSuite Contact ID and some basic contact information transfers. For details, see NetSuite to OpenAir Data Mapping.

You can enable the transfer of all contacts associated with customers and projects you Export to OpenAir. If a project exports to OpenAir, the project's customer has also been exported. The customer associated with the contact is the customer of the project. Attach a contact to a customer and/or project designated for Export to OpenAir and the contacts associated with those customers and/or projects transfer from NetSuite to OpenAir.

To enable this feature, check the following boxes on the NetSuite Connector administration form:

- Import all NetSuite contacts associated with NetSuite customers marked 'Export to NS OpenAir'
- Import all NetSuite contacts associated with NetSuite projects marked 'Export to NS OpenAir'

If you would like to limit the subsequent transfer of NetSuite contact data to OpenAir, you can use advanced settings to designate fields as initial-only export. See the instructions for Field Declarations.

To make a contact inactive, select the check box on the NetSuite Contact form for Contact is Inactive and clear the Active check box on the OpenAir Edit demographic information form for contacts.

To add additional information to the Contact record in OpenAir:

1. Go to Administration > Global Settings > Customers > Contacts.
2. Click the Name.

   Demographic information displays including a NetSuite Contact ID.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Project Records in OpenAir

Project records in NetSuite transfer to OpenAir as Projects. Once you establish a project in NetSuite and transfer it to OpenAir, you can update the project in NetSuite with OpenAir project management information. You can also create a project template in OpenAir, apply it to a new project in NetSuite, and when it transfers to OpenAir, it contains the tasks, assignments, billing rules, loaded costs, revenue recognition rules, auto-bill settings, and project approvers available in the template. For more information, see Projects (Projects) Import Workflow.

To enable the project features, check the following boxes on the NetSuite Connector administration form and relevant workflow settings form:

- NetSuite Connector administration form — Exports:
  - Update NetSuite Projects with OpenAir Project Data
- Projects (Projects) import workflow settings form:
  - Map the OpenAir project stage from the NetSuite project
  - Create new projects based on the project template specified in NetSuite
  - Enable project workspace creation

Once in OpenAir, each project receives a NetSuite Project ID and currency and subsidiary information transfer, depending on the integration. Project receipts can be set to either billable or non-billable on the

Edit project form in OpenAir. This allows expense receipts to transfer to NetSuite, but it sets all receipts for that project as billable or non-billable in NetSuite. For details, see NetSuite to OpenAir Data Mapping.

If you are using the NetSuite Classifications to OpenAir Hierarchies integration, you can transfer department, class, location, and subsidiary information. See Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.

If you would like to limit the subsequent transfer of NetSuite project data to OpenAir, you can use advanced settings to designate fields as initial-only export. See the instructions for Field Declarations.

To add additional information to the Project in OpenAir:

1. Go to Projects > Projects and select the project. The Edit project form displays.
2. Review the project information and make any additions.
3. Save the project.
4. Click the project link again. Additional links display, allowing you to use OpenAir to manage the project.

   You can assign and track issues, establish and track tasks, assign or book resources, enter a budget and track financial information, establish billing and recognition rules, establish a baseline report, and run overview reports.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Task Records in OpenAir

Project Task records in NetSuite transfer to OpenAir as Tasks. Once you establish a task in NetSuite and transfer it to OpenAir, you can update the task in NetSuite with OpenAir task information. For more information, see Tasks (Project Tasks) Import Workflow and Update NetSuite Tasks with OpenAir Task Data.

To enable the task features, check the **Enable NetSuite task import and update** box on the NetSuite Connector administration form under additional integrations.

Once in OpenAir, each task receives a NetSuite Project Task ID and name . For details, see NetSuite to OpenAir Data Mapping.

If you would like to limit the subsequent transfer of NetSuite task data to OpenAir, you can use advanced settings to designate fields as initial-only export. See the instructions for Field Declarations.

To add additional information to the Task in OpenAir:

1. Go to Projects > Tasks and select the task. The Edit task form displays.
2. Review the task information and make any additions.
3. Save the task.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Item Records in OpenAir

In NetSuite, items are the goods and services you sell to customers. Depending upon your integration, you can transfer service items to OpenAir or all items. You can transfer them to be used in OpenAir as activities and services that hold billing information. Items used on NetSuite purchase orders can transfer to OpenAir as products. There are two different integrations for transferring items from NetSuite to

OpenAir: Items to Activities or Services and Items to Products. You can use both of them simultaneously and distinguish between their use by selecting the applicable check boxes for export on the NetSuite Item form. Each is described as follows.

Items to Services

Once in OpenAir, the service is assigned a NetSuite Item ID. If revenue recognition is enabled in the integration, a NetSuite Income Account ID and NetSuite Deferred Revenue Account ID also transfer. For details, see NetSuite to OpenAir Data Mapping.

To enable the transfer of all items to OpenAir, check the **Disable NetSuite service type filtering for activities in NetSuite integration** box on the Services (Service Items) import workflow settings form. Only service items transfer if the box is not checked. To mark an item for transfer to OpenAir as an service, check the **Export to OpenAir** box on the NetSuite Item form.

If you would like to limit the subsequent transfer of NetSuite item-related data to OpenAir, you can use advanced settings to designate fields as initial-only export. See the instructions for Field Declarations.

To add additional information to the Activity record in OpenAir:

1. Go to Administration > Application Settings > Invoices > Activities.
2. Select the activity. The Edit activity form displays.

> ⚠ **Important:** If revenue recognition is enabled in the integration and Direct Revenue Posting is used for Item records in NetSuite, additional set up is required for invoices to export successfully from OpenAir to NetSuite. For more information, see NetSuite Direct Revenue Posting.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Items to Products

Once in OpenAir, the product is assigned a NetSuite Purchase Item ID and the Product Type and Subtype display. If revenue recognition is enabled in the integration, a NetSuite Income Account ID and NetSuite Deferred Revenue Account ID also transfer. A check box indicates if the NetSuite Product Is Fulfillable. For more information on configuring the Items to Products integration, see Implementing the NetSuite Purchase Order Integration. For details, see NetSuite to OpenAir Data Mapping.

To enable the transfer of Items to OpenAir as Products, check the **Products (Purchase Items)** box on the NetSuite Connector administration form under Imports. To mark an item for transfer to OpenAir as a product, check the **Export to OpenAir Products** box on the NetSuite Item form. You may use NetSuite Inventory, Kit/Package, Non-inventory, Other charge, and Service items. Custom mappings may also be used when transferring Items to Products. See Advanced Mapping and Filter Notation Reference.

To add additional information to the Product record in OpenAir:

1. Go to Administration > Application Settings > Purchases > Products.
2. Select the product. The Edit product form displays.

> ⚠ **Important:** If revenue recognition is enabled in the integration and Direct Revenue Posting is used for Item records in NetSuite, additional set up is required for invoices to export successfully from OpenAir to NetSuite. For more information, see NetSuite Direct Revenue Posting.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

NetSuite Direct Revenue Posting

In NetSuite, if Revenue Recognition is enabled, items must either reference a Deferred Revenue Account ID or have the Direct Revenue Posting box checked on the Item > Accounting tab. An Income Account ID must also be referenced.

When the integration is used to export invoices created in OpenAir to NetSuite, a Service or a Product associated with one or more charges on an invoice in OpenAir must either reference a NetSuite Deferred Revenue Account ID or have the NetSuite Direct Revenue Posting box checked. A NetSuite Income Account ID must also be referenced. The invoice integration from OpenAir to NetSuite will fail otherwise.

OpenAir NetSuite Connector does not currently support mapping for the Direct Revenue Posting field in NetSuite. The Activity or Service records and Product records must be updated manually to enable the integration of Invoices using Direct Revenue Posting.

To enable the integration of Invoices using Direct Revenue Posting:

1. In OpenAir, go to Administration > Application Settings > Invoices > Activities or Administration > Application Settings > Purchases > Products, as applicable.
2. Select the Activity or Product corresponding to the NetSuite Item record. The Edit Activity form or Edit Product form displays.
3. Check the **NetSuite Direct Revenue Posting** box.
4. Click Save.

> **Note:** Direct Revenue Posting must be enabled for the Item record in NetSuite.
>
> In OpenAir, if revenue recognition is enabled in the integration, the NetSuite Direct Revenue Posting check box is available for:
>
> - The Edit Activity or Edit Service form, if the Activities (Activity Items) or Services (Service Items) integration workflow is active.
> - The Edit Product form, if the Products (Purchase Items) integration workflow is active.

Working with Opportunities as Projects in OpenAir

A NetSuite opportunity can be associated with a NetSuite project and transfer to OpenAir. When in OpenAir, you can scope the project effort and schedule tasks using OpenAir's project task breakdown structure. When you click the link to update the NetSuite opportunity line items, the task assignments are added together and items transfer back to the NetSuite opportunity.

To enable the opportunity to project integration, check the **Enable OpenAir project to NetSuite opportunity integration** box on the NetSuite Connector administration form under Additional integrations. See Configuring the Opportunity to Project Integration.

To add additional information to the Opportunity as a Project in OpenAir:

1. Go to Projects > Projects and select the project. The Edit project form displays and includes the NetSuite Opportunity ID field.
2. Add task assignment and scheduling information.
3. Save the project.

4. Click the link in tips to: Update the NetSuite opportunity line items. It displays on the Projects application Phases/Tasks: List and Phases/Tasks: Outline.
5. Opportunity information for the project transfers to NetSuite and displays under items.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Expense Categories in OpenAir

Expense categories in NetSuite are actually an item type rather than a record. NetSuite Expense Category data transfers to OpenAir as an expense item. For details, see NetSuite to OpenAir Data Mapping.

You should export expense categories to OpenAir to ensure continuity within your projects. However, since an Expense Category is not an entity (record) in NetSuite, there is no check box to Export to OpenAir on Expense Category pages in NetSuite. See Expense Items (Expense Categories) Import Workflow.

If you would like to limit the subsequent transfer of NetSuite expense category data to OpenAir, you can use advanced settings to designate fields as initial-only export. See the instructions for Field Declarations.

To add additional information to the expense item in OpenAir:

1. Go to Administration > Application Settings > Expenses > Expense items.
2. Select the expense item. The Edit expense item form displays.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Vendors in OpenAir

There are two different integrations for transferring vendors from NetSuite to OpenAir: Vendors to Vendors and Vendors to Users. You can use them both simultaneously. For the purchase order integration, you transfer vendor records from NetSuite to vendor records in OpenAir. For the projects integration, you may transfer vendors to OpenAir as users so they can be assigned as resources to projects and their expense reports can export to NetSuite as vendor bills. Each is explained as follows. To enable both integrations, open a support ticket. See Contacting NetSuite and OpenAir.

Vendors to Vendors Transfer

NetSuite Vendor records transfer to OpenAir as Vendor records. If you are using the Expense Report to Vendor Bill integration, you must also identify the vendor as a vendor. For details, see NetSuite to OpenAir Data Mapping.

Vendor name, active status, payment terms, currency, and NetSuite Vendor ID transfer from NetSuite to OpenAir and display on the Edit vendor form. For more information on configuring the Vendors to Vendors transfer, see Implementing the NetSuite Purchase Order Integration and Implementing the Expense Report to Vendor Bill Integration.

To enable the transfer of NetSuite vendor records to OpenAir vendors, check the **Vendors (Vendors)** box on the NetSuite Connector administration under Imports. See Contacting NetSuite and OpenAir.

To add additional information to the Vendor record in OpenAir:

1. Go to Administration > Application Settings > Purchases or Expenses.

2. Click Vendors.

   Vendor list view displays. Click a vendor to display the Edit vendor form.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Vendors to Users Transfer

Vendor records transfer to OpenAir as User accounts. For details, see NetSuite to OpenAir Data Mapping.

In NetSuite, the Vendor Type must be Individual, or if you are using the Expense Report to Vendor Bill integration, you must identify the vendor as a user. After the integration is run, the vendor is listed as a user in OpenAir and can be assigned to projects and can use time tracking functionality in OpenAir. When vendors submit expense reports, the integration recognizes them as vendors and creates a vendor bill in NetSuite for approved OpenAir expense reports.

To enable the transfer of vendor records to OpenAir, check the **Employees (Vendors)** box on the NetSuite Connector administration under Imports.

To enable the transfer of expense reports to vendor bills, check the **Enable OpenAir expense report to NetSuite vendor bill integration** box on the NetSuite Connector administration under Additional integrations. See Expense Reports (Expense Reports) Export Workflow. In OneWorld accounts, please keep in mind that the vendor and the Customer : Project must be in the same subsidiary.

If you would like to limit the subsequent transfer of NetSuite vendor data to OpenAir, you can use advanced settings to designate fields as initial-only export. See the instructions for Field Declarations.

To add additional information to the User/Vendor record in OpenAir:

1. Go to Administration > Global Settings > Users.
2. Click the User ID.

   Demographic information displays along with a number of links for entering additional user information.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Purchases/Vendor Bills in OpenAir

There are two mutually exclusive integrations you can use for integrating NetSuite purchases and NetSuite vendor bills with OpenAir purchase orders. They are:

- NetSuite Vendor Bills to OpenAir Purchase Orders
- NetSuite Purchase Order Integration

Each is described in the following sections.

NetSuite Vendor Bills to OpenAir Purchase Orders

Using the NetSuite Vendor Bills to OpenAir Purchase Order integration, you can transfer vendor bills that contain items marked for export to OpenAir into OpenAir as purchase orders. The vendors and items or products used in each line item must also be marked for export to OpenAir and it must be for a customer and project that are already established in OpenAir and have a NetSuite ID. For more information on the necessary configuration, see NetSuite Vendor Bills to OpenAir Purchase Orders.

To enable the transfer of purchase orders to OpenAir, check all three following boxes on the NetSuite Connector administration under Imports:

- **Vendors (Vendors)**
- **Products (Purchase Items)**
- **POs (Vendor Bills)**

After you run the integration to transfer a NetSuite vendor bill to OpenAir, you can view it in OpenAir. Vendor bills transfer to OpenAir as request items, purchase requests, purchase items, purchase orders (POs), and fulfillments. Only NetSuite vendor bills with the following statuses transfer: Open or Paid in Full. Therefore, vendor bills that are Not Approved do not import to OpenAir.

You can bill and rebill items in a vendor bill transferred to OpenAir if they were originally marked as billable on the NetSuite vendor bill. You can also rebill vendor bills transferred to OpenAir even though they were not marked as billable, however, the linkage to the original NetSuite vendor bill is not maintained. See Rebilling Vendor Bills Marked Billable in NetSuite and Rebilling Vendor Bills NOT Marked Billable in NetSuite.

Because OpenAir purchase requests and request items are associated with a single project, each item on the NetSuite vendor bill becomes an OpenAir purchase item and request item. A NetSuite vendor bill maps to one OpenAir purchase order, and each item marked for export to OpenAir maps to an OpenAir purchase item.

To view the Vendor Bill to Purchase Order import in OpenAir:

1. Go to Purchases > POs > All.
2. Select a PO# that is similar to NS-VB-#.

   The PO displays with the NetSuite vendor bill purchase items.
3. Click on a Purchase item to view the contents.
4. To view the OpenAir Invoice that was exported to NetSuite, go to Invoices > Approved and click on the related invoice that was exported to NetSuite.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

NetSuite Purchase Order Integration

Using the NetSuite Purchase Order integration, you can transfer Purchase orders that contain items marked for export to OpenAir into OpenAir as purchase orders. The vendors and items or products used in each line item must also be marked for export to OpenAir and it must be for a customer and project that are already established in OpenAir and have a NetSuite ID. For more information on the necessary configuration, see the NetSuite Purchase Order Integration.

To enable the transfer of purchase orders to OpenAir, check all three following boxes on the NetSuite Connector administration under Imports:

- **Vendors (Vendors)**
- **Products (Purchase Items)**
- **POs (Purchase Orders)**

After you run the integration to transfer a purchase order to OpenAir, you can view it in OpenAir, but you continue to update it in NetSuite. Purchase orders transfer to OpenAir as purchase requests, request items, purchase orders (POs), purchase items, and fulfillments. Only NetSuite purchase orders with the following statuses transfer: Fully Billed, Partially Received, Pending Bill, Pending Billing / Partially Received, and Pending Receipt. The integration does not transfer a NetSuite purchase order with any other status.

Because OpenAir purchase requests and request items are associated with a single project, each item on the NetSuite purchase order becomes an OpenAir purchase request and request item. A NetSuite purchase order maps to one OpenAir purchase order. Relevant lines on the NetSuite purchase order become purchase items on the OpenAir purchase order. Vendor bills do not transfer.

To view the purchase order in OpenAir:

1. Go to Purchases > Application Settings > Purchases Settings.
2. Select one of the following links to display the list view for a state: Purchase requests, POs, or the Fulfillment.
3. Click the transferred purchase order to view it. Each displays with a NetSuite PO #.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Sales Orders in OpenAir

Sales Order records transfer to OpenAir as sales order line items. During the data transfer, associated line item discount information for project data also transfers. For details, see NetSuite to OpenAir Data Mapping and Configuring the Sales Order Integration.

OpenAir creates a unique Customer PO for each sales order line item you enter. In addition, any specified related billing rule information also transfers to OpenAir and is used in subsequent invoices. Sales orders with discounts and mark ups also transfer. You can select one of the following OpenAir Billing Rules:

- Time
- Fixed fee % complete
- Fixed fee on date or milestone

If enabled, you can also create revenue recognition rules along with billing rules in the NetSuite sales order that transfer to OpenAir. You can select one of the following OpenAir billing rules:

- % Complete - OpenAir type = P
- As Billed - OpenAir type = A
- Fixed Amount for % Complete - OpenAir type = F
- Fixed Amount for Date or Milestone - OpenAir type = F
- Incurred vs. Forecast - OpenAir type = I
- Time - OpenAir type = T

To enable the transfer of sales orders to OpenAir, check the following boxes on the NetSuite Connector administration form and relevant workflow settings form:

- NetSuite Connector administration — Imports:
  - Customer POs (Sales Orders). See also Importing NetSuite Sales Orders and Creating OpenAir Invoices.
- Customer POs (Sales Orders) import workflow settings form:
  - Create time billing rules in sales order integration with a service filter. Map sales order line description to billing rule Charge description — See also Importing NetSuite Sales Orders and Creating OpenAir Invoices.

  - Enable recognition rule creation in sales order integration
- Invoices (Invoices) export workflow settings form:
  - Adjust quantity based on % invoiced for time charges related to NetSuite sales order lines with quantities that do not reflect hours — See also Calculating Quantity from % Invoice.
  - Always respect the NetSuite sales order customer and project when exporting invoices — Allows the transfer of NetSuite sales order information such as Sales Team when you export invoices back to NetSuite. See also Exporting NetSuite Sales Order Information with OpenAir Invoice.
  - Create billing rules for NetSuite discount and markup lines on sales orders — See also Creating OpenAir Billing Rules for NetSuite Discounts and Mark Ups
  - Enable charges to map to NetSuite description items — See also Exporting OpenAir Charge Detail to NetSuite Description Item. For custom mapping, see Mapping Description Items When Exporting Invoices.
  - Initialize invoices from corresponding sales orders when exporting invoices to NetSuite — See also Initializing an OpenAir Invoice. For custom mappings, see Mapping Service Sales Order Custom Fields When Exporting Invoices.

If you would like to limit the subsequent transfer of NetSuite sales order data to OpenAir, you can use advanced settings to designate fields as initial-only export. See the instructions for Field Declarations.

Custom mappings are also available in Advanced Mapping and Filter Notation Reference for Mapping Description Items When Exporting Invoices.

To use Sales Order records in OpenAir:

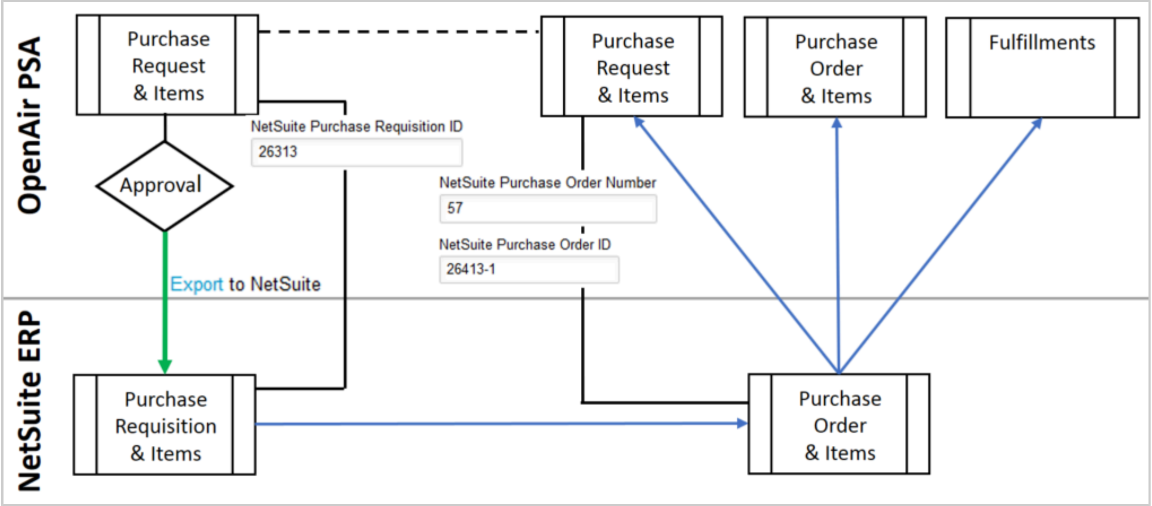
1. Generate charges to use on invoices.
   - If you are using a time or fixed fee OpenAir Billing Rule, go to Projects > Project > Financials > Billing. Click on the billing rule to display and edit the information transferred from NetSuite. Edit the dates on the billing rule. If it is a time billing rule, select either an hourly rate or if it is an other type rate, select a rate card or another non-hourly rate. Click Run billing and create project billing transactions and charges.
   - If you are creating a one-off charge, go to Invoices > Charges and select a new hourly, flat rate, or new other rate charge. Associate the charge with the correct Customer PO. Complete the information and click Save.
2. Go to Invoices > Invoices > New invoice and create an Invoice. Select the customer and sales order item and create the invoice.
3. Generate recognition records and transfer them to NetSuite. Go to Projects > Project > Financials > Recognition. Click on the recognition rule to display and edit the information transferred from NetSuite. Edit the dates on the recognition rule. Click Auto-run, complete requested information, and Click Save. When the rule is run, it creates recognition transactions. Run the integration to send these as journal entries to NetSuite.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Purchase Requisitions in OpenAir

Purchase requisitions can be created in OpenAir and then sent to NetSuite for fulfillment. Purchase requisitions can be created in OpenAir even if the user does not know the exact price or vendor name. This information can later be entered in NetSuite by the requisitions team.

When an OpenAir purchase requisition is integrated with and processed in NetSuite, associated NetSuite purchase orders, purchase items, request items, and fulfillments can be integrated back to the project in OpenAir.



This workflow allows buyers to select or change the vendor or price, and to split or combine requisitions. In addition, buyers can set up automatic purchasing using preferred vendors and pricing.

Users can mark multiple requests as exported or unexported using bulk actions. Access to these bulk actions is controlled by assigning the following user settings to an employee:

- The employee can unexport purchase requests
- The employee can mark purchase requests exported

Note: To enable this feature, check the **Purchase Requests (Purchase Requisitions)** box under Exports.

You can also allow non-administrators to export their own purchase requests to NetSuite based on their role. Go to Administration > Global Settings > Users > Roles and select a role which you would like to be able to export purchase requests. In the "Integration" section, select "Allow employee to export purchase requests to NetSuite".

Once these features are enabled, you can add the Purchase request (Purchase requisition) chapter to your integration mappings.

Working with Expense Report Reimbursements in OpenAir

When an OpenAir user's expense report transfers to NetSuite and is reimbursed in NetSuite, that information transfers back to OpenAir and displays under reimbursements associated with the user's expense report. The date the payment is made, the amount, and the NetSuite Payment ID # all transfer and display in OpenAir. Reimbursements are only applied to expense reports that have a balance that is greater than 0.

In addition, the NetSuite internal ID is stored in a custom field that is available in detail reports. If multiple reimbursements are made for one expense report, they import as multiple transactions.

To enable the transfer of expense report reimbursements to OpenAir, check the **Reimbursements (Expense Report Reimbursements)** box on the NetSuite Connector administration form under Imports.

To add additional information to Expense Report Reimbursement records in OpenAir:

1. Go to Expenses > Expense reports and select the expense report that was reimbursed.
2. Click on the Reimbursements link.

The Reimbursements form displays and lists the date, amount, and NetSuite Payment ID # of the NetSuite reimbursement.

3. To run a detail report, go to Reports > Detail > Reimbursements under Expenses. Under Report layout, select NetSuite Expense Report Reimbursement ID and click OK. Run the report.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Invoice Payments in OpenAir

When you record a customer payment in NetSuite for an invoice generated in OpenAir, a corresponding payment transaction displays in the OpenAir invoice when the integration is run. The date, amount, currency, and NetSuite payment ID transfer. Remember that an invoice must be approved and it must be for a project that has a NetSuite ID. Payments are only applied to OpenAir invoices that have a balance due that is greater than 0. The NetSuite Payment # displays in the OpenAir Notes field.

In addition, the NetSuite internal ID is stored in a custom field that is available in detail reports. If multiple payments are made for an invoice, they import as separate payment transactions.

To use the invoices integration, it must be enabled in both NetSuite and OpenAir. For more information, see Invoices (Invoices) Export Workflow.

To add additional information to Invoice Payment records in OpenAir:

1. Go to Invoices > Invoices > Approved and select the invoice for which payment was made.
2. Click the Payments link.

   The Payments form displays and lists the date, amount, currency, and NetSuite payment ID # of the NetSuite customer payment.
3. To run a detail report, go to Reports > Detail > Payments under Invoices. Under Report layout, select NetSuite Invoice Payment ID and click OK. Run the report.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Customer Deposits Applied to Invoices in OpenAir

When you record a customer deposit in NetSuite and apply it to an invoice generated in OpenAir, the amount transfers to OpenAir when the integration is run and creates a retainer for the deposit amount. Then, OpenAir draws payment for the OpenAir invoice against that retainer. Customer deposits are only applied to approved OpenAir invoices with NetSuite IDs that have outstanding balances greater than 0. The NetSuite Customer Deposit # displays in the OpenAir Notes field. In addition, the NetSuite internal ID is stored in a custom field that is available in detail reports.

If multiple customer deposits are applied to the NetSuite invoice, they import to OpenAir as separate retainer transactions.

To use the invoices integration, it must be enabled in both NetSuite and OpenAir. For more information, see Invoices (Invoices) Export Workflow.

To add additional information to Invoice Retainer records in OpenAir:

1. Go to Invoices > Invoices and select the invoice that is associated with a retainer.
2. Click the Retainer link.

   The Retainer form displays and lists the date, amount, and NetSuite Payment ID # of the NetSuite customer deposit.

3. To run a detail report, go to Reports > Detail > Payments under Invoices. Under Report layout, select NetSuite Deposit Applied To ID and click OK. Run the report.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Credit Memos Applied to Invoices in OpenAir

When you create a credit memo in NetSuite that applies to an invoice transferred from OpenAir, the credit memo transfers back to OpenAir when the integration is run. OpenAir creates a negative invoice that contains charges for each line on the credit memo. In addition, OpenAir displays the total credit amount on the original OpenAir invoice. The integration only imports credit memos applied to approved OpenAir invoices with NetSuite IDs that have outstanding balances greater than 0. The Applied NetSuite credit memo # and amounts display on the OpenAir negative invoice and the NetSuite internal ID is stored in a custom field.

To use the invoices integration, it must be enabled in both NetSuite and OpenAir. For more information, see Invoices (Invoices) Export Workflow as well as the Credit Memo Integration for Invoices.

If you would like charge types to display on NetSuite credit memos, check the **Map the charge type from the NetSuite credit memo** box on the Credits (Applied Credit Memos) import workflow settings form.

To add additional information to credit memo invoices in OpenAir:

1. Go to Invoices > Invoices >Approved and select the negative invoice that is associated with a credit memo.

   The Applied NetSuite credit memo # displays on the invoice and the amount of the credit is subtracted.
2. Click the Change charges link.

   The form displays and lists invoice charges.
3. Click the edit link next to the invoice charge.

   The Edit form for the charge displays.
4. Make your changes and click Save.
5. To run a detail report, go to Reports > Detail > Charges under Invoices. Under Report layout, select NetSuite Applied Credit Memo ID and click OK. Run the report.

Working with OpenAir Data in NetSuite

After the integration is run, OpenAir data transfers to NetSuite. Each record's data can then be viewed and modified in NetSuite. Note that the terminology may differ for the record when in NetSuite. See Terminology Mapping for the Integration for more information.

Review the following topics for details of how to work with each OpenAir record type in NetSuite:

- Working with Expense Reports in NetSuite
- Working with Vendor Bills in NetSuite
- Working with Timesheets in NetSuite
- Working with Invoices in NetSuite
- Working with Revenue Recognition Data in NetSuite
- Working with Customers from Salesforce.com in NetSuite
- Working with Projects from Salesforce.com in NetSuite

Before you can view these records in NetSuite, you must first export the data. See Exporting OpenAir Data to NetSuite for more information.

Working with Expense Reports in NetSuite

You can transfer OpenAir expense reports that have been approved to NetSuite as Expense Reports. They can include receipts that are both billable and non-billable, reimbursable and non-reimbursable, as well as receipts in foreign currencies.

OpenAir receipts have a Reimbursable (`R`) or Non-reimbursable (`N`) **Status**. For non-reimbursable receipts, you can:

- Use advanced mapping to map to mark the corresponding NetSuite expense line as non-reimbursable.
- Configure OpenAir NetSuite Connector to export non-reimbursable receipts as negative expense lines in NetSuite and show the associated Customer : Project on these negative expense lines. When an expense is billed to a customer but not reimbursable to an employee, a negative expense line will be created in addition to the positive expense line in NetSuite. The negative expense lines do not show on the OpenAir expense report, only in NetSuite. To enable these features, check the following boxes on the Expense Reports (Expense Reports) export workflow settings form:
  - **Automatically create negative Expense report lines in NetSuite for non-reimbursable expenses**.
  - **Keep customer reference on negative Expense report lines in NetSuite for non-reimbursable expenses**.

When the OpenAir expense report is transferred to NetSuite, it includes the user, date of the report and the report number, cash advance information as well as notes, currency and approvals. Two multi-select lists of OpenAir expense report approvers are also exported to NetSuite. One is for the supervisor and the other for the project approver. For details, see OpenAir to NetSuite Data Mapping.

You can also use advanced settings and transfer the expense report line tax. All of the fields in the OpenAir tax_location table are available for custom mapping. Refer to Advanced Mapping and Filter Notation Reference and review the tax_location table in the OpenAir data dictionary. :
To view the OpenAir data dictionary, use the following URL: `https://<account-domain>/database/single_user.html#tax_location`.
You can also access the OpenAir data dictionary from the OpenAir Help Center using the link in the navigation bar, if you have the View Help Center role permission.

User currency is set for an employee when the record is transferred as a user account to OpenAir. That becomes the expense report currency that is transferred back to NetSuite on the expense report. OpenAir users can include receipts in an expense report that are in a foreign currency. To suppress setting foreign currency in NetSuite when transferring an expense report, check the **Do not set the foreign currency in NetSuite for foreign currency receipts when exporting expense reports** box on the Expense Reports (Expense Reports) export workflow settings form. This ensures that expense reports that transfer to NetSuite display all receipts in the user's base currency.

If it is helpful to transfer NetSuite transactional data back to OpenAir, you can use advanced settings to do so. See the instructions for Mapping a Transaction Number from NetSuite to an OpenAir Custom Field.

When the feature to export tax on expense reports is enabled, the specific taxes, tax codes, rates and amounts transfer based on the nexus the user is assigned to and the tax location of the receipts in the expense report. Your OpenAir account must be configured to use tax locations on receipts and the Advanced Taxes feature must be enabled for your NetSuite account. See Enable tax export on expense reports.

If you are using the tax export feature, you can also request that the following configuration for your OpenAir account: Allow manual override of Canadian taxes on Expense report export. This feature applies only to Canadian taxes manually entered on receipts in OpenAir that export to NetSuite. When this feature is enabled, you can enter Canadian taxes in a pop-up window that launches from the OpenAir receipt form. See Allow manual override of Canadian taxes on Expense report export.

⚠ **Important:** The Expense Reports (Expense Reports) export workflow is not fully compatible with the SuiteTax feature:

- Tax amounts are calculated correctly when exporting OpenAir expense reports to NetSuite as vendor bills.
- When exporting OpenAir expense reports to NetSuite as expense reports, however, tax details need to be entered manually in NetSuite when using the SuiteTax engine. See Entering Expense Reports with Tax Impact in SuiteTax (external link to NetSuite SuiteAnswers | Answer ID: 65664)

Note that SuiteTax compatibility testing is currently in progress. You should test the NetSuite <> OpenAir integration with the SuiteTax feature on sandbox accounts before you enable the feature in your NetSuite production account. See Known Limitations of SuiteTax (external link to NetSuite SuiteAnswers). See also Legacy Tax or SuiteTax.

To add information to the Expense Report record in NetSuite:

1. Go to Transactions > Employees > Enter Expense Reports > List.
2. Click the Expense Report.

   The expense report number displays in the OpenAir Expense Report Number field and the NetSuite Expense Report Number is set to the same number if you enable expense report numbering to Allow Override. To do so, see Configuring OpenAir for the Integration.

If you receive an error, refer to Troubleshooting the Integration for suggestions on error resolution.

Working with Vendor Bills in NetSuite

You can transfer OpenAir expense reports that have been approved to NetSuite as Vendor Bills for NetSuite vendors or parent vendors/subcontracting companies. OpenAir expense reports can include receipts that are both billable and non-billable as well as receipts in foreign currencies. However, in OneWorld accounts, the vendor and the Customer : Project must be in the same subsidiary.

When the OpenAir expense report is transferred to NetSuite, it includes the vendor, date of the report, currency, exchange rate, due date, amount, expense report number, and OpenAir expense report creator. Two multi-select lists of OpenAir expense report approvers are also exported to NetSuite. One is for the supervisor and the other for the project approver. For details, see OpenAir to NetSuite Data Mapping.

Vendor currency is set for a vendor when the record is transferred to OpenAir as a user account. That becomes the expense report currency that is transferred back to NetSuite on the expense report. OpenAir users can include receipts in an expense report that are in a foreign currency. To suppress setting foreign currency in NetSuite when transferring an expense report, check the **Do not set the foreign currency in NetSuite for foreign currency receipts when exporting expense reports** box on the Expense Reports (Expense Reports) export workflow settings form. This ensures that expense reports that transfer to NetSuite display all receipts in the user's base currency.

If it is helpful to transfer NetSuite transactional data back to OpenAir, you can use advanced settings to do so. See the instructions for Mapping a Transaction Number from NetSuite to an OpenAir Custom Field.

When the feature to export tax on expense reports is enabled, the specific taxes, tax codes, rates and amounts transfer to the vendor bill based on the nexus the user is assigned to and the tax location of the receipts in the expense report. Your OpenAir account must be configured to use tax locations on receipts and the Advanced Taxes feature must be enabled for your NetSuite account. See Enable tax export on expense reports.

> **Note:** If the SuiteTax feature is enabled for your NetSuite account, tax amounts are calculated correctly when exporting OpenAir expense reports to NetSuite as vendor bills.
>
> Note that SuiteTax compatibility testing is currently in progress. You should test the NetSuite <> OpenAir integration with the SuiteTax feature on sandbox accounts before you enable the feature in your NetSuite production account. See Known Limitations of SuiteTax (external link to NetSuite SuiteAnswers). See also Legacy Tax or SuiteTax.

To add information to the Vendor Bill record in NetSuite:

1. Go to Transactions > Purchases/Vendors > Enter Bills > List.
2. Click Edit for the Vendor Bill.

   The vendor bill displays. Expenses are listed under the Expenses & Items tab.

If you receive an error, refer to Troubleshooting the Integration for suggestions on error resolution.

Working with Timesheets in NetSuite

You can transfer OpenAir timesheets that have been approved to NetSuite. They display as time entries in the Time Tracking form and Weekly Time Sheet form. You can include hours that are both billable and non-billable and distinguish between them. If you do not designate a time entry as billable, it defaults to non-billable. Optionally, you can also send the NetSuite Time Entry ID. When the OpenAir timesheet transfers to NetSuite, in addition to time entries, the Employee, Customer : Project, and Service Item also transfer.

To enable the feature to export time entries to NetSuite, check the **Time Entries (Time)** box on the NetSuite Connector administration form under Exports.

To add information to the Time Tracking record in NetSuite:

1. Go to Transactions > Employees > Time Tracking or Weekly Time Sheet.
2. Enter employee and date information.
3. In the Weekly Time Sheet, click the View Time tab.

   OpenAir approved time entry information displays.

If you receive an error, refer to Troubleshooting the Integration for suggestions on error resolution.

Working with Invoices in NetSuite

You can transfer invoices to NetSuite if they are approved and are linked to a customer record and project that contain a NetSuite ID. Information sent includes customer and project ID, invoice date, invoice number, service, rate, and amount information. You can also send project-specific information to the NetSuite invoice header. NetSuite calculates all totals and taxes, and if enabled, can transfer the total tax amount on the invoice back to OpenAir, including GST and PST taxes. In addition, you can initialize the invoice so that the integration gathers additional information from the NetSuite sales order, adds it to the OpenAir invoice information, and transfers it to NetSuite. You can also add charge details from OpenAir to the NetSuite invoice description items.

When an expense charge is manually created in OpenAir, it can be sent to NetSuite if there is a service assigned to it, however, it displays in NetSuite as a service charge instead of a billable expense.

If it is helpful to transfer NetSuite transactional data back to OpenAir, you can use advanced settings to do so. See the instructions for Mapping a Transaction Number from NetSuite to an OpenAir Custom Field.

Custom mappings are also available for Mapping Service Sales Order Custom Fields When Exporting Invoices.

If the Multiple Prices feature is enabled in NetSuite, the NetSuite price level is set to custom. If not, the NetSuite price level is not set.

If the import invoice tax feature is enabled, the total tax amount imports back to OpenAir with the invoice. Check the **Import taxes applied to exported invoices** box on the Invoices (Invoices) export workflow settings form. See the instructions for Invoices (Invoices) Export Workflow and Importing Invoice Taxes as well as for Configuring the Sales Order Integration if you are going to initialize invoices and add charge descriptions.

⚠ **Important:** If the SuiteTax is enabled for your NetSuite account, NetSuite OpenAir SRP Integration bundle 369637 (previously 2851) to version 2.11 or later is required in order for the tax amount calculated in NetSuite to be imported into OpenAir. With earlier versions, the tax amount calculated in NetSuite with SuiteTax cannot be imported back into OpenAir.

Note that SuiteTax compatibility testing is currently in progress. You should test the NetSuite <> OpenAir integration with the SuiteTax feature on sandbox accounts before you enable the feature in your NetSuite production account. See Known Limitations of SuiteTax (external link to NetSuite SuiteAnswers). See also Legacy Tax or SuiteTax.

If the OpenAir invoice URL mapping is enabled, a custom field captures the OpenAir hypertext link on the NetSuite invoice form. See Enable the OpenAir invoice link mapping to NetSuite.

If you enable the feature to combine non-expense charges by category, service, or activity on invoices , non-expense charges roll up and display as one line item under the Items subtab on the NetSuite invoice. First, the integration combines the charges by project, and then it rolls them up by category, service, or activity. There was a previous limitation that involved using this feature in conjunction with the Sales Order integration. That limitation no longer exists. You can use the features in the Invoices integration in conjunction with the features in the Sales Order integration. See Invoices (Invoices) Export Workflow and Configuring the Sales Order Integration.

To add information to the invoice in NetSuite:

1. Go to Transactions > Sales > Create Invoices > List.
2. Click on the Invoice.

   The invoice number displays in the OpenAir Invoice Number field and the NetSuite Invoice # is set to the same number if you enable invoice numbering to Allow Override. For more information on setting, refer to Configuring Additional Settings, Preferences, and Customizations in NetSuite.

If you receive an error message, refer to Troubleshooting the Integration for suggestions on resolution.

Working with Revenue Recognition Data in NetSuite

Send OpenAir project information about revenue recognized from paid invoices to NetSuite. When you generate project billing transactions and charges and then run recognition for a project, the information can be transferred to NetSuite.

If it is helpful to transfer NetSuite transactional data back to OpenAir, you can use advanced settings to do so. See the instructions for Mapping a Transaction Number from NetSuite to an OpenAir Custom Field.

To use the revenue recognition integration, it must be enabled in both NetSuite and OpenAir. For more information, see Configuring Revenue Recognition.

If you are using the revenue recognition integration and would like to transfer revenue recognition transactions incrementally throughout the life of a project to create intercompany journal entries, check the following boxes on the NetSuite Connector administration form under Additional integrations:

- **Enable NetSuite intercompany journal creation for recognition transaction and invoice export**
- **Enable NetSuite subsidiary import**

To add information to the Journal Entry in NetSuite:

1. Go to Transactions > Financial > Make Journal Entries > List.
2. Click on the Journal Entry.

   An invoice is created. All lines are marked as Delayed Rev Rec to indicate a deferred revenue account.

   > **Note:** If the Delayed Rev Rec does not display on the Invoice form, customize the default Invoice form before running the integration. Click on the Customize tab, select Screen Fields and then select Columns. Scroll down and select the Delay Rev. Rec. field.

Working with Customers from Salesforce.com in NetSuite

> **Important:** The 3-way Salesforce, OpenAir and NetSuite integration for customers and projects is deprecated and will be removed at a future date to be announced.
>
> You should use the Projects (Projects) and Customers (Customers) export workflows instead. See Replacing the Deprecated 3-Way Salesforce, OpenAir and NetSuite Integration With the Customers (Customers) or Projects (Projects) Export Workflows.
>
> The **Enable 3-way Salesforce.com, OpenAir and NetSuite integration** setting has been moved to the Legacy compatibility section of the NetSuite Connector administration form. You may continue using the 3-way Salesforce, OpenAir and NetSuite integration if it is already enabled for your account. However, it can no longer be enabled.

You can transfer customers that have been pulled into OpenAir from Salesforce.com to NetSuite. Once the customer record is in NetSuite, it is assigned a NetSuite ID, which is subsequently added to the OpenAir customer record. While information can be updated in OpenAir and Salesforce.com, those updates are not exported back to NetSuite through the integration. For details about data that transfers, see OpenAir to NetSuite Data Mapping.

For users with the Salesforce.com Integration in OpenAir, this feature adds the ability to send customers through OpenAir to NetSuite. For information on this feature and available options, refer to (DEPRECATED) Configuring the Salesforce.com Integration for Customers and Projects.

To add information to a Customer in NetSuite:

1. Go to Lists > Relationships > Customers.
2. Click on the Customer.

Working with Projects from Salesforce.com in NetSuite

⚠ **Important:** The 3-way Salesforce, OpenAir and NetSuite integration for customers and projects is deprecated and will be removed at a future date to be announced.

You should use the Projects (Projects) and Customers (Customers) export workflows instead. See Replacing the Deprecated 3-Way Salesforce, OpenAir and NetSuite Integration With the Customers (Customers) or Projects (Projects) Export Workflows.

The **Enable 3-way Salesforce.com, OpenAir and NetSuite integration** setting has been moved to the Legacy compatibility section of the NetSuite Connector administration form. You may continue using the 3-way Salesforce, OpenAir and NetSuite integration if it is already enabled for your account. However, it can no longer be enabled.

You can transfer projects that have been pulled into OpenAir from Salesforce.com to NetSuite. Once the project is transferred to NetSuite, it becomes a NetSuite record and is assigned a NetSuite ID, which is subsequently added to the OpenAir project record. While information can be updated in OpenAir and Salesforce.com, those updates are not exported back to NetSuite through the integration. For details about data that transfers, see OpenAir to NetSuite Data Mapping.

For users with the Salesforce.com Integration in OpenAir, this feature adds the ability to send projects through OpenAir to NetSuite. For information on this feature and available options, refer to (DEPRECATED) Configuring the Salesforce.com Integration for Customers and Projects.

To add information to a Project in NetSuite:

1. Go to Lists > Relationships > Projects.
2. Click on the Project.

Frequently Asked Questions (FAQs)

The following information provides more detail about how the integration works between the two applications.

Regarding the name and rate of an item sent to OpenAir, what happens if there is no base price in NetSuite?

If there isn't a NetSuite base price, no price is set on the item. However, there could be user-specific additional logic added to the advanced settings in OpenAir for this situation. For example, you might specify that the pricing available in advanced search in web services should be used.

What happens in multi-currency situations when prices on the base currency tab are blank? If base currency is blank and base price is identified on several tabs, how do you find the price? If price is picked up from a foreign currency, does price get converted to base currency?

Currently what happens is that no currency is set for the item. Therefore, the currency is assumed to be the base currency of the OpenAir account. We are addressing this situation in future releases.

When I look at the Mapping table for NetSuite Items, it looks like a custom field is created in OpenAir for netsuite_category_id. What is this?

The netsuite_category_id is for the NetSuite ID of NetSuite items. In OpenAir, items are stored in something called a category. They are also called activities and services. In other words, in OpenAir, a category = service = activity. The actual database table happens to be called category and so the field name created is netsuite_category_id.

In the Entity and Field Mapping table for NetSuite Expense Categories, why isn't there a netsuite_category_id for NetSuite expense categories?

In the OpenAir database, they are called expense items, so the field name is netsuite_item_id. It holds the ID of the NetSuite expense category ID.

OpenAir transactions such as invoices, expense reports, and revenue rec entries are posted into a closed period in NetSuite. How can this be prevented?

This is likely due to the role permissions assigned to the user configured for the integration. If the integration user has the “Override Period Restrictions” role permission enabled, transactions may be posted to a period closed but not locked. For example, assume your accounting period closes on the 28th on a given month, e.g. April, and an employee submits an expense reports on the 29th. If the integration runs on the 30th, the expense report would be posted to April when you would expect all expense reports submitted / approved after the 28th of the month to be posted in May.

- **Solution:** create a new role that limits the ability to override the period restrictions. Ensure that all functional role permissions are set to use the integration properly. See Creating a Custom Role for the Integration in NetSuite.
- **Other considerations:**
  - Accounting Periods in OpenAir will only affect the dates of the transaction itself, not the accounting period. To prevent wrong dates being used on the transaction, you should setup the accounting periods in OpenAir and use accounting dates wherever possible when using complex month closing procedures.
  - Form permissions are crucial to having expenses and invoices flow across the integration properly. Make sure that in addition to the above permissions in the role, all additional permissions that are customized directly for the particular account are included. All entities will fail if custom fields are required but not available due to form permissions.

Data Mapping in the Integration

Introduction

NetSuite and OpenAir exchange records and transactional information. The integration eliminates the need for re-entering data in each application, thereby ensuring accuracy and streamlining workflow.

The following tables show the default entity and field mapping between NetSuite and OpenAir. Although terminology is somewhat different, the specific NetSuite record is identified and the location and fields transferred are listed along with the location of the NetSuite records and transactions in OpenAir and vice-versa.

Note: You can set up user-defined field mapping and filtering on the NetSuite integration form. For more information, see Advanced Mapping and Filter Notation Reference.

Terminology Mapping for the Integration

NetSuite and OpenAir use slightly different terminology. The table below shows the differences between NetSuite and OpenAir terms. For terminology consistency, you may wish to change the default terms in one application or the other.

NetSuite Term	OpenAir Term
Employee	User
Customer	Customer
Contact	Contact
Vendor	Depending on implementation: Vendor User
Project	Project
Project Task	Task
Item	Depending on implementation: Product Activity or Service
Purchase Order	Purchase Order
Billing Schedule	Billing Rule
Invoice	Invoice
Revenue Recognition Transaction	Revenue Transaction
Expense Category	Expense Item
Expense Report	Expense Report
Purchase requisition	Purchase request

Note: Billing schedules in NetSuite do not map to billing rules in OpenAir and are not used in the integration. All invoices created in OpenAir use OpenAir billing rules to determine when and how to bill a customer.

NetSuite to OpenAir Data Mapping

OpenAir pulls records from NetSuite for employees, customers including leads and prospects if enabled, contacts if enabled, projects, service items or all items if enabled, expense categories, vendors if enabled, sales orders if enabled, expense report reimbursements if enabled, invoice payments if enabled, and applied customer deposits if enabled. The following are the default data mappings.

NetSuite Record	Data Transferred	OpenAir Application
Employees	OpenAir User: custom field for netsuite_user_id, indication of active status, nickname, email, and first and last name. If this is a new user, the password is established using a random string and the role assigned is the default new user role. When active status changes to inactive status, the change does not transfer from NetSuite to OpenAir, but must be performed manually in each application.	Administration > Global Settings > Users
Customers Leads Prospects	OpenAir Customer: custom field for netsuite_customer_id, type, active status, name, company email, and address including street address, city, state, and zip code. If multiple addresses exist, the first address listed in NetSuite is used. If the OpenAir account is multi-currency, the customer currency will be set to the NetSuite customer currency. When active status changes to inactive status, the change does not transfer from NetSuite to OpenAir, but must be performed manually in each application.	Administration > Global Settings > Customers > Customers
Contacts	OpenAir Contact: custom field for netsuite customer (must exist before contact is created), custom field for netsuite_contact_id, active status, nickname, email, first and last name, phone, fax, mobile, and job_title. Contact addresses are stored in an address table, not the contacts table, and include address1, address2, address3, city, state, zip, and country. When active status changes to inactive status, the change does not transfer from NetSuite to OpenAir, but must be performed manually in each application.	Administration > Global Settings > Customers > Contacts
Projects	OpenAir Project: custom field for netsuite_project_id, name, customer ID, and start date. If the OpenAir account is multi-currency, the project currency will be set to the project's NetSuite customer currency. This respects the requirement in NetSuite that the project currency should be the same as the customer currency. If setting the OpenAir project stage in NetSuite is enabled, project stage transfers from NetSuite. Also, if revenue recognition is enabled and the NetSuite account is a OneWorld account, the project subsidiary and the subsidiary currency will also be set for the OpenAir project in custom fields. If you are using multi-currency customers in NetSuite, please see the project section in NetSuite Records Available for Export to OpenAir.	Projects > Projects > All

NetSuite Record	Data Transferred	OpenAir Application
Project Tasks	OpenAir Project: custom field for netsuite_project_task_id, name, customer_id, project_id, start_date, currency, and planned_hours. When the NetSuite project task has a constraint type "Fixed Start" the start date will be populated on the OpenAir project task. Otherwise, it is NOT imported into OpenAir. The OpenAir task planned hours are only populated when the task is not a milestone and the hours are not set at the task assignment level.	Projects > Tasks > All
Service Items/Items (all types)	OpenAir Activities: custom field for netsuite_category_id, indication of active or inactive status, name and rate, which is the first item from the price list in NetSuite. See Frequently Asked Questions (FAQs) for more information.	Administration > Application Settings > Invoices > Activities
Items for purchase orders	OpenAir Product: custom field for netsuite_product_id, name, and the following custom fields when revenue recognition is enabled: netsuite_product_inc_acct_id, netsuite_product_def_rev_acct_id, netsuite_product_expense_acct_id, netsuite_product_def_expense_acct_id, and netsuite_product_subtype	Administration > Application Settings > Purchases > Products
Expense Categories	OpenAir Expense Type: custom field for netsuite_item_id, indication of active or inactive status, name of the category, and type. If the NetSuite expense category has mileage in the name, it translates to a type=M, OpenAir mileage expense item. Otherwise, it becomes type=R. Either can be edited.	Administration > Application Settings > Expenses > Expense items
Vendors	OpenAir User: custom field for netsuite_vendor_id or netsuite_vendor2_id, indication of active or inactive status, nickname, email, first and last name, netsuite_export_envelope_to_vendorbill, netsuite_vendor_parent_id, and netsuite_user_tax_nexus (if tax export enabled).	Administration > Global Settings > Users
Vendors	OpenAir Vendor: custom field for netsuite_vendor_id, indication of active or inactive status, name, email, fax, phone, address1, address2, address3, city, state, zip, URL, and currency.	Administration > Application Settings > Purchases > Vendors. Can also access Vendors from Expenses Settings.
Purchase Orders for Purchase Requests	OpenAir Purchase Request: customer_id and project_id for customer and project specified, date, date_required, date_submitted, date_approved, currency, prefix for NS, number, approval_status, notes, and name, which is NS + tranID + line ID	Purchases > Purchase requests > Approved
Purchase Orders for Request Items	OpenAir Request Item: name, vendor_id, product_id, customer_id, project_id, purchaesrequest_id, currency, cost, um (each), quantity, date, notes, user_id, and total	Purchases > Purchase requests > Request items
Purchase Orders for POs	OpenAir POs: date, vendor_id, terms, currency, user_id (OA purchaser), prefix for NS, number (tranID), name (NS + tranID), auto_track_payable_with_fulfilled (yes), carrier_id, accounts_payable_id, receiving_id, and ship_complete_only	Purchases > POs > Approved
Purchase Orders for Purchase Items	OpenAir Purchase Item: name, vendor_id, request_item_id, product_id, customer_id, project_id, pruchaserequest_id, pruchaseorder_id, currency, cost, approved_cost, um (each), quantity, date, notes, user_id (OA purchaser), total, and purchaser_id	Purchases > POs > Purchase items

NetSuite Record	Data Transferred	OpenAir Application
Purchase Orders for Fulfillment	OpenAir Fulfillment: quantity, purchaseorder_id, purchaserequest_id, request_item_id, purchase_item_id, and date	Purchases > Fulfillmenet > Outstanding POs
Expense Report Reimbursements	OpenAir Expense report Reimbursement: netsuite_reimbursement_id, date transaction is created in NetSuite, total, and currency. By default, notes displays the NetSuite Payment # followed by the NetSuite transaction number. This transaction number is established when created, but is never updated.	Expenses > Expense reports > Reimbursed
Invoice Payments	OpenAir Invoice Payment: netsuite_payment_id, date transaction is created in NetSuite, total, and currency. By default, notes displays the NetSuite Payment ID # followed by the NetSuite internal ID of the transaction. This transaction number is established when created, but is never updated.	Invoices > Invoices > Approved invoice > Payment
Applied Customer Deposits	OpenAir Invoice Retainer: netsuite_retainer_id, netsuite_deposit_applied_to_id, customer_id, date transaction is created in NetSuite, total, and currency. By default, notes displays the NetSuite customer deposit ID # followed by the NetSuite internal ID of the transaction. This transaction number is established when created, but is never updated.	Invoices > Retainers
Sales Orders See more Sales Order Info in following: - Billing Rules for Newly Created POs - Revenue Recog. Rules for Newly Created POs	OpenAir Customer Purchase Order - the following custom fields are created: netsuite_sales_order_id for entire sales order, netsuite_sales_order_line_id for a specific line on the sales order, netsuite_sales_order_net_amount for net amount of sales order after discounts or markups, netsuite_sales_order_project_id for the NetSuite internal ID of the associated project for the sales order line, netsuite_sales_order_item_id for the NetSuite internal ID of the item on the sales order line, netsuite_sales_order_line_quantity for the quantity of the sales order line, and netsuite_discount_po for the OpenAir internal ID of a customer PO that is created from an applicable NetSuite line or subtotal discount. Check box fields include: netsuite_sales_order_quantity_is_not_hours for calculate quantity from % invoiced, netsuite_sales_order_line_discount_po for the NetSuite sales order line discount customer PO, and netsuite_sales_order_header_discount_po for the NetSuite sales order header discount customer PO. Also transferred are item name (default name is NetSuite sales order number. If it appears more than once, name is appended with a number 2, 3, 4, etc.), customer_id, and number, total (grossAmount), currency, an indication of active or inactive status, and date (tranDate). Transfer Charge description and activity to Fixed fee billing rules and set a Billing rule filter for Activity on Time billing rules.	Projects >Projects > Edit project and scroll down to Customer POs > Edit Administration > Application Settings > Invoices > Customer POs Invoices > Invoices > New invoice
Sales Orders Billing Rules for Newly Created POs	OpenAir Project Billing Rule: name (default is same as customer PO name), customer_id, project_id, category_id, customerpo_id, an indication of active or inactive status, currency, notes, start_date, end_date, type (NetSuite custom field custcol_openair_billing_rule_type), cap (grossAmount for time rules), amount (grossAmount for fixed fee rules), percent_how (set to A for fixed fee % complete rules), and percent (set to 100 for fixed fee % complete rules). Date on billing rule is automatically set to 1/1/2050 to ensure it is edited before running the rule.	Projects > Projects > select the project > Financials > Billing > select a billing rule from the sales order line item

NetSuite Record	Data Transferred	OpenAir Application
Sales Orders Revenue Recog. Rules for Newly Created POs	OpenAir Project Revenue Recognition Rules: name (default is same as customer PO name), customer_id, project_id, category_id, customerpo_id, an indication of active or inactive status, currency, notes, start_date, end_date, type (NetSuite custom field custcol_oa_rev_rec_rule_type), project_billing_rule_filter if time project billing rule corresponds to OpenAir internal ID, percent_how set to A for % complete rules, percent_trigger set to A for fixed fee % complete rules, percent set to 100 for fixed fee % rules, lc_level set to 0 for incurred vs. forecast rules, and extra_data: - For as billed rules, populate hours in recognition transactions is selected by default. - For incurred vs. forecast rules, approved hours multiplied by user cost is selected by default under Incurred and approved and booked hours multiplied by user cost is selected by default under Forecast. Date on recognition rule is automatically set to 1/1/2050 to ensure it is edited before running the rule.	Projects > Projects > select the project > Financials > Recognition > select a recognition rule from the sales order line item

OpenAir to NetSuite Data Mapping

OpenAir pushes transactional data from OpenAir to NetSuite for invoices, including line items for non-expense charges and line items for expense charges, expense reports, expense items, time entries on timesheets, revenue recognition transactions, and opportunities. Customer and project information in OpenAir can also be pushed to NetSuite. The following are the default data mappings.

OpenAir Record	Data Transferred	NetSuite Application
Invoices	NetSuite Invoices: custom field for customer_id or project_id, date of invoice, number, project_id, and if specified, a custom field for OpenAir Invoice number.	Transactions > Sales > Create Invoices > List
Invoice Line Items (non-expense charges)	NetSuite Invoices: quantity, rate, and project_id.	Transactions > Sales > Create Invoices > List
Invoice Line Items (expense charges)	NetSuite Invoices: custom field for netsuite_envelope_id and total.	Transactions > Sales > Create Invoices > List
Expense Reports	NetSuite Expense Reports: OpenAir custom field for NetSuite user ID, date of expense report, expense report number, OpenAir custom fields for expense report number and expense report approver, advance, notes, indication of multi-currency, and accounting approval and supervisor approval by default.	Transactions > Employees > Enter Expense Reports > List
Expense Receipts	NetSuite Transaction Lines in Expense Reports: OpenAir custom field for NetSuite ID from expense item, date of expense item, notes, customer ID, total from receipt, whether receipt is missing, indication of multi-currency, and indication of whether billable or non-billable based on selected item on OpenAir project form. A workflow setting lets you disable expense report references on invoices.	Transactions > Employees > Enter Expense Reports > List
Expense Reports to Vendor Bills	NetSuite Bills under Expenses & Items: OpenAir custom field for NetSuite ID or NetSuite Parent ID, expense report date, expense report number, expense report approver, OpenAir expense report creator, vendor currency, and notes.	Transactions > Purchases/ Vendors > Enter Bills > List

OpenAir Record	Data Transferred	NetSuite Application
Time Entries on Timesheets	NetSuite Time Entries: OpenAir custom field for NetSuite project ID from project for project and customer, NetSuite employee ID from user, NetSuite service item ID from category/activity/service, date of time entry, and hour and minute from time entry converted to decimal hour. Notes associated with a timesheet transfer, but only approved timesheets export.	Transactions > Employees > Track Time or Weekly Time Sheet and select View Time tab
Revenue Recognition Transaction	NetSuite Revenue Recognition Transactions: OpenAir custom fields for NetSuite Rev Rec journal entries from either the NetSuite Income Account ID or the Netsuite Deferred Revenue Account ID depending upon whether it is a debit or credit line on the journal entry, revenue recognition transaction amount for credit or debit, NetSuite project ID, and any revenue transaction notes. In addition, if the account is multi-currency, NetSuite project exchange rate is transferred. If the account is both OneWorld and multi-currency, NetSuite project subsidiary and NetSuite subsidiary currency are also transferred.	Transactions > Financial > Make Journal Entries > List
Project associated with NetSuite Opportunity	NetSuite Opportunity: OpenAir fields for default_category and planned_hours (the total planned hours in the work breakdown structure by default_category.	Transactions > Sales > Create Opportunities > List
Customers	NetSuite Customers: OpenAir company name, address1 and address2, city, state, zip and country (if the relevant configuration setting is enabled), and currency (if the NetSuite account is multicurrency). The **Export customer to NetSuite** box must be checked on the customer record in OpenAir. **Note:** When exporting customer records to NetSuite using the deprecated 3-way Salesforce, OpenAir and NetSuite integration, the OpenAir custom field `customer_sf_id` is mapped to the NetSuite custom field `CUSTENTITY_OA_SFA_ID` to store the internal ID for the customer record in Salesforce. If the OpenAir custom field `customer_sf_id` is populated, the customer record is automatically marked for Export to NetSuite.	Lists > Relationships > Customers
Projects	NetSuite Projects: OpenAir company name, Netsuite customer ID, and the start date. The **Export project to NetSuite** box must be checked on the project record in OpenAir. **Note:** When exporting customer records to NetSuite using the deprecated 3-way Salesforce, OpenAir and NetSuite integration, the OpenAir custom field `project_sf_id` is mapped to the NetSuite custom field `CUSTENTITY_OA_SFA_ID` to store the internal ID for the project record in Salesforce. If the OpenAir custom field `project_sf_id` is populated, the project record is automatically marked for Export to NetSuite.	Lists > Relationships > Projects
Purchase requisitions	OpenAir Purchase Requests including request items, once approved through the common approval system, are exported to NetSuite as purchase requisitions and purchase requisition	Transactions > Purchases > Enter Requisitions > List

OpenAir Record	Data Transferred	NetSuite Application
	items associated with a NetSuite Purchase Requisition ID, which is linked to the associated Purchase Request in OpenAir. **Note:** NetSuite Purchase Requisitions can be linked to NetSuite Purchase Orders and Items. Purchase Orders and Items can be pulled to OpenAir using the NetSuite Purchase Order & Items to Purchase Requests and Items, Purchase Order and Items, and Fulfillments workflows.	
Revenue Recognition Rule	Data about rule is transferred to a NetSuite Custom Record called **customrecord_oa_arm_rules**	Customization > List, Records & Fields > Record Types
Revenue Recognition Transaction	Data about recognition transaction, including which rule created it, is transferred to a NetSuite Custom Record called **customrecord_oa_arm_rrts**	Customization > List, Records & Fields > Record Types

Advanced Mapping and Filter Notation Reference

If you have complex filtering or field mapping requirements, you can use the OpenAir NetSuite Connector advanced mapping and filter notation for your filter and field mapping definitions.

The following topics provide reference information about how to use advanced mapping and filter notation:

- Advanced Mapping and Filter Notation Syntax Rules
- Legacy Advanced Settings Notation — Notation used in the previous NetSuite Connector, accessible from the Tip menu option **Show relevant old advanced mapping**.
- Field Declarations — Field referenced in your filter and mapping definitions must first be declared.
- Advanced Field Mapping Definition
- Filter Definition
- Schema Reference Documentation

Advanced Mapping and Filter Notation Syntax Rules

The OpenAir NetSuite Connector advanced mapping and filter notation use markup language syntax similar to XML to define hierarchy and query language syntax similar to SQL to define query or filter conditions.

- Advanced mapping and filter notation syntax is case sensitive.
  - All advanced mapping and filter notation syntax must be written using uppercase except when referencing a record type or field name in NetSuite or OpenAir.
  - The case for NetSuite and OpenAir record types and field names must follow the case used in the application. Refer to the NetSuite SOAP Schema Browser and OpenAir Data Dictionary — See Schema Reference Documentation.
- Markup language tags similar to XML tags are used for:
  - Direction and integration point tags (used by the legacy advanced mapping and filter notation. See Legacy Advanced Settings Notation.
  - Mapped field value lists. See Advanced Field Mapping Definition.
- Where markup language tag is used, each opening tag must have a corresponding closing tag and elements must be properly nested.

Legacy Advanced Settings Notation

In the previous NetSuite Connector UI, you had to use direction tags and integration point tags when setting filters and field mapping definitions.

- Direction Tags: <PULL> and <PUSH>
- Integration Point Tags

Direction tags and integration point tags are no longer required, as the advanced mapping is set in the context of the integration workflow and mapping group. This help topic is provided for reference only. When you click **Show relevant old advanced mapping** in the Tips menu, the popup window shows your filters and field mapping definitions in the legacy advanced settings notation.

Direction Tags: <PULL> and <PUSH>

In the legacy advanced settings notation, the direction for the flow of information is specified in the code using the PULL and PUSH commands:

- The PULL command is used to import information from NetSuite into OpenAir. All information you want to import from NetSuite into OpenAir must be defined between a <PULL> opening tag and a </PULL> closing tag.
- The PUSH command is used to export information from OpenAir to NetSuite. All information you want to export from OpenAir to NetSuite must be defined between a <PUSH> opening tag and a </PUSH> closing tag.

Important: There cannot be more than one PULL element and one PUSH element.

Integration Point Tags

In the legacy advanced settings notation, the code specifies which integration point field mapping definitions correspond to. Each integration point defines the NetSuite record type and the OpenAir record type you are transferring information to or from.

Note: Each integration point corresponds to a mapping group in the current NetSuite Connector UI.

- Integration point tags must be used either between a direction opening tag and a direction closing tag.
- Each opening integration point tag must have a corresponding closing tag.
- All field mapping information for an integration point in a specific direction must be defined between the opening tag and the closing tag for this integration point. For example, between <EMPLOYEE_TO_USER> and </EMPLOYEE_TO_USER>.
- See Integration Points Importing NetSuite Information into OpenAir and Integration Points Exporting OpenAir Information to NetSuite.

Integration Points Importing NetSuite Information into OpenAir

Note: This help topic is provided for reference only. Integration point tags were used in the legacy advanced settings notation but are no longer required.

When you click **Show relevant old advanced mapping** in the Tips menu, the popup window shows your filters and field mapping definitions in the legacy advanced settings notation.

The following table lists the integration point you can use to import NetSuite records into OpenAir in execution order. The table also indicates the corresponding integration workflow and the equivalent mapping group on the Mapping tab in the current OpenAir NetSuite Connector UI.

- Use these integration points between a <PULL> opening tag and a </PULL> closing tag.

- All field mapping information for an integration point in a specific direction must be defined between the opening tag and the closing tag for this integration point. For example, between <EMPLOYEE_TO_USER> and </EMPLOYEE_TO_USER>.
- The integration workflow must be enabled before you can define custom field mappings and filters for an integration point.
- Integration points support field mapping definitions and filters, unless otherwise specified.
- For billing rule and revenue recognition integration points:
  - Filters are not supported. Filters are created automatically through the integration.
  - Only valid values should be used. For more information, see the OpenAir data dictionary.
  - Extreme caution should be used when customizing billing rule creation.
  - See also Customizing Revenue Recognition Rules.

Integration Point	Workflow	Mapping Group	Additional information
EMPLOYEE_TO_USER	Employees (Users)	Employee	
CUSTOMER_TO_CUSTOMER	Customers (Customers)	Customer	
JOB_TO_PROJECT	Projects (Projects)	Project	
ITEM_TO_CATEGORY	Services (Service Items)	Service	
EXPENSE_CATEGORY_TO_ITEM	Expense Items (Expense Categories)	Expense Item	
CONTACT_TO_CONTACT	Contacts (Contacts)	Contact	
VENDOR_TO_USER	Employees (Vendors)	Vendor	
VENDOR_TO_VENDOR	Vendors (Vendors)	Vendor	Does not support filters
ITEM_TO_PRODUCT	Products (Purchase Items)	Product	
SO_TO_CUSTOMERPO	Customer POs (Sales Orders)	Customer PO	
REVREC_RULE_FF_DATE	Customer POs (Sales Orders)	Date/Milestone Fixed Fee	▪ Used to customize fixed fee on date or milestone revenue recognition rule creation. ▪ Does not support filters.
REVREC_RULE_PERCENT_COMPLETE	Customer POs (Sales Orders)	Percent Complete	▪ Used to customize % complete revenue recognition rule creation. ▪ Does not support filters.
REVREC_RULE_AS_BILLED	Customer POs (Sales Orders)	As Billed	▪ Used to customize as billed revenue recognition rule creation. ▪ Does not support filters.
REVREC_RULE_TIME	Customer POs (Sales Orders)	Time Project	▪ Used to customize time project revenue recognition rule creation. ▪ Does not support filters.
REVREC_RULE_FF_PERCENT_COMPLETE	Customer POs (Sales Orders)	Fixed Fee Percent Complete	▪ Used to customize fixed fee % complete revenue recognition rule creation. ▪ Does not support filters.

Integration Point	Workflow	Mapping Group	Additional information
REVREC_RULE_INC_FORECAST	Customer POs (Sales Orders)	Incurred vs. Forecast	■ Used to customize incurred vs. forecast revenue recognition rule creation. ■ Does not support filters.
PROJECT_BILLING_RULE_FIXED_FEE	Customer POs (Sales Orders)	Project Fixed Fee Billing	■ Used to customize project fixed fee billing rule creation. ■ Does not support filters.
PROJECT_BILLING_RULE_TIME	Customer POs (Sales Orders)	Time Project Billing	■ Used to customize project time billing rule creation. ■ Does not support filters.
BILLPAYMENT_TO_REIMBURSEMENT	Reimbursements (Expense Report Reimbursements)	Expense Reimbursement	
PAYMENT_TO_PAYMENT	Payments (Invoice Payments)	Invoice Payment	
NS_PR_TASK_TO_OA_PR_TASK	Tasks (Project Tasks)	Task	
PURCHASEORDER_TO_PURCHASEORDER	POs (Purchase Orders)	PO	POs (Purchase Orders) and POs (Vendor Bills) workflows are mutually exclusive.
PURCHASE_ITEM	POs (Purchase Orders)	Purchase item	■ POs (Purchase Orders) and POs (Vendor Bills) workflows are mutually exclusive. ■ Does not support filters.
VENDORBILL_TO_PURCHASEORDER	POs (Vendor Bills)	PO	POs (Purchase Orders) and POs (Vendor Bills) workflows are mutually exclusive.
VENDORBILL_TO_PURCHASE_ITEM	POs (Vendor Bills)	Purchase Item	■ POs (Purchase Orders) and POs (Vendor Bills) workflows are mutually exclusive. ■ Does not support filters.
DEPOSIT_TO_RETAINER	Retainers (Applied Customer Deposits)	Retainer	
CREDIT_MEMO_TO_CREDIT	Credits (Applied Credit Memos)	Credit Memo	
SUPPORTCASE_TO_TASK	Tasks (Support Cases)	Support Case	
VENDORCREDIT_TO_PURCHASEORDER	POs (Vendor Credits)	PO	
VENDORCREDIT_TO_PURCHASE_ITEM	POs (Vendor Credits)	Purchase Item	Does not support filters.
NS_INVOICE_TO_OA_INVOICE	Invoices (Invoices)	Invoice	■ Pullback integration point. Used to import NetSuite invoice information back into OpenAir when exporting an invoice from OpenAir to NetSuite. For example, you can use this to import NetSuite transaction numbers to an OpenAir custom field. ■ Does not support filters.

Integration Point	Workflow	Mapping Group	Additional information
NS_EXPENSE_REP_TO_OA_EXPENSE_REP	Expense Reports (Expense Reports)	Expense report	■ Pullback integration point. Used to import NetSuite expense report information back into OpenAir when exporting an expense report from OpenAir to NetSuite. For example, you can use this to import NetSuite transaction numbers to an OpenAir custom field. ■ Does not support filters.
JOURNAL_TO_REVENUE	Recognition Transactions (Revenue Recognition Transactions)	Recognition transaction	■ Pullback integration point. Used to import NetSuite journal entry information back into OpenAir when exporting a revenue recognition transaction from OpenAir to NetSuite. □ The target OpenAir field must be a custom field for each field mapping. □ For example, you can use this to import a standard field in NetSuite, such as transaction numbers, to an OpenAir custom field. Transaction.tranID (NetSuite field) —> Revenue_recognition_transaction.custom_field__c (OpenAir field) ■ Does not support filters.

Integration Points Exporting OpenAir Information to NetSuite

Note: This help topic is provided for reference only. Integration point tags were used in the legacy advanced settings notation but are no longer required.

When you click **Show relevant old advanced mapping** in the Tips menu, the popup window shows your filters and field mapping definitions in the legacy advanced settings notation.

The following table lists the integration point you can use to export OpenAir records to NetSuite in execution order. The table also indicates the corresponding integration workflow and the equivalent mapping group on the Mapping tab in the current OpenAir NetSuite Connector UI.

- Use these integration points between a <PUSH> opening tag and a </PUSH> closing tag.
- All field mapping information for an integration point in a specific direction must be defined between the opening tag and the closing tag for this integration point. For example, between <OA_CUSTOMER_TO_NS_CUSTOMER> and </OA_CUSTOMER_TO_NS_CUSTOMER>.
- The integration workflow must be enabled before you can define custom field mappings and filters for an integration point.
- Integration points support field mapping definitions and filters, unless otherwise specified.

Integration Point	Workflow	Mapping Group	Additional information
OA_CUSTOMER_TO_NS_CUSTOMER	Customers (Customers)	Customer	—
PROJECT_TO_JOB	Projects (Projects)	Project	—

Integration Point	Workflow	Mapping Group	Additional information
PURCHASE_REQUEST_TO_PURCHASE_REQUISITION	Purchase request (Purchase requisition)	Purchase request	Used to export OpenAir purchase requests as NetSuite purchase requisitions.
PURCHASE_REQUEST_ITEM_TO_PURCHASE_REQUISITION_ITEM	Purchase request (Purchase requisition)	Request item	■ Used to export OpenAir purchase request items as line items on NetSuite purchase requisitions. ■ Does not support filters.
EXPENSE_REP_TO_EXPENSE_REP	Expense Reports (Expense Reports)	Expense Report	Can be used to define filters for the EXPENSE_REP_TO_VENDORBILL integration point.
EXPENSE_REP_TO_VENDORBILL	Expense Reports (Expense Reports)	Vendor Bill	■ Used to export OpenAir expense reports as NetSuite vendor bill headers. ■ Does not support filters. Can use filters defined for the EXPENSE_REP_TO_VENDORBILL integration point.
EXPENSE_REP_LINE	Expense Reports (Expense Reports)	Expense Line	■ Used to export OpenAir receipts as NetSuite expense lines. ■ Does not support filters. This is to prevent inconsistent expense reports in NetSuite and OpenAir. ■ DO NOT map the nonreimbursable OpenAir field. This would cause significant impact downstream when rebilling expenses on invoices.
EXPENSE_REP_LINE_TO_VENDORBILL_LINE	Expense Reports (Expense Reports)	Vendor Bill Line	■ Used to export OpenAir receipts as NetSuite vendor bill lines. ■ Does not support filters.
EXPENSE_REP_LINE_TAX	Expense Reports (Expense Reports)	Tax Line	■ Used to export OpenAir receipts as NetSuite tax lines (related to the tax_location table). ■ Does not support filters.
INVOICE_TO_INVOICE	Invoices (Invoices)	Invoice	■ Used to export invoice header information.
INVOICE_PROJECT	Invoices (Invoices)	Project	■ Used to export OpenAir project information as NetSuite invoice header information. ■ Requires the **One project per invoice** Invoices application setting to be enabled (Administration > Application Settings > Invoices > Other settings). ■ Does not support filters.
INVOICE_LINE	Invoices (Invoices)	Invoice Line	■ Used to export OpenAir charges as line items on NetSuite invoices. ■ Does not support filters.
INVOICE_EXPENSE_LINE	Invoices (Invoices)	Expense Line	■ Used to export OpenAir expense item charges as billable expense lines on NetSuite invoices. ■ Can be used to export OpenAir expense item charges created from expense reports exported to NetSuite as expense reports or vendor bills. ■ If an expense charge was created manually in OpenAir, it can only be exported to NetSuite if there is a service assigned to it. It will be exported using the INVOICE_LINE integration point and will display as a line item (service

Integration Point	Workflow	Mapping Group	Additional information
			charge) instead of a billable expense line on the NetSuite invoice. - Does not support filters.
INVOICE_PURCHASE_LINE	Invoices (Invoices)	Purchase Line	- Used to export OpenAir product charges as NetSuite line items. - Can be used to include purchase item information to the NetSuite line item. - Does not support filters.
INVOICE_BILLABLE_ITEM_LINE	Invoices (Invoices)	Bill Line	- Used to export OpenAir product charges as NetSuite billable items lines when OpenAir charges are created from NetSuite vendor bill lines imported into OpenAir as purchase items. - Can be used to export charge information to the NetSuite billable items lines. - Used to rebill NetSuite vendor bills to the end customer through OpenAir when at least one of the following conditions applies: ◦ The **NetSuite vendor bill line billable** box (custom field) on the Purchase Order form in OpenAir is NOT checked — the vendor bill is NOT marked billable in NetSuite. ◦ The **Disable vendor bill linkage on invoices in NetSuite integration** setting is enabled on the **NetSuite Connector administration** form. - Does not support filters.
INVOICE_PURCHASE_VB_LINE	Invoices (Invoices)	Vendor Bill Line	- Used to export OpenAir product charges as NetSuite billable items lines when OpenAir charges are created from NetSuite vendor bill lines imported into OpenAir as purchase items. - Can be used to export purchase item information, including any information from the original NetSuite vendor bill line, to the NetSuite billable line items. - Used to rebill NetSuite vendor bills when both the following condition apply: ◦ The **NetSuite vendor bill line billable** box (custom field) on the Purchase Order form in OpenAir is checked — the vendor bill is marked billable in NetSuite. ◦ The **Disable vendor bill linkage on invoices in NetSuite integration** setting is NOT enabled on the **NetSuite Connector administration** form. - Must not be used to export OpenAir expense item charges created from expense reports exported to NetSuite as vendor bills. Use INVOICE_EXPENSE_LINE instead. - Does not support filters.
INVOICE_LINE_TIME_GROUP_BY	Invoices (Invoices)	Group by Time	- Used only together with the Customer POs (Sales Order) import integration workflow and if the **Enable charges to map to NetSuite description items** workflow setting is enabled.

Integration Point	Workflow	Mapping Group	Additional information
			▪ Used to export OpenAir time charges (for example, hourly charges) as NetSuite description line items on NetSuite invoices. ▪ If the **Combine charges by service on invoices** workflow setting is enabled, all OpenAir time charges, including time charges that are not associated with a sales order, are grouped under one description line item on the NetSuite invoice. ▪ Does not support filters.
INVOICE_LINE_OTHER_TIME_GROUP_BY	Invoices (Invoices)	Group by Other	▪ Used only together with the Customer POs (Sales Order) import integration workflow and if the **Enable charges to map to NetSuite description items** workflow setting is enabled. ▪ Used to export OpenAir other charges (for example, daily charges) as NetSuite description line items on NetSuite invoices. ▪ If the **Combine charges by service on invoices** workflow setting is enabled, all OpenAir other charges, including other charges that are not associated with a sales order, are grouped under one description line item on the NetSuite invoice. ▪ Does not support filters.
INVOICE_LINE_FIXED_FEE_GROUP_BY	Invoices (Invoices)	Group by Fee	▪ Used only together with the Customer POs (Sales Order) import integration workflow and if the **Enable charges to map to NetSuite description items** workflow setting is enabled. ▪ Used to export OpenAir fixed fee charges as NetSuite description line items on NetSuite invoices. ▪ If the **Combine charges by service on invoices** workflow setting is enabled, all OpenAir fixed fee charges, including fixed feecharges that are not associated with a sales order, are grouped under one description line item on the NetSuite invoice. ▪ Does not support filters.
TIMESHEET	Time Entries (Time)	Timesheet	Used to export OpenAir timesheets as NetSuite time entries
TIMEENTRY_TO_TIMEBILL	Time Entries (Time)	Time Entry	Used to export OpenAir time entries as NetSuite time entries.
REVENUE_TO_JOURNAL	Recognition Transactions (Revenue Recognition Transactions)	Recognition Transaction	Used to export information from OpenAir revenue recognition transactions as header information on NetSuite journal entries.
REVENUE_PURCHASE_TO_JOURNAL	Recognition Transactions (Revenue Recognition Transactions)	Purchases Revenue Recognition	▪ Used to export information from OpenAir revenue recognition transactions as header information on NetSuite journal entries when revenue recognition transactions are created from OpenAir purchases revenue recognition rules. ▪ Should be used only if you need to export purchase item information to the NetSuite

Integration Point	Workflow	Mapping Group	Additional information
			journal entry header (if the `fulfillment_id` field on revenue recognition transactions has a non-null value).
`JOURNAL_LINE`	Recognition Transactions (Revenue Recognition Transactions)	Journal Entry Line	■ Used to export information from OpenAir revenue recognition transactions as journal entry lines on NetSuite journal entries. ■ Does not support filters.
`JOURNAL_LINE_CREDIT_ PURCHASE`	Recognition Transactions (Revenue Recognition Transactions)	Credit Purchase Journal Entry Line	■ Used to export information from OpenAir revenue recognition transactions as credit journal entry lines on NetSuite journal entries when revenue recognition transactions are created from OpenAir purchases revenue recognition rules. ■ Should be used only if you need to export purchase item information to the NetSuite journal entry line (if the `fulfillment_id` field on revenue recognition transactions has a non-null value). ■ Does not support filters.
`JOURNAL_LINE_CREDIT_ PURCHASE`	Recognition Transactions (Revenue Recognition Transactions)	Credit Purchase Journal Entry Line	■ Used to export information from OpenAir revenue recognition transactions as credit journal entry lines on NetSuite journal entries when revenue recognition transactions are created from OpenAir purchases revenue recognition rules. ■ Should be used only if you need to export purchase item information to the NetSuite journal entry line (if the `fulfillment_id` field on revenue recognition transactions has a non-null value). ■ Does not support filters.
`JOURNAL_LINE_DEBIT_ PURCHASE`	Recognition Transactions (Revenue Recognition Transactions)	Debit Purchase Journal Entry Line	■ Used to export information from OpenAir revenue recognition transactions as debit journal entry lines on NetSuite journal entries when revenue recognition transactions are created from OpenAir purchases revenue recognition rules. ■ Should be used only if you need to export purchase item information to the NetSuite journal entry line (if the `fulfillment_id` field on revenue recognition transactions has a non-null value). ■ Does not support filters.
`JOURNAL_LINE_CREDIT`	Recognition Transactions (Revenue Recognition Transactions)	Credit Journal Entry Line	■ Used to export information from OpenAir revenue recognition transactions as credit journal entry lines on NetSuite journal entries (income account or revenue). ■ Does not support filters.
`JOURNAL_LINE_DEBIT`	Recognition Transactions (Revenue Recognition Transactions)	Debit Journal Entry Line	■ Used to export information from OpenAir revenue recognition transactions as debit journal entry lines on NetSuite journal entries (deferred revenue). ■ Does not support filters.

Integration Point	Workflow	Mapping Group	Additional information
`CREDIT_TO_CREDIT_MEMO`	Send OA Credit (negative invoice) as NS credit memo	OA Credit (negative invoice)	■ Used to export OpenAir invoices with a negative total amount as NetSuite credit memos. ■ Does not support filters.
`PR_TASK_TO_OPP_ITEM`	WBS (Opportunity)	Task	■ Used to export OpenAir project tasks with manually entered billing and cost projections as line items on NetSuite opportunities. ■ Does not support filters.
`PTA_JOB_CODE_TO_OPP_ITEM`	WBS (Opportunity)	Task assignment	■ Used to export OpenAir project task assignments with calculated billing and cost projections as line items on NetSuite opportunities. ■ Does not support filters.

Field Declarations

Before you reference NetSuite or OpenAir field names to set filter or field mapping definitions using the advanced mapping and filter notation, you must first declare these fields.

Field declaration statements must:

- Use the correct syntax. See Field Declaration Syntax.
- Specify the field type. See Field Types Used for Field Declaration.
- Specify at least one field name.
- Be used between an integration point opening and closing tags if you are using the **Advanced Settings** text box on the legacy NetSuite integration form in OpenAir to set filters and field mapping definitions.

Field Declaration Syntax

Use the following syntax for field declaration statements.

```
FIELD_TYPE field_name_1, field_name_2, Field_name_3
```

- Field declaration statements include the field type followed by at least one space or blank line and a comma-separated list of field names.
- The field type must be specified using uppercase
- The case for NetSuite and OpenAir field names must follow the case used in the application. Refer to the NetSuite SOAP schema browser and OpenAir data dictionary.
- The list of field names must not include line breaks. Only commas and spaces can be used.
- Field declaration statements do not use opening and closing tags.

Single field declaration example:

```
OA_FIELDS project_stage_id
```

Multiple field declaration example:

```
OA_FIELDS address1, address2, address3, city, state, zip
```

Field Types Used for Field Declaration

The following table lists the different field types used for field declaration and their usage. Note that fields must be declared more than once if they fall under different field types

Field type (notation)	Field Type (plain)	Usage
NS_FIELDS	NetSuite fields	Used to declare all NetSuite fields, both standard and custom fields.
NS_CUSTOM_FIELDS	NetSuite custom fields	Used to declare all NetSuite custom fields. NetSuite custom fields must also be declared using NS_FIELDS.
OA_FIELDS	OpenAir fields	Used to declare all OpenAir fields, both standard and custom fields.
OA_CUSTOM_FIELDS	OpenAir custom fields	Used to declare all OpenAir custom fields. OpenAir custom fields must also be declared using OA_FIELDS.
OA_FIELDS_INITIAL_ONLY	OpenAir fields for the initial import only	Used to declare OpenAir fields for which a value should be set only the first time the record is imported into OpenAir (when the OpenAir record is created). These OpenAir fields must also be declared using OA_FIELDS.
OA_CUSTOM_FIELDS_ INITIAL_ONLY	OpenAir custom fields for the initial import only	Used to declare OpenAir custom fields for which a value should be set only the first time the record is imported into OpenAir (when the OpenAir record is created). These OpenAir custom fields must also be declared using OA_FIELDS, OA_FIELDS_INITIAL_ONLY and OA_CUSTOM_FIELDS.
OA_FIELDS_GROUP_BY	Group by OpenAir fields	**Important:** Supported only in the following mapping groups (integration points) when used in conjunction with the Customer POs (Sales Order) import integration workflow. ■ Invoices (Invoices) export workflow □ Group by Time (INVOICE_LINE_TIME_GROUP_BY) □ Group by Other (INVOICE_LINE_OTHER_TIME_GROUP_BY) □ Group by Fee (INVOICE_LINE_FIXED_FEE_GROUP_BY) Used to declare the OpenAir fields used to group charges by when exporting OpenAir charges to description line items on NetSuite invoices. All charges with the same values for the declared fields will be grouped and exported as one description line item on the NetSuite invoice. Can be used to declare OpenAir charge (slip) fields only. Declared fields must also be declared using OA_FIELDS and, for custom fields, OA_CUSTOM_FIELDS.

Field type (notation)	Field Type (plain)	Usage
OA_FIELDS_SORT_BY	Sort by OpenAir fields	**Important:** Supported only in the following mapping groups (integration points) when used in conjunction with the Customer POs (Sales Order) import integration workflow. ■ Invoices (Invoices) export workflow □ Group by Time (INVOICE_LINE_TIME_GROUP_BY) □ Group by Other (INVOICE_LINE_OTHER_TIME_GROUP_BY) □ Group by Fee (INVOICE_LINE_FIXED_FEE_GROUP_BY) Used to declare the OpenAir fields used to sort charges by when exporting OpenAir charges to description line items on NetSuite invoices. Description line items on the NetSuite invoice will be sorted in ascending order of values for each declared field. Can be used to declare OpenAir charge (slip) fields and related record type fields only. Use the following format to declare fields from related record types: table_name.field_name. For example charge fields include user_id, you can use OA_FIELDS_SORT_BY user.name to sort by the name of the employees instead of their internal ID. Declared fields, or the foreign key (if the declared field is from a related record type) must also be declared using OA_FIELDS and, for custom fields, OA_CUSTOM_FIELDS.
NS_CUSTOM_FIELDS_FROM_SO_INVOICE_HEADER	—	**Note:** Supported only in the following mapping group when used in conjunction with the Customer POs (Sales Order) import integration workflow, and with the **Initialize invoices from corresponding sales orders when exporting invoices to NetSuite** workflow setting enabled. ■ Invoices (Invoices) export workflow □ Invoice (INVOICE_TO_INVOICE) Used to declare custom fields from header information on NetSuite sales orders that should be set explicitly in the header information of the NetSuite invoice when exporting invoices from OpenAir to NetSuite. Should only be used if the default value on the NetSuite sales order can be overridden.
NS_CUSTOM_FIELDS_FROM_SO_INVOICE_LINE_ITEM	—	**Note:** Supported only in the following mapping group when used in conjunction with the Customer POs (Sales Order) import integration workflow, and with the **Initialize invoices from corresponding sales orders when exporting invoices to NetSuite** workflow setting enabled. ■ Invoices (Invoices) export workflow □ Invoice Line (INVOICE_LINE) Used to declare custom fields from items lines on NetSuite sales orders that should be set explicitly on the items lines of the NetSuite invoice when exporting invoices from OpenAir to NetSuite. Should only be used if the default value on the NetSuite sales order can be overridden.

Advanced Field Mapping Definition

You can use the advanced mapping notation to set field mapping definitions on the OpenAir NetSuite Connector Mapping tab.

The following table lists the field mapping definition type with a description of usage and syntax, including the syntax used on the legacy NetSuite integration form if different. Review also the Advanced Mapping Best Practice Guidelines.

Source value	Usage	Syntax	Legacy syntax (if different)
Field value	Simple field mapping — Set the field in the target application to the same value as the field in the source application.	*Not required* unless used with conditional branching. The Mapping tab lets you select fields from a dropdown lists informed by the schema. Use the legacy syntax with conditional branching.	Target field name, followed by at least one space and source field name. `target_field source_field`
Fixed value	Set a field value in the target application to a fixed value.	Unless used with conditional branching, enter the equal sign followed by a value contained within single quotation marks in the Source field column. Example: `='123'` Use the legacy syntax with conditional branching.	Target field name, followed by at least one space and a value contained within a pair of single quotation marks. `target_field '123'`

Combined field and fixed string values	Set the field value in the target application to a concatenated string combining strings from multiple field values in the source application or fixed values.
	`\\n` can be used in fixed value strings to insert line breaks. Example: `'String\\n\\with\\nline\\nbreaks'`.
		Target field name, followed by at least one space and a concatenation expression. The concatenation operator is a comma and can be used to combine any number of field string values as well as fixed string values contained within a pair of single quotation marks.
		`target_field source_field_1,'fixed string value',source_field_2`
		For example, use this syntax to map URL custom fields in OpenAir to URL custom fields in NetSuite for import. URL custom field values in NetSuite include the URL only, whereas URL custom field values in OpenAir include both the visible link text and the link URL. If the NetSuite URL field value is `https://www.example.com/example`, the corresponding URL field value in OpenAir should be `[Example of visible link text](https://www.example.com/example)`. To import URL custom field from NetSuite into OpenAir, use the following advanced field mapping:
		`target_field '[Example of visible link text](',source_field,')'`
			—

Source value	Usage	Syntax	Legacy syntax (if different)
		Concatenation can be used with looked up OpenAir field values. The following example maps a list/record custom entity field holding the internal ID of the parent project in NetSuite to a pick list custom field in OpenAir. Pick list custom field values sourcing list items from projects include the internal IDs of the customer [`customer_id`] and the project [`project_id`] separated by a colon [`:`]. The mapping involves two looked up OpenAir field values and some concatenation. `parent_project lookup=custentity_parent_project:lookup_table=project:lookup_by=netsuite_project_id:lookup_return=customer_id,':',lookup=custentity_parent_project:lookup_table=project:lookup_by=netsuite_project_id:lookup_return=id`	
Today's date (Macro)	Set the field value in the target application to today's date.	Target field name, followed by at least one space and `_TODAY` `target_field _TODAY`	—
Empty (Macro)	Exclude a mapping. Can be used to exclude default mapping, for example.	*Not required* unless used with conditional branching. The Mapping tab lets you deactivate the field mapping row. Use the legacy syntax with conditional branching.	Target field name, followed by at least one space and `_REMOVE` `target_field _REMOVE`

Field value mapped to target field value	Set the target field values according to a list of mapped source field value – target field value pairs. In the example given, if the source field value is 127, the target field value will be set to `4970`.
	`<target_field source_field>`
	`'126' '4971'`
	`'127' '4970'`
	`</target_field>`
		Opening tag with the target field name followed by a space and the source field name. List of mapped target and source field values, one target field value and one source field value per line, separated by one space and with each value contained within single quotation marks. Closing tag with the target field name.
		`<target_field source_field>`
		`source_value_1 target_value_1`
			—

Source value	Usage	Syntax	Legacy syntax (if different)

	⚠ **Important:** Mapped values in the left column cannot contain spaces. You can refer to the values using internal ID of the record instead. For example, use the internal ID on Department records to map the department name field values. Alternatively, you can use conditional branching.	`    source_value_2 target_value_2`
		`</target_field>`	

Looked up OpenAir field value	Look up a field value in a related OpenAir table and set the target field to this looked up value, if a matching record exists in the related table and if the looked up value exists.
	Can be used with concatenation (See “Combined field and fixed string values” entry in this table).
	Can be used with conditional branching (See “Conditional branching value” entry in this table).
		Target field name, followed by at least one space, and all required lookup parameter-value pairs. Each parameter-value pair must be written in the format `parameter=value` and separated by colons (`:`).
		Four parameters are required:
		- `lookup` - The lookup field. The field in the source application containing the value to be looked up.
		- `lookup_table` - The lookup table. The OpenAir table containing the looked up information.
		- `lookup_by` - The field in the lookup table containing the lookup field value.
		- `lookup_return` - The field in the lookup table containing the value the target field should be set to.
		`job_code_id   lookup=custentityuser_free_text:`
		`              lookup_ta`
		`ble=job_code:lookup_by=name:`
		`              lookup_return=id`
			—

Conditional branching value	Set the target field value using different mapping definitions depending on different conditions. Conditions can be set to compare a source field value with a fixed value or with another source field value. Conditions can be set to compare a looked up OpenAir field value with a fixed value or with another source field value.	Conditional branching block opening statement `IF <<END` (space between `IF` and `<<`, no space between `<<` and `END`), followed by a line break, a series of `IF logical_expression THEN mapping ELSE alt_mapping` each separated by a line break, and a conditional branching block closing statement `END` on a separate line. Logical expressions can be formed of multiple conditions (relational expressions) combined using logical operators.	—

<table>
<thead>
<tr><th>Source value</th><th>Usage</th><th>Syntax</th><th>Legacy syntax
(if different)</th></tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Conditions can be combined using the AND and OR logical operators.
Conditional branching mappings are always applied after standard mappings when the integration runs.
Generally, custom fields referenced in the default mapping definitions cannot be overridden with these custom mappings.</td>
<td>
<ul>
<li>Each relational expression consists of two operands separated by a relational operator and must be contained within parentheses.</li>
<li>Available relational operators: =, <=, >=, <, >, and <> (not equal)</li>
<li>Relational operands can be source field values, including looked up OpenAir field values, or fixed values.</li>
<li>When using a looked up OpenAir field value as a relational operand, include the full lookup syntax between parentheses.</li>
<li>Available logical operators in order of precedence: AND, OR.</li>
<li>Parentheses can be used to modify operator precedence.</li>
</ul>
<pre>
IF <<END
   IF (email = 'mcollins@example.com') OR (email
 = 'admin@example.com') THEN email lastName
   IF (firstName = 'John') THEN email entityID
 ELSE email 'nothing'
END
</pre>
Example of conditional branching using looked up OpenAir field values:
<pre>
IF << END
   IF (lookup=user_id:lookup_ta
ble=user:lookup_by=id:lookup_return=net
suite_vendor2_id) <> '' THEN custcol
inv_line_emp lookup=user_id:lookup_ta
ble=user:lookup_by=id:lookup_return=netsuite_ven
dor2_id
   IF (lookup=user_id:lookup_ta
ble=user:lookup_by=id:lookup_return=net
suite_user_id) <> '' THEN custcolin
v_line_emp lookup=user_id:lookup_ta
ble=user:lookup_by=id:lookup_return=netsuit
e_user_id
END
</pre>
⚠ <b>Important:</b> There cannot be more than one conditional branching block on the same mapping group tab. Include all conditional branching (IF ... THEN ...) mapping definitions on the same mapping definition row. Leave the destination field empty on that row.
</td>
<td></td>
</tr>
</tbody>
</table>

Advanced Mapping Best Practice Guidelines

Review the following best practice guidelines:

- You should contact OpenAir Professional Services if you need to make changes to your integration configuration and field mapping definitions.
- Make sure you familiarize yourself with the NetSuite and OpenAir technical documentation before attempting to set field mapping definitions using the advanced notation — See Schema Reference Documentation.
- Declare each NetSuite or OpenAir field you reference in your advanced field mapping definitions — See Field Declarations.
- **Checkbox fields** — You can use an advanced mapping to map a checkbox field in NetSuite with a checkbox field in OpenAir. Note that the values corresponding to the state of the checked or cleared state of the box are different in NetSuite and OpenAir. Refer to the following table and example:

Checkbox State	NetSuite Value	OpenAir Value
Checked	`'true'`	`'1'`
Cleared	`'false'`	`''` (empty value)

```
NS_FIELDS custentity_my_netsuite_checkbox
NS_CUSTOM_FIELDS custentity_my_netsuite_checkbox
OA_FIELDS my_openair_checkbox
OA_CUSTOM_FIELDS my_openair_checkbox
IF <<END
   IF (my_openair_checkbox = '1') THEN custentity_my_netsuite_checkbox 'true'
   IF (my_openair_checkbox = '') THEN custentity_my_netsuite_checkbox 'false'
END
```

Filter Definition

You can use the advanced filter notation to define filter conditions for mapping groups (integration points) on the OpenAir NetSuite Connector Mapping tab.

This topic describes the syntax and guidelines to follow when defining filters.

- Fields referenced in your filter conditions must be declared first. See Field Declarations.
- Filter conditions must be contained within an opening and closing statement. See Filter Opening and Closing Statement.
- A logical expression can be used to combine multiple filter conditions. See Filter Conditions.

Filter Opening and Closing Statement

You need to contain your filter conditions within an opening and closing statement.

- Opening filter block statement: `FILTER <<END` – Must be on a separate line and include one space between `FILTER` and `<<`, no space between `<<` and `END`.
- Closing filter block statement: `END` – Must be on a separate line.

> **Note:** The filter block contains only filter conditions. Fields referenced in your filter conditions must be declared before the filter block. See Field Declarations.

```
      . . . field declarations go here . . .

FILTER <<END

      . . . filter conditions go here . . .

```

```
END
```

Filter Conditions

You can use a logical expression to define multiple filter conditions (relational expressions) combined using logical operators.

- Each relational expression consists of two operands separated by a relational operator and must be contained within parentheses.
- Available relational operators: =, <=, >=, <, >, and <> (not equal).
- The first operand can be:
  - A single source field value (the value of a standard or custom field native to the record type in the source application). This fields must be declared first. See Field Declarations.

    > **Note:** When the first operand is a NetSuite field of type RecordRef, the second operand should be a single NetSuite Internal ID or a list of NetSuite internal IDs.

  - A field from a related record type in OpenAir. Use the following format to reference a field from a related record type: `table_name.field_name`. The foreign key for the related OpenAir table must be declared first. See Field Declarations.
- The second operand can be:
  - A fixed value. The fixed value must be contained within single quotation marks — For example, `'123'`.
  - A list of comma–separated fixed values. The entire list must be contained within a pair of single quotation marks. Spaces are permitted. Containing each listed value within quotation marks is not valid and will return an error. For example:
    - Correct: `'123, 456, 789'`
    - Incorrect: `'123’,'456,'789'`
- Available logical operators in order of precedence: `AND`, `OR`.
- Parentheses can be used to modify operator precedence.

Example 7. Example 1 – Filter records based on multiple conditions

In the following example, only records that have a `department` value 1 or 3, or a `firstName` value John are imported from NetSuite into OpenAir.

```
NS_FIELDS department,firstName

FILTER <<END
      (department = '1,3') OR firstName='John'
END
```

Example 8. Example 2 – Filter records based on a Boolean custom field

In the following example, a custom field is used to filter invoices exported to NetSuite. The check box custom field `do_not_export_invoice__c` is associated with Invoice records. When the box is checked, the OpenAir invoice is not exported to NetSuite.

```
OA_FIELDS           do_not_export_invoice__c
OA_CUSTOM_FIELDS    do_not_export_invoice__c

FILTER <<END
      do_not_export_invoice__c <> '1'
END
```

Example 9. Example 3 – Exclude records based on related records

In the following example, records associated with the OpenAir projects with internal ID 8, 9 or 105 are not exported to NetSuite.

```
OA_FIELDS project_id

FILTER <<END
      project_id <> '8,9,105'
END
```

Example 10. Example 4 – Exclude records based on field values on related records

In the following example, OpenAir records (in this case, time entries) associated with OpenAir projects that have the `do_not_export_project_time__c` box (custom field) checked are not exported to NetSuite.

```
OA_FIELDS project_id

FILTER <<END
      (project.do_not_export_project_time__c <> '1')
END
```

Schema Reference Documentation

NetSuite and OpenAir provide technical documentation for identifying record types and field names.

- NetSuite Schema Browser and WSDL
- OpenAir Data Dictionary

NetSuite Schema Browser and WSDL

The NetSuite SOAP Schema Browser provides a web-based summary of all records, sublists, and other objects available in NetSuite SOAP web services. The SOAP Schema Browser should be used in conjunction with the Records Guide, which is available in the NetSuite Help Center. See SuiteTalk SOAP Web Services Records Guide (external link to NetSuite SuiteAnswers | Answer ID: 86929).

Make sure you use the SOAP Schema Browser corresponding to the correct communication endpoint.

- Use the 2020.1 SOAP Schema Browser (Online| Zipped).
- Use the 2023.1 SOAP Schema Browser (Online| Zipped).

Note: Refer to the object and field names on the Schema Browser tab and not the internal IDs on the Records Browser. In particular, the object names on the Schema Browser tab are different or have different case to the internal IDs on the Records Browser.

When importing transaction records, or pulling back information from transaction records, the NetSuite fields are listed as `Transaction.<fieldName>` in the field mapping definition tables when in fact the SOAP web services object corresponding to the related searches are used to fetch the information from NetSuite.

For more details about the SOAP Schema Browser, see the following:

- Finding a Record or Subrecord (external link to NetSuite SuiteAnswers | Answer ID 38194)
- Finding Other Objects (external link to NetSuite SuiteAnswers | Answer ID 38193)
- Default View (external link to NetSuite SuiteAnswers | Answer ID 38195)
- Namespace View (external link to NetSuite SuiteAnswers | Answer ID 38196)

In addition, the NetSuite SOAP web services WSDL provides a complete description of available services, operations, and messages.

- Use the 2020.1 SOAP WSDL (Online| Zipped).
- Use the 2023.1 SOAP WSDL (Online| Zipped).

OpenAir Data Dictionary

The OpenAir Data Dictionary provides a summary of all tables, and standard fields in OpenAir.

The Database Guide also provides an overview of the corresponding tables and their relationships for most record types in OpenAir with cross-references to the OpenAir Data Dictionary for details. See Database Guide.

Note: To view the OpenAir Data Dictionary, use the following URL: `https://<account-domain>/database/single_user.html`.

- The URL includes the domain name for your OpenAir account `<account-domain>`. For more information about your account-specific domain name, see the help topic Your OpenAir Account URLs.
- To view the details of a specific table, append a hash symbol # followed by the table name to the end of the data dictionary URL. For example, use `https://<account-domain>/database/single_user.html#project` to view the details of the Project table.
- You can access the data dictionary from the OpenAir Help Center using the link in the navigation bar if you have the View Help Center role permission.

Support and Troubleshooting

Support for the Integration

Contacting NetSuite and OpenAir

In order to use the integration, you must have sign-in access to NetSuite and OpenAir.

If you are experiencing difficulties or would like additional information, please create a support case and submit it through your OpenAir account. Our support staff and engineers will work with you to find a solution to your problem.

To create a support case:

1. Sign in to your OpenAir account and select **Support** from the User Center
2. Click on the **Go to SuiteAnswers** button.
3. From the **SuiteAnswers** site home page, click **Contact Support Online**.
4. Enter your question keywords and click **Search**.

> **Note:** If you do not have a question, i.e. you need an optional feature or account configuration option, just click Search.

5. Very often the answer to your question will be displayed. If you still want to create a support case click **Continue to Create Case**.
6. Fill out the **Create Case** form and then click the **Submit**. You will receive an email confirmation with **Your OpenAir Customer Care #**.

> **Note:** An asterisk * displays after required fields.

Getting Help

All OpenAir users have searchable access to a rich repository of answers with SuiteAnswers. SuiteAnswers is your one-stop destination for support, training, and documentation inquiries. In direct response to customer demand for more self-service, SuiteAnswers provides searchable access to support solutions, best practices, training videos and technical documentation.

The SuiteAnswers support and learning site is available from the NetSuite and the OpenAir Customer Support pages.

Troubleshooting the Integration

With user accounts in NetSuite and OpenAir, you should be able to exchange information between the two applications. Review the following integration guidelines for reading error messages and diagnosing integration errors, as well as the error messages, possible causes, and steps to correct. If you are still having problems, contact OpenAir Customer Support. The support staff and engineers will work with you to find a solution to your problem.

Reading Error Messages

When an error occurs, the integration returns an error message that contains information to help you determine the cause of the error. Refer to the following error message and table that explains its contents.

Error Message: NetSuite web service error: Export failed for “1251”. Error occurred while trying to add record #0: Invalid doc, line reference key 6050,1.

This code...	refers to...
Export failed for 1251	1251 is the OpenAir invoice number. When you tried to export the OpenAir invoice to NetSuite, the export failed.
Invalid doc, (6050)	6050 is the NetSuite ID for the expense report. The reason the invoice export failed is that it includes charges from the NetSuite expense report 6050. There is an error in the NetSuite expense report 6050.
line reference key 1	1 is the tracking number of the receipt within the NetSuite expense report that is causing the problem. Go to NetSuite expense report 6050 and look at receipt 1. When you correct the problem there, you can try to export the invoice through the integration and it should be successful.

Note: By default, additional debugging information is enabled. The integration attempts to debug export errors and returns a message to help you fix an issue. The integration processes each error and makes additional calls to NetSuite web services. While this adds to total processing time when you run the integration if there are errors, it does not impact processing time when there are no errors.

Diagnosing Integration Errors

The integration uses NetSuite web services, which is a form-based system. This means that, in general, the behavior you see in the NetSuite user interface (UI) is the same behavior as is available in NetSuite web services. If the integration gives you an error, you are usually not able to perform the same action in the NetSuite UI.

A simple way to diagnose an integration error is to attempt to perform the same action manually in the NetSuite UI. Refer to the following example.

Example: If an expense report fails to export, try to recreate that expense manually in NetSuite as the integration user. This can often lead to the root cause of the error such as an expense category is inactive or an employee no longer exists.

Note: It is important that you attempt do this as the integration user or as a user with a role and permissions that are the same as the integration user. See Creating and Assigning an Access Token for the Integration for integration user information.

Error Messages

The following are error messages you might encounter as you use the integration. Suggestions are provided for resolving them. In the OpenAir application, they display in red. For ease of reading, they are in regular type with the main focus of the error in bold.

We urge you to search for a portion of the message you receive. That way, you can quickly find a possible cause and solution.

Error Message	Possible Causes	Steps to Correct
Default NetSuite forms - You do not have permissions to **set a value** for element *field1, etc...* due to one of the following reasons: 1) The field is read-only 2) An associated feature is disabled 3) The field is available either when a record is created or updated, but not in both cases.	Default NetSuite forms are respected when the integration is run, including all permissions on the form. Generally, if a user gets this error, it is because the form does not have a *field1* available or the fields are read-only.	Verify that the fields in questions are available and not read-only on the user's default form. The standard field mappings in the integration use fields that are all exposed on the standard NetSuite form. A user-defined field mapping (advanced settings) can also be used to remove those fields from the export.
Invoice export error - You do not have permissions to **set a value** for element *doc,line* due to one of the following reasons: 1) The field is read-only 2) An associated feature is disabled 3) The field is available either when a record is created or updated, but not in both cases.	1) This error is specific to the invoice export. All non-manually created billable expenses on an OpenAir invoice should come from an expense report that was exported to NetSuite. 2) The integration references expense reports and requires corresponding expenses on the NetSuite expense report be marked billable. 3) Expense lines might not be marked billable or may have already been billed on another invoice. Also, the customer needs to be the same on the expense report and invoice.	1) Find the OpenAir expense reports that have the associated charges on the invoice. 2) Find the corresponding expense reports in NetSuite and ensure that the corresponding expenses have been marked billable. 3) Ensure the expenses have not already been billed on another invoice. 4) Ensure the customers on the invoice is the same customer as on the expense report.
Invoice billing rule error - Please enter a value for OpenAir Billing Rule.	When the integration was run, the bundle installed a transaction column custom field: OpenAir Billing Rule. It may display on the default invoice form used in the integration.	Check the NetSuite default invoice form. If the OpenAir Billing Rule field is a column on the Items subtab, customize the form so that it does not display that field.
Invoice service error - Export failed for "XXXX". The service *"ABC"* does not have a NetSuite ID.	In OpenAir, the activity, category, or service on an invoice charge does not have a NetSuite ID.	In OpenAir, add a NetSuite ID to the activity/ category/service.
Invoice service error - Export failed for "XXXX". The service *"service"* does not have a NetSuite ID	1) In OpenAir, one or more charges on an invoice do not have an activity, category, or service associated with them. 2) To export, all charges on an invoice must reference a NetSuite activity, category, or service and have a NetSuite ID. 3) This error can occur when an expense charge is manually created in OpenAir via Invoices > Charges instead of from an expense billing rule in the Projects application.	1) Assign a service (with a NetSuite ID) to the manually created expense charge and export the invoice again. While the invoice exports, the expense charge does not display on the NetSuite invoice as a billable expense. It displays as a service charge in the Items subtab. 2) Delete the charge in OpenAir, create an expense report with the expense, export the expense report to NetSuite, and re-run the expense billing rule on the project. This re-creates the expense charge associated with an expense report and the charge can be added back to the invoice XXXX so you can export the invoice.
Invoice export error — Export failed for "XXXX": The service *"Service_Name "* (internal ID YYYY) does not have a NetSuite Deferred	In OpenAir, a Service or a Product associated with one or more charges on an invoice	Manually update Service or Product records in OpenAir if Direct Revenue Posting is enabled for the corresponding Item records

Error Message	Possible Causes	Steps to Correct
Revenue Account ID and/or NetSuite Income Account ID.	does not reference a NetSuite Deferred Revenue Account ID or NetSuite Income Account ID. This error can occur when Direct Revenue Posting is enabled for the Item record in NetSuite but not for the corresponding Service or Product record in OpenAir.	in NetSuite. See NetSuite Direct Revenue Posting.
Invoices export error - Export failed for "1042". Error occurred while trying to add record #0: Invalid **job** reference key 28972 for **entity 26897**	There is a mismatch between customer and projects on Invoices exported from OpenAir. This may happen when a customer is deleted in NetSuite and it is not reflected in OpenAir.	In OpenAir, look for the project with NetSuite ID 28972. Find the correct customer for this project in NetSuite. Take the NetSuite internal ID of the customer and overwrite the NetSuite ID of that customer in OpenAir.
Invoices export error for billable expense report reference - NetSuite web service error: Export failed for "**1040**". Error occurred while trying to add record #0: No line items match the entered id(s) **68737, 1.** Non-manually created billable expenses must reference an expense report and the line of that expense report in NetSuite.	1) In the example, OpenAir invoice **1040** has billable expenses on an OpenAir expense report that has a **NetSuite ID 68737** . The first (**1**) expense is causing an issue, i.e., 68737,1. 2) When expense reports are exported, each line on the expense report has an internal ID in NetSuite. Those internal IDs are just ordinals; the first expense line ID = 1, the second line ID = 2, etc... 3) However, in this case, the expense report lines in NetSuite no longer correspond directly to the expense lines in OpenAir. Possible causes: 1) "Combine Detail Items on Expense Reports" is enabled in NetSuite (Setup --> Accounting --> Accounting Preferences --> Time and Expenses (subtab) --> Expenses section. It must be disabled. 2) Or, lines were added/deleted on the expense report in NetSuite.	1) If possible, delete the expense report in NetSuite and then re-export it from OpenAir. 2) If not possible, manually remove the expense charges from the OpenAir invoice and export the invoice to NetSuite, and then re-enter the expense charges manually in NetSuite.
Invoice export error - Export failed for "XXXXX". Error occurred while trying to add record #0: Invalid price reference key - **1** for item 68.	1) The price level with internal ID -1 is not present on the NetSuite default invoice form on the items tab. The price level with internal ID -1 is the "Custom" price level, which is set for all service items when the multiple prices preference is enabled in NetSuite. 2) The price level column is either not present on the default invoice form or the rate field is hidden on the	1) Check that the price level column is present on the NetSuite default invoice form. 2)Then, check that the "Custom" value is available. If it is not available, make sure that rate column is present on the default invoice form. Once it is on the form, the price level should have the "Custom" value.

Error Message	Possible Causes	Steps to Correct
	invoice form, which causes the "Custom" price level from appearing on the price level drop-down field.	
Invoice export error - The total cannot be negative.	This error occurs when exporting invoices that have a total < 0. NetSuite does not support negative invoices.	Create the invoice manually in NetSuite and mark it as exported. Contact OpenAir Customer Support and request engineering assistance with this task. See Getting Help.
Invoice export error - Export failed for "3261". Error occurred while trying to add record #0: Please enter a value for Tax Code	This error occurs when taxes are not set up properly in NetSuite. Taxes are required for the customer, but the tax code is not automatically set in NetSuite. 1) Tax nexus of customer with associated tax codes and/or groups may not be properly set up. 2) Customer is not properly set up in NetSuite for the taxes although taxes are required for that customer.	1) Properly setup the tax nexus of the customer in NetSuite. 2) Also, make sure that the customer is set up to use the correct tax code. Contact NetSuite Customer Support for assistance. See Getting Help.
Invoice export error - NetSuite web service error: Export failed for "499". You cannot initialize invoice: invalid reference **2322**	The sales order with internal ID **2322** is unable to be billed for some reason. You can see this in the NetSuite UI by navigating to the sales order. The "Next Bill" button is not available. Causes: 1) The sales order has already been fully billed. 2) The sales order is potentially pending approval. 3) The maximum quantity on the sales order for a line has already been billed even though the dollar amount has not been fully billed.	Once you determine the cause of the error, you can take steps to resolve it. 1) If the sales order is pending approval, see if it can be approved. 2) Verify the maximum quantity has not already been fully billed.
Invoice export error - NetSuite web service error: Export failed for "**PS1036**". Error occurred while trying to add record #0: Please specify an email address to send to	This could occur with the invoice export when the customer record in NetSuite has a preference enabled to send invoices to that customer via email but no email address is available for that customer.	Check the customer record in NetSuite. 1) Navigate to the "Info" tab and check the "Send Transactions Via" preference on that tab. 2) If transactions are to be sent by email, check that an email address has been provided for that customer.
Invoice fax number error - Export failed for "INOA1050". Error occurred while trying to add record #0: Please specify an fax number to send to.	The NetSuite customer associated with this invoice requests that their invoices are faxed, but the invoice does not set the fax number.	The NetSuite customer should either: 1) be modified NOT to have invoices faxed to them. 2) specify a fax number using a user-defined field mapping. See Advanced Mapping and Filter Notation Reference.
NetSuite web service error: Export failed for "XXX". The charge with NS OpenAir ID YYYY on this invoice references NetSuite expense	1) The **Apply invoice line in NetSuite to expense using OpenAir ticket id** NetSuite	1) Check the **Apply invoice line in NetSuite to expense using OpenAir ticket id** box on the NetSuite Connector administration form

Error Message	Possible Causes	Steps to Correct
report with ID ZZZZ line 2. The NS OpenAir receipt has reference number 1. This line does not exist in NetSuite because the line has been deleted or the internal IDs do not match. Please check to see if the expense line still exists.	Connector administration setting is not enabled in OpenAir and the Credit Card feature is used in NetSuite. 2) The expense report is not approved in NetSuite (approval routing is enabled in NetSuite). 3) The intercompany time and expenses feature is not enabled for the NetSuite account but the Expense is from a different subsidiary than the associated Customer. 4) The expense report is already used for an invoice in NetSuite.	in OpenAir. Make sure the **Expense Report line ID** is visible on the default Expense Report form in NetSuite. See Apply invoice line in NetSuite to expense using OpenAir ticket id. 2) Manually approve the expense report in NetSuite, or disable the approval routing feature for expenses in NetSuite (Setup > Accounting preferences — Approval routing subtab). 3) In NetSuite, enable the intercompany time and expenses feature (Setup > Setup manager > Enable features — Accounting subtab), or make sure the customer and expense report are within the same subsidiary. 4) Link the invoices between NetSuite and OpenAir manually, or remove the invoice from NetSuite and export it again from OpenAir.
Sales order-related NetSuite web service error - Export failed for "643". Error occurred while trying to add record #0: That is not a valid record transformation.	The sales order is for a cash sales, not invoice. The error indicates that you cannot invoice the sales order because it is for a cash sale.	Change the cash sale associated with this OpenAir invoice to a sales order.
Expense Item or Service error -NetSuite internal ID for an expense item or service is no longer valid - Invalid *something* reference key 5	1) In OpenAir, look for the expense item or service with NetSuite ID = 5. Verify the expense category or service item also **exists** and is **active** in NetSuite. 2) If the item does not exist in NetSuite, create it. 3)change the item on the expense report or invoice in OpenAir to something else or update the NetSuite ID on the existing item in OpenAir.	1) If the item is inactive in NetSuite, mark it as active and run the integration again. 2) If the item does not exist in NetSuite, create it and run the integration again. If you want to correlate it with the OpenAir item, change the NetSuite ID = # to the new number.
Expense report export error - Export failed for "**Expense report for** 03/09/09 to 03/15/09 - number 159" Error occurred while trying to add record #0: Invalid **customer reference key 37177 for entity 33**.	1) This is an expense report error. There is confusing terminology in that *customer* actually means *project*. 2) This error means that the NetSuite employee with internal ID 33 does not have access to enter expenses against the NetSuite project with an internal ID 37177.	1) This could be caused by the flag "Limit Time to Assignees" on the NetSuite project. By default, projects in NetSuite should have this flag disabled when the bundle is installed. However, users may reset the flag. The user should verify that this flag is disabled (not checked) on the corresponding project in NetSuite. In this example, locate the project with NetSuite ID = 37177 in OpenAir. Look for that project in NetSuite. In NetSuite, the flag is on the "General" subtab of the project. 2) It is also possible that the employee and the project are in separate subsidiaries. 3) Or, there may be some other role or permission issue.
Expense report export error - Project associated with an Expense report does not have NetSuite project ID - Export failed: "Expense report for 03/16/09 to 03/22/09" is **not associated with a NetSuite job.**	The project associated with this expense report does not have a NetSuite ID.	1) Find the project in NetSuite, make sure it transfers to OpenAir, and type the NetSuite internal ID of the project into the OpenAir project.

Error Message	Possible Causes	Steps to Correct
		2) Or, create a new project in NetSuite and copy the internal ID of the project to the NetSuite ID field in OpenAir.
Expense report approver and/or supervisor error - Invalid **custbody_oa_exp_approver_supervisor** reference key 12200.	Expense report export sets both the supervisor and project approver if these are available in OpenAir. These are mapped to custom fields on the expense report transaction in NetSuite. These two fields source their data from an approvers list in NetSuite. 1) This error means that the NetSuite employee with internal ID 12200 is not available in the supervisor approver custom field in NetSuite. 2) This could be due to the employee being inactive or some other reason.	1) In OpenAir, find the user with NetSuite ID 12200. Remember, you can customize the User list view to list the NetSuite ID. 2) Find this user/employee in NetSuite and make sure that this person is flagged as an approver to appear in that custom field.
Expense report export error - Invalid category reference key **XXX** for entity **YYY**.	This may appear when expense reports are exported. The error means that the user with NetSuite ID **YYY** does not have access to the expense category (OpenAir expense item) with NetSuite ID **XXX**. One possible cause is that the G/L account associated with expense category **XXX** is not visible to the subsidiary of the user **YYY**.	In NetSuite, make sure that the subsidiary of user YYY is able to see the expense category XXX.
Expense report tax export - Export failed for “Envelope/Expense Report Name - number X”. Error occurred while trying to add record #0. Please enter a value for Tax Code.	This error may occur with the expense report export when taxes are required on the expense report line in NetSuite but it was either not set up in NetSuite or it is not being passed from OpenAir. If taxes are set up in OpenAir for the expenses, you may need to enable the feature “Enable tax export on expense reports” on the Expense Reports (Expense Reports) workflow settings form.	Make sure that the NetSuite Tax Code ID on the OpenAir Tax Location form matches the NetSuite Internal ID of the Tax Code or Tax Group. To verify: ■ In OpenAir, go to Administration > Global Settings > Organization > Tax Locations. Verify the ID on the list view or edit the location form to change the code. ■ In NetSuite, go to Setup > Accounting > Tax Codes or Tax Groups. Make sure that the OpenAir Expense Report includes appropriate location and tax fields. To modify the form, go to Administration > Application Settings > Expenses > Expense report layout. The shipping address or billing address on the NetSuite Employee form establishes the NetSuite Tax Nexus Type for each employee. Verify the address is present in NetSuite. The NetSuite Tax Nexus Type is populated when the user is transferred to OpenAir and the nexus displays on the OpenAir User Demographic form.

Error Message	Possible Causes	Steps to Correct
		Note: Ensure the following internal account configuration setting is not enabled: Get the receipt tax location from the expense item. This conflicts with the Tax calculation method selection: Tax locations.
Employee ID error - Export failed for "Jan 5-8 at Pepsico in Dallas - number 134" Error occurred while trying to add record #0: Invalid **entity reference key 12470**	1) Errors like this may occur on expenses when there is no longer an employee with an internal ID 12470 in NetSuite. 2) This may also be caused by the vendor to user integration. That integration allows the import of NetSuite vendors and creates OpenAir users. Expenses entered for those users cannot be exported to NetSuite since they are not employee entities but vendor entities. 3) Another possible scenario is that employee with NetSuite ID 12470 is inactive or there is a date in the release date field of the employee in NetSuite.	1) In OpenAir, find the user with NetSuite ID 12470. Look for the corresponding entity in NetSuite. 2) Ensure the corresponding entity in NetSuite is a NOT vendor. Vendors cannot submit OpenAir expense reports to NetSuite. 3) Ensure the user is assigned as a resource to the project. Make sure the Limit Time to Assignees is NOT selected and Allow Expenses is selected. 4) If the employee does not exist at all in NetSuite, a new employee needs to be created in NetSuite and the internal ID of that new employee should be used as the NetSuite ID for the corresponding OpenAir user.
Advanced settings error - Export failed for "267". Error occurred while trying to add record #0: org.xml.sax. SAXException: 'value' on {urn:core_2008_2.platform.webservices. netsuite.com}StringCustomFieldRef is required	Check the advanced settings. There is likely a bad field mapping. Possibly a field name is misspelled.	1) Make sure that the NetSuite field names match what is available in the NetSuite schema. 2) See NetSuite developers documentation for the schema browser.
Advanced settings error - ARRAY(0x27f92b30) is missing IF/THEN for EXPENSE_REP_LINE	The error indicates that you are using more than one IF block in a specific mapping section. In the example here, it is the EXPENSE_REP_LINE section.	All of the IF statements should be in one block within the <EXPENSE_REP_LINE> section in this example. Make sure you have all of the IF statements in one IF block, not multiple IF blocks in one mapping section.
Advanced settings error - NetSuite web service error: org.xml.sax.SAXException: is not a legal value for {urn:types.core_2008_2.platform.webservices.netsuite.com}RecordType	This is related to a bad user-defined custom mapping, most likely a field in a SOAP request is set to a value of blank when blank is not a valid value. For example, on an export, a NS drop-down field is set to blank but in the NS UI, it cannot be set to blank.	Review the user-defined custom mappings. 1) Try to isolate the error to a specific transaction which can often be seen by simply running the integration. Once you have done that, look at the field mappings. You may need to capture the SOAP request, examine which fields are set to a blank internal ID. Then add some additional logic to map a value to the field only when there is a value, not when it is blank. 2) Also, make sure that you are not attempting to map from a custom list in NetSuite; only a custom field.
Employee/User error - Invalid payrollitem reference key **98** for employee **693**	The payroll type with a NetSuite internal ID **98** on a time entry for the OpenAir user with NetSuite internal ID **693** is:	1) Make sure that the NetSuite payroll item ID is correct.

Error Message	Possible Causes	Steps to Correct
	1) possibly invalid 2) Or, the employee in NetSuite does not have access to this payroll item.	2) Also check this employee in NetSuite and make sure that the NetSuite payroll item with internal ID 98 is valid for this employee.
Project integration error - *Project name* was not updated because the NetSuite customer for this project has changed.	The NetSuite internal ID associated with *project name* is already associated with a different customer/project association in OpenAir.	This should never occur unless: 1) multiple NetSuite accounts have been used to sync against this OpenAir account 2) internal IDs have been manually changed on projects. This will require manual cleanup of the internal IDs of the projects in the OpenAir account.
Server SuiteScript error - Unable to retrieve OpenAir billing rule types from NetSuite.	This error may occur when Server SuiteScript has not been enabled in the NetSuite account. This is required for the integration to function properly.	Check that Server SuiteScript has been enabled. Setup -> Company -> Enable Features -> SuiteFlex. See Contacting NetSuite and OpenAir for assistance.
Customer PO ERROR: Invalid revenue recognition rule type: '`XYZ`' (NetSuite id `N`).	The custom records defining the supported revenue recognition rule types on NetSuite — **OpenAir: Revenue Rules** (`customrecord_oa_rev_rec_rules`) must be set correctly.	In NetSuite, go to Customization > Records > OpenAir: Revenue Rules, view the list and verify that OpenAir: Revenue Rules records are listed correctly. See Customizing Revenue Recognition Rules.
ERROR: Could not add Recognition rule 'NetSuite Sales Order xxxxxx ' to project ' xxxxxx- xxxxxx' - row_1: Please enter an amount.	The custom records defining the supported revenue recognition rule types on NetSuite — **OpenAir: Revenue Rules** (`customrecord_oa_rev_rec_rules`) must be set correctly.	In NetSuite, go to Customization > Records > OpenAir: Revenue Rules, view the list and verify that OpenAir: Revenue Rules records are listed correctly. See Customizing Revenue Recognition Rules.
SuiteSignOn error - Account mismatch from initial call to verify call.	This is a SuiteSignOn error as a result of being unable to clearly identify the OpenAir account and user in question.	Make sure you have at least Internal ID and Account as part of the "User Identification" tab on the SuiteSignOn record in NetSuite. If you have been in multiple OpenAir accounts, clear out your cache and browser cookies and then try it again.
Web service error - Connection to webservices.netsuite.com timed out.	1) There may be a network or system issue. 2) The NetSuite account is heavily customized so when the integration attempts to retrieve NetSuite account features and preferences to be used in the integration, the request times out.	Verify there aren't any network or system issues by signing in directly into the NetSuite UI.
Web service error - custentity_oa_rate is not a valid search custom field.	In this example, custentity_oa_rate is a custom field in the NetSuite account, but the following applies to all similar messages. 1) The custom field does not store values and so the field is not available for query in web services.	Check that the custom field actually stores a value in the database. 1) In NetSuite, go to Setup -> Customization -> type of custom field -> name of your custom field. 2) Look for the "Store Value" checkbox. It should be checked if this field is to be used in web service queries. 3) Verify that the custom field is valid for searching. To do so, create a “Saved Search”

Error Message	Possible Causes	Steps to Correct
	2) Custom fields in NetSuite can be set up to only display a value, but nothing is stored in that database. 3) The field references another field and just displays without storing anything. 4) When the web service query is issued, there is nothing in the database for this field and so there is nothing to return.	and include the field in the saved search. If you cannot do it, the field is not available.
Web service error - Export failed for "PS-5499". Error occurred while trying to add record #0: Invalid doc,line reference key *17519,1*.	The error means that the line 1 on the expense report with NetSuite internal ID 17519 in NetSuite, does not appear in the "Billable Expenses" tab when you try to create an invoice in the NetSuite UI. There are several possible causes: 1) Has it already been rebilled on another charge/invoice? 2) Has the expense report in NetSuite or OpenAir been altered? after the initial approval and export to NetSuite? 3) Is the expense non-billable in NetSuite?	1) If the expense has already been re-billed, remove that charge from the OpenAir invoice. 2) If the expense is non-billable but should be, change this on the NetSuite expense report. 3) If the expense reports have been altered in either or both systems, the simplest would be to delete the NetSuite version of the expense report, un-export and re-export the OpenAir expense report, and re-export the invoice again. 4) If you cannot re-export the invoice, manually create that invoice in NetSuite and then have engineering mark the OpenAir invoice as exported to NetSuite. See Contacting NetSuite and OpenAir.
Web Service error - Export Failed for "Idaptive media courrier fees Dec 2010 - number 112" Error occurred while trying to add record#0: Invalid custbody_oa_exp_approver_project reference key *280*.	1) The expense report export is attempting to set the expense report project approver in NetSuite. 2) The project approver is set in a custom field in NetSuite called custbody_oa_exp_approver_project but the employee corresponding to ID *280* does not appear in that list.	Add the employee with ID 280 to the list of project approvers in NetSuite.
Web Service error - {urn:core_2008_2.platform.webservices.netsuite.com}SearchColumnSelectCustomField must be used to search custom field custentity_projectmanage	1) The error is related to the custom field changing its type in NetSuite. 2) NetSuite custom field attributes are cached in OpenAir after the first time that custom field is used in the integration. So if the type is changed, the attributes cached in OpenAir need to be updated.	Try manually resynchronizing the custom fields. Go to the OpenAir NetSuite Connector administration form and check the **Manually synchronize NetSuite custom fields used in the integration** box. When enabled, The Tips options include **Synchronize NetSuite custom fields**.
NetSuite error - Error occurred while trying to add record #0: An unexpected error occurred. Error ID: grrq2pstj9tt3tcr6qj1	That error is something unexpected that occurred in NetSuite directly. It is outside of OpenAir. Those errors are analogous to OpenAir server errors when you get that "Page not found, support has been contacted" response.	Create a support case and request that the NetSuite web services support team look at the support case. Someone who works directly with NS should look at it since OpenAir support cannot help. Sometimes these errors are genuine bugs, but often they are caused by bad data in NetSuite.

Error Message	Possible Causes	Steps to Correct
OpenAir NetSuite Connector logs show this error for all new and already integrated records of a certain type: `ERROR: The currency "" for NetSuite <record type> "<name>" has not been defined in NS OpenAir.`	OpenAir NetSuite Connector does not handle the currency for the NetSuite record correctly. There may be different reasons for this: - The Multiple Currencies feature may not enabled in NetSuite. - Both OpenAir and NetSuite accounts were using a single currency previously and are now using multiple currencies but the subsidiary information was not updated. - SOAP web services preferences prevent the SuiteScript which OpenAir NetSuite Connector uses to fetch the integration preferences set in the NetSuite account from running. OpenAir NetSuite Connector expects the NetSuite account to use a single base currency account when in fact it uses multiple base currencies.	1. Verify that the Multiple Currencies feature is enabled in your NetSuite account. See Enabling the Multiple Currencies Feature (External link to NetSuite SuiteAnswers — Answer ID: 7714). 2. Verify that the `NSOA Connector` role or custom role you created for the integration includes `Full` access permission for `Currency` (Lists subtab). See NetSuite Permissions Required for the Integration. 3. If there is only the root subsidiary in your NetSuite account, create another subsidiary. See Creating Subsidiary records (External link to NetSuite SuiteAnswers — Answer ID: 7714). 4. Ensure NetSuite subsidiary information is mapped to an OpenAir hierarchy. See Map NetSuite employee subsidiary to employee hierarchy, Map NetSuite customer subsidiary to customer hierarchy, and Map NetSuite project subsidiary to project hierarchy. 5. Run a full synchronization and include the import workflow for the NetSuite record type. 6. Go to Setup > Integration > SOAP Web Services Preferences. Check the **Run server SuiteScript and trigger workflows** box. Click **Save**. See Enabling the OpenAir Integration in NetSuite.

Troubleshooting the Creation of Duplicate Records in OpenAir Using the Real-Time Integration

The script execution order of user event scripts may change when you update a bundle in NetSuite. In some cases, the real-time integration may create duplicate records in OpenAir unless the OpenAir user event script for real-time integration is the last in the script running order. To avoid this, check the script running order for all relevant scripted records in NetSuite (Customization > Scripting > Scripted Records > [*Select a scripted record*] > User Event Scripts), and change it if necessary, after any bundle update. The following afterSubmit script should always be the last in the script running order:

- **OpenAir: Send Record to OpenAir**— if NetSuite OpenAir SRP Real-time Add-on bundle 369630 (previously 12610) version 1.x is installed.
- **OpenAir: Send Record to OpenAir 2.0** — if NetSuite OpenAir SRP Real-time Add-on bundle 369630 (previously 12610) version 2.0 or later version is installed.

The **OpenAir: Send Record to OpenAir** triggers the real-time integration process. If an afterSubmit script called after **OpenAir: Send Record to OpenAir** modifies the record in NetSuite, this causes the real-time integration process to be triggered again before the previous process finishes, resulting in the creation of duplicate records.

> **Note:** Effective with the October 9, 2021 OpenAir release, additional validation and a temporary record lock prevent the creation of duplicate records in OpenAir when importing a record from NetSuite to OpenAir using the real-time integration. OpenAir NetSuite Connector now checks for any processes already importing the record and locks the record until the import process completes so the same record cannot be imported by more than one process at the same time. This removes the script running order requirement and prevents the creation of duplicate records in OpenAir. However, it is best practice to verify that the OpenAir user event script for real-time integration is the last in the script running order.

For more information about changing the script running order of user event scripts, see The Scripted Records Page (external link to NetSuite SuiteAnswers | Answer ID: 10555).

For more information about the real-time integration, see Configuring Real-Time Record Import from NetSuite into OpenAir.

NetSuite Bundle Version History

This section gives the version history of NetSuite bundles required for the NetSuite <> OpenAir integration and optional integration features.

- NetSuite OpenAir SRP Integration – Bundle ID 369637 (Previously 2851)
- NetSuite OpenAir SRP Single Sign-On – Bundle ID 369619 (Previously 13512)
- NetSuite OpenAir SRP Real-time Add-on – Bundle ID 369630 (Previously 12610)
- NetSuite OpenAir SRP ARM – Bundle ID 370366 (Previously 217172)

NetSuite OpenAir SRP Integration – Bundle ID 369637 (Previously 2851)

Version	Summary of changes
4.0	■ Added the following settings to the `OpenAir: SRP Integration Settings` custom record type. These settings are required to set up the upgraded versions of the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) and Real-Time Record Import features. The upgraded features replace the SuiteSignOn integration with a NetSuite as OIDC Provider integration and certificate-based authentication. □ OpenAir URL □ OpenAir Database ID □ Display OpenAir in Project subtab □ Display OpenAir in Customer subtab □ Display OpenAir in Employee subtab □ Display OpenAir in Item subtab □ Display OpenAir in Contact subtab □ Display OpenAir in Rate card subtab □ Display OpenAir in Expense report subtab □ Display OpenAir in Invoice subtab □ Display OpenAir in Journal entry subtab ■ Added the following permission to the `NSOA Connector` role: □ OIDC Provider Setup (Full) □ OAuth 2.0 Authorized Applications Management (Full) □ Log in using OAuth 2.0 Access Tokens (Full) □ Certificate management (Full)
3.05	■ Added custom field to prevent duplicate exports of credit notes (negative invoices).
3.04	■ Changed **Execute as role** permission for `OpenAir: Add Timestamp v2.1` to the `NSOA Connector` role. ■ Fixed defect causing the **Export to OpenAir** box to be cleared when modifying a vendor bill record using inline editing in lists.
3.03	■ Modified the default **Preference** for SuiteScript `OpenAir: Add Timestamp v2.1` to `Update Deployments` when updating the bundle.
3.02	■ Fixed defect in `OpenAir: Rev. Rec. % Complete v2.1` script that caused an error on project exports when updating **% Complete**.

Version	Summary of changes
3.01	■ Changed visibility of `OpenAir: Project ID`. The custom field shows on custom forms by default.
3.0	■ Added SuiteScript 2.1 versions of all scripts. **Note:** For information about switching to SuiteScript 2 versions of the scripts, see Switching to SuiteScript 2 Versions of Integration Bundle Scripts. ■ Added custom record type `OpenAir: SRP Integration Settings` to hold NetSuite integration settings. ■ Added installation script that copies setting from script parameters to a `OpenAir: SRP Integration Settings` record. ■ Added option to **Duplicate bookings** to copy the bookings from an OpenAir project template when you import a new project from NetSuite into OpenAir. ■ Added monitored fields support for employee and project records ■ Added custom fields to prevent duplicate exports of customers and projects.
2.14	■ Added permission for `NSOA Connector` role (Custom records).
2.13	■ Updated `NSOA Connector` role to properly handle all other necessary staff (transactions, access tokens, SSO, currencies).
2.12	■ Updated `NSOA Connector` role to handle custom fields correctly.
2.11	■ Added field `custrecord_oa_suitetax_on` to Features and Preferences custom record to support SuiteTax.
2.10	■ Fixed monitored fields for conversion from Prospect / Lead to Customer.
2.09	■ Added custom fields to prevent duplicate exports of invoices and expenses. ■ Changed Bundle ID from 2851 to 369637 (due to NetSuite repository deprecation).
2.08	■ Added monitored fields support for customer records.
2.07	■ Added new integration record “Internal NS Application OPENAIR”.
2.06	■ The oa_revenue_recognition.js undeployed by default. (NetSuite hides the Revenue recognition feature in some accounts and this script then unintentionally hides the OpenAir Id) ■ The NSOA Connector role updated to enable TBA token creation under this role
2.05	■ Custom timestamp added for hierarchy objects (department, class, location, subsidiary)
2.04	■ New custom record for Monitored fields – holds fields to be updated to OpenAir. ■ New hidden custom field for project task, add timestamp for Project Task if some of monitored fields was changed
2.03	■ Added SuiteScript for adding current Timestamp before submit ■ New custom field and deployment of “add timestamp” script for Expense Category
2.02	■ Added support of searching record to oa_record_api.js RESTlet script
2.01	■ Removed Center tab 'OpenAir' as it's required only for SSO bundle
2.0	■ Added subtab 'OpenAir' ■ Moved all custom fields to OpenAir subtab for all custom fields managed by this bundle

Version	Summary of changes
1.39	■ Added deployments for Vendor Credit type on scripts: □ OpenAir: PO/VB/VC Save □ OpenAir: PO/VB/VC Validate ■ Added new Custom preferences parameter on script “OpenAir: PO/VB/VC Validate” □ Vendor Credit to OpenAir PO Integration – custscript_oa_vc_oa_po_export □ Added support for Vendor Credit Script oa_purchase_integration.js
1.38	■ Scripts □ Added RESTlet scripts ▬ oa_get_field_info.js – get all fields name and labels for specified field and sublist ▬ ping.js – test script to test RESTlet communication ▬ oa_record_api.js – record api (get,add,update,upsert,delete)
1.37	■ Added “NSOA Connector” role to be used for script execution. ■ customdeploy_oa_credit_tax
1.36	■ Scripts □ Added deployment of customscript_oa_invoice_tax for Credit Memo
1.35	■ Custom fields: □ custentity_oa_pco_value (OpenAir: Percent Complete Override) ■ hidden custentity_oa_pco_date (OpenAir: Percent Complete Override Date)Scripts □ setPrjPcoSublistBeforeSubmit.js ▬ added setPrjPcoSublistBeforeSubmit method for project percent complete integration with NetSuite “Advanced Revenue Management” feature
1.34	■ Custom fields: □ custentity_oa_ns_purchaser (OpenAir: Employee is NS purchaser) □ custevent_oa_export_to_openair (Export to OpenAir; exposed on support cases) ■ Scripts □ Updated oa_export_to_openair_flags.js ▬ Added beforeSubmitSetSupportCaseFields method for support case integration. ▬ Added company custom preference “Export Support Cases to OpenAir” ▬ Added setExportToOpenAirSupportCaseClient method to automatically set the “Export to OpenAir” on support cases. It is deployed as client script.
1.33	■ Custom fields: □ custcol_oa_wbs_phases_ancestry (OpenAir: Phases Ancestry)
1.32	■ Custom fields □ custevent_oa_export_err_msg (OpenAir: Export Error) □ custevent_oa_export_to_openair (Export to OpenAir) □ custevent_oa_milestone (OpenAir: Milestone) □ custevent_oa_project_task_name (OpenAir: Project Task Name)

Version	Summary of changes
	■ Scripts □ Updated oa_export_to_openair_flags.js ▬ Added beforeSubmitSetProjectTaskFields method for project task integration ▬ Added company custom preference “Enable OpenAir Task Integration”
1.31	■ Custom fields □ custentity_oa_project_currency (OpenAir: Project Currency) ■ Custom record new fields □ Features and Preferences custom record ▬ custrecord_oa_mc_customer (Multi-currency customer) ■ Scripts □ Updated oa_export_to_openair_flags.js ▬ When advanced projects enabled, copy project currency to custentity_oa_project_currency ▬ Renamed setLimitTimeAndExpensesFalseUE to setProjectFieldsUE ■ Updated features_and_preferences.js □ Detect the multi-currency customer feature
1.30	■ Custom fields □ custentity_oa_export_err_msg (OpenAir: Export Error) □ custitem_oa_export_err_msg (OpenAir: Export Error) □ custbody_oa_export_err_msg (OpenAir: Export Error) □ custrecord_oa_export_err_msg (OpenAir: Export Error)
1.29	■ Custom fields: □ custentity_oa_copy_expense_policy (Duplicate expense policy)
1.28.1	■ Updated name of custom record from customrecord_oa_project_rate to customrecord_oa_project_rate_card
1.28	■ Add support for adding a rate card to a project □ Custom fields ▬ custentity_oa_project_rate_card ■ Add support for creating rate cards using a custom record □ Custom records ▬ customrecord_oa_project_rate_card
1.27.1	■ Fix issue with journal entry creation from time entries where project was not set on the journal entry

Version	Summary of changes
1.27	■ Add support to export OpenAir project task assignments to NetSuite opportunity line items □ Custom fields: - custcol_oa_line_from_oa (OpenAir: Line from OpenAir) - custcol_oa_wbs_phase_name (OpenAir: Phase Name) - custcol_oa_wbs_task_name (OpenAir: Task Name) - custcol_oa_wbs_assignees (OpenAir: Assigned Users) - custcol_oa_wbs_planned_hours (OpenAir: Task Assignment Planned Hours) - custcol_oa_project_task_id (OpenAir: Project Task ID) - custcol_oa_task_assignment_id (OpenAir: Task Assignment ID) - custcol_oa_project_task_id_number (OpenAir: Project Task ID Number)
1.26	■ Add support for including OpenAir loaded costs on time entries

Version	Summary of changes
	■ Scripting to allow creating journal entries from time entries □ Custom records: ▬ customrecord_oa_cost_center (OpenAir: Cost Center) ▭ custrecord_oa_cost_center_debit (Debit Account) ▭ custrecord_oa_cost_center_credit (Credit Account) ▭ custrecord_oa_cost_center_subsidiary (Subsidiary) ▭ custrecord_oa_cost_center_department (Department) ▭ custrecord_oa_cost_center_class (Class) ▭ custrecord_oa_cost_center_location (Location) □ Custom fields: ▬ custbody_oa_je_from_time_id (OpenAir: Created From Time Entry) ▬ custcol_oa_lc_0 (OpenAir: Primary Loaded Cost) ▬ custcol_oa_lc_0_curr (OpenAir: Primary Loaded Cost Currency) ▬ custcol_oa_lc_1 (OpenAir: Secondary Loaded Cost) ▬ custcol_oa_lc_1_curr (OpenAir: Secondary Loaded Cost Currency) ▬ custcol_oa_lc_2 (OpenAir: Tertiary Loaded Cost) ▬ custcol_oa_lc_2_curr (OpenAir: Tertiary Loaded Cost Currency) ▬ custcol_oa_lc_department (OpenAir: Loaded Cost Department) ▬ custcol_oa_lc_class (OpenAir: Loaded Cost Class) ▬ custcol_oa_lc_location (OpenAir: Loaded Cost Location) ▬ custcol_oa_lc_subsidiary (OpenAir: Loaded Cost Subsidiary) ▬ custcol_oa_lc_debit_account (OpenAir: Loaded Cost Debit Account) ▬ custcol_oa_lc_credit_account (OpenAir: Loaded Cost Credit Account) ▬ custcol_oa_lc_timesheet_date (OpenAir: Timesheet Start Date) ▬ custcol_oa_je_from_time_error (OpenAir: Journal Entry Error) ▬ custcol_oa_je_from_time_error_msg (OpenAir: Journal Entry Error Message) ▬ custcol_oa_je_from_time_created (OpenAir: Journal Entry Creation Date) □ Subtab: ▬ OpenAir: Cost Data (appears on journal entry) □ Custom lists: ▬ customlist_oa_je_from_time_cost_level (OpenAir: Cost Level for JE) ▬ customlist_oa_je_from_time_classes (OpenAir: JE From Time Classes) ▬ customlist_oa_je_from_time_use_date (OpenAir: JE Date to Use) □ Saved search: ▬ customsearch_oa_cost_center_search (OpenAir: Cost Center Search) □ Scheduled script: ▬ customscript_oa_create_je_from_time (OpenAir: Create JE from Time) □ Script deployment: ▬ customdeploy_oa_create_je_from_time ▬ The following parameters are on the script deployment: ▭ custscript_oa_je_from_time_error_email (OpenAir: Journal Entry Error Email) ▭ custscript_oa_je_from_time_error_from (OpenAir: Journal Entry Error Email From) ▭ custscript_oa_je_from_time_level (OpenAir: Cost Level To Use for JE) ▭ custscript_oa_je_from_time_location (OpenAir: Location for JE)

Version	Summary of changes
	▭ custscript_oa_je_from_time_dept (OpenAir: Department for JE) ▭ custscript_oa_je_from_time_class (OpenAir: Class for JE) ▭ custscript_oa_je_from_time_ow (OpenAir: This is a OneWorld Account) ▭ custscript_oa_je_from_time_use_date (OpenAir: JE Date to Use)
1.25	■ Add support for creating revenue recognition rules through the sales order integration ■ The following were added: □ Custom fields: ▬ custcol_oa_rev_rec_rule (OpenAir: Rev Rec Rule) □ Custom records: ▬ customrecord_oa_rev_rec_rules (OpenAir: Revenue Rules) □ General preference: ▬ customsearch_oa_rev_rec_rule_type (OpenAir: Revenue Rules Search) □ Script deployment: ▬ customdeploy_oa_rev_rec_rule (deployed on sales orders)
1.24	■ Add support to export NetSuite vendor bills to OpenAir purchases orders □ oa_purchase_integration.js modifications □ “Vendor Bill to OpenAir PO Integration” company custom preference □ Add two more script deployments ▬ customdeploy_oa_vb_integration (client script) ▬ customdeploy_oa_vb_integration_save (user event)
1.23.1	■ Fix issue with credit charge type records not included in the custom record.
1.23.	■ Allow expense line Canadian overridden taxes to be displayed. □ Custom fields: ▬ custcol_oa_gst_override ▬ custcol_oa_pst_override
1.22.1	Minor tweak to ensure to check for a non-empty string, non-null value in the Canadian tax override custom fields prior to copying to tax fields.
1.22	■ Fix credit memo import issue by allowing charge type to be specified. □ Saved search: ▬ customsearch_ oa_credit_charge_type □ Custom record: ▬ customrecord_oa_credit_charge_type □ Custom field: ▬ custcol_oa_credit_charge_type ■ Add ability to create project workspace in OpenAir on initial export of NetSuite project. □ Custom field: ▬ custentity_oa_create_project_workspace

Version	Summary of changes
	■ Add ability to override calculated tax on Canadian expense reports and vendor bills. □ Script: - oa_expense_tax_override.js - OpenAir: Canadian Tax Override □ Custom fields: - custbody_oa_gst_override - custbody_oa_pst_override
1.21.1	Fix issue with default description item that has null value and “Enable billing rules for SO discounts”
1.21	■ Added ability to indicate the OpenAir project stage in NetSuite via custom field. This can set the project stage in OpenAir. □ Saved search: - customsearch_oa_project_stages □ Custom record: - customrecord_oa_project_stages □ Custom field: - custentity_oa_project_stage
1.20.2	■ Restored OpenAir: Opportunity custom field on project record
1.20.1	Removed OpenAir: Opportunity custom field on project record
1.20	■ Add hyperlink custom field for the feature to store the URL of an OpenAir invoice in NetSuite □ Custom fields: - custbody_oa_invoice_link, OpenAir: Invoice Link (invoice record)
1.19	■ Adding ability to update NetSuite opportunity based on the OpenAir project work breakdown structure □ Custom fields: - OpenAir: Opportunity (project record)

Version	Summary of changes
1.18	■ Added ability to integrate OpenAir expense reports with NetSuite vendor bills. Relaxed restriction on exporting vendors so that vendors may be exported to OpenAir users and vendors □ Custom fields: - OpenAir: Parent Vendor (vendor record) - OpenAir: User Currency (vendor record) - OpenAir: Map Exp. Rep. To Parent Vendor (vendor record) - OpenAir: Map Exp. Rep. To Vendor Bill (vendor record) - OpenAir: User or Vendor (vendor record) - OpenAir: User Tax Nexus Type (vendor record) - OpenAir Expense Report Creator (vendor bill record) □ SuiteScript: - oa_export_to_openair_flag.js (updated) - user event script = customscript_oa_set_vendor_fields - client script = customscript_oa_vendor_for_vb_int □ Custom preferences: - Expense Report - Vendor Bill Integration □ Custom record: - customrecord_oa_record_type_synch - customrecord_oa_vendor_nexus_type □ Saved search: - customsearch_oa_user_or_vendor - customsearch_oa_vendor_nexus_type
1.17.5	Fix issue with tax imported to NetSuite when there is both PST and GST tax.
1.17.4	■ Add adjustments for time zones and new custom preference “OpenAir Forex Time Zone Adjust” script parameter for the exchange rate integration.
1.17.3	Additional error handling for the exchange rate integration for OneWorld accounts. Fix issue with currencies being skipped due to incorrect internal ID comparison.
1.17.2	■ Add additional custom record for handling accounts with very large numbers of currencies that goes beyond the scheduled script usage limit. Script will be rescheduled immediately and generate other cross rates.
1.17.1	■ Fix date handling in exchange rate synchronization.
1.17	■ Added ability to copy exchange rates to a custom record so that all cross rates based on each subsidiary base currency can be synched to OpenAir. This is a OneWorld specific feature. □ SuiteScript: - openair_exchange_rate.js - script = custscript_oa_forex_rate; it is a scheduled script □ Custom preferences: - Exchange Rate Integration - OneWorld □ Custom record: - custrecord_oa_forex_daily □ Saved search: - customsearch_oa_forex_rate

Version	Summary of changes
1.16.1	■ fixes purchase order scripting issue by adding Validate OpenAir Purchase Order user-level custom preference. This will validate purchase order line data for export to OpenAir when enabled.
1.16	■ adds the following custom fields and scripts to handle the NetSuite/OpenAir purchase order integration. □ Custom fields: – custbody_oa_purchase_project_id (copies customer/project in header, shipTo, to a custom field) – custbody_oa_export_to_openair (header-level export to OpenAir - need this for transaction search to get header fields) – custcol_oa_po_rate (this is the custom column that holds the copied rate) – custcol_oa_po_line_tax (line-level tax - the tax is copied here) – custcol_oa_export_to_openair (line-level export field - specifies which specific lines to be exported to OpenAir) – custitem_oa_export_to_openair_product (this is an additional custom field used for exporting items to OpenAir as products) – custentity_oa_export_to_openair - we need to associate this with vendors too □ SuiteScript: – oa_purchase_integration.js □ Custom preferences: – Enable OpenAir Purchase Integration – account-level custom preference to enable the integration □ Custom record: – Features and preferences has a new Advanced Receiving field to check for the Advanced Receiving feature
1.15	■ adds new custom preference “custscript_oa_discount_billing_rules” to handle feature to create billing rules related to NetSuite discount lines on sales orders.
1.14	■ adds custcol_oa_time_entry_id for the time entry export and custcol_oa_quantity_not_hours for the OpenAir calculate quantity on % invoiced features. Also updates validation on the custcol_oa_quantity_not_hours field to be valid only on Time billing rules and hides that field on invoices and cash sales.
1.13	■ makes the billing rule column not mandatory

Version	Summary of changes
1.10	■ adds the following custom fields, subtabs and saved search for use with the project template feature: □ Custom fields: - custentity_oa_project_template (OpenAir Project Template) - custentity_oa_copy_revrec_rules (Duplicate revenue recognition rules) - custentity_oa_copy_prbill_rules (Duplicate project billing rules) - custentity_oa_copy_prbill_auto_settings (Duplicate project auto-bill settings) - custentity_oa_copy_project_pricing (Duplicate project pricing) - custentity_oa_copy_custom_fields (Duplicate phase and task custom fields) - custentity_oa_copy_loaded_cost (Duplicate loaded costs) - custentity_oa_copy_approvers (Duplicate project approvers) - custentity_oa_revrec_auto_settings (Duplicate recognition auto-run settings) - custentity_oa_copy_issues (Duplicate issues) - custentity_oa_copy_notification_settings (Duplicate notification settings) - custentity_oa_copy_dashboard_settings (Duplicate dashboard settings) - custentity_oa_copy_invoice_layout (Duplicate invoice layout settings) - custentity_oa_copy_project_budget (Duplicate project budget) - custentity_oa_copy_cost_type (Duplicate cost types) □ Subtabs: - OpenAir Project Template □ Saved Searches: - customsearch_oa_project_template (OpenAir Project Templates) □ Custom record: - customrecord_oa_project_templates (OpenAir Project Templates)
1.9	■ adds fields for credit memo and tax integrations □ OpenAir Invoice Line ID (CUSTCOL_OA_INVOICE_LINE_ID): Stores the internal ID of the OpenAir charge □ OpenAir Expense Report Line ID (CUSTCOL_OA_EXPENSE_REPORT_LINE_ID): Stores the internal ID of the OpenAir receipt
1.8.8	■ updates the handling of the Export to OpenAir field.
1.8.2	■ adds the “Combine Detail Items on Expense Reports” preference.
1.8.1	■ adds the fields for sales order linking support: □ Export to OpenAir (CUSTBODY_OA_EXPORT_TO_OPENAIR): Expose the Export to OpenAir field on the sales order. □ Export to OpenAir (CUSTCOL_OA_EXPORT_TO_OPENAIR): Expose the Export to OpenAir field on the sales order line. □ OpenAir Billing Rule (CUSTCOL_OA_BILLING_RULE_TYPE): Field to specify what type of billing rule should be generated from this sales order line. - OpenAir Billing Rules Custom List (CUSTOMLIST_OA_BILLING_RULES): Enumerates the supported OpenAir billing rules.

Version	Summary of changes
1.8	- adds two fields: - OpenAir Revenue Recognition ID (custbody_oa_rev_rec_id): This field is only available when the Revenue Recognition feature is enabled. - OpenAir SFA ID (custentity_oa_sfa_id on both customer and job): There is an additional preference for enabling the SFA ID in Setup > Company > General Preferences > Custom Preferences: SFA Integration Enabled.
1.7	- exposes the Revenue Recognition feature.
1.6	sets ‘Limit Time and Expenses’ to false for projects that have the ‘Export to OpenAir’ flag set.
1.5	adds the filtering fields and script parameters to set their default values.
1.4.4	fixes the location of the supervisor approval fields.
1.4.3	makes the script released.
1.4.2	makes the search public.
1.4.1	simplifies the saved search.
1.4	adds the saved search for relevant features and preferences.
1.3	adds the two multi-select fields for expense approvers.

NetSuite OpenAir SRP Single Sign-On – Bundle ID 369619 (Previously 13512)

Version	Summary of changes
7.0	⚠ **Important:** Update NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) to Version 4.0 or later before you update NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619) to this version. NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) 4.0 updates the `OpenAir: SRP Integration Settings` custom record type with settings required for using the NetSuite as OIDC Provider for the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. - Added support for using NetSuite as OIDC Provider instead of SuiteSignOn to enable access to OpenAir from the NetSuite UI. The OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature uses NetSuite as OIDC Provider instead of SuiteSignOn if an **OpenAir URL** and an **OpenAir Database ID** are specified on the `OpenAir: SRP Integration Settings` record. - Updated SuiteScript 2 versions of all Portlet and Suitelet scripts to support the NetSuite as OIDC Provider integration. - Added script `OpenAir: Load subtab`. The script uses the NetSuite as OIDC Provider integration to show the OpenAir record as a subtab on the NetSuite record page if the corresponding box is checked on the `OpenAir: SRP Integration Settings` record. - Updated the installation script to: - Undeploy SuiteScript 1.0 versions of all scripts in the bundle automatically, if previously deployed. - Deploy SuiteScript 2 versions of all scripts in the bundle automatically. - Update the center category links for the OpenAir (PSA) center tab to point to the SuiteScript 2.1 Suitelet script deployments automatically.

Version	Summary of changes
6.0	**Important:** Update the NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) to Version 3.0 or later before you update NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619) to this version. NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) 3.0 creates a custom record type and a custom record that are required to update the NetSuite OpenAir SRP Single Sign-On (Bundle ID 369619) to version 6.0 or later. ▪ Added SuiteScript 2.1 versions of all scripts except `OpenAir: Home module portlet`. ▪ Added SuiteScript 2.0 version of the `OpenAir: Home module portlet` script. **Note:** For information about switching to SuiteScript 2 versions of the scripts, see Switching to SuiteScript 2 Versions of Integration Bundle Scripts. ▪ Added installation script that copies setting from script parameters to a `OpenAir: SRP Integration Settings` record. ▪ Added menu items to PSA center tab for new Suitelets.
5.0	▪ Bundle ID changed from 13512 to 369619 (due to NetSuite repository deprecation).

NetSuite OpenAir SRP Real-time Add-on – Bundle ID 369630 (Previously 12610)

Version	Summary of changes
3.0	**Important:** Update NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) to Version 4.0 or later before you update NetSuite OpenAir SRP Real-time Add-on (Bundle ID 369630) to this version. NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) 4.0 updates the `OpenAir: SRP Integration Settings` custom record type with settings required for using certificate-based authentication for the Real-Time Record Import feature. ▪ Added support for using certificate-based authentication instead of a SuiteSignOn integration to enable real-time record import from NetSuite into OpenAir. The Real-Time Record Import feature uses certificate-based authentication instead of a SuiteSignOn if an **OpenAir URL** and an **OpenAir Database ID** are specified on the `OpenAir: SRP Integration Settings` record. ▪ Added the `custcertificate_oa_realtime` certificate record required by the real-time integration. The default certificate `openair-certificate.pem` associated with the `custcertificate_oa_realtime` certificate record is not valid – it is a required placeholder that will be replaced in a later step. A new certificate is generated in OpenAir and saved in NetSuite and OpenAir as part of the initial set up, and certificate rotation is done automatically thereafter.
2.03	**Important:** Update the NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) to Version 3.0 or later before you update to this version. NetSuite OpenAir SRP Integration bundle (Bundle ID 369637) 3.0 creates a custom record type and a custom record that are required to update the NetSuite OpenAir SRP Real-time Add-on (Bundle ID 369630) to version 2.03 or later. ▪ Added installation script that copies setting from script parameters to a `OpenAir: SRP Integration Settings` record.
2.01	▪ Unnecessary calls removed from initial check ▪ Bundle ID changed from 12610 to 369630 (due to NetSuite repository deprecation).
2.0	Added SuiteScript 2.0 version of real-time script

Version	Summary of changes
	Note: For information about switching to SuiteScript 2 versions of the scripts, see Switching to SuiteScript 2 Versions of Integration Bundle Scripts.
1.7	Add support for integrating Vendor credits in real time
1.6	Added “NSOA Connector” role for script execution
1.5.2	Fix issue with error on synchronizing project tasks.
1.5.1	Fix issue with error when saving vendor payments.
1.5	■ Add support for integrating the following in real time □ Project tasks □ Items □ Expense categories □ Vendors □ Vendor bills □ Purchase orders □ Sales orders □ Credit memos □ Invoice payments □ Customer deposit applications □ Expense reimbursements and vendor bill payments □ Cost center custom record
1.4	Add support for integrating support cases
1.3	■ Add support for rate cards (custom record)
1.2	■ Add support to export leads and prospects based on a status change □ When a lead or prospect changes to customer status, the integration will be triggered □ When a lead becomes a prospect and the prospect integration is enabled, the integration will be triggered
1.1	Add support to export leads and prospects
1.0	Initial version

NetSuite OpenAir SRP ARM – Bundle ID 370366 (Previously 217172)

Version	Summary of changes
0.96	■ SuiteScript 2.1 versions of scripts downgraded to SuiteScript 2.0 (The NetSuite ARM plug-in does not support SuiteScript 2.0).
0.95	■ Added permissions for NSOA Connector role for ARM Custom records.
0.94	■ Additional logging for ARM creating Revenue Recognition Event

Version	Summary of changes
0.93	■ Adds script for creating Revenue Recognition Event in SuiteScript 2.0 ■ Bundle ID changed from 217172 to 370366 (due to NetSuite repository deprecation).
0.92	Added support for setting exchange rate
0.91	Added support for Accounting book value
0.9	Added support for setting department, class, location
0.8	Initial stable version

What's New

This chapter documents new features and functionality as they are added to the integration. Since these additions often occur when there is a new OpenAir release, they are categorized under the release date.

April 13, 2024

Summary of changes:

- Added support for custom CRM field for NetSuite resource allocation records in field mapping definitions. See NetSuite custom fields for resource allocation records in field mapping for custom export and Exporting Records from OpenAir to NetSuite Using Custom Workflows (under Best Practice Guidelines).
- Added automated certificate rotation and health check for the upgraded Real-time Record Import feature which uses certificate-based authentication instead of SuiteSignOn to validate real-time record import requests. See Real-Time Record Import Certificate and Certificate Rotation
- Added OpenAir NetSuite Connector setting Use OAuth 2.0 for NetSuite Single Sign-On Integration and Real-Time Record Import. This setting replaces the optional features introduced in the OpenAir 2023.2 release and required for removing the dependency on the SuiteSignOn feature.
- Added support for importing NetSuite custom record types OpenAir: Cost Center as Cost center records in OpenAir and OpenAir: Project Rate Card as Rate card records in OpenAir in real time. See Importing or Mapping NetSuite Classifications to OpenAir Hierarchies, Real-Time Record Import and Configuring Real-Time Record Import from NetSuite into OpenAir.

October 7, 2023

Summary of changes:

- Added support for the NetSuite 2023.1 communication endpoint. See Use 2023.1 communication endpoint and NetSuite 2023.1 Communication Endpoint.
- Introduced upgraded versions of the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) and Real-Time Record Import features. The upgraded features use the NetSuite as OIDC Provider feature and certificate-based authentication instead of the SuiteSignOn feature, for which support is scheduled to end with the NetSuite 2024.2 upgrade.
    - Updated background information and steps to upgrade the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) feature. The upgraded feature replaces the SuiteSignOn integration with a NetSuite as OIDC Provider integration to enable access to OpenAir from within the NetSuite UI.

      Updated help topics:

        - OpenAir Within NetSuite
        - NetSuite Features Required for the Integration
        - NetSuite Bundles for the Integration
        - Creating a Custom Role for the Integration in NetSuite
        - Enabling Access to OpenAir From Within NetSuite
        - Controlling Access to the PSA Center Tab
        - Configuring Integration Preferences and Terminology Preferences in NetSuite
        - Connecting OpenAir to NetSuite for the Integration
        - Enable NetSuite single sign-on (OpenAir NetSuite Connector administration setting).

    - Enabling the OpenAir Within NetSuite Feature in OpenAir
  - Updated background information and steps to upgrade the Real-Time Record Import feature. The upgraded feature replaces the SuiteSignOn integration with certificate-based authentication to validate real-time record import requests.

    Updated help topics:

    - Real-Time Record Import
    - NetSuite Bundles for the Integration
    - Creating a Custom Role for the Integration in NetSuite
    - Configuring Real-Time Record Import from NetSuite into OpenAir
    - Configuring Integration Preferences and Terminology Preferences in NetSuite
    - OpenAir NetSuite Connector Contextual Tips
    - Enable NetSuite real-time integration (OpenAir NetSuite Connector administration setting).

    New help topic:

    - Added Enabling the Real-Time Record Import Feature in OpenAir — Steps to enable the real-time import of records from NetSuite into OpenAir.
  - Updated background information about the previous versions of the OpenAir Within NetSuite (NetSuite Single Sign-On Integration) and Real-Time Record Import features relying on the NetSuite Outbound Single Sign-On (SuiteSignOn) feature. This content was subsequently deleted.
  - Update the version history for the following NetSuite bundles. Release versions supporting the upgraded OpenAir Within NetSuite and Real-time Record Import features are available as of February 7, 2024.
    - NetSuite OpenAir SRP Integration – Bundle ID 369637 (Previously 2851)
    - NetSuite OpenAir SRP Single Sign-On – Bundle ID 369619 (Previously 13512)
    - NetSuite OpenAir SRP Real-time Add-on – Bundle ID 369630 (Previously 12610)

  > ⓘ **Note:** Steps for setting up the NetSuite Outbound Single Sign-On (SuiteSignOn) integration have been relocated. This content was subsequently removed.

- The Autorecovery feature is available for the Send OpenAir Credit (negative invoice) as NetSuite Credit Memo export workflow. See Verify credit (negative invoice) export and NetSuite OpenAir SRP Integration – Bundle ID 369637 (Previously 2851).
- Added support for exporting booking records from OpenAir to NetSuite as resource allocation records using a custom export workflow. See Custom Export Workflows and Exporting Records from OpenAir to NetSuite Using Custom Workflows.
- Added workflow setting for Customer POs (Sales Orders) import workflow Set OpenAir Customer PO total value to NetSuite sales order net amount.
- Updated Expose NetSuite full synchronization control since date, Running a Full Synchronization on Demand, and OpenAir NetSuite Connector Contextual Tips — Restrict the synchronization to records last modified on or after a specific date.

April 15, 2023

Summary of changes:

- Added Custom Import (Administration form setting) and Custom Import Workflows (detailed feature description) — Add custom workflows to import data from NetSuite service items or custom record types into a supported record type in OpenAir using your own field mapping definitions.

- Updated Custom Export Workflows — Time entry information can be exported from OpenAir to NetSuite using custom export workflows.
- Updated Update NetSuite Projects with OpenAir Project Data Export Workflow — Custom filters are supported for the Update Project mapping group.
- Updated Update NetSuite Tasks with OpenAir Task Data — Custom filters are supported for the Update Task mapping group.
- Added Time Entries to Time Entries Custom Export Best Practice.
- Added Customer POs (Sales Orders) import workflow setting Only update last modified date if no error.
- The New NetSuite Connector UI feature is required to use the NetSuite <> OpenAir integration, and OpenAir NetSuite Connector uses the NetSuite 2020.1 communication endpoint is used as standard. Help topics have been updated throughout the guide.

October 8, 2022

Summary of changes:

- Extended filter builder support to import workflows. See Filter Builder.
- Custom export workflows can now be used to create new records in NetSuite matching the OpenAir records and to update records already exported to NetSuite. See Custom Export Workflows and Exporting Records from OpenAir to NetSuite Using Custom Workflows.
- Context sensitive help extended to workflow settings. See Workflow Settings.
- The Send OpenAir Credits (Negative Invoices) as NetSuite Credit Memos export workflow now features a Project mapping group, which can be used to include project-specific information from the project record in OpenAir as header information on the NetSuite credit memo transaction record. See Project Mapping Group.
- The Monitored Fields feature supports the Employees (Employees) and Projects (Projects) import workflows. See Monitored Fields and Enabling Monitored Fields for Import Workflows.
- Ability to copy the bookings from an OpenAir project template when you import a new project from NetSuite into OpenAir. See Create new projects based on the project template specified in NetSuite.
- The Autorecovery feature is available for the Customers (Customers) and Projects (Projects) export workflows. The Autorecovery feature for the Expense Reports (Expense Reports) workflow has been extended to prevent the creation of duplicated vendor bills when exporting expense reports from OpenAir to NetSuite as vendor bills. See Verify customer export, Verify project export, and Verify expense report export.
- End of Support for the 2016.2 NetSuite Communication Endpoint and the Legacy NetSuite Connector UI. See NetSuite 2023.1 Communication Endpoint and New NetSuite Connector User Interface.
- All four NetSuite bundles for the NetSuite <> OpenAir integration will soon include SuiteScript 2 versions of all scripts. See NetSuite Bundles for the Integration, and Switching to SuiteScript 2 Versions of Integration Bundle Scripts.

April 9, 2022

Summary of changes:

- Added performance feature for export workflows. See Use direct filters.
- Extended filter builder support to all export workflows. See Filter Builder.
- The following export workflows support advanced mapping when reporting values are excluded.
  - Update NetSuite Projects with OpenAir Project Data (see also Exclude reporting values from Update NetSuite projects with OpenAir project data)

  - Update NetSuite Tasks with OpenAir Task Data (see also Exclude reporting values from Update NetSuite tasks with OpenAir task data)
- Added best practice tips for creating custom record type that will hold the information you export from OpenAir published reports. See Viewing Tips for Custom Export of OpenAir Reports to NetSuite.
- The OpenAir NetSuite Connector Guide link in the Tips menu now points to the online version of this guide in the OpenAir Help Center.

October 9, 2021

Summary of changes:

- Changes to Custom Export Workflows:
  - Added ability to export OpenAir published report data to a NetSuite custom record type (custom export workflow). See Exporting OpenAir Published Report Data To NetSuite Custom Record Types.
  - Added a filter builder UI to create and validate filters for custom export workflows exporting OpenAir published report to NetSuite. See Filter Builder.
  - Built-in Autorecovery feature preventing the creation of duplicated records in NetSuite if the integration run is interrupted.
  - Added support for advanced mappings — see Advanced Mappings for Custom Export Workflows.
  - Added support for custom exports to Contact records in NetSuite.
  - Increased the default maximum number of custom export workflows to 4.
  - Added ability to view the setting details for existing custom export workflows. See Viewing Custom Export Workflows Settings.
- Added ability to review the history of OpenAir NetSuite Connector configuration changes. See Reviewing NetSuite Connector Settings History.
- Added the following autorecovery setting to the NetSuite Connector administration form: Verify timesheet export.
- Changes to OpenAir NetSuite Connector Health Check:
  - Added checks for the following recommended features:
    - Direct TBA.
    - Use NetSuite 2020.1 communication endpoint.
  - Added ability to click each recommendation and view the relevant help content in the OpenAir Help Center.
- Added ability to click each setting label on the NetSuite Connector administration form and view the relevant help content in the OpenAir Help Center. See NetSuite Connector Administration Settings.
- SuiteTax compatibility verified for the following workflows:
  - Invoices (Invoices) export workflow
  - Send OpenAir Credit (Negative Invoice) as NetSuite Credit Memo export workflow
  - Expense Reports (Expense Reports) export workflow, when expense reports are exported from OpenAir to NetSuite as vendor bills.

  See Legacy Tax or SuiteTax.
- Preliminary online version of this guide available in the OpenAir Help Center. The content of this guide is undergoing a full review and restructure. See About this Guide.
- Added steps to enable the Customers (Customers) and Projects (Projects) export workflows instead of the 3-way Salesforce, OpenAir and NetSuite integration. See Replacing the Deprecated 3-Way

Salesforce, OpenAir and NetSuite Integration With the Customers (Customers) or Projects (Projects) Export Workflows.

- Added real-time import validation to prevent the creation of duplicate records creation in OpenAir due to NetSuite script execution order when using the real-time integration. See Configuring Real-Time Record Import from NetSuite into OpenAir.

April 10, 2021

Summary of changes:

- Added Custom Export Workflows — Add custom workflows to export OpenAir records to NetSuite, including custom record types in NetSuite. See also the new NetSuite Connector administration setting Custom Export to enable the Custom Export Workflows feature on your account.
- Changes to NetSuite Connector administration form:
  - The NetSuite Connector administration form is now self-service. Designated employees can edit the form without entering a password on the form — See NetSuite Connector Administration Settings.
  - The “NetSuite integration admin” form was renamed to “NetSuite Connector administration”.
  - Additional form validation has been added to avoid common errors when editing the NetSuite Connector administration settings.
  - The following settings are controlled by OpenAir Customer Support and no longer available on the form:
    - **Enable multithreading support. To be used with SuiteCloud+. Maximum used threads** — See Optional Features.
    - **Log warning when 00:00:00 not detected in date value** — Used for troubleshooting purposes.
    - **Disable logging of detailed runtime performance data** — Used for troubleshooting purposes.
  - If the New NetSuite Connector UI feature is enabled for your account, the settings controlling the availability of each integration workflow are now grouped under the new Imports and Exports sections. These settings were previously listed under the “Additional integrations” section of the form. The labels of the settings under the new Imports and Exports sections now match the workflow names as they appear on the Mapping screen and in the same order. The following table summarizes the changes.

Import / Export	Previous setting	New setting
Import	*N/A*	Employees (Employees) — *The workflow is always available*
Import	*N/A*	Customers (Customers) — *The workflow is always available*
Import	*N/A*	Projects (Projects — *The workflow is always available*)
Import	*N/A*	Services (Service Items) — *The workflow is always available*
Import	*N/A*	Expense Items (Expense Categories) — *The workflow is always available*
Import	*N/A*	Contacts (Contacts — *The workflow is always available*)

Import / Export	Previous setting	New setting
Import	Enable NetSuite vendor to NS OpenAir employee integration	Employees (Vendors)
Import	Enable NetSuite vendor to NS OpenAir vendor integration	Vendors (Vendors)
Import	Enable NetSuite product integration	Products (Purchase Items)
Import	Enable NetSuite sales order to NS OpenAir customer PO integration	Customer POs (Sales Orders)
Import	Enable NetSuite expense report/ vendor bill reimbursement integration	Reimbursements (Expense Report Reimbursements)
Import	Enable NetSuite payment integration for invoices	Payments (Invoice Payments)
Import	Enable NetSuite PO integration	POs (Purchase Orders)
Import	Enable NetSuite vendor bills to NS OpenAir POs integration	PO (Vendor Bills)
Import	Enable NetSuite customer deposit integration for invoices	Retainers (Applied Customer Deposits)
Import	Enable NetSuite support case import to OpenAir tasks	Tasks (Support Cases)
Import	Enable NetSuite vendor credits to NS OpenAir POs integration	POs (Vendor Credits)
Export	Enable Customer export to NetSuite	Customers (Customers)
Export	Enable Project export to NetSuite	Projects (Projects)
Export	Enable NetSuite Purchase requisition integration for purchase requests	Purchase request (Purchase requisition)
Export	*N/A*	Expense Reports (Expense Reports) — *The workflow is always available*
Export	*N/A*	Invoices (Invoices) — *The workflow is always available*
Export	Enable time entries export to NetSuite	Time Entries (Time)
Export	Enable NS OpenAir to NetSuite recognition transaction integration	Recognition Transactions (Revenue Recognition Transactions) Export Workflow Settings
Export	Enable NS OpenAir Loaded Cost to NetSuite custom record integration. Custom record Internal ID: xxx	Loaded Cost (Custom Record: <CustomRecordTypeID>)
Export	Enable workflows for Advanced Revenue Management feature	Recognition Transactions (Custom Record: OpenAir: Revenue Transactions for ARM) and Recognition Rules (Custom Record: OpenAir: Revenue Rules for ARM)
Export	Enable workflows for Advanced Revenue Management feature	Recognition Transactions (Custom Record: OpenAir: Revenue Transactions for ARM) and Recognition Rules (Custom Record: OpenAir: Revenue Rules for ARM)
Export	Export NS OpenAir project data to existing NetSuite projects.	Update NetSuite Projects with OpenAir Project Data

Import / Export	Previous setting	New setting
Export	Enable NS OpenAir project work breakdown structure to NetSuite opportunity integration	WBS (Opportunity)
Export	*N/A*	Custom Export— *New setting*

- Added an autorecovery feature to prevent duplicate record exports when the integration run is interrupted. See the following NetSuite Connector administration settings under the Auto recovery section:
  - Verify expense report export
  - Verify invoice export
  - Verify recognition transaction export
  - The feature requires NetSuite bundle 369637 (previously 2851) – NetSuite OpenAir SRP Integration version 2.09 or later.
- Added the following legacy compatibility settings to the NetSuite Connector administration form: Do not protect workflows and Hide workflow groups.
- Deprecation of the 3-way Salesforce, OpenAir and NetSuite integration for Customers and Projects. See Replacing the Deprecated 3-Way Salesforce, OpenAir and NetSuite Integration With the Customers (Customers) or Projects (Projects) Export Workflows and the NetSuite Connector administration setting DEPRECATED — Enable 3-way Salesforce.com, NS OpenAir and NetSuite integration now moved to the Legacy compatibility section of the form.
- Added NetSuite Connector administration setting Apply invoice line in NetSuite to expense using OpenAir ticket id.
- The NetSuite bundles used for the integration were replaced with follow-on bundles on January 20, 2021. The first time you update any of the bundles listed in the following table, NetSuite will install the follow-on bundle automatically. Each follow-on bundle has a new **Bundle ID**. The **Installed from** field, indicating the source of the bundle, shows “Production” for follow-on bundles instead of “Repository”. The **Owner** shows “OpenAir (6993781)” for follow-on bundles, instead of “OpenAir Bundle Account (660883)” for legacy bundles.

  Legacy integration bundles (with legacy bundle IDs) are deprecated. You can continue using the legacy bundles, if they are installed on your NetSuite account, until you update them to later versions.

Bundle name	Legacy bundle ID	New bundle ID
NetSuite OpenAir SRP Integration	2851	369637
NetSuite OpenAir SRP Real-time Add-on	12610	369630
NetSuite OpenAir SRP Single Sign-on	13512	369619
NetSuite OpenAir SRP ARM	217172	370366

October 10, 2020

Summary of changes in this release:

- Added NetSuite Connector administration settings **Log warning when 00:00:00 not detected in date value** (Removed in April 2021 OpenAir release) and Set time zone used by integration user setup on NetSuite to.
- Added Set & Forget Integrations feature:

  - Schedule Integration runs by selected workflows — see Scheduling the Integration to Run Automatically.
  - Manual Runs by selected workflows — see Running the Bulk Import and Export Integration on Demand.
  - Full synchronization runs for selected workflows — see Running a Full Synchronization on Demand.
  - Workflow protection when selected for scheduled runs or real-time integration — see Protected workflows.
- Added NetSuite Connector administration settings Enable Direct TBA — Double the speed of your integration with an optimized link between OpenAir and NetSuite.
- Added support for column filters and advanced filters on the Status list view. See Reviewing the Integration Run History and Status screen in the Navigating the OpenAir NetSuite Connector UI.
- Added OpenAir NetSuite Connector Health Check — A new Tips menu option lets you review whether there are any recommended actions to improve the NetSuite Connector configuration and to take full advantage of recommended features.
- Added support for the NetSuite 2020.1 communication endpoint.

  This release makes the following changes to OpenAir NetSuite Connector configuration settings:
  - End of support for the NetSuite 2013.1 communication endpoint.
  - NetSuite Connector uses the NetSuite 2016.2 endpoint by default.
  - Token-based authentication is now the only supported authentication method for the integration. The optional feature **Use Token-Based Authentication exclusively** is no longer available. See Creating and Assigning an Access Token for the Integration
  - NetSuite Connector Credentials form — Removed the **Email**, **Password**, **Confirm Password** and **Web Service Role** fields, changed the position of the **Application ID** field. See Connecting OpenAir to NetSuite for the Integration.
  - The NSOA Connector custom role added by NetSuite bundle 369637 (previously 2851) – OpenAir Integration Version 2.06 supports token-based authentication. See Creating and Assigning an Access Token for the Integration.
- Added Monitored fields for Customers workflow — See Enabling Monitored Fields for Import Workflows.

April 18, 2020

Summary of changes in this release:

- Updated the navigation path to the NetSuite Connector administration screen throughout.

  To access the NetSuite Connector administration screen, go to Administration > NetSuite Connector.
- Added the following integration settings enabling the **integration user error management** feature under NetSuite Connector Administration Settings:
  - Show last NetSuite integration error on expense reports
  - Show last NetSuite integration error on invoices
  - Show last NetSuite integration error on timesheets
- Added the ability to select an active administrator as the **designated integration user**. See Connecting OpenAir to NetSuite for the Integration.
- Added **multithreading support for the main integration**. See Multithreading.
- Added Exporting Revenue Recognition Transactions to NetSuite in Batches under Configuring Revenue Recognition. See also NetSuite recognition transaction batch size (maximum of 100) under NetSuite Connector Administration Settings > Performance features.

- Added a note about **optimizing the performance of integration workflows importing hierarchies from NetSuite**. See Importing or Mapping NetSuite Classifications to OpenAir Hierarchies.
- Added **quotation mark validation** in Validating Your Field Mapping Definitions.
- Added integration setting Use Service item instead of Product item during invoices export for purchase charges.

October 12, 2019

New features include:

- Added Validating Your Field Mapping Definitions features available in the Navigating the OpenAir NetSuite Connector UI:
  - Added **Validate mapping** link in the Tips menu. See Validating the Mapping Manually.
  - Added NetSuite Connector administration setting Enable strict mapping validation. See also Enforcing Front-End Mapping Validation.
- Added Reviewing the Integration Run History and Status screen in the Navigating the OpenAir NetSuite Connector UI.
- Added Customers (Customers)
- Added **Real-time support for Project export to NetSuite**. See Projects (Projects) and Configuring Real-Time Record Import from NetSuite into OpenAir.

April 13, 2019

New features include:

- OpenAir Customer Support can now set **Use TBA authentication exclusively** (Removed — As of October 10, 2020 OpenAir release token-based authentication is the only supported authentication method).
- Added **Run NetSuite full synchronization** from the Tips menu. See OpenAir NetSuite Connector Contextual Tips.
- Added **To map the OpenAir project task closed checkbox to the NetSuite EXPORT TO OPENAIR checkbox**, See Implementing the Export of OpenAir Task Data to NetSuite.
- Added **To map the NetSuite Project Task STATUS field to the OpenAir Closed Task checkbox**, See Implementing the Export of OpenAir Task Data to NetSuite.
- Added Enabling Monitored Fields for Import Workflows
- **Export NetSuite integration settings** and **Import NetSuite integration settings** now include inactive as well as active workflows. See OpenAir NetSuite Connector Contextual Tips.
- Added Performance features in the **NetSuite Connector administration form**:
  - Exclude reporting values from Update NetSuite projects with OpenAir project data
  - Exclude reporting values from Update NetSuite tasks with OpenAir task data.
- Added NetSuite Direct Revenue Posting under Working with Item Records in OpenAir in the Using the Integration section.

October 13, 2018

New features include:

- Added Projects (Projects)
- Added Enable import of inactive employees

April 14, 2018

New features include:

- Added OpenAir Revenue Recognition with NetSuite Advanced Revenue Management.
- Added Always attempt to send all viable time entries.
- Added Enable use of NetSuite time units of measure.
- Added **Time Project** rule to OpenAir Revenue Recognition with NetSuite Advanced Revenue Management.
- Added **Fixed Amount Percent Complete** rule to OpenAir Revenue Recognition with NetSuite Advanced Revenue Management.
- Advanced Revenue Management Revenue Recognition rules used in NetSuite Connector now support custom mappings. See OpenAir Revenue Recognition with NetSuite Advanced Revenue Management.
- OpenAir Customer Support can now enable the 2016.2 Communication Endpoint. As of October 10, 2020 OpenAir release, NetSuite Connector uses the NetSuite 2016.2 endpoint by default.

October 14, 2017

New features include:

- Added Token-Based Authentication. As of October 10, 2020 OpenAir release token-based authentication is the only supported authentication method.
- Added ability to integrate OpenAir projects and tasks on save in real-time. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- Added ability to change the number of threads used by NetSuite Connector for integrations with SuiteCloud Plus. See Multithreading Support for Main Integration Export Workflows.
- Added As Billed, Fixed Amount on Date or Milestone, Percent Complete, and Incurred vs. Forecast Revenue Recognition Rules for Advanced Revenue Management. See OpenAir Revenue Recognition with NetSuite Advanced Revenue Management.

April 15, 2017

New features include:

- Added workflow for integrating Vendor Credits from NetSuite to OpenAir. See POs (Vendor Credits) Import Workflow.
- Added support for the NetSuite 2016.2 endpoint. As of October 10, 2020 OpenAir release, NetSuite Connector uses the NetSuite 2016.2 endpoint by default.
- Added the ability to integrate OpenAir Purchase Requests on approval in real-time.

October 15, 2016

New features include:

- Added workflow for OpenAir Purchase Requests and Purchase Request Items to NetSuite Purchase Requisitions and Purchase Requisition Items. See Working with Purchase Requisitions in OpenAir, Terminology Mapping for the Integration, and OpenAir to NetSuite Data Mapping.
- Added Skip merge of matching customers with NetSuite Internal ID

April 16, 2016

New features include:

- User interface now supports advanced mapping in the visual mapping grid. See Navigating the OpenAir NetSuite Connector UI.
- Real-time screen allows you to set rules for integration runs on OpenAir approvals. See Configuring Real-Time Record Export from OpenAir to NetSuite.
- Chapters now include field grouping tabs which organize mapping fields. Tabs which aren't visible due to screen space can be viewed by clicking the ... icon. See Navigating the OpenAir NetSuite Connector UI.
- Added new role permissions. See Integration Role Permissions in OpenAir.
- Added import / export settings from the Tips menu. See OpenAir NetSuite Connector Contextual Tips.
- Added Real-time sync options, including on saving NetSuite records, approval of OpenAir records. See Configuring Real-Time Record Import from NetSuite into OpenAir and OpenAir NetSuite Connector Contextual Tips.
- Added the ability to sync on-demand from the Tips menu or using the API. See OpenAir NetSuite Connector Contextual Tips, and To Sync On-Demand from the Tips Menu.
- Added the ability to disable NetSuite Connector's check for the NetSuite Revenue Recognition feature. See Disable NetSuite revenue recognition feature check for recognition transaction integration.
- Added the ability to include taxes applied to exported OA credit invoices. See Import taxes applied to exported OA Credit (negative invoice).

October 17, 2015

New features include:

- New user interface available with a visual mapping grid. See Navigating the OpenAir NetSuite Connector UI

> ⚠ **Important:** Enabling this feature will not convert any advanced mappings to the new interface. This must be done manually. Implementing this feature should be done with PS assistance.

- Add ability to import NetSuite Cases as OpenAir project tasks. See Tasks (Support Cases) Import Workflow.
- Enhanced NetSuite purchase/vendor bill integration. See Enable purchaser integration.
- Add ability to run the integration more than six times a day.
- Support for multiple, simultaneous runs of the integration. See Multithreading

> ⓘ **Note:** It's necessary to have active NetSuite SuiteCloud Plus license which allows up to ten multiple webservices threads.

- Support added for Advanced Revenue Management. See Configuring Integration for Percent Complete Rules with NetSuite Advanced Revenue Management

April 18, 2015

No new features were added for this release.

October 18, 2015

No new features were added for this release.

May 17, 2014

New features include:

- Add ability to use NetSuite exchange rate in revenue transaction export. See Do not use OpenAir project exchange rate on invoice and recognition transaction export.
- Add support for multi-phase depth in OpenAir WBS export to NetSuite opp lines. See WBS (Opportunity).
- Add ability to pass OpenAir loaded cost information to a NetSuite custom record. See Loaded Cost (Custom Record: <CustomRecordTypeID>).

February 15, 2014

New features include:

- Add ability to import project tasks from NetSuite into OpenAir. See Working with Task Records in OpenAir.
- Add ability to export project tasks from OpenAir into NetSuite. See Implementing the Export of OpenAir Task Data to NetSuite.
- Mapping NetSuite transaction number to the OpenAir purchase request and purchase order number is now optional. See Other Options.

November 16, 2013

No new features were added for this release.

August 17, 2013

New features include:

- Add support in SRP integration to automatically know what data center URL to use for an account.
- Add support for 2013.1 endpoint. Removed — As of October 10, 2020 OpenAir release, NetSuite Connector supports the NetSuite 2016.2 and 2020.1 endpoints.
- Add support for setting the role for the integration user.
- Add support for NetSuite multi-currency customer feature.
- Add support for advanced mapping (including filtering) of WBS export. See Integration Points Exporting OpenAir Information to NetSuite.

May 18, 2013

New features include:

- Add “Email log for manual integration run” option to email the integration log when the integration is run manually. See Configuring Result and Error Notification Email.
- Add option to set integration errors on NetSuite records. See Enable feature to set import errors on NetSuite records.

January 19, 2013

New features include:

- A tip from the NetSuite integration form is available to get the data center specific URL.
- Support project expense policy feature in NetSuite integration. "Duplicate expense policy" is provided as an available checkbox when creating a project from a template in NetSuite. See Implementing OpenAir Project Templates.

November 17, 2012

New features include:

- Option to move the “Update the NetSuite opportunity line items” link for the Tips section to the header section and rename it as “Finalize WBS”. See Enable NetSuite opportunity export link in project navigation bar
- Add capability to set project-specific rate card through project integration from NetSuite to OpenAir. See Map the OpenAir rate card from the NetSuite project.

September 15, 2012

New features include:

- Add support for portfolio projects when used with the project template feature in NetSuite integration. See Create new projects based on the project template specified in NetSuite.
- Add support for running saved searches for items and saving as job codes.
- Export project task assignments to NetSuite opportunity. See WBS (Opportunity)

July 14, 2012

New features include:

- Add ability to export loaded cost with time entries and import cost centers. See Enable loaded cost export with the timesheet export.
- Add ability to use saved searches to import list data.

May 12, 2012

New features include:

- Added new functionality to Configuring Optional Features for Configuring the Sales Order Integration. Added the ability to generate OpenAir revenue recognition rules when you create a new PO that has a corresponding billing rule. See Configuring the Sales Order Integration and Customizing Revenue Recognition Rules.
- Added new functionality to Configuring Optional Features for Configuring the Real-Time Integration. You can update customer, employee, and project records in NetSuite. When you save, the changes transfer to OpenAir real-time. You can also include Leads and Prospects. See Configuring Real-Time Record Import from NetSuite into OpenAir.
- Added new functionality to Configuring Optional Features for Configuring the PSA Center Tab now available with version 4.0 of the OpenAir: Single Sign-on feature. See Enabling Access to OpenAir From Within NetSuite.
- Added more detail for configuring custom preferences for the NetSuite OpenAir SRP Integration, NetSuite OpenAir SRP Single Sign-On, NetSuite OpenAir SRP Real-Time Add-On, and OpenAir: WIP Accounting.
- Updated Using the Integration to include revenue recognition rules in Working with Sales Orders in OpenAir and custom mappings for tax location information in Working with Expense Reports in NetSuite.
- Updated Data Mapping in the Integration to include mapping of NetSuite sales orders to OpenAir Purchase Orders (POs). Revenue recognition rules are optionally transferred to OpenAir when new sales orders are created. See NetSuite to OpenAir Data Mapping.
- Updated Advanced Mapping and Filter Notation Reference integration points to add mapping syntax for revenue recognition rule types and expense report line tax information for NetSuite expense report exports. See Direction Tags: <PULL> and <PUSH>.
- Updated Advanced Mapping and Filter Notation Reference to include a Frequently Asked Questions section. See Frequently Asked Questions (FAQs).

March 17, 2012

New features include:

- Added new functionality to Configuring Optional Features for Configuring the Purchases/Vendor Bills Integration. You can transfer vendor bills from NetSuite to OpenAir purchase orders. See POs (Vendor Bills) Import Workflow.
- Updated Using the Integration to include Working with Purchases/Vendor Bills in OpenAir to include NetSuite Vendor Bills to OpenAir Purchase Orders. See Working with Purchases/Vendor Bills in OpenAir and specifically NetSuite Vendor Bills to OpenAir Purchase Orders.
- Added new functionality to Configuring Optional Features for the Email alerts. Receive email notifications every time you run the integration or only when an error occurs. See Configuring Result and Error Notification Email.
- Updated Advanced Mapping and Filter Notation Reference to include a new conditional transformation. Use a comparison of fields in conditions and set a value based on results of the comparison. See Advanced Field Mapping Definition.
- Updated Advanced Mapping and Filter Notation Reference integration points used to exchange data from NetSuite vendor bills to OpenAir purchase items and purchase orders. See Direction Tags: <PULL> and <PUSH>.

Enhancements to the document include an expansion of Error Messages to include all known errors to date. Also added more information such as possible causes, in addition to steps for correcting an error condition.