



ORACLE  
NETSUITE

# Billing and Invoices Guide

---

2022.2

March 22, 2023



Copyright © 2005, 2022, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

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, software documentation, data (as defined in the Federal Acquisition Regulation), 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," "commercial computer software documentation," or "limited rights data" 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®, Java, and MySQL 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 <https://www.oracle.com/corporate/accessibility>.

### **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=trs> if 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](http://www.netsuite.com/tos).

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.

### **Beta Features**

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

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.

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.

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, and timing of any features or functionality described in this document 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.

# Send Us Your Feedback

We'd like to hear your feedback on this document.

Answering the following questions will help us improve our help content:

- Did you find the information you needed? If not, what was missing?
- Did you find any errors?
- Is the information clear?
- Are the examples correct?
- Do you need more examples?
- What did you like most about this document?

Click [here](#) to send us your comments. If possible, please provide a page number or section title to identify the content you're describing.

To report software issues, contact NetSuite Customer Support.

# Table of Contents

Billing .....	1
Customer Billing Workflow Charts .....	2
Deciding Between Invoices and Cash Sales .....	3
Statement Charges .....	3
Adding Billing Information to a Transaction .....	4
Using Billing Classes .....	4
Creating Billing Classes .....	5
Using Consolidated Billing for Projects and for Payments .....	6
Creating Terms of Payment .....	7
Using Terms of Payment .....	9
Remittance Forms .....	9
Printing Remittance Forms .....	10
Invoices .....	11
Custom Workflow Based Invoice Approval .....	12
Invoice Approval Workflow .....	14
Using the Billing Tab and Dashboard .....	21
Global Invoicing Preferences .....	22
Creating an Invoice .....	26
Choosing an Invoice Type .....	28
Invoice Groups Overview .....	29
Invoice Groups Workflow Diagram .....	30
Invoice Groups Statuses and State Transitions .....	30
Invoice Groups and Installments .....	31
Invoice Groups Setup .....	31
Enabling the Invoice Groups Feature .....	31
Numbering Invoice Groups .....	32
Using Billing Operations to Generate and Group Invoices .....	32
Customizing the Invoice Group Form .....	33
Setting a Subsidiary for an Invoice Group .....	33
Using SuiteFlow to Create an Invoice Groups Workflow .....	34
Invoice Groups Management .....	34
Creating an Invoice Group .....	35
Viewing the Invoice Group Record .....	36
Viewing an Individual Invoice in an Invoice Group .....	37
Adding an Invoice to an Invoice Group From the Invoice Groups Record .....	37
Adding an Invoice to an Invoice Group From the Invoice Record .....	38
Removing an Invoice From an Invoice Group .....	38
Grouped Invoices Due Date .....	39
Invoice Groups and Closed Periods .....	39
Accepting Payment for an Invoice Group .....	39
Generating and Printing an Invoice Group PDF .....	40
Invoice Groups and Customer Statements .....	40
Invoice Group Reporting and the A/R Aging Report .....	41
Billing Operations .....	41
Billing or Invoicing a Sales Order .....	46
Invoicing Billable Customers .....	46
Closing or Voiding an Invoice .....	48
Creating Installments .....	48
Printing an Invoice .....	50
Giving Customers Access to Invoices .....	51
Show the Customer Account Balance Due on Invoices .....	53
Displaying Deposit Balance on Customer Statements and Remittance Slips .....	54
Progress Invoices .....	55

Using Progress Invoicing .....	55
Creating Invoices from Progress Sales Orders .....	56
Creating Invoices from Fulfilled Progress Sales Orders .....	56
Billing Costs to Customers .....	57
Billing Items to Customers .....	59
Billing Time to Customers .....	61
Billing Expenses to Customers .....	62
Customer Statements .....	64
Generating Statements .....	64
Printing a Statement .....	66
Printing Mini Statements .....	68
SuiteBilling Overview .....	70
SuiteBilling Terminology .....	70
SuiteBilling Process Diagrams .....	71
Subscription Plan Creation .....	71
New Stand-Alone Subscription Creation .....	72
Subscription Creation from a Sales Order .....	73
Renew and Extend a Subscription from a New Subscription Record .....	75
Renew a Subscription with a New Subscription Record .....	75
SuiteBilling Roles and Permissions .....	76
SuiteBilling Change Orders .....	78
Maximum Values in SuiteBilling .....	79
Decimal Precision in SuiteBilling .....	80
Revenue Recognition for SuiteBilling .....	80
Revenue Recognition Rules for SuiteBilling .....	81
Calculations for Revenue Plans Created on Subscription Events .....	82
Revenue Allocation for One-Time and Recurring Subscription Lines .....	84
SuiteBilling Setup .....	87
Setting the SuiteBilling Preferences .....	87
Enabling SuiteBilling Features .....	88
Optional Preferences for SuiteBilling .....	88
Enabling the Multi-Subsidiary Customer Feature for SuiteBilling .....	92
Creating Items for Subscription Plans .....	92
Class, Department, and Location Fields on Subscription Lines .....	93
Configuring Subscription Items for Revenue Recognition .....	94
Creating Subscription Plans .....	95
Creating Price Books .....	97
Creating Price Plans .....	101
Time-Based Pricing .....	103
Creating Billing Accounts .....	105
Renaming SuiteBilling Records .....	108
SuiteBilling Management .....	110
Creating a New Stand-Alone Subscription .....	110
Creating a Subscription from a Transaction .....	113
Updating a Subscription on a Transaction .....	114
Using Item Records .....	115
Activating Subscription Line Items .....	115
Adding Line Items to an Existing Subscription .....	116
Charges Based on a Fixed Amount .....	118
Modifying the Class, Department, or Location of a Subscription Line .....	119
Changing a Subscription Line's Status on a Subscription Record .....	119
Removing an Add-On Item .....	119
Activating a Subscription .....	120
Creating Usage Records .....	120
Voiding a Usage Record .....	121

Managing Rating Runs .....	122
Applying Discounts to Subscriptions .....	125
SuiteBilling Subscription Revisions .....	125
Viewing Subscription Revisions .....	125
Clearing Revisions .....	127
Voiding a Change Order .....	127
Viewing a Voided Change Order .....	128
Change Order Type-Specific Revenue Impacts .....	128
Terminating a Subscription Line Item .....	128
Suspending a Subscription Line Item .....	129
Reactivating a Subscription Line Item .....	129
Removing One-Day Charges .....	130
Modifying Pricing .....	132
Closed Subscriptions .....	133
Renewing a Subscription .....	133
Adding Renewal Information to Subscription Plans .....	136
Deleting a Subscription .....	136
SuiteBilling Reports .....	138
Monthly Recurring Revenue Report .....	138
Total Contract Value Report .....	138
Billings To Date Report .....	139
Churn By Customer Report .....	139
Churn By Revenue Report .....	140
Advanced Billing .....	142
Advanced Billing Overview .....	142
Enabling Advanced Billing .....	143
Billing Schedules .....	143
Creating Billing Schedules .....	144
Creating Billing Schedules From an Estimate or Sales Order .....	147
Applying Billing Schedules .....	148
Discount and Markup Items and Billing Schedules .....	150
Billing Customers Using Billing Schedules .....	151
Viewing Invoice Schedules .....	152
Advanced Billing and Advanced Shipping .....	154
Billing Orders Using Advanced Shipping with Advanced Billing .....	154
Electronic Invoicing .....	156
Electronic Invoicing Overview .....	158
E-Document Audit Trail and Statuses .....	164
Electronic Invoicing Permissions and Access Levels .....	164
Electronic Invoicing SuiteApp Availability and License Client .....	167
Electronic Invoicing Limitations and Best Practices .....	168
Electronic Invoicing Administrator Guide .....	170
Installing and Setting Up Electronic Invoicing .....	171
Defining the E-Document Country for Free Use .....	174
Advanced PDF/HTML Template .....	175
Granting Access Permission to the E-Documents Portlet .....	175
Understanding E-Documents and E-Document Packages .....	176
Creating E-Document Packages .....	176
Multi-subsidiary Support in the Outbound Process .....	177
Creating E-Document Templates .....	177
Editing E-Document Templates .....	251
E-Document Certification in the Outbound Process .....	251
E-Document Network Status Overview .....	252
Creating E-Document Sending Methods .....	253
E-Document Email Custom Templates .....	261

Selecting a Designated E-Document Sender .....	263
Setting Up Custom Roles to Send E-Documents .....	263
Customizing Roles to Restrict E-Document Generation or Sending .....	264
Deploying the Bulk Generation Script for E-Documents .....	265
Deploying the Script for Scheduled Sending of E-Documents .....	266
Scheduled Script for Getting Network Status of E-Documents .....	268
Updating E-Document Certification Statuses .....	270
Processing E-Documents Automatically for Individual Transactions .....	272
Electronic Invoicing Inbound Email Capture .....	272
Using SOAP Web Services for Inbound Processing .....	274
Setting Up Custom Roles that can Convert Inbound E-Documents .....	277
Inbound Validation Plug-ins .....	278
Deploying Automatic Bulk Conversion Script for Inbound E-Documents .....	281
Electronic Invoicing User Guide .....	282
Displaying the E-Documents Portlet on the Home Page .....	283
Overview of Outbound E-Document Process .....	284
Outbound E-Document Statuses .....	285
Assigning E-Document Packages to Customer or Vendor Records .....	286
Defining E-Document Email Recipients .....	286
Selecting E-Document Packages, Templates and Sending Methods on Transactions .....	287
Enabling PDF File Reference Generation .....	287
Generating and Regenerating E-Documents .....	288
Sending and Resending E-Documents .....	291
E-Document Network Status .....	294
Overview of Inbound E-Document Processing .....	294
Inbound E-Document Statuses .....	295
Receiving Inbound E-Documents by Email Capture .....	295
Receiving E-Document XML Files from Web Service .....	296
Uploading Received XML Files as Inbound E-Documents .....	297
Converting Inbound E-Documents into Transaction Records .....	298
Common Scenarios in Vendor Bill Conversion .....	301
Converting Individual Inbound E-Documents into Vendor Bills .....	303
Converting Failed Inbound E-Documents .....	304
Compatibility of Approval Workflows with Vendor Bill Conversion .....	305
Canceling Inbound E-Documents .....	305
Electronic Invoicing Errors .....	305

# Billing

## Billing Overview

To receive payment from customers for the correct amount they owe, bill customers for any goods and services that you sell. Customer bills track exactly what goods and services were sold to the customer and the exact amount the customer owes you for each one.

After you have entered customer bills in NetSuite, send the bill to your customer so they know how much they owe you. You can print bills and then mail or fax them, or you can send bills to your customers by email.

If you enter a bill for all items you sell, you can accurately track the amount you charged for each item, the amount each customer owes you, and the total amount of funds due as income.

## Billing Forms

You can bill customers in several ways: using invoices, cash sales and statement charges. These are transactions that record any items or services provided to the customer and how much you charge for each item or service. For more information on billing customers using these forms, read the following:

- [Deciding Between Invoices and Cash Sales](#)
- [Invoices](#)
- [Choosing an Invoice Type](#)
- [Cash Sales](#)
- [Statement Charges](#)

Customer bills can also be created by converting a sales order into a bill. This enables you to create the bill without re-entering data. For more information, read the help topic [Creating Invoices or Cash Sales from Sales Orders](#).

## Terms of Payment

Terms are used to specify when payment is due on your customers' bills. You can create different payment terms for different customers. For example, one customer has terms of Net 30 and must pay the balance due within 30 days.

Using terms helps you track expected income over time. For more information on terms, read [Creating Terms of Payment](#) and [Using Terms of Payment](#).

## Advanced Billing

Advanced billing enables you to create billing schedules to invoice sales over a range of time or a contract term. For example, a service company can use advanced billing to manage recurring billing on sales such as a one-year membership billed monthly. Billing schedules manage the billing process to track when to invoice customers and how much to bill. For more information, read the following:

- [Advanced Billing](#)
- [Advanced Billing and Advanced Shipping](#)

## Invoice in Advance of Fulfillment

If you want to create invoices for a whole order even if the order has not been completely fulfilled, enable the **Invoice in Advance of Fulfillment** preference. Then, even if you ship only part of an order, you can

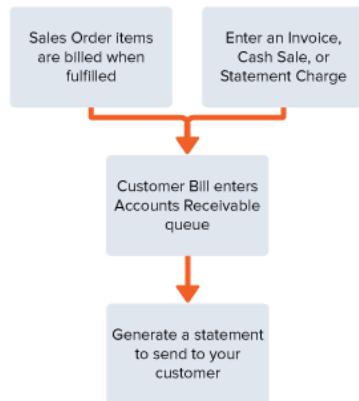
create an invoice or cash sale for the entire order. For information on enabling this preference, read the help topic [Accounting Preferences](#).

## Customer Billing Workflow Charts

This section includes high-level overview diagrams of the billing process, with and without the Advanced Shipping feature enabled.

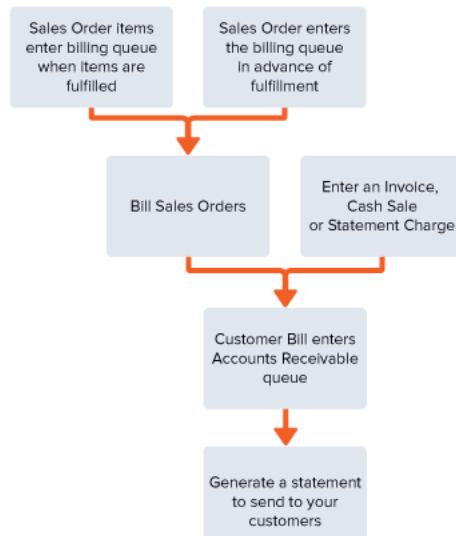
### Billing Without Advanced Shipping

The following diagram shows the workflow for standard billing without Advanced Shipping. For information on how to bill customers without advanced shipping, see [Adding Billing Information to a Transaction](#).



### Billing with Advanced Shipping

The following diagram shows the workflow for billing with Advanced Shipping enabled. To enable Advanced Shipping, see the help topic [Advanced Shipping](#). For information on billing customers using the Advanced Billing and Advanced Shipping features, see the help topic [Advanced Billing and Advanced Shipping](#).



# Deciding Between Invoices and Cash Sales

Generally, you should enter a cash sale only if you receive assured payment at the time of the sale. This could be in the form of credit card, check or cash.

If, on the other hand, you provide terms for your customers and let them pay within a certain period of the delivery of the product or service, you should issue an invoice so you can keep track of who owes you and how much.

## Statement Charges

Statement charges are similar to invoices because they track payments you expect from customers. Use statement charges instead of invoices when you want to enter several charges that will accumulate before you send a billing statement to the customer.

For example, a landscaping company offers weekly lawn service to its customers. Each weekly service incurs a \$50 charge. A \$50 statement charge is entered each week that the lawn service is performed. Then, one time each month a statement is sent to the customer showing the total due for services performed.

### To enter a statement charge:

1. Go to Transactions > Sales > Create Statement Charges (Administrator).
2. In the **A/R Account** field, choose the account you want to post this statement charge to.
3. In the **Customer or Project** field, select the appropriate customer or project.

The currency selected on the customer's record appears here as well as the exchange rate for this customer's currency.



**Note:** NetSuite enables you to create a statement charge for intercompany customers and inventory items.

4. Enter a new rate to update this currency's exchange rate.
5. In the **Date** field, accept, enter or pick the correct date.
6. In the **Posting Period** field, select the month and year you want this charge to post.
7. In the **Ref. No.** field, accept or enter a reference number.
8. In the **Due Date** field, enter or pick a date this charge is due.
9. In the **Item** field, select the item to which this charge applies.
10. In the **Quantity** field, enter the appropriate quantity.
11. In the **Description** field, accept or enter a description for this customer charge.  
In the Rate field, the rate for the item automatically appears.  
In the Total field, the amount in the Quantity field is multiplied by the amount in the Rate field.
12. If you track departments, select the appropriate department for this charge in the **Department** field.
13. If you use NetSuite OneWorld and the selected customer is shared with multiple subsidiaries, you can choose any subsidiary assigned to the selected customer. If you have not shared the customer record with multiple subsidiaries, the read-only primary subsidiary displays.

For information about sharing customer records with multiple subsidiaries, see the help topic [Assigning Subsidiaries to a Customer](#).

14. If you track classes, select the appropriate class for this charge in the **Class** field.
15. Click **Save**.

After you create statement charges, you can send them to your customers by generating statements. How you generate statements depends on your role.

- To generate statements as an Administrator, go to Transactions > Customers > Generate Statements.
- To generate statements using the A/R Clerk role, go to Billing > Sales > Generate Statements.

## Adding Billing Information to a Transaction

Select the Billing subtab to add billing related information to a transaction.

### To add billing information to a transaction:

1. If you are creating a progress billing sales order, select billing terms for this order in the **Terms** field.



**Important:** On a sales order, if you enter both a payment method and terms, the payment method takes precedence, and a cash sale is generated upon billing.

2. If you use the Advanced Billing feature, in the **Billing Schedule** field select the schedule you want to apply to this order.
3. The customer's default billing address shows in the **Bill To** text field.
  - Select **New** in the **Bill To Select** field, or click the **+** icon, to add an address to be used for this sales order and saved with the customer record.
  - Select **Custom** in the **Bill To Select** field to add an address to be used for this sales order only and not saved with the customer record.
  - Click the **Edit** icon to make changes to the address selected in the **Bill To Select** field. These changes are saved with the customer record.

## Using Billing Classes

You can use billing classes to set billable time rates for service items based on employees.

When different employees charge different rates for the same service, create billing classes and assign a billing class to employees. Then, determine pricing for each billing class on service item records. Then, the correct pricing defaults when an employee enters billable time to invoice the customer.

You can create a billing classes for individual employees or groups of employees and charge a different rate for each.

For example, a law office can use billing classes to charge specific hourly rates for time billed by partners in the firm:

For example, you run a law firm and create two billing classes called Junior Partner and Senior Partner. You associate the Junior Partner and Senior Partner billing classes with the service item Initial Consultation and set the appropriate billing rates for each. Also, you select the Junior Partner billing class on John Smith's employee record. Then, when John Smith enters billable time for an initial consultation, the time is billed at the junior partner rate by default.

Using Billing Classes requires the four following steps:

1. Enable the Bill Costs to Customers feature
2. Creating Billing Classes
3. Specify Billing Classes on Employee Records
4. Track Billing Class Pricing on Service Items

## Enable the Bill Costs to Customers feature

1. Go to Setup > Company > Enable Features (Administrator).
2. Click the **Transactions** subtab and check the **Bill Costs to Customers** box under Billing.
3. Click the **Employees** subtab and under Time & Expenses, check the **Time Tracking** box .
4. Check the **Per-Employee Billing Rates** box.
5. Click **Save**.

Now that these features are enabled, you can create billing classes. Create a billing class for each category you want to create prices for.

## Creating Billing Classes

1. Go to Setup > Accounting > Billing Classes > New.
2. On the Billing Class page, enter the name of the billing class.  
For example, enter Junior Partner or Senior Partner.
3. If you use Multiple Units of Measure and Charge-Based Billing, you can select a Price, Units Type, and Sale Unit to use custom interval billing rates. For more information, see the help topic [Custom Interval Billing Rates](#).
4. Click Save.

After you have created billing classes, they appear in the billing class lists on service item records and employee records.

**Note:** If you use Project Management and Charge-Based Billing for projects, you can also enable billing rate cards to define different billing rates for groups of billing classes. These rate cards can then be used to set billing rates on charge-based projects using time-based charge rules. For more information, see the help topic [Using Billing Rate Cards](#). If you use rate cards, billing class records will no longer contain a field for defining rates. You must define rates for billing classes used for projects on billing rate cards. A default billing rate card is automatically created after the feature is enabled with your existing billing classes and default rates.

Next, you can select the default billing class to determine the rate billed for each employee.

## Specify Billing Classes on Employee Records

1. Go to Lists > Employees > Employees.
2. Click **Edit** next to an employee.
3. Under Classification, in the **Billing Class** field, choose a billing class.
4. Click **Save**.

Now, when billable time is entered in Time Tracking for this employee, the default rate charged is the rate set for the billing class on the service item record.

On service item records, enter rates for each billing class to determine pricing.

## Track Billing Class Pricing on Service Items

1. Go to Lists > Accounting > Items.
2. Click **Edit** next to a service item. The item type must be Service for Sale or Service for Resale.
3. On the item record, click the **Sales/Pricing** subtab.
4. Under Sales, select a billing class.  
For example, select Senior Partner to set up senior partner rates.
5. Under Pricing, verify that you have entered default prices on your base currency subtab.
6. Enter the appropriate pricing for that billing class.  
Billing classes cannot have negative rates.
7. Click **Add**.
8. Set additional billing classes and rates as necessary.
9. Click **Save**.

When the service item is entered in Time Tracking for an employee with a billing schedule, the appropriate pricing defaults.

For more information on invoicing customers for billable time, read [Billing Time to Customers](#).



**Note:** Because the Billing Classes feature is not compatible with the Quantity Pricing feature, you cannot use both features at the same time. If both are enabled, then billing classes replace quantity pricing on service item records.

## Using Consolidated Billing for Projects and for Payments

Consolidated billing consists of a set of accounting features and preferences that affect the way you invoice your customers and accept payments toward those invoices.

These features and preferences are described below:

- **Consolidate Projects on Sales Transactions** – When you enable this preference, you can create sales orders and bill customers for charges associated with multiple projects on a single invoice. A Project column appears on sales transactions in the list of items so that each item or expense can be attributed to a particular project.
  - You cannot issue sales transactions directly to projects. You can only issue sales transactions to customers and subcustomers.
  - If you use Project Management, one project is created for each project-tagged line-item on sales transactions.

When you clear this box, you can associate all items on a sales transaction with only one customer or project. You can issue sales transactions to both customers and projects. If you use Project Management, one project is created that contains all project-tagged line-items on the order.

- **Consolidated Payments** – Enable this feature to accept payments for projects or subcustomers either through top-level customer or the individual subcustomer. Multiple invoices for different projects and subcustomers can be paid at the same time. For more information, see the help topic [Consolidated Payments](#).

You can use these preferences together or independently. What you choose to enable determines how your sales transactions and payments are handled.

#### **Consolidate Projects on Sales Transactions only:**

- Projects are not available in the Project list when creating sales transactions.
- A project column appears in the item list on sales transactions.

#### **Consolidated Payments feature only:**

- Sales transactions can be created for projects.
- Any balances for projects and subcustomers may be paid through the parent customer as well as through the subcustomer.

#### **Both Consolidate Projects on Sales Transactions and Consolidated Payments:**

- Sales transactions can only be created for customers and subcustomers.
- A project column appears in the item list on sales transactions.
- Payments may be made through either the parent customer or the subcustomer.



**Important:** Projects are not tracked as subcustomers when the Project Management feature is enabled. For more information, read the help topic [Using Project Management](#).

You can enable the Consolidate Projects on Sales Transactions preference at Setup > Accounting > Preferences > Accounting Preferences . You can enable the Consolidated Payments feature at Setup > Company > Setup Tasks > Enable Features. You can enable the Consolidated Payments feature at Setup > Company > Setup Tasks > Enable Features on the Accounting subtab.

To accept a customer payment, go to Transactions > Customers > Accept Customer Payments.

## Creating Terms of Payment

Terms are used to specify when payment is due on customer invoices and vendor bills. Define the specific requirements of a term of payment by creating a term record. You can create different payment terms for different customers or vendors.

#### **To create a standard term record:**

1. Go to Billing > Setup > Payment Terms > New.
2. On the Add to Accounting List page, select **Term**.
3. On the Term page, select **Standard**.



**Note:** Standard is selected by default, as opposed to Date Driven.

4. Enter a name for the term in the **Terms** field.

For example, you can name a term **Net 30**.

5. In the **Days Till Net Due** field, enter the number of days until the net amount of the invoice or bill becomes due.
  6. In the **% Discount** field, enter the percentage discount if the invoice or bill is paid early.
  7. In the **Days Till Discount Expires** field, enter the number of days the early payment discount is available.
- For example, if the early payment discount is available for 15 days after the bill is issued, enter **15** in this field.
8. Click **Save**.

### To create a date driven term record

1. Go to Billing > Setup > Payment Terms > New.
2. On the Term page, select **Date Driven**. Date driven terms are used when payment is due on a particular calendar date; for example, the 15th of the month.
3. Enter a name for the term in the **Terms** field.
4. In the **Day of Month Net Due** field, enter the day of the month when the net amount of the invoice is due.  
If you enter a date that does not exist in the month the transaction is due, the last day of the month becomes the due date.
5. In the **Due Next Month If Within Days** field, enter the number of days before the due date during which issued invoices will be carried forward to the next month.  
Use this field to ensure that customers are always given a minimum number of days to pay after they are invoiced.  
For example, if invoices are due on the 30th of each month and you enter **5** in this field, then invoices issued between the 25th and 30th are due for payment by the 30th of the following month.
6. In the **% Discount** field, enter the percentage discount if the invoice is paid early.
7. If you offer a discount for early payment, enter the last day of the month the early payment discount is available in the **Day Discount Expires** field.  
For example, if the early payment discount is available through the 20th of each month, enter **20** in this field.
8. Check the **Preferred** box if you want to use this term for customers by default.
9. Click **Save**.

Now this term can be used on customer records, vendor bills and other transactions.

You can create several term records to apply the appropriate terms to individual customers, vendor bills and other transactions. Standard terms include the following:

- 1% 10 Net 30
- 2% 10 Net 30
- Due on receipt
- Net 15
- Net 30
- Net 60

You can apply the terms to invoices by choosing a term on individual invoices or by setting default terms on customer and vendor records. For more information, read [Using Terms of Payment](#).

# Using Terms of Payment

After you have created a term record, you can select that term on invoices and customer records to set payment terms. When you select a term on a customer's record, that term defaults on bills to the customer. You can also set terms on each invoice individually.

If you always require the same payment terms for a customer, you can set the default terms on the customer record. Then, when you select the customer on an invoice, their terms automatically appear.

## To set default terms on an existing customer record:

1. Go to Lists > Relationships > Customers.
2. Click **Edit** next to the customer's name.
3. On the customer record, click the **Financial** subtab.
4. Under Account Information, in the **Terms** field, select the term you want to default for this customer.
5. Click **Save**.

Now, the term you chose defaults on the customer's invoices. After you have set default terms for a customer, you can still change the terms on any individual invoice.

## To set payment terms on an invoice:

1. Go to Customers > Sales > Create Invoices.
2. On the Invoice in the **Customer** field, choose the customer you are billing and fill in the appropriate information.
3. Click the **Billing** subtab.
4. Under Billing Information, in the **Terms** field, choose the terms of payment for this invoice.
5. Click **Save**.

Now, the terms of payment for this invoice are set according to the term you defined.

# Remittance Forms

A remittance slip prints on invoices and statements to help customers make payments to you. It also helps you track incoming payments.

The remittance slip details payment information the customer needs to make a payment. Remittance slips include:

- your company name
- your customer's name
- a blank box for the customer to enter amount paid
- the invoice number (on invoices) or amount due (on statements)
- a box for the customer to enter credit card information
- the name to make checks payable to
- the address to mail the payment to
- an optional message or disclaimer

If the Deposit Balance and Net Amount Due fields are enabled for the customer, these fields also display on the remittance slip. For more information, see [Displaying Deposit Balance on Customer Statements and Remittance Slips](#).

You can activate the remittance slip only on forms that use a standard template. Remittance slips do not appear on customized forms that use advanced PDF/HTML templates.

Customers complete the information and detach the remittance slip to remit with their payment.

When you receive the payment, the enclosed remittance slip details the payment method and how the customer intends for the payment to be applied, such as to specific invoices or to the statement balance.

### To print a remittance slip with your invoices and statements:

1. Go to > Setup > Company > Preferences > Printing and Fax (Administrator).
2. Check the **Print Remittance Form with Invoices & Statements** box in the **Transactions** column.
3. Click **Save**.

## Printing Remittance Forms

You can print a remittance slip on invoices and statements to help customers make payments to you. A remittance slip also helps you track incoming payments.

The remittance slip details information the customer needs to make a payment. Customers complete the information and detach the remittance slip to remit with their payment.

When you receive the payment, the enclosed remittance slip details the payment method and how the customer intends for the payment to be applied, such as to specific invoices or their statement balance.

### To activate Print Remittance Slip with Invoices & Statements:

1. Go to Setup > Company > Preferences > Printing and Fax (Administrator).
2. Click the **Printing** subtab.
3. Check the **Print Remittance Slip with Invoices & Statements** box.
4. Click **Save**.

Now, your printed invoices and statements show a remittance slip.

To customize remittance slips, go to Customization > Forms > Transaction Forms . On the Custom Transaction Forms page, click Customize next to Standard Remittance Slip.

# Invoices

- [Custom Workflow Based Invoice Approval](#)
- [Custom Workflow Based Invoice Approval](#)
- [Using the Billing Tab and Dashboard](#)
- [Global Invoicing Preferences](#)
- [Creating an Invoice](#)
- [Choosing an Invoice Type](#)
- [Invoice Groups Overview](#)
- [Billing Operations](#)
- [Billing or Invoicing a Sales Order](#)
- [Invoicing Billable Customers](#)
- [Closing or Voiding an Invoice](#)
- [Creating Installments](#)
- [Printing an Invoice](#)
- [Progress Invoices](#)
- [Billing Costs to Customers](#)
- [Displaying Deposit Balance on Customer Statements and Remittance Slips](#)

## Invoices Overview

An invoice is a record of a sale to a customer. Invoicing is the process of creating bills for goods and services that customers receive. Record an invoice when payment is not received at the time of delivery. The invoice amount is the total due to be paid to you by the customer.

The invoice indicates terms of payment that specify when you expect to receive the money owed to you. Invoice aging tracks how long the amount has been due as payable. NetSuite keeps track of your outstanding invoices in an account called Accounts Receivable.

All sales in NetSuite involve items from your Items list. Each invoice consists of multiple line items whose sales amounts add up to the total of the invoice. In companies with the most basic billing procedures, items are chosen at the time of sale by clicking Items in the middle of the Invoice page.

 **Note:** Partially fulfilled Item Groups and partially fulfilled Kit/Packages display a quantity of zero (0) on printed invoices.

Companies that bill costs back to customers have additional sections where they can identify billable costs, mark them up and add them to invoices.

 **Important:** The total number of lines per transaction cannot exceed 200. For more information about line limitations in NetSuite, see the help topic [Line Limitations for Transactions](#).

To enter an invoice, go to Billing > Sales > Create Invoices.

## Invoice in Advance of Fulfillment

If you want to create invoices for a whole order even if the order hasn't been completely fulfilled, enable the **Invoice in Advance of Fulfillment** preference. Then, even if you ship only part of an order, you can create an invoice or cash sale for the entire order. For information about enabling this preference, read the help topic [Accounting Preferences](#).



**Note:** If an item is manually added to an invoice, the on-hand quantity displayed for the item on the invoice and on any linked fulfillments shows the corresponding reduction.

## Custom Invoice Forms

To create a custom invoice form or modify your invoice layout, go to Customization > Forms > Transaction Forms. For detailed instructions, read the help topic [Creating Custom Entry and Transaction Forms](#).

# Custom Workflow Based Invoice Approval

You can create a custom workflow or use the standard workflow for invoice approvals. A custom workflow offers the maximum flexibility for processing approvals. For more information, read the help topic [Using Custom SuiteFlow Workflows for Approval Routing](#).

You can use SuiteFlow to create your own custom workflow to process invoice approvals.

- To use SuiteFlow to create a custom workflow for invoice approvals, you must first enable approval routing for invoices. To enable approval routing for invoices, go to Setup > Accounting > Accounting Preferences. On the **Approval Routing** subtab, check the **Invoices** box. For more information about approval routing, read the help topic [Approval Routing](#).
- To create workflows using SuiteFlow, you must enable SuiteFlow. To enable SuiteFlow, go to Setup > Company > Setup Tasks > Enable Features. Click the **SuiteCloud** tab, and then check the **SuiteFlow** box. Be sure to click **Save**.

For workflow based invoice approvals, you must create a custom workflow.

Consider including the following options in your custom invoice approval workflow:

- Hierarchical or custom routing rules
- Email notifications that include links to drill down to records for approval
- Approve and Reject buttons
- Respecting approval limits
- Updating the Approval Status and Next Approver fields on records
- Preventing pending approval records from being edited
- Designating an alternate approver.

### To complete a basic invoice approvals workflow:

1. Go to Customization > Scripting > Workflows > New (Administrator).
2. Under Basic Information:
  - In the **Name** field, enter **Invoice Approval**
  - In the **Record Type** field, select **Transaction**
  - In the **Sub-Types** field, select **Invoice**
  - In the **Release Status** field, select **Released**
3. Under Event Definition:
  - Check the **On Create** box
  - In the **Context** field, select **CSV Import** and use Ctrl+click to also select **User Interface**
4. Click **Save**.
5. Double-click **State 1** to open the Workflow State page.
6. In the **Name** field, enter **Pending Approval**, and then click **Save**.

7. In the bottom, right side of the page, click **New Action**.
8. Click **Set Field Value**.
9. On the Workflow Action page:
  - Under Parameters, in the **Field** field, select **Approval Status**
  - Under Value, select **Static Value**
  - In the **Selection** field, click the double arrows, select **List**, and then select **Pending Approval**
  - Click **Save**
10. Click **New Action**.
11. Click **Add Button**.
12. In the **Event Type** field, select **View**.
13. In the Parameters section, enter **Approve** in the **Label** field.
14. Click **Save**.
15. In the Workspace, click **New State**.
16. Double-click **New State** to open the Workflow State page.
17. In the **Name** field, enter **Approved**, and then click **Save**.
18. In the Diagram frame, click **Approved**.
19. Click the **Actions** subtab, and then click **New Action**.
20. Click **Set Field Value**.
21. Under Parameters, select **Approval Status** in the **Field** field.
22. Under Value:
  - Select **Static Value**
  - In the **Selection** field, click the double arrows, select **List**, and then select **Approved**
23. Click **Save**.
24. Click the icon at the bottom of the **Pending Approval** state then drag the arrow to create a transition to the **Approved** state.
25. Double-click the arrow to open the Transition window and edit the workflow transition properties as needed.



**Note:** If you use SuiteFlow for invoice approval, all invoices are created with a Pending Approval status by default.

You can always manually set an invoice to an approved status for basic approvals.



**Note:** The SuiteFlow conditions used for Bulk Approvals for records using custom approvals have been updated for the invoice record. Prior to 2016.2, both the workflow button's condition and the Next Approver condition were used to filter records for bulk approvals for the invoice record. The Next Approver condition checks that the **Next Approver** field on the record contains the current logged in user. Beginning in 2016.2, the Next Approver condition is no longer used to filter records for bulk approvals for this record. Records are now filtered for bulk approvals using only the workflow button's condition. If you would like to continue to filter records based on the **Next Approver** field, you must add the condition to the workflow button condition. If you require assistance, contact Customer Support.

## Transactional Impact for Workflow Based Invoice Approvals

Sales Orders – For invoices created against a sales order, the sales order status changes from Pending Billing to Fully Billed.

Payments – If an invoice is not approved, it does not appear in the customer payment queue. When approved, the invoice displays in the customer payment queue.

Inventory-standalone invoices – When pending approval invoices are created, there is no inventory impact. When the approval status changes to approved, quantity on-hand and quantity available are both decremented.

Inventory-Advanced Shipping – When Advanced Shipping is on, pending approval invoices created against sales orders have no inventory impact. When Advanced Shipping is on and when the approval status of an invoice changes from pending approval to approved, quantity on-hand is decreased at the location, inventory number, and bin level. Quantity available is decreased at the inventory numbers level if inventory numbers are not entered on the sales order.

Accounting – Transaction are non-posting until their state changes from pending approval to approved.

Revenue recognition – When invoices require approval, related revenue recognition schedules are not created until the approval is obtained.

## Invoice Approval Workflow

The Invoice Approval Workflow manages the validation and approval routing of an invoice before it is issued to a customer. It executes predefined criteria to determine whether an invoice requires approval. Only incorrect or questionable invoices are required for review by a supervisor or billing manager.

The following validations are executed by the Invoice Approval Workflow:

- Identification of a customer's first invoice
- Customers with a credit hold on the account
- Validation of payment terms on the invoice against the customer record
- Validation of tax amount on the invoice
- Identification of flagged customers

For more information about the validations, see [Invoice Approval Workflow Exception Criteria](#).

You have the option of customizing the workflow and select only the criteria that suits your business needs. As an additional process in invoice processing, learn about the impact of workflow-based approval on other transactions in the topic, [Transactional Impact for Workflow Based Invoice Approvals](#).

## Availability

The Invoice Approval Workflow is included in the NetSuite Approvals Workflow SuiteApp that must be installed in your account. For the steps, see [Installing the NetSuite Approvals Workflow SuiteApp](#).

## Setting Up the Invoice Approval Workflow

Before you can use the Invoice Approval Workflow, you need to complete the following actions:

1. Enable features and a preference. See [Prerequisites](#).
2. Install the NetSuite Approvals Workflow SuiteApp. See [Installing the NetSuite Approvals Workflow SuiteApp](#).
3. Set up roles and permissions. See [Setting Up Roles and Permissions](#).

- Run the Invoice Approval Workflow. See [Running the Invoice Approval Workflow](#).

## Prerequisites

Before using the Invoice Approval Workflow, as an Administrator, be sure to enable the following features and preference:

- Go to Setup > Company > Setup Tasks > Enable Features, and enable the following:
  - On the **Employees** tab, check the box for **Approval Routing**.
  - On the **SuiteCloud** tab, check the box for **SuiteFlow**.
- Go to Setup > Accounting > Accounting Preferences. On the **Approval Routing** tab, check the box for **Invoices**.

For more information about enabling features and preferences, see the help topic [Enabling Features](#).

## Installing the NetSuite Approvals Workflow SuiteApp

The Invoice Approval Workflow is included in the NetSuite Approvals Workflow SuiteApp. Users with the existing SuiteApp can verify if they have the upgraded version by going to Customization > SuiteBundler > Search & Install Bundles > List. On the Installed Bundles page, look for the SuiteApp with the following details:

- Bundle Name: **NetSuite Approvals Workflow**
- Bundle ID: **240841**

Click the link for the SuiteApp to open the Bundle Details page and look for the Invoice Approval Workflow under the **Components** tab. If you still need to install the SuiteApp, use the same details to search for it on the Search & Install Bundles page. For more information about installing bundles, see the help topic [Installing a Bundle](#).



**Note:** This is a managed SuiteApp and is automatically updated whenever there are upgrades. Issue fixes and enhancements are available after the SuiteApp is updated in your account.

## Setting Up Roles and Permissions

After you install the SuiteApp, the following roles are automatically provided full permissions to Invoice Approval:

- CEO
- CFO
- Accountant
- Accountant (Reviewer)
- Bookkeeper
- Warehouse Manager



**Important:** Be sure to designate a supervisor, to take the role of approver, for each invoice creator. For information about assigning supervisors, see the help topic [Assigning a Supervisor to an Employee](#).

For more information about assigning roles and permissions, see the help topic [Customizing or Creating NetSuite Roles](#).

## Running the Invoice Approval Workflow

After installing or creating your custom Invoice Approval Workflow, you have to run it by updating the release status.

### To run the Invoice Approval Workflow:

1. Go to Customization > Scripting > Workflows.
2. On the Workflows list, do either of the following:
  - If you are running the default workflow, click the workflow name, Invoice Approval Workflow.
  - If you are running your custom workflow, click the Edit link for your workflow.
3. On the **Workflow** subtab, located at the right side of the page, click the pencil icon.
4. In the Workflow popup window, do either of the following:
  - If you are running the default workflow, click **Change Status**.
  - If you are running your custom workflow, in the **Release Status** field, select **Released**.



**Important:** Before running your custom workflow, be sure to disable the original Invoice Approval Workflow by setting the release status to **Not Running**.

5. Click **Save**.

After running the workflow, when you return to the list of workflows, the release status for Invoice Approval Workflow should display **Released**. For more information about running a workflow, see the help topic [Release Status](#).

## Customizing the Invoice Approval Workflow

You can modify the Invoice Approval Workflow by enabling only the features that suit your specific business requirements. You can choose which exception criteria to execute in your custom workflow, reorder the sequence of actions, and add more components to the workflow.

When planning your customization, you can review the states and exception criteria in these topics:

- [Invoice Approval Workflow States](#)
- [Invoice Approval Workflow Exception Criteria](#)

To begin the customization, you must create a copy of the workflow. On the workflow page, point to the More link and then click Make Copy. The default workflow states, actions, and other components are copied to your custom workflow. The exception criteria and fields appear in their default settings. For more information, see the topic [Copying a Workflow](#). For instructions on customization, see the following topics:

- [Creating a Custom Invoice Approval Workflow](#)
- [Disabling Exception Criteria](#)

After completing your customization, you must run the workflow. For instructions on running the workflow, see [Running the Invoice Approval Workflow](#).

## Invoice Approval Workflow States

The Invoice Approval Workflow is initiated on creation of an invoice record. It verifies that the account of the invoice creator has a designated supervisor before the invoice can proceed to the initial state. After the invoice has entered the workflow, it is further validated and processed, depending on the conditions that are set for each state.



**Note:** Invoices without an identified approver are automatically approved because they cannot be routed for approval.

## Approval Routing Feature Check

In the Approval Routing Feature Check state, the workflow verifies that the **Approval Routing** feature is enabled in the account used to submit the request. If the feature is enabled, the invoice is processed for validation using the exception criteria.

For more information about the required features and preferences, see [Setting Up the Invoice Approval Workflow](#).

## Invoice Validation

In the Invoice Validation state, the approval status of the invoice is set to **Pending Approval**. The invoice is verified against the exceptions. If one or more exceptions are identified, the invoice is routed for approval. The corresponding message for each exception is added to the transaction, which can be reviewed by the approver. Transactions with no exception are automatically approved.

For more information about the exceptions, see [Invoice Approval Workflow Exception Criteria](#).

## Pending Approval

When it has been determined that an invoice requires approval, it is transferred to the Pending Approval state. The supervisor of the invoice creator is notified through email of the request to be approved.

Both the **Approve** and **Reject** buttons are enabled for the approver. The **Show Exceptions** button is also enabled to display the list of identified exceptions. The invoice is transferred to another state depending on the action of the approver. The record is locked for editing while the invoice is on **Pending Approval** status. At this point, only the administrator can make changes to the record.

## Rejected

Requests that are not approved by the supervisor are sent to this Rejected state. The invoice creator is sent an email to be notified of the status of the request.

A rejected invoice can be updated by another user with access to the record, besides the invoice creator. The **Resubmit** button is enabled for these users. When resubmitted, the invoice goes through the Approval Routing Feature Check state again to start the validation.

## Approved

Requests are transferred to this state when any of the following conditions have been met:

- The invoice has been approved by the supervisor.
- When no exceptions have been identified on the invoice.

The invoice creator is notified of the approval through email.

## Show Exceptions

From the Pending Approval state, the invoice can transition to the Show Exceptions state to allow the approver to view the list of identified exceptions. The **Show Exceptions** button is enabled on the specific invoice record, at the Pending Approval state. After an invoice goes through the Show Exceptions state, it is returned to the Pending Approval state.

## Invoice Approval Workflow Exception Criteria

When a request enters the Invoice Approval Workflow, it is validated against a specific set of exceptions to determine if it requires approval. The list of exceptions described in this section are all executed

within the Invoice Validation state. For more information about the workflow states, see [Invoice Approval Workflow States](#).

- **Credit Hold:** The workflow validates the account of the customer associated with the invoice. If there is a credit hold on the account, invoices for this customer will require approval.
- **Terms Validation:** Comparison is made between the invoice and the customer record. If the payment terms are not the same, the invoice is routed for approval.
- **Tax Amount Exception:** The workflow verifies the tax amount on the invoice. If there is no value for the tax amount, the invoice will require approval.
- **Flagged Customers:** The customer record associated with the invoice is validated. Invoices from those flagged as a new customer, for example, are routed for approval.
- **First Invoice:** The workflow verifies that there are no previous invoices for the customer. If an invoice is tagged as the customer's first invoice, it will require approval.

If you do not need all of these exceptions to be run by the workflow, you have the option of disabling some of them. For instructions on how to disable exceptions, see [Disabling Exception Criteria](#).

## Creating a Custom Invoice Approval Workflow

The instructions in this section show you how to edit or add conditions, actions, transitions, and other components of the Invoice Approval Workflow. The customization is done mostly within the workflow states. To learn more about the states, [Invoice Approval Workflow States](#). For more information about workflow customization, see the help topic [Editing a Workflow](#). For more options on customization, see [Disabling Exception Criteria](#).

**Note:** When modifying the workflow components or reordering the sequence of actions, take note of any dependent conditions or triggers that might affect the flow of processing.

After creating your custom workflow, you have to disable the original Invoice Approval Workflow in case it is still running. To do so, change the release status to **Not Running**. When the original workflow is already disabled, you can run your custom workflow by setting its release status to **Released**. For instructions on running a workflow, see [Running the Invoice Approval Workflow](#).

### Modifying a condition

You can edit or add a condition for a workflow, group, action, and transition. On the workflow or state page, click the Edit link for the component to be customized. The specific component page contains the Condition section where you can edit or add a condition.

The screenshot shows the 'Workflow Action' screen with the 'Condition' tab selected. The 'Basic Information' tab displays the following details:  
 - WORKFLOW: Invoice Approval Workflow (2)  
 - STATE: Pending Approval  
 - TYPE: Send Email  
 - SCRIPT ID: workflowaction\_iaw\_act\_emailnxtapp  
 The 'Condition' tab shows:  
 - USE: VISUAL BUILDER (radio button selected)  
 - CONDITION: Next Approver Is Not Empty

On the Condition section, you have the option to use the Visual Builder or enter your own formula. To edit or add using the Visual Builder, click the Visual Builder icon located beside the **Condition** field. To use the formula, make sure that the **Custom Formula** radio button is selected. In the **Formula** field, enter the new formula or edit the existing one. For more information about adding and editing conditions, see the help topic [Working with Conditions](#).

### Reordering an action

Actions are listed within the state where they are executed. On the Workflow State page, drag and drop the action before or after another one.

<a href="#">Actions</a> • <a href="#">Transitions</a> • <a href="#">Fields</a> •								
	<a href="#">Move To Top</a>	<a href="#">Move To Bottom</a>	<a href="#">New Action</a>	<a href="#">New Group</a>				
EDIT	NAME	PARAMETERS	TRIGGER ON	EVENT TYPE	CONTEXT	CONDITION	FORMULA	SAVED SEARCH
⋮ <a href="#">Edit</a>	<a href="#">Set Field Value</a>	Approval Status=Pending Approval	Entry					
⋮ <a href="#">Edit</a>	<a href="#">Set Field Value</a>	State : invoice creator=Workflow : [IAW] Invoice Creator	Entry					
⋮ <a href="#">Edit</a>	<a href="#">Send Email</a>	To: Next Approver, Subject: Invoice {tranid} for Approval	Entry			Next Approver Is Not Empty		

Another way to reorder is by editing the specific Action page. On the Workflow State page, click the Edit link for the action to be reordered. On the Workflow Action page, in the **Insert Before** field, select the action that should run after the one to be transferred. For more information, see the help topic [Reordering Actions](#).

On the Workflow State page, you can also add a new action to the state. Click the **New Action** button to start creating an action. For more information about workflow actions, see the help topic [Working with Actions](#).

### Changing an exception message

You can set up or modify an existing exception message within the workflow state where its action is executed.

#### To change an exception message:

1. On the Workflow State page, click the Edit link for the NAW Create Exception Record WS action.
2. On the Workflow Action page, in the Parameters section, enter the new message in the Value column for the **exception** field.

Workflow State		
⋮ <a href="#">Edit</a>	<a href="#">Set Field Value</a>	Workflow : [IAW] exception flag=T
⋮ <a href="#">Edit</a>	<a href="#">NAW Create Exception Record WS</a>	exception=There is no tax amount on the invoice.,record ID=Workflow : [IAW] Exceptions Record ID (Into Workflow : [IAW] Exceptions Record ID)

3. Click **Save**.

## Disabling Exception Criteria

By default, all exception criteria are enabled and executed by the workflow. You have the option of disabling those that do not suit your business requirements. In effect, this shortens the duration of the workflow run for each invoice. This is done by changing the property of the field that is associated with the exception.

Other guidelines and instructions for customizing the workflow can be found in these topics:

- [Customizing the Invoice Approval Workflow](#)
- [Creating a Custom Invoice Approval Workflow](#)

### To disable exception criteria:

1. On the **Workflow** tab of the context panel, click the **Fields** subtab.
2. Click the pencil icon for the exception criteria field to be edited.

Exception fields are identified by the [PREF] prefix, for example, [PREF] Terms Validation.



**Important:** The **Fields** subtab also shows fields with the [IAW] prefix. To avoid disrupting the workflow process, make sure that those fields are not changed or deleted.

The screenshot shows the 'Workflow' tab selected in the top navigation bar. Under the 'Workflow' tab, the 'Fields (10)' subtab is active. A list of fields is displayed, including:  
 - [IAW] Exceptions Record ID  
 - [IAW] Invoice Approval Routing  
 - [IAW] Invoice Creator  
 - [IAW] Is Customers First Invoice  
 - [IAW] exception flag  
 - [PREF] Customer 1st Invoice Validation  
 - [PREF] Customer Invoice Needs Approval  
 - [PREF] Customer Validation  
 - [PREF] Tax Amount check  
 - [PREF] Terms Validation  
 At the bottom of the list, there is a button labeled '+ New Workflow Field'.

3. On the Workflow Field page, click the **Validation & Defaulting** subtab.
4. Clear the **Default Checked** box.
5. Click **Save**.

To help you decide which criteria to execute in your custom workflow, refer to the complete list in the topic, [Invoice Approval Workflow Exception Criteria](#). For more information about workflow fields, see the help topic [Workflow Custom Fields](#).

# Using the Billing Tab and Dashboard

Use the Billing tab and dashboard to increases productivity for accounts receivables (A/R) professionals. The billing dashboard provides a consolidated view and place to process, analyze, and review a large number of invoices.

Using the A/R Clerk role, access the billing dashboard by going to Billing > Billing Overview. The Billing tab is available to all standard and customized roles created from the Accounting Center. Depending on your permissions, the list of tasks, menu options, and reminders available is varied. This feature does not have to be enabled as it is automatically available with the A/R Clerk role. Like other NetSuite dashboards, you can customize the portlets, meters, and metric that show in your billing dashboard.

Depending on your permissions, you can access the following options from the Billing tab:

- [Saved Searches](#)

From here you can access the following reports:

- [Customers by Open A/R Balance](#)
- [Customers by Outstanding Balance](#)
- [Customers by Unbilled Orders](#)

- [Sales](#)

From here you can:

- [Creating an Invoice](#)
- [Entering a Cash Sale](#)
- [Invoicing Sales Orders](#)
- [Invoicing Billable Customers](#)
- [Generating Statements](#)

- [Billing Operations](#)

From here you can:

- [Billing Operations](#)
- [Billing Operations](#)

- [Creating Terms of Payment](#)

- [Access to Reports](#)

By default and based on your permissions, the billing dashboard shows the following reminders:

- [Invoices to Approve](#)

 **Note:** This reminder shows the number of invoices pending approval regardless of whether you are the next approver. This is opposed to the Invoices Pending My Approval reminder in the Worklist portlet, which only shows invoices that are pending approval and you are the next approver.

- [Invoices Overdue](#)
- [Invoices to Print](#)
- [Invoices to Reject](#)

New reminders available as of Version 2016 Release 2 but do not display by default include:

- [Invoices to Approve Due to Credit Hold](#)

- Invoices to Approve Due to Unknown Tax Amount
- Invoices to Approve to Inconsistent Payment Terms

Based on your permissions, the billing dashboard Quick Links provide access to:

- Customer saved searches
- Billing-related tasks
- Billing-related setups
- Reports

Based on your permissions, billing dashboard KPIs and Trend Graphs include:

- Receivables KPI meter
- Receivables KPI list
- Monthly receivables trend graph

Based on your permissions, the billing dashboard Worklists include:

- Invoices Pending My Approval

**Note:** This reminder shows only invoices that are pending approval and you are the next approver. This is opposed to the Invoices to Approve reminder, which shows the number of invoices pending approval regardless of whether you are the next approver.

- Billing Operations From Last 5 Days

For additional information about dashboards and portlets in NetSuite, see the help topic [Dashboard Personalization](#).

## Global Invoicing Preferences

Use the Invoicing Preferences page to determine the way billable lines appear on invoices. To set invoicing preferences, go to Setup > Accounting > Preferences > Invoicing Preferences.

As you select options in various combinations, a preview at the bottom of the page reflects how invoices appear using the settings. You can select and clear options until the Preview section reflects the way you prefer for invoices to show.

When the Preview section accurately reflects your preference for the appearance of invoices, click Save.

The subtabs that appear on the Invoicing Preference page are determined by the settings that you have enabled. If you have charge-based billing enabled, the Charge subtab appears on the page.

The following options are available from the Invoicing subtab.

### ■ Combine Time Items on Invoices

Check this box to group time items into one total on invoices.

For example, if one employee reports 8 hours of data entry for 20 days to the same customer, the customer will see 160 hours of time at the billable rate on the invoice.

**Note:** You must enable this preference to also enable the Group Time by Employees preference.

### ■ Group Time by Employee

Check this box to group billable time by employees when you combine time items on invoices.

When you enable this preference, you can select Employee as a grouping criteria on invoices when billable time exists with the same employee, service item, and rate. Then, when you print the invoice, the time for each employee is combined in one line.



**Note:** You must enable Combine Time Items on Invoices to use this preference.

#### ■ **Combine Expense Items on Invoices**

When you enable this preference, expense items are combined into one total on invoices.

If billable expenses have the same expense category, they are grouped on the printed invoice. Also, the memo field can be edited for only the first entry listed in each category.

- You can enter a different rate for each line.
- If a billable expense line does not identify a category, it is not grouped.

When you clear this box, each billable expense is separate on the invoice and they are not grouped on the printed invoice. You can enter different rates and memos for each expense.

#### ■ **Group Expenses by Employee**

This preference groups billable expenses by employees when you combine expense items on invoices.

When you enable this preference, you can select Employee as a grouping criteria on invoices when billable expenses exist with the same employee, service item and rate. Then, when you print the invoice, the expenses for each employee are combined in one line.

#### ■ **Combine Billable Items on Invoices**

Check this box to combine expense items into one total on invoices.

For example, if one employee enters a \$50 expense for 5 days that is billable to the same customer, the customer will see \$250 in expenses on the invoice

#### ■ **Group Billable Types on Invoices**

When you check this box, similar billable charges are grouped together with their subtotals on your customers' invoices. All billable items, time, and expenses are grouped together by type.

For example, an invoice will show Billable Group as a heading with line items for billable time and a billable time total.

A Billable Group heading and total will also show for billable expenses and for billable items.

#### ■ **Show Billable Subtotals on Invoices**

Check this box to show subtotals for grouped billable lines on invoices. Billable time, expenses or items are grouped together by item or category name. Then, these items are grouped with subtotals when you print, e-mail or fax the invoice.

For example, if you enter two expense reports for air travel to bill back to a customer, they are grouped as one total. If you enter another expense report for meals, the expense is still listed separately from the air travel charges.

Billable lines are grouped and subtotalized regardless of whether a markup or discount is applied.



**Note:** When you choose to add billable subtotals, the subtotal is calculated for the column used to sort the billable list. The secondary sort is by the previously sorted column. For example, to sort and subtotal by project then by category, click on the category header then the project header.

You can change the order of billable items, expenses, and time by customizing the invoice form to reorder the lists under the lists tab. To customize invoices, open the invoice and click the Customize button.

The following options are available on the Subscription Management subtab:



**Note:** The **Subscription Management** subtab does not appear until after you enable SuiteBilling.

■ **When Sales Order Gets Approved, Auto-Change Subscription Status to "Pending Activation"**

Check this box to automatically change the subscription lines in the subscription from Draft to Pending Activation when the associated sales order is approved.

■ **First Activation Change Order Creation Sets the Subscription Start Date**

Check this box to set the subscription start date to the effective date of the first Activation change order. Note: When you change the effective date of the first Activation change order, this preference no longer applies to that change order.

■ **Request Credit Memo for Off-Cycle Change Orders**

Check this box to automatically generate credit memos for negative charges that occur when using the billing operations off-cycle functionality.

■ **Align Charge Amounts with Subscription**

Check this box to align subscription charges with the subscription start date. When you check the Align Charge Amounts with Subscription preference, all lines within an activated subscription align charges with the subscription header start date instead of its bill dates. Multiple-year subscriptions with multiple charge frequencies, such as monthly, quarterly, and annually, recur based on the subscription start date for better alignment of charge periods. Existing subscriptions activated before setting this preference may display the original recurrence pattern. When you activate a subscription, the current preference setting dictates the subscription charge pattern.

■ **Months To Extend Estimated Revenue Recognition End Date**

For recurring subscriptions with an evergreen term, enter an integer from 1 to 120. Suggested values are:

- If you bill subscriptions monthly, set this preference to 1 or greater.
- If you bill subscriptions quarterly, set this preference to 3 or greater.
- If you bill subscriptions annually, set this preference to 12 or greater.
- If you enter a value for this preference, you also must enter a value between 1 and 120 for the following preference:

■ **Months Prior To Estimated Revenue Recognition End Date To Initiate Auto-Extend**

For recurring subscriptions with an evergreen term, enter an integer from 1 to 120. This value pushes the estimated revenue recognition end date by this many months each time a subscription matches the auto extend.

■ **Create Delta Charges for Changes to Invoiced Service Periods**

If changes to invoiced service periods result in a change to what the customer owes while this preference is enabled, the system generates a Delta charge. The amount on a Delta charge is the difference (+ or -) between the invoiced amount and the new amount for the service period.

To see how to set this preference, view the video:



[Enabling Delta Charges](#)

For example, you create an invoice of \$1000 on January 1 for an annual recurring service with monthly proration. On July 1, a Modify Pricing change order increases the quantity from 1 to 2. The following chart shows you how enabling the **Create Delta Charges for Changes to Invoiced Service Periods** preference simplifies the billing for this change:

Charge Period	Preference Off	Preference On
1/1-12/31	\$1000.00	\$1000.00

Charge Period	Preference Off	Preference On
	-\$1000.00	
1/1-6/30	\$500.00	
6/30-12/31	\$1000.00	\$500.00

The following options are available on the Charge subtab.

■ **Group Charges By**

Use the drop down list to chose the field you want to use to group charges on invoices. When you generate a customer invoice, all charges with the same information in the selected field are combined on a single line on the invoice.

**Note:** When you bill charges on the Invoice Billable Customers page, a separate invoice is created for each grouping. The invoices created include line items for each grouped charge.

■ **Map To**

If you are grouping charges based on a custom field, use the drop down list to select the invoice field to which you want to map that custom field. When an invoice is created, the information in that custom field populates the field you select.

■ **Automatically Add Charges to As-Of Date to Invoices, Cash Sales, Credit Memos, and Cash Refunds**

Check this box to provide a distinction between to transaction date (Date field) and the as-of date. Selecting this box may have an impact on system performance.

Use the As-of Date field when you need to separate the date when charges are added to an invoice from the transaction date. The transaction date impacts the general ledger. The as-of date does not impact the general ledger.

**Note:** If you do not check this box, you have the option to add charges on these transaction types by clicking the Add Charges button on the transaction form.

If you create a sales order after you run charges, charges created by that run are not linked to the sales order, and therefore are not transferred from the sales order to the invoice. To view the charges that were created by the sales order, delete the charges already created, and rerun the charge.

## Invoicing Preferences for Individual Forms

In addition to the company-level settings for invoicing preferences, you can set preferences for invoicing options on customized forms.

For example, you can customize a service invoice form that uses invoicing preferences that are different than the company-level preferences set for other invoice forms. The customized service invoice can group time items even if all other invoices do not.

To set invoicing preferences for individual forms, go to Setup > Customization > Forms > Transaction Forms (Administrator), and click Customize next to the form. On the Preferences subtab, invoicing preferences for this form can be chosen based on the setting for the **Use Defaults** box.

- Clear the Use Defaults box to allow invoice presentation settings that are individual to this form. The company settings are selected by default, but you can change the settings by checking boxes on this tab.
- If you check the Use Defaults box, the invoice presentation determined by company-level settings are also used on this custom form. Then, the settings cannot be changed on this subtab, they must be set at the company level only.



**Note:** Company level settings are set at Setup > Accounting > Preferences > Invoicing Preferences.

If the Use Defaults box is clear, you can choose invoice settings for the form.

## Classes and Preferences

The billable lines on an invoice get assigned the class from the corresponding to the purchase transaction if you **do not** enable these preferences:

- Combine Time Items on Invoices
- Combine Expense Items on Invoices
- Combine Billable Items on Invoices

When these are disabled, each billable line gets its own general ledger line. You can see these billable lines in the GL Impact page. To view this, click the GL Impact link at the top of the form.

However, if you do enable these preferences, no class is populated on billable lines unless you assign a class to the entire invoice at the transaction body level.

## Creating an Invoice

An invoice is a record of sale to a customer. Enter an invoice when payment for goods or services is not received at the time of delivery.

Record sales on invoices by adding each item sold from your Items list. Each invoice consists of line-items and their sales amounts. The sum of all sales amounts on an invoice equals the total amount the customer owes.



**Note:** The fields described below show on standard invoice forms. To create a custom invoice form, go to Setup > Customization > Forms > Transaction Forms.

### To create an invoice:

1. Go to Billing > Sales > Create Invoices.
2. Under Primary Information:
  - a. In the **Custom Form** field, select the invoice form you want to use. Invoice forms can be standard or customized.
  - b. Accept or enter the invoice number in the **Invoice #** field.



**Note:** You can edit an automatically generated invoice number only if your company permits you to do so.

- c. In the **Customer or Project** field, select the appropriate customer or project.



**Note:** If you add, edit, or delete a project after a sales order is billed, the sales order becomes a standalone sales order and is no longer associated with the invoice. Also, the invoice is removed from the Related Records subtab on the sales order.

- d. Accept or enter the date in the **Date** field.



**Note:** If you later edit the date on an invoice, the new date also becomes the due date, whether the due date value was previously set.

- e. In the **Start Date** field, if you use Advanced Billing, enter the creation date for the first invoice.
- f. In the **End Date** field, enter the date this invoice is no longer valid.
- g. If you use accounting periods, select the posting period for this invoice.
- h. In the **Due Date** field, enter the invoice due date.
- i. In the **PO#** field, enter any customer purchase order number associated with this invoice.
- j. In the **Memo** field, enter a memo to identify this invoice.

When you use the Search Transactions feature, you can search for specific words and phrases in the Memo field.

3. Under Sales Information:

- a. Select the sales reps associated with this invoice.

Associate sales reps with this transaction in one of the following ways:

- If you do not use the Team Selling feature, select the sales rep or sales group in the **Sales Rep** field.

The sales rep or sales group associated with the customer on this transaction is chosen by default.

- If you use the Team Selling feature, click the **Sales Team** subtab.

Select the sales team for this transaction. For more information, see the help topic [Associating Sales Teams with Customers and Transactions](#).

- b. In the **Opportunity** field, select the opportunity associated with this invoice.

You can change the sales effective date for this transaction. The sales effective date determines which commission plan and historical sales team this transaction applies to.

- c. Check **Exclude Commissions** to exclude this transaction and its subordinate transactions from inclusion in all commission calculations. For example, selecting this box on a sales order excludes the sales order and the resulting invoice from all commission calculations for all sales people.

- d. In the **Lead Source** field, select the source of the lead for this invoice. If you use Marketing Automation, select the campaign you want to associate with this transaction's revenue.

- e. The partner associated with this customer appears in the Partner field. You can choose a different partner.

If you have enabled the Multi-Partner Management feature, you can associate partners with this transaction on the Partners subtab. For more information, see the help topic [Associating Partners With Customers and Transactions](#).

- f. Under **Classification**, select a department, class, or location to associate with the invoice.



**Note:** If you use NetSuite OneWorld and the selected customer is shared with multiple subsidiaries, you can choose any subsidiary assigned to the selected customer. For information about sharing customer records with multiple subsidiaries, see the help topic [Assigning Subsidiaries to a Customer](#).

The Invoice page includes several subtabs. For more information about invoice subtabs, see the links in the following list:

- The [Adding Items on a Sales Transaction](#) subtab.

- The [Entering Shipping Information on a Sales Transaction](#) subtab.
- The [Adding Billing Information to a Transaction](#) subtab.
- The [Entering Accounting information about a Sales Transaction](#) subtab.
- The [Entering Relationships on a Transaction](#) subtab.
- The [Associating Sales Teams with Customers and Transactions](#) subtab.
- The [Entering Communication Information on a Transaction](#) subtab.

You can apply terms to invoices by choosing a term on the invoice record or by setting default terms on the customer record. For more information, see [Using Terms of Payment](#).

For information about enabling tax lookup, see the help topic [Enabling U.S. Tax Lookup](#).

## Choosing an Invoice Type

NetSuite includes the following standard invoices types:

- **Finance Charge** – Use this only for assessing finance charges, not for normal products and services. It includes the date, finance charge number, bill-to address, terms, due date, project, item, description, amount, and total.



**Note:** This invoice type does not show in the list for UK accounts.

- **Product** – This includes all the information from the standard service invoice. In addition, it includes the ship-to address, sales representative, ship date, FOB, and unit price.
- **Professional** – This is the simplest design. It includes the date, invoice number, bill-to address, terms, due date, project, item name, description, amount, tax, and total.
- **Service** – This includes all the information from the standard professional invoice. In addition, it includes the PO# and itemizes a quantity and rate for each line item in the body of the form.
- **Progress** – Use progress billing to bill for portions of projects as they are completed. It includes the date, invoice number, bill-to address, terms, due date, PO#, sales rep, end date, shipping information, item name, current percentage completed, description, amount, and a percentage of the total cost.

### To select an invoice type:

1. Go to Billing > Sales > Create Invoices.
2. Under Primary Information, in the **Custom Form** field, select a standard invoice or an invoice form you have customized.
3. Continue to fill out details on the Invoice page.
4. When you have finished, choose one of three options to post the information to your NetSuite account:
  - Click **Save** to submit the information and return to the Transactions page.
  - Click **Save & New** to submit the information and return to another blank Invoice page.
  - Click **Print** to save the information and immediately print the invoice form you created.

## Custom Invoice Forms

To create a custom invoice form, go to Customization > Forms > Transaction Forms.

# Invoice Groups Overview

The Invoice Groups feature lets you group invoices for your customers. It gives you greater flexibility in meeting your customer's expectations for how and when they receive invoices by combining multiple invoices into one invoice group.

Each customer can have multiple invoice groups.

After [Enabling the Invoice Groups Feature](#) for a customer, you must check the **Group Invoices** box located on the **Financial** subtab of the customer record. If you have a sales order for which you do not want the invoice grouped for this customer, clear the **For Invoice Grouping** box. If there is a sales order associated with an invoice you want to group, the For Invoice Grouping box must be checked on the sales order.

An invoice that is eligible for grouping is not required to have an associated sales order. Examples of when an invoice does not always have a sales order include invoices generated from subscription or projects and invoices that are imported to NetSuite.

An invoice group does not post to the general ledger. Any general ledger posting occurs from the individual invoices that are part of the group.

You can apply payments to an invoice group. Payments are applied to all the invoices that are in that group. NetSuite supports both full and partial payments for invoice groups. When accepting a partial payment, choose the invoices in the group to which you want to apply the payment. Invoice groups also support the Consolidated Payments feature. For more information, see the help topic [Consolidated Payments](#).

Each user has one set of permissions and restrictions for accessing invoice groups. Invoice groups inherit permissions and restrictions from the invoices that are in the group. If there are fifty invoices in an invoice group and you have permission to only view ten of the invoices, you only see those ten invoices. You also see the full total of the invoice group, including any invoices for which you do not have permission.

This book includes the following chapters and topics:

## [Invoice Groups Overview](#)

- [Invoice Groups Workflow Diagram](#)
- [Invoice Groups Statuses and State Transitions](#)
- [Invoice Groups and Installments](#)

## [Invoice Groups Setup](#)

- [Enabling the Invoice Groups Feature](#)
- [Numbering Invoice Groups](#)
- [Using Billing Operations to Generate and Group Invoices](#)
- [Customizing the Invoice Group Form](#)
- [Setting a Subsidiary for an Invoice Group](#)
- [Using SuiteFlow to Create an Invoice Groups Workflow](#)

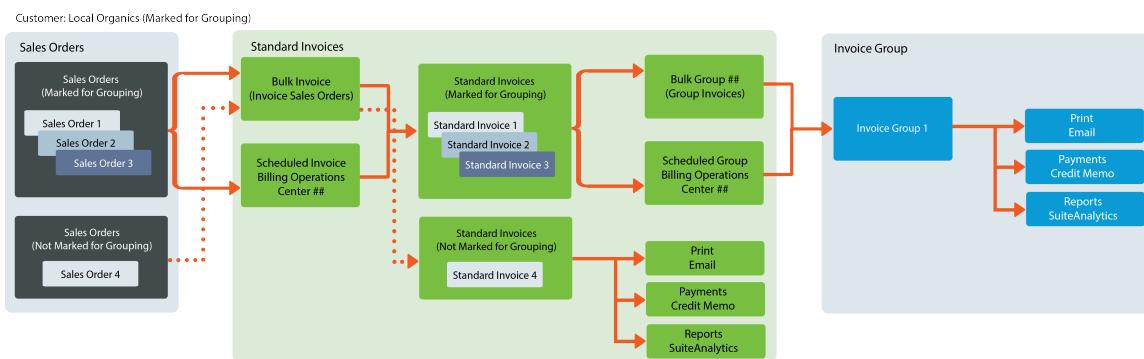
## [Invoice Groups Management](#)

- [Creating an Invoice Group](#)
- [Viewing the Invoice Group Record](#)
- [Viewing an Individual Invoice in an Invoice Group](#)

- Adding an Invoice to an Invoice Group From the Invoice Groups Record
- Adding an Invoice to an Invoice Group From the Invoice Record
- Removing an Invoice From an Invoice Group
- Grouped Invoices Due Date
- Invoice Groups and Closed Periods
- Accepting Payment for an Invoice Group
- Generating and Printing an Invoice Group PDF
- Invoice Groups and Customer Statements
- Invoice Group Reporting and the A/R Aging Report

## Invoice Groups Workflow Diagram

The following diagram shows the workflow for the Invoice Groups feature.



## Invoice Groups Statuses and State Transitions

The table below shows the sales order, invoice, and invoice group status in various transaction states. The states are listed in order, from creating a sales order to paid invoices.

State	Sales Order Status	Invoice Status	Invoice Group Status
Sales order is created, fulfilled, and billed	Pending fulfillment Fulfilled/Billed	Open-ungrouped Invoice status is open. Label is open-ungrouped	Not applicable Record not created
Invoice group is created	Pending fulfillment Fulfilled Billed	Open-grouped Invoice status is open. Label is open-grouped Invoice 1: Open, grouped Invoice 2: Open, grouped Invoice 3: Open, grouped	Open
Invoices 1 and 2 are paid. Invoice 3 is unpaid	Pending fulfillment Fulfilled Billed	Invoice 1: Paid in full Invoice 2: Paid in full	Open-partially paid

State	Sales Order Status	Invoice Status	Invoice Group Status
		Invoice 3: Open, grouped. Unpaid or partially paid	
All invoices in the group are paid	Pending fulfillment Fulfilled Billed	Invoice 1: Paid in full Invoice 2: Paid in full Invoice 3: Paid in full	Paid in full

## Invoice Groups and Installments

You can enable the Invoice Groups feature if the Installments feature is enabled, but an invoice with an installment term cannot be grouped. If you select an installment term on a sales order or invoice for a customer that has invoice grouping enabled, the transaction is not saved and an error message appears. To save a transaction with an installment term, you must clear the For Invoice Grouping box. If the term changes from installments to a different term, the Group Invoices box is checked, and the invoice is eligible for grouping.

If an invoice is marked for grouping, the Term field is dimmed.

For more information about the Installments feature, see [Creating Installments](#)

## Invoice Groups Setup

This chapter includes the steps required to enable and setup the Invoice Groups feature. This chapter includes the following topics:

- [Enabling the Invoice Groups Feature](#)
- [Numbering Invoice Groups](#)
- [Using Billing Operations to Generate and Group Invoices](#)
- [Customizing the Invoice Group Form](#)
- [Setting a Subsidiary for an Invoice Group](#)
- [Using SuiteFlow to Create an Invoice Groups Workflow](#)

## Enabling the Invoice Groups Feature

Enable the Invoice Groups feature from the Enable Features page.

 **Warning:** After the Invoice Groups feature is enabled and an invoice group has been created, it can no longer be disabled. Before enabling this feature in your production account, it is important to test this feature in your sandbox or release preview account. If you need further assistance, contact your NetSuite account manager.

### To enable the Invoice Groups feature:

1. Go to Setup > Company > Enable Features.
2. Click the **Transaction** subtab.
3. From the **Billing** section, check the **Invoice Groups** box.



**Tip:** If you can't see the Invoice Groups feature, switch to an administrative role or contact your NetSuite administrator.

## Numbering Invoice Groups

If you do not set up custom numbering for your invoice groups, they are numbered sequentially, beginning with the number one (1).

An invoice group is not a transaction, therefore it does not appear in transaction searches. However, invoice groups number similar to transactions, and they appear in the Transaction Numbering Audit log.

You can setup prefixes, suffixes, and a minimum number of digits for an invoice group number.

### To customize invoice group numbering:

1. Go to Setup > Company > Auto-Generated Numbers.
2. Click the **Document Numbers** subtab.



**Note:** Because an invoice group is not a transaction, it appears under the Document Numbers subtab instead of the Transactions subtab.

3. Locate Invoice Group in the **Type** column.
4. Enter information in the following column fields:
  - Prefix
  - Suffix
  - Minimum Digits
5. If you want to restart numbering, check the **Update Initial Number** box.
6. Click **Save**.

To create subsidiary-set numbering for an invoice group number, check the Use Subsidiary box.

For more information about numbering, see the help topic [Set Auto-Generated Numbers](#).

Even though an invoice group is not a transaction, you can use the Transaction Numbering Audit Log to view information about an invoice group. For more information, see the help topic [Transaction Numbering Audit Log](#).

## Using Billing Operations to Generate and Group Invoices

You can use the Billing Operations feature to group invoices based on common criteria. When using this option, you cannot group invoices by PO #. For information about enabling the Billing Operations feature, see [Billing Operations](#).

### To use Billing Operations to generate and group invoices:

1. Go to Transactions > Billing > Process Billing Operations.  
Alternatively, go to Transactions > Billing > Schedule Billing Operations.
2. In the Billing Operation Options section, do the following:
  - a. To generate or update charges during the billing operation, check the **Rating Run** box.
  - b. To generate credit memos during the billing operation, check the **Credit Memo Run** box.

- c. To generate invoices during the billing operation, check the **Bill Run** box.
- d. To group invoices during the billing operation, check the **Group Invoices** box.
- e. (Optional) To prevent grouping new invoices generated during the billing operation, check the **Do Not Apply Grouping for Bill Run** box.

For new invoices created during the billing operation, the **For Invoice Grouping** box on the invoice record will be cleared. These invoices will not be grouped.

If you also checked the **Group Invoices** box, preexisting invoices created before the billing operation will still be grouped.

3. Click **Save**.



**Important:** You can use the Billing Operations feature to create invoice groups without billing the sales orders. To only group invoices, clear the **Bill Run** box. When you click **Run** on the Billing Operations page, or **Save** on the Billing Operations Schedule page, invoices are grouped, but sales orders are not billed.

## Customizing the Invoice Group Form

NetSuite lets you customize forms, including the Invoice Group form. You can customize the following components:

- Actions
- Custom code
- Fields
- Field groups
- Roles
- Sublists
- Subtabs

### To customize the Invoice Group form:

1. Go to Customization > Forms > Entry Forms.
2. Click the **Customize** link on the Standard Invoice Group Form line.
3. Use the Custom Entry Form page to customize the form.
4. You can change the text in the **Name** field to name this customized form.
5. Click **Save**.

You cannot use this customized form to generate a PDF statement because it is an entry form and not a transaction form.

For more information, see the help topic [Creating Custom Entry and Transaction Forms](#).

## Setting a Subsidiary for an Invoice Group

Use the Invoicing Preferences page to set a default subsidiary for customers who want invoices grouped. You can set this preference for new customers.

### To set a subsidiary for a new invoice groups customer:

1. Go to Setup > Accounting > Invoicing Preferences.
2. Select the **Invoice Groups** subtab,
3. Select the appropriate subsidiary from the **Group Invoices for New Customers** list.



**Note:** The list only shows active subsidiaries. If you are not a OneWorld customer, the only option is Parent Company.

4. Click **Save**.

## Using SuiteFlow to Create an Invoice Groups Workflow

Use SuiteFlow to define and run a workflow for the invoice group record. Any workflows that include the invoice group record appear under the System Information subtab on the invoice group record. You cannot create an invoice group record using SuiteFlow.

### To create an invoice groups workflow:

1. Go to Customization > Worfklow > Workflows > New.
2. From the **Record Type** field, select **Invoice Group**.
3. Complete the following steps:



**Note:** For detailed information about completing these steps, see the help topic [SuiteFlow \(Workflow\)](#).

- a. Define basic information for your workflow.
  - b. Define initiation information for your workflow.
  - c. Define conditions for your workflow.
  - d. Create workflow states.
  - e. Create actions.
  - f. Create a transition.
  - g. Initiate and validate the workflow.
4. Click **Save**.

## Invoice Groups Management

This chapter describes tasks that you perform to manage invoice groups.

For instructions on enabling and setting up the Invoice Groups feature, see:

- [Creating an Invoice Group](#)
- [Viewing the Invoice Group Record](#)
- [Viewing an Individual Invoice in an Invoice Group](#)
- [Adding an Invoice to an Invoice Group From the Invoice Groups Record](#)
- [Adding an Invoice to an Invoice Group From the Invoice Record](#)
- [Removing an Invoice From an Invoice Group](#)
- [Grouped Invoices Due Date](#)
- [Invoice Groups and Closed Periods](#)

- Accepting Payment for an Invoice Group
- Generating and Printing an Invoice Group PDF
- Invoice Groups and Customer Statements
- Invoice Group Reporting and the A/R Aging Report

## Creating an Invoice Group

When you enable invoice groups on the customer record, you can set sales orders for invoice grouping by checking the **For Invoice Grouping** box. When the sales order is invoiced, the invoice is eligible for invoice grouping. When you group invoices, you can do the following actions:

- print the invoice group information.
- accept payments or apply a credit memo.
- use saved searches for invoice group reporting.

You can create an invoice group with a maximum of 100 invoices. If you attempt to group more than 100 invoices, a new invoice group is created. If you require more than 100 invoices per invoice group, contact your account manager.

Invoices are grouped based on the following criteria:

- Customer
- Terms
- Currency
- Billing Address
- Subsidiary
- PO # (optional)



**Warning:** After the Invoice Groups feature is enabled and an invoice group has been created, it can no longer be disabled. Before enabling this feature in your production account, it is important to test this feature in your sandbox or release preview account. If you need further assistance, contact your NetSuite account manager.

### To create an invoice group:

1. Go to Lists > Relationships > Customers.
2. Click **Edit** to open the appropriate customer record.
3. Click the **Financial** subtab on the customer record.
4. Check the **Group Invoices** box.

When you check the Group Invoices box, all new invoices that are eligible for grouping for this customer appear on the Group Invoices page. If you do not want to group an invoice for this customer, clear the **For Invoice Grouping** box on the associated invoice.



**Tip:** Before continuing to step 5, you must create or generate a new invoice that is eligible for grouping. If you mark a sales order for grouping, you can also transform that sales order into an invoice that is eligible for grouping.

5. Go to Transactions > Sales > Group Invoices to view the list of invoices ready for grouping.
6. From the **Customer** list, select the customer for which you want to group invoices. Select **All** to group invoices for all customers.

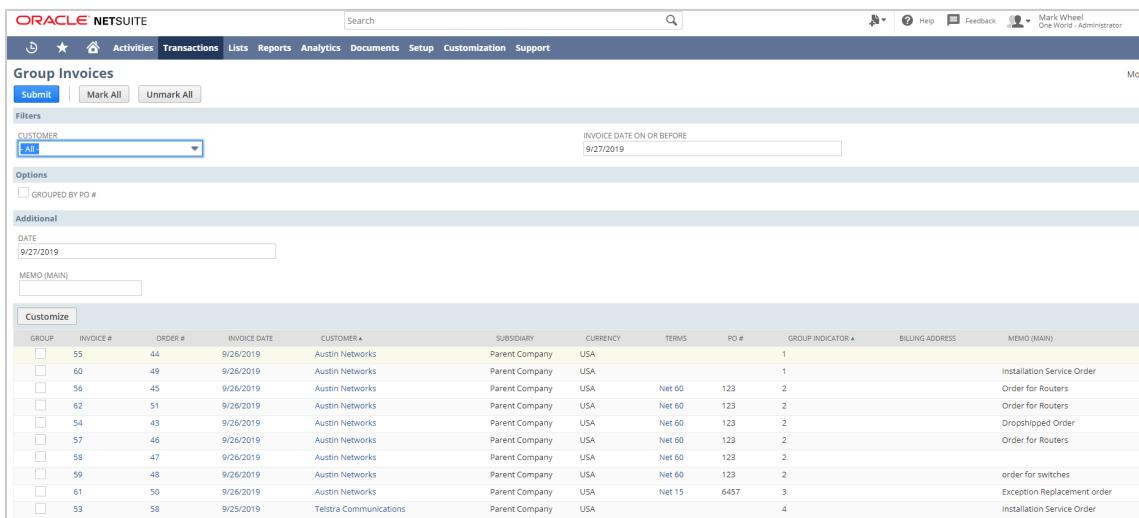
7. If you want to group by purchase order number, check the **Grouped by PO #** box.
8. Check the **Group** box next to the invoices that you want to group.

**Note:** The Group Indicator column shows you which invoices are grouped together. For example, a list of 10 invoices with two groupings:

- Invoices with an indicator number of 1, associated with one purchase order
- Invoices with an indicator number of 2, associated with a different purchase order

This example assumes that you are using the Group by PO# criteria.

9. Click **Submit**.



The screenshot shows the Oracle NetSuite interface for 'Group Invoices'. The top navigation bar includes 'Activities', 'Transactions' (which is selected), 'Lists', 'Reports', 'Analytics', 'Documents', 'Setup', 'Customization', and 'Support'. A 'Mark Wheel' link is also present. The main area has a title 'Group Invoices' with buttons for 'Submit', 'Mark All', and 'Unmark All'. It features 'Filters' for 'CUSTOMER' (set to 'A/R') and 'INVOICE DATE ON OR BEFORE' (set to '9/27/2019'). Under 'Options', there is a checked checkbox for 'GROUPED BY PO #' and an unchecked checkbox for 'Additional'. The 'DATE' field is set to '9/27/2019'. The 'MEMO (MAIN)' field is empty. The 'Customize' section at the bottom contains a table with columns: GROUP, INVOICE #, ORDER #, INVOICE DATE, CUSTOMER▲, SUBSIDIARY, CURRENCY, TERMS, PO #, GROUP INDICATOR▲, BILLING ADDRESS, and MEMO (MAIN). The table lists 12 invoices, grouped into two main categories based on their 'GROUP' value (1 and 2). The first group (Group 1) consists of invoices 55, 60, 56, 62, 54, 57, 58, 59, 61, and 53. The second group (Group 2) consists of invoice 49. The 'MEMO (MAIN)' column for Group 1 contains descriptions like 'Installation Service Order', 'Order for Routers', 'Order for Routers', 'Dropshipped Order', 'Order for Routers', 'order for switches', 'Exception Replacement order', and 'Installation Service Order'.

**Note:** The Invoice Group feature does not support custom addresses.

## Viewing the Invoice Group Record

The invoice group record has three different available views: A/R, All Details, and Summary.

### To view the invoice group record:

1. Go to Transactions > Sales > Group Invoices > List.
2. Click the **View** link next to the invoice group you want to view.
3. To change your view, click the **Details** subtab, and then select the view you want from the **View** list.

Use these different views to show information in the format that you want:

- **A/R** – The A/R view only shows lines that post to the general ledger. Posting discounts are included in the view. All other discounts are included in the product price. Subtotals are not included.
- **All Details** – The details view shows tax information, subtotals, and discount amounts.
- **Summary** – The summary view shows a list of invoices linked to the invoice group.

The screenshot shows the Oracle NetSuite interface for managing invoice groups. The top navigation bar includes links for Activities, Transactions, Lists, Reports, Analytics, Documents, Setup, Customization, and Support. The user is logged in as 'Mark Wheel' with the role 'OneWorld Administrator'. The main page title is 'Invoice Group'. Below the title, there are tabs for 'Edit', 'Back', 'Accept Payment', and a download icon. A search bar is located at the top right.

**Primary Information**

CUSTOMER Local Organics	DATE 9/10/2019	INVOICE GROUP NUMBER 7
TERMS Net 60	DUUE DATE 11/9/2019	STATUS Open-Partially Paid
CURRENCY USA	PO #	SUBSIDIARY Parent Company
BILL TO US Poly	MEMO	<input type="checkbox"/> GROUPED BY PO #

**BILL TO**  
US Poly  
United States Map

**Summary**

TOTAL AMOUNT 29,803.08	AMOUNT PAID 12,075.58	AMOUNT DUE 17,727.50
ITEM TOTAL 27,000.00	DISCOUNT TOTAL -834.48	
TAX TOTAL 2,184.56	SHIPPING COST 1,453.00	

**Details** System Notes Related Records

**Invoices**

VIEW	INVOICE NUMBER	LINE SEQUENCE NUMBER	ITEM	ITEM RATE	QUANTITY	AMOUNT
<a href="#">View</a>	Invoice #11	0	1 SM-Security Camera	800.00	15	\$12,000.00
<a href="#">Edit</a>	Invoice #11	1	2 Tier-1 partner discount	-3.56		(\$42.20)
<a href="#">View</a>	Invoice #11	2	4 Tier-1 partner discount	-10.00		(\$1,157.28)
<a href="#">Edit</a>	Invoice #11	3				
<a href="#">View</a>	Invoice #11	4				

## Viewing an Individual Invoice in an Invoice Group

You can view individual invoices that are grouped. From the invoice group record, click the invoice number or the Edit link to view an individual invoice.

### To view an individual invoice in an invoice group:

1. Go to Transactions > Sales > Group Invoices > List.
2. For the invoice group containing the invoice you want to view, click the **Edit** link.
3. To view an invoice in the list, click the associated **View** or **Edit** link.

You can also click the text under the Invoice Number column to open the invoice in View mode.

The status of the invoice appears with the customer name. For example, an unpaid grouped invoice has a status of Open – Grouped.

After an invoice is grouped, you can make edits to the invoice, but there are limitations. You cannot edit the following fields:

- Due Date
- PO #
- Terms (located on the Billing subtab)

## Adding an Invoice to an Invoice Group From the Invoice Groups Record

After you have created an invoice group, you can add more invoices to the group. Any invoices that you want to add to an existing invoice group must have the same criteria as the invoices that are already in the invoice group. For example, if the invoices in the group are all part of the same subsidiary, then any invoices you want to add must be in the same subsidiary.

### To add an invoice to an invoice group:

1. Go to Transactions > Sales > Group Invoices > List.

2. Open the invoice group to which you want to add an invoice by clicking the associated **Edit** or **View** link.
3. Click **Add to Group**.
4. From the Add Invoices to Group dialog, check the box to the left of the invoices you want to add to the group.
5. Click **Add to Group**.



**Note:** The total number of invoices must be 100 or less. If the number of invoices that you are adding plus the existing invoices exceeds 100, a message appears to indicate that you must clear the box for some of the invoices. While you are adding invoices to a group, a popup message indicates the number of invoices you can add to the group.

You can also add invoices to an invoice group from the Group Invoices page. To add invoices from this page, select the box associated with the invoices you want to add, then click Add to Group. If you attempt to add more than 100 invoices to a group from the Group Invoices page, a message appears indicating that multiple invoice groups will be created.

When you add an invoice to an invoice group, the total amount due for the invoice group is updated. The payment due date does not change.

## Adding an Invoice to an Invoice Group From the Invoice Record

From the invoice record, you can add an invoice to an existing invoice group by using the Add to Group button. The Add to Group button appears on invoices that are not part of an invoice group but are eligible for grouping.

### To add an invoice to an invoice group using the Add to Group button:

1. Go to Transactions > Sales > Create Invoices > List.
2. Open the invoice you want to add to a group.
3. Click the Add to Group button.
4. Select the group to which you want to add the invoice.



**Note:** The only invoice groups that display are those that match the same criteria as the invoice you want to group. You can only select one invoice group.

5. Click Submit to add the invoice to the invoice group.



**Note:** If there are no invoice groups that match the criteria for the invoice, you cannot create a new invoice group from the invoice record.

## Removing an Invoice From an Invoice Group

If an invoice is accidentally grouped or needs to be removed from a group, you can remove it if no payments are applied to the invoice. If a payment is partially or fully applied to an invoice that you want to remove from an invoice group, you must first remove the payment. For more information, see the help topic [Reversing or Deleting Customer Payments](#).

After removing the invoice from the invoice group, you can reapply the payment to the remaining invoices.

## To remove an invoice from an invoice group:

1. Go to Transactions > Sales > Group Invoices > List.
2. Open the invoice group from which you want to remove an invoice by clicking the associated **View** or **Edit** link.
3. Click **Remove From Group**.
4. From the Remove Invoices From Group dialog, check the box to the left of the invoice you want to remove from the invoice group.
5. Click **Remove From Group**.



**Note:** You can also remove an invoice from a group by opening the invoice from the invoice group record. Click the Edit link associated with the invoice or click the invoice number, and then click Remove From Group on the invoice record.

When you remove an invoice from an invoice group, the total amount of the invoice group is updated. The payment due date does not change.

To delete an invoice group, you must first remove all invoices from the group.

When you remove an invoice from a group, it remains available for grouping, but you cannot regroup it to the same group. The ability to regroup the invoice to the same group will be available in a future release.

If you do not want the invoice available for future invoice grouping, click Unmark for Grouping.

Refresh the invoice group record to view updated information after you remove an invoice from a group.

Currently, you cannot void an invoice group.

## Grouped Invoices Due Date

A common criteria required for grouping invoices is the invoice terms. For example you can group invoices with Net 30 terms if the other common criteria is met. When you have multiple invoices that are grouped, even though their terms are the same, they might have varying due dates. When the invoice group is created and billed, a new due date is created for all grouped invoices based on their common term and the invoice group creation date. The invoice with the latest due date determines the due date for the invoice group.

## Invoice Groups and Closed Periods

Invoices that are grouped follow the same accounting rules as invoices that are not included in an invoice group. You should not change invoices that can impact previous periods of the general ledger. Scenarios in which invoices from a closed period need editing include:

- An invoice needs to be removed from an invoice group that is in a closed period.
- An invoice from a closed period needs to be added to an invoice group.
- An invoice group includes invoices from the current period and a closed period.

For more information about accounting rules for closed periods, see the following topics:

- [Using the Period Close Checklist](#)
- [Non-Posting Transactions in Locked and Closed Periods](#)

## Accepting Payment for an Invoice Group

You can accept a full or partial payment for an invoice group.

## To accept payment for an invoice group:

1. Go to Transactions > Sales > Group Invoices > List.
2. Click the **Edit** or **View** link associated with the invoice group for which you want to accept payment.
3. On the invoice group record, click **Accept Payment**.
4. On the Payment page, click **Pay All** to pay all invoices.  
All invoices linked to the invoice group are automatically selected. If you are accepting a partial payment, you can clear the Apply box for any invoices for which you do not want to accept payment.
5. Click **Save**.

You cannot create a credit memo from an invoice group. If a credit memo exists, you can apply a credit to an invoice that is linked to the invoice group. The credit memo is visible from the Related Records subtab on the invoice group.

An invoice that is part of an invoice group cannot accept payments outside of the invoice group. All payments must go through the invoice group.

After applying a payment to an invoice, the invoice cannot be removed from the invoice group.

Any discounts associated with the invoices that are part of the group are carried over to the group. Discount information appears under the Disc. Available and Disc. Taken columns on the invoice group record.

If there is an automatically applied deposit associated with a sales order that is invoiced, that deposit carries over to the invoice group. However, you cannot apply a deposit after the invoice is grouped.

For more information about customer payments, see the help topic [Customer Payments](#).

Consolidated payments for invoice groups follows the same rules as consolidated payments. For information about consolidated payments, see the help topic [Consolidated Payments](#).

 **Note:** You cannot use the Consolidated Payments feature to generate customer statements for invoice groups.

## Generating and Printing an Invoice Group PDF

You can generate and print an invoice group PDF statement. By default, the PDF is the summary view of the invoice group record, and you can customize it.

## To print an invoice group PDF:

1. Go to Transactions > Sales > Group Invoices > List.
2. Click the **View** link for the invoice group you want to print.
3. Point your mouse to **Actions**, and then click **Print**.

 **Note:** The format of the PDF is determined by the Advanced PDF/HTML template. For more information, see the help topic [Advanced PDF/HTML Templates](#).

## Invoice Groups and Customer Statements

You can create and send customers a statement that includes invoice groups.

## To generate a customer statement that includes invoice groups:

1. Go to Transactions > Customers > Generate Statements.

**Note:** You can also go to Transaction > Customers > Individual Statements and continue with the following steps:

2. Enter a date in the **Statement Date** field.
3. Check the box associated with the customer to which you want to send a statement.
4. From the Form field, select **Standard Statement**.

**Note:** The Standard Statement PDF/HTML form includes a list of the invoices that are in the invoice group. You can customize the template to only show one line for the entire invoice group.

5. Click **Print**.

## Invoice Group Reporting and the A/R Aging Report

Invoice group information is available on the A/R aging report and a customized version of the Open Invoices report. Use the Report Builder feature to add invoice grouping fields to the report. For more information about the Report Builder feature, see the help topic [Financial Report Builder](#).

The following invoice group fields are available:

- Bill To
- Date
- Date Due
- Grouping Status
- Invoice Group: Invoice Group Number
- Invoice Group Number

Open invoices that are not in a group also appear on the report, but information only appears in the Grouping Status field (Not for Grouping).

## Billing Operations

Billing Operations provide automation tools for billing managers and account receivables professionals. You can access Billing Operations as an administrator two different ways:

- Transactions > Billing > Process Billing Operations
- Transactions > Billing > Schedule Billing Operations

To access Billing Operations using the A/R clerk role, go to Billing > Billing Operations.

You must enable Billing Operations to use these features.

### To enable billing operations:

1. Go to Setup > Company > Enable Features.
2. Select the **Transactions** subtab.
3. Under **Billing**, check the **Billing Operations** box.

4. Click **Save**.

If you are an administrator, the menu options for Process Billing Operations and Schedule Billing Operation are available from the Transactions tab. If you are using the A/R clerk role, the options are available from the Billing menu.

When billing operations run, rating runs and credit memo runs occur before invoicing billable customers.

To process Billing Operations for an Intercompany A/R account, make sure an Intercompany A/R account is selected on the Billing Operations page. An A/R account is designated as an Intercompany A/R account if the Eliminate Intercompany Transactions box is checked for that account. To set the Eliminate Intercompany Transactions box, go to Setup > Accounting > Chart of Accounts > New. For more information about Intercompany A/R accounts, see the help topic [Creating Intercompany Accounts](#).

## Viewing and Creating Bill Runs and Credit Memo Runs

You can view and create bill runs by going to Billing > Billing Operations > Process Billing Operations. After bill run schedules are created, you can review them. In addition, you can view all bills being created during the times you scheduled.

**Note:** You cannot schedule a billing operation to run after the 28th of the month.

When viewing a list of bill runs, you can view a list of all invoices and cash sales by clicking the Submission ID column.

You can also create a Credit Memo run from the Billing Operations page. By default, both the Credit Memo Run and Bill Run boxes are checked.

When processing scheduled billing operations, NetSuite uses the database maintenance time zone. In the image below, a user is located on the east coast of the United States (U.S.). The company is located on the U.S. west coast. On the Billing Operations Schedule page, after a **Next Time** is entered, an in-line message appears indicating when the billing operation will run in the database maintenance time zone. Use the **Next Bill Run As Of Date** to set the date for which you want charges billed. In the image below, the billing operation is scheduled for January 26 and the **Next Bill Run As Of Date** is January 31. When the January 26 billing operation runs, it will include future charges up to January 31. You can also enter a date that occurs before the scheduled billing operation. If you are unsure of your company time zone, you can find this information by going to Setup > Company Information > Time Zone.

The screenshot shows the 'Billing Operation Schedule' page. At the top, there are 'Save' (with a dropdown), 'Cancel', and 'Reset' buttons. Below is a 'Primary Information' section with a 'NAME' field containing 'Billing Operations Time Zone' and an 'INACTIVE' checkbox. Under 'Schedule', 'RECURRANCE FREQUENCY' is set to 'Monthly'. 'REPEAT EVERY' is set to '1 month(s)'. The 'NEXT DATE' is '1/26/2018'. The 'NEXT TIME' is '12:00 pm noon (GMT-05:00) Eastern Time (US & Canada)'. A note indicates: 'This will run on 1/26/2018 9:00 am in the time zone ((GMT-08:00) Pacific Time (US & Canada)).' The 'NEXT BILL RUN AS OF DATE' is '1/31/2018'. At the bottom right, there are 'List' and 'More' buttons.

When entering an immediate bill run (Process Billing Operations), the billing operation occurs in the user's time zone.

## To create a new bill run or credit memo run:

1. Go to Billing > Billing Operations > Process Billing Operations.
2. Under **Primary Information**, enter an optional description for the bill run schedule.
3. Under **Billing Operation Options**, indicate whether you want to create a credit memo run, a bill run, or both.
4. If you are creating an off-cycle bill run, check the **Bill Run (Off-Cycle Only)** box.

 **Note:** If you are creating an off-cycle bill run, several fields are grayed-out and uneditable.

5. If you are in a One World account, choose an option from the Subsidiary list.
6. Under Customer, select a **Customer** or **Group** from the list or check the **All Customers** box.
7. Under Billables, select the types of transactions for which you want to create bills.
8. Under Accounting, indicate if a credit card is approved.

 **Note:** Check this box if a credit card outside of NetSuite paid for the transaction, and you want to record the transaction in NetSuite.

9. Under Communication, indicate whether bills should be printed, emailed, or faxed.
10. Click **Run**.

The Next Bill Run As Of Date field indicates the date until which charges are included in the bill run. For example, you are scheduling a bill run to occur on January 15, but you want to bill charges through the end of the month. The date in the Next Bill Run As Of Date field would be 1/31/YYYY.



**Important:** If a billing operation does not process all invoices, you have two options:

- Run billing operations a second time
- Schedule a billing operation for a convenient time for your business

Reasons why not all invoices are processed during a billing operation include:

- A charge is created after billing operations begins processing charges
- A single billing operation can process up to 100,000 processing lines

When using billing operations to process billing accounts, you might reach up to 100,000 processing lines. For more information about billing accounts, see [Creating Billing Accounts](#).

## Creating Billing Groups

A billing group is a group of customers for which you want to create bills. Billing groups can be dynamic or static. For more information about groups, read the help topic [Working with Groups](#). You can change any group into a billing group by checking the Billing Group box on the group.

Dynamic group membership changes to include anyone that meets the criteria defined in a saved search. You can create a dynamic group from an existing saved search or you can create a new saved search when you create the group.

The membership of a static group remains constant. You can create static groups by selecting members from lists and with a search. You can add or remove members from a static group at any time (see the help topic [Adding and Removing Members of Groups](#)).



**Note:** You cannot create a static group with more than 10,000 members. For groups larger than 10,000, you must create a dynamic group.

### To create a dynamic billing group:

1. Go to Lists > Relationships > Billing Groups.
2. On the **Create Group** page, choose **Dynamic**.
3. Choose the kind of members you would like this group to contain.
4. Click **Continue**.
5. Enter a name for this group.
6. From the **Saved Search** field, select an existing saved search. For more information about saved searches, see the help topic [Saved Searches](#).
7. NetSuite pre-fills the **Owner** field with your name. If the owner of this group is someone else, select that person from the list.  
Only the owner can add and remove members or delete this group.
8. (Optional) If this group shares an email alias, enter the email address in the **Email** field.
9. (Optional) In the **Comments** field, enter other information about this group.
10. If you want to restrict viewing (and also editing) of this group to its members, check the **Private** box.
11. If you want to restrict access to this group to members of another group, select the group from the **Restrict To** list.



**Note:** You cannot mark restricted groups as private.

12. To permit only the owner of the group to modify the group, check the **Restrict Group Editing to Owner** box. Otherwise, anyone with access to the group can modify it.



**Note:** The **Billing Group** box is checked by default and is read-only.

13. Click **Save**.

Now, every time a record of this type is created, it is added to this group if it matches the criteria in the saved search.

### To create a static billing group

1. Go to Lists > Relationships > Billing Groups.
2. On the **Create Group** page, choose **Static**.
3. Choose the kind of members you would like this group to contain.
4. Click **Continue**.
5. Enter a name for this group.
6. NetSuite pre-fills the **Owner** field with your name. If the owner of this group is someone else, select that person from the list.  
Only the owner can add and remove members or delete this group.
7. (Optional) If this group shares an email alias, enter the email address in the **Email** field.
8. (Optional) In the **Comments** field, enter other information about this group.
9. If you want to restrict viewing (and also editing) of this group to its members, check the **Private** box.

10. If you want to restrict access to this group to members of another group, select the group from the **Restrict To** list.



**Note:** You cannot mark restricted groups as private.

11. To permit only the owner of the group to modify it, check the **Restrict Group Editing to Owner** box. Otherwise, anyone with access to the group can modify it.



**Note:** The **Billing Group** box is checked by default and is read only.

12. On the Members subtab, do one of the following:

- To add individual members, in the **Name** column, select a member from the list, and click **Add**. Repeat these steps for each member.
- To add multiple members, click **Add Multiple**, then click on each member to add them, or use the **Search** field. When you are finished, click **Done**.
- Click **Add With Search** to enter search criteria for the members you want to add.

You can use more than one of these methods to add members to the same group. For example, you can select a few members individually and then add all the members that meet search criteria.



**Note:** If you select the same member using multiple methods, this adds the member multiple times.

13. Click the **Files** subtab if you would like to attach a file to the group.
14. Click the **Notes** subtab to add any notes about this group.
15. Click **Save**.

To add or remove members from a static group, see the help topic [Adding and Removing Members of Groups](#).

To deactivate a billing group, open the billing group record, check the **Inactive** box, and then save the record.

## Creating and Managing Bill Run Schedules

After creating Billing Groups, you can create bills for a group of customers on a predefined, scheduled basis. When creating a bill run schedule, you can define parameters such as what bills get created, which customers should be selected for invoicing, and the frequency for invoice creation. After you have set a schedule, bills are created at the scheduled time. You can view all bills being created by going to Billing > Billing Operations > Process Billing Operations > Status.

### To create a bill run schedule:

1. Go to Billing > Billing Operations > Process Billing Operations
2. Under Primary Information, enter a name and description for the bill run schedule. If you do not want to make the bill run schedule active right away, check the **Inactive** box.
3. Under Schedule, setup a schedule for the bill run.
4. Under Billing Operation Operations, check the **Bill Run** box.
5. Under Billables, select the types of transactions for which you want to create bills.
6. Under Accounting, indicate whether the credit card is approved..
7. Under Communication, indicate whether bills should be printed, emailed, or faxed.
8. Click **Save**.

## Billing or Invoicing a Sales Order

Billing a sales order creates a record of debt owed to you by your customer for a sale. When you bill a sales order, your accounts receivable ledger increases by the amount of the bill.

Usually, customers are billed for a sale when the items on the order are shipped. When you ship the items, you bill the customer. The process you use to bill customers depends on if you use the Advanced Shipping feature:

- **Without Advanced Shipping**, you use a single transaction to fulfill and bill sales orders at the same time. Fulfilling a sales order automatically creates a bill based on the fulfillment. You can partially fulfill a sales order and any items fulfilled will also be billed.
- **With Advanced Shipping**, you use separate transactions to fulfill and bill sales orders. First, you fulfill the sales order to indicate the items are being shipped. Next, you create a bill for the shipped items. Even if you ship only part of an order, you can create an invoice or cash sale for the entire order.

To bill orders, fulfill part or all of a sales order, and then create an invoice or cash sale for the shipped items and rendered services. You can choose to bill a single sales order or multiple sales orders.

You cannot include a partially processed sales order when bulk billing orders. Partially processed sales orders must be billed individually.

**Note:** You cannot fulfill or bill sales orders with the status Pending Approval. If you need to make changes to an order that is pending approval, you can remove line items.

Lines from both a vendor bill and any associated bill credit appear on the Billable Items list located on the Items subtab.

To convert a sales order to an invoice or cash sale, go to Transactions > Sales > Bill Sales Orders.

If you ship only part of an order, you can later ship and create an invoice or cash sale for the rest of the order.

After you have billed a sales order, you can view the Items subtab of the order and the Invoiced column shows the quantity of each item that has been billed.

To enable Advanced Shipping, go to Setup > Company > Setup Tasks > Enable Features > Transactions > Transactions (Administrator). Check the Advanced Shipping box, and click Save.

To turn off Advanced Shipping, go to Setup > Company > Setup Tasks > Enable Features > Transactions > Transactions (Administrator). Clear the Advanced Shipping box, and click Save.

**Note:** After you enable Advanced Shipping, you cannot turn it off unless all fulfilled sales orders are billed.

You may also choose to use the Advanced Shipping feature with the Advanced Billing feature. For more information, read [Advanced Billing and Advanced Shipping](#).

## Invoicing Billable Customers

You can bulk invoice customers for billable expenses, items, and time when you use the Bill Costs to Customers feature. To invoice billable customers by using bulk billing can decrease the time you spend billing customers.

If you use the Multiple Currencies feature, you can generate invoices only in each customer's primary currency. For more information, see the help topic [Customers and Multiple Currencies](#).

The Invoice Billable Customers page includes lines from the vendor bill as well as any applied vendor credit, providing visibility into lines already billed with an included credit.



**Note:** You cannot fulfill or bill sales orders with the status Pending Approval. You also cannot close lines on a sales order with this status. If you need to make changes to an order that is pending approval, you can remove line items.

## To bulk bill customers:

1. Go to Billing > Sales > Invoice Billable Customers.
2. Enter the Invoice date to apply to all invoices you are creating.
3. From the **Account** list, select the accounts receivable account to post to.  
To set the default A/R account, go to Setup > Accounting > Accounting Preferences (Administrator). You can select **Respect Customer Preference** if you want the invoices created to use each customer's default. You can set the default A/R account for a customer on the Financial subtab of the customer record.
4. From the **Form** list, select the form you want to use for the invoices.
5. Choose to print, email or fax the invoices.  
In the fields **To Be Printed**, **To Be Emailed**, and **To Be Faxed**, set your preferences for sending the completed form. You can make a selection in each field individually.
  - Select **Yes** if you want to send the form using the method indicated.  
For example, select Yes in the To Be Faxed field to fax the form.
  - Select **No** if you do NOT want to send the form using the method indicated.  
For example, select No in the To Be Emailed field to not email the form.
  - Select **Respect Customer Preference** to send the form based on the default preference on the customer record.  
For example, if the customer's record indicates to send forms by fax, then the completed form is faxed.
 Set the preference on the Info subtab of a customer record in the Send Transactions Via field.
6. Filter the transactions that show in the list by checking the **Expenses** box, **Items** box, or **Time** box. For example, when you check the Expenses box, the list shows only transactions for customers with billable expenses.
7. Check the **Charges** box if you want to invoice billable charges.
8. Check the **Credit Card Approved** box if the credit card transaction occurred outside of NetSuite. For example, a charge at a card-swipe terminal.
9. Check the **Show Invoices** box to show a list of invoices created when you submit this page.
10. In the **Bill Date** field, filter the list of transactions by selecting a date range. Then, you can filter the list of transactions to process billable transactions for a particular time frame.  
For example, a lawyer can filter the transaction list for billable time during the previous month. Then, customers can be invoiced monthly for billable time accrued during the previous calendar month.
11. Select the invoices to create by checking the **Invoice** box next to the transaction.
12. Click **Submit**.

For each transaction you checked, NetSuite creates an invoice using your preferred invoice form. The Results page lists all invoices created in this bulk process.



**Note:** When billable time is entered using the Vendor Center against a service item with a price, the billable time shows when you invoice billable customers. The rate shown is the price on the item record.

## Closing or Voiding an Invoice

An invoice closes when it is paid in full. However, the status is Paid in Full rather than Closed.

Invoices that are in a closed period cannot be changed.

Depending on your company policy or location, deleting an invoice may not be an option.

If your company policy allows, you may be able to void an invoice or issue a credit memo and apply the credit memo to the invoice.

For more information, see the help topic [Voiding, Deleting, or Closing Transactions](#).

## Creating Installments

You can create installments for your customers who pay for items over time using installment-based payments. The Installments feature is part of the term record. When the Installment box on the term record is checked, installment-specific fields appear. Fields that are not installments-specific appear, but are not available for editing.

Before using the Installments feature, enable it by going to Setup > Company > Enable Features . Select the Accounting subtab, and check the Installments box.

### To create an installment term

1. Access the term record by going to Setup > Accounting > Accounting Lists > New > Term.
2. On the term record, check the **Installment** box.

The screenshot shows the 'Term' creation screen in Oracle NetSuite. At the top, there are 'Save' and 'Cancel' buttons. The 'TERMS' field contains 'STANDARD'. Under 'RECURRANCE FREQUENCY', 'Monthly' is selected. In the center, there are two sections: 'STANDARD' (selected) and 'DATE DRIVEN'. The 'STANDARD' section includes fields for 'DAYS TILL NET DUE', '% DISCOUNT', and 'DAYS TILL DISCOUNT EXPIRES'. The 'DATE DRIVEN' section includes fields for 'DAY OF MONTH NET DUE', 'DUE NEXT MONTH IF WITHIN DAYS', '% DISCOUNT', and 'DAY DISCOUNT EXPIRES'. At the bottom right, there are checkboxes for 'SPLIT EVENLY' and 'PAY TAX UPFRONT', both of which are checked. The bottom of the screen has 'Save' and 'Cancel' buttons again.



**Note:** After you check the Installment box, new fields appear. Any non-installment fields are unavailable.

3. In the **TERMS** field, enter the installment terms. For example, **Three Month Term**.

4. In the **Days Till Net Due** field, enter the number of days after the invoice date that the payment is due. For payments due upon receipt, enter **0**.
5. From the list, select a recurrence frequency. Options are:
  - Annually
  - Daily
  - Monthly
  - Weekly
6. in the **Recurrence Count** field, enter the number of recurring payments. For example, 3.
7. To split the payments into equal amounts, check the **Split Evenly** box. If you do not want the payments split evenly, clear the box. A list of installment lines appears that enables you to enter the amount of payment for each installment.
8. To include the total tax amount with the first installment, check the **Pay Tax Upfront** box.



**Note:** If the Pay Tax Upfront box is checked, any shipping fees are split across the installments. However, the total of any shipping taxes is included in the first installment.

9. Click **Save**.

Any fields that are not supported by installments are unavailable after the Installment box is checked. Specifically, discounts are not supported.

After saving the installment term, you can add it to a sales order. On a sales order, installment details are available for viewing only. Installments cannot be edited on a sales order. Any editing must be done on the invoice.

To add an installment term to a sales order, click the **Billing** subtab, and then select your term from the Terms list. When the sales order is fulfilled and then billed, the installment terms appear on an invoice.

To see installment information, click the **Billing** subtab on the invoice. If creating a standalone invoice, select a term from the Terms list.

The screenshot shows the Oracle NetSuite interface for managing invoices. The top navigation bar includes 'Items', 'Shipping', 'Billing' (which is the active tab), 'Accounting', 'Relationships', 'Communication', 'System Information', and 'Custom'. Below the navigation bar, there are two main sections: 'Billing Information' and 'Billing Address'. Under 'Billing Information', there is a dropdown for 'TERMS' containing '3 Month Term' and a checkbox for 'OVERRIDE INSTALLMENTS'. Under 'Billing Address', there is a dropdown for 'BILL TO SELECT' showing 'Default Address' and a list for 'BILL TO' with 'Hahn & Associates' and 'CA'. Below these sections is a 'Map' button. The bottom half of the screen is titled 'Installments' and contains a table with three rows of due dates and amounts. At the bottom of the 'Installments' section are buttons for 'Add' and 'Cancel'. At the very bottom of the page are buttons for 'Save', 'Cancel', 'Reset', and 'Actions'.

DUE DATE *	AMOUNT *
7/11/2018	2.36
8/11/2018	2.68
9/11/2018	4.47

You have the option to override the installments. To make changes, check the **Override Installments** box to make changes. If you override any installments, you must ensure that the total adds up to the total amount of the invoice.

On the invoice, the header-level due date always indicates the next installment due date. In edit mode, the header-level due date cannot be changed. However, when in edit mode, unpaid installments can be changed on the invoice. Paid installments cannot be edited. On the sales order, installments are only shown in view mode and cannot be edited.

When you accept payments, the status of the payment changes. If a customer pays more than the amount due for an installment, the overage amount is applied to the next installment. When the overage amount does not completely cover the amount due for an installment, the status for that installment is partially paid. The same is true for the aging balance. The aging balance shows a late installment as past due, not the entire balance.

When an installment is unpaid after the due date, only the late installment is marked as past due, not the entire invoice.

## Setting the Tax Allocation for a Subsidiary

If you have a OneWorld account, you can set the tax allocation method for each subsidiary by going to the Invoice Preferences page. When you set the tax allocation method for a subsidiary, all invoices with an installment term use the same method. Each subsidiary can only use one method. When you click the Installments subtab, your available subsidiaries appear in the Upfront Tax Allocation list and the Split Tax Allocation list. Select the subsidiaries that you want to use for one of the two allocation methods.

### To set the tax allocation method for a subsidiary:

1. Go to Setup > Accounting > Preferences > Invoicing Preferences
  2. Click the **Installments** subtab.
  3. From the Split Tax Allocation list or the Upfront tax allocation list, select the subsidiaries you want to associate with a tax allocation method.
- Note:** You can select more than one subsidiary for each method. Each subsidiary can only have one tax allocation method.
4. Click **Save**.

If you need to change the tax allocation method for a subsidiary, you can do so from the Invoicing Preferences page. The change is applied from that date forward.

## Printing an Invoice

When printing, invoice formats you can use are PDF or HTML. Invoice formatting depends on your set preferences:

- If you have not enabled the Download PDF Files preference at Home > Set Preferences page, a preview of your invoices pops up in the default Adobe application.
- If you have enabled the Download PDF Files preference, you must first save your invoices and then open them with the Adobe application.

To print invoices in PDF format, you need Adobe® Reader®. Visit the [Adobe Web site](#) to download the latest version at no charge.

### To print a single invoice:

1. Go to Billing > Sales > Create Invoices.

2. On the Invoice page, complete the information as appropriate.
3. Click **Save & Print**.  
NetSuite shows the printout for your invoice in an Adobe application document.
4. Click the printer button in the Adobe application frame.
5. If you want to print multiple copies, specify the number on the print message that appears.
6. Click **OK**.

You can also print invoices by going to Billing > Sales > Create Invoices > List. Then, on the Invoices page, click **Print** next to the invoice.

When printing an invoice through Advanced PDF/HTML Template, all items, time, and billable items are listed in the Items sublist. For more information, see the help topic [Advanced PDF/HTML Templates](#).

You can print invoices for more than one transaction at a time. To save an invoice in your print queue to print later, on the Communication subtab, check the **To Be Printed** box. Then, follow these steps to print invoices in bulk.



**Tip:** The page sequence of the printed document depends on how the list appears on the Print Invoices page. Before printing, you can sort the list based on your preference by clicking any of the column headers.

### To bulk print invoices from multiple transactions:

1. Go to Customers > Accounts Receivable > Print Checks and Forms.
2. On the Print Checks and Forms page, click **Invoices**.
3. On the Print Invoices page, select the invoices you want to print from the list.
  - a. (Optional) If you use the Locations feature, choose the location you want to print invoices from.
  - b. (Optional) To customize the Print Invoices page view and filter the results, click **Customize**. Use the following subtabs to show the list of fields available on the invoice record.
    - **Additional Columns** – To include a field as a column in the sublist, check the **Include** box for the field, then click **Save**.  
For example, if you want to display the status of the invoices you want to print, select the **Status** field.
    - **Additional Filters** – To include a field as an additional search filter on the page, check the **Include** box for the field, then click **Save**.  
For example, if you want to print only paid invoices, select the **Status** field. On the Print Invoices page, you can use the status filter to show only the invoices with Paid In Full status.
  - c. Check the **Print** box next to each invoice you want to print.
4. Click **Print**.

## Giving Customers Access to Invoices

Giving customers access to their invoices allows them to view past invoices. If you set up the appropriate preferences, you can also allow your customers to pay their open invoices online.

You need to complete the following steps to give customers access to their invoices. Detailed instructions follow:

1. Enable the appropriate features.
2. Setup access on customer records.

### To enable features:

1. Go to Setup > Company > Enable Features (Administrator).
2. Click the **Transactions** subtab.
3. Under Payment Processing, check the **Credit Card Payments** box if you want to allow customers to pay their invoices online.
4. Click the **Web Presence** subtab.
5. Under Access, check the **Customer Access** box.
6. Click **Save**.
7. Go to Setup > Accounting > Preferences > Accounting Preference.
8. Click the **Items/Transactions** subtab.
9. To let customers pay their invoices online, under Payment Processing, check the **Customers Can Pay Online** box.
10. Click **Save**.

## Setting Up Access on Customer Records

You can use customer records in NetSuite to give access to customers and assign roles.

You should check the **Send New Access Notification Email** box to send an email so customers can set up a NetSuite password for themselves. However, if you prefer to set their passwords yourself, use the procedure in [Manually Set a Password](#) instead.

### To set up access on customer records:

1. Go to Customers > Lists > Customers.
2. Click **Edit** next to the name of the customer you want to give access to.
3. Under Email | Phone | Address, enter your customer's email address if you haven't already done so.
4. Click the **Access** subtab.
5. In the **Role** field, select the Customer Center role you want to assign to this customer.
6. Check the **Give Access** box.
7. To notify your customer of this new access, check the **Send New Access Notification Email** box. The notification email includes the email address (used for logging in to NetSuite) and explains login procedures. It also contains a URL so that the customer can set up a NetSuite password.
8. When you have finished entering information, click **Save**.

## Manually Set a Password

You should use the **Send New Access Notification Email** feature that lets customers set up a NetSuite password for themselves. However, if you prefer to set their passwords yourself, use the following procedure instead.

### To manually set a password:

1. Go to Lists > Relationships > Customers.
2. Next to the customer to whom you want to assign a role, click **Edit**.
3. In the **Email** field, enter the customer's email address.  
The customer uses this email address to log in to NetSuite.
4. On the **Access** subtab, check the **Give Access** box.
5. In the **Role** field, select the Customer Center role you want to assign to this customer.
6. Clear the **Send New Access Notification Email** box.
7. Check the **Manually Assign or Change Password** box.
8. Enter a password for your customer.



**Note:** For details about password requirements, see the help topic [NetSuite Password Requirements](#).

9. Enter the password again for verification.
10. When you have finished entering information, click **Save**.
11. Next, tell your customer to go to your customer center login page. To find the URL, a user with an Administrator role can go to Setup > Company > Company Information. The URL is in the Customer Center Login field. Your customer can log in with the email address and the password you entered on the customer's record. Do not send the customer the password by email.

After your customers log in to NetSuite, they can view and pay their invoices.

## Show the Customer Account Balance Due on Invoices

You can inform customers of the outstanding balance due on their account by showing their total account balance on each invoice. This is a good way to keep customers aware of their balance due between statements.

The balance that shows on an invoice is the customer's total payable amount due on their account, including the amount of the invoice it is printed on.

For invoices to show the customer account balance, you need to customize your standard invoice form. Then, you can choose this custom form when creating an invoice.

### To customize an invoice to show the Balance field:

1. Go to Customization > Forms > Transaction Forms (Administrator).
2. Next to the invoice form to which you want to add the Balance field, click **Customize**.  
In the **Name** field, enter a name for this form. For example, you might call this form **Balance Field Invoice**.
3. Click the **Screen Fields** subtab.  
On the **Billing** subtab, check the box in the **Show** column next to Balance.  
If you use the Consolidated Payments feature, check the **Consolidated Balance** box to show the consolidated balance.

4. Click the **Printing Fields** subtab.
5. Click the **Body** subtab.
6. Check the box in the **Print/Email** column next to Balance.  
If you use the Consolidated Payments feature, check the **Consolidated Balance** box to show the consolidated balance.
7. If you want this to be the preferred form, check the **Form is Preferred** box in the header.
8. Click **Save**.

Now, when you create invoices using this form, the account balance due for the customer being billed shows in the Balance field. This balance shows on the invoice when it is viewed, printed, faxed or emailed.

 **Note:** Viewing a previously created invoice shows the balance due at the time the invoice was created. To update the balance on the invoice, you must edit the invoice.

## Displaying Deposit Balance on Customer Statements and Remittance Slips

You can add a Deposit Balance field to customer statements or remittance slips so that customers have an understanding of the unapplied deposit amount remaining on their account. By default, the Deposit Balance field does not show on customer statements nor on remittance slips.

Similar to the Deposit Balance field, you can also add a Net Amount Due field to customer statements and remittance slips. By exposing Net Amount Due, it is assumed that your company applies the deposit balance to any open invoices on or before the payment due date; therefore, Net Amount Due is the Statement Balance less Deposit Balance. If your intention is not to apply the deposit balance to any open invoices, for example a security deposit on hold, do not expose the Net Amount Due field. Please note that even if enabled on customer statements or remittance slips, these fields do not show on printed invoices.

The Deposit Balance and Net Amount Due fields on the statement are for the same currency as the statement. In addition, for consolidated statements, the Deposit Balance on the statement is the consolidated deposit balance for the parent and all sub-customers and the Net Amount Due is the amount due of all customers less the consolidated deposit balance.

For more information, see either [Customer Statements](#) or see [Remittance Forms](#).

### To enable the Deposit Balance or Net Amount Due field on customer statements:

1. Go to Customization > Forms > Transaction Forms.
2. Click the **Customize** link for Standard Statement.  
If you cannot see the link, contact your administrator.
3. On the **Custom Transaction Form** record, click the **Body** subtab, and then check the **Print/Email** box for Deposit Bal.
4. To enable the Net Amount Due field, check the **Print/Email** box for Net Amount Due.
5. Click **Save**.

### To enable the Deposit Balance or Net Amount Due field on the remittance slip:

1. Go to Customization > Forms > Transaction Forms.
2. Click the **Customize** link for Standard Remittance Slip.  
If you cannot see the link, contact your administrator.
3. On the **Custom Transaction Form** record, click the **Body** subtab, and then check the **Print/Email** box for Deposit Bal.
4. To enable the Net Amount Due field, check the **Print/Email** box for Net Amount Due.
5. Click **Save**.

## Progress Invoices

- Using Progress Invoicing
- Creating Invoices from Progress Sales Orders
- Creating Invoices from Fulfilled Progress Sales Orders

### Using Progress Invoicing

Many service businesses use progress billing to bill for portions of projects as they are completed.

When using progress billing, a progress sales order is created and approved. When a portion of the project has been completed, the progress sales order is fulfilled. Then, a progress invoice is created for a percentage of the total cost.

To create a progress invoice, you must first create a progress sales order. Without the corresponding sales order, a progress invoice is not recorded correctly. For more information about creating a progress sales order, read the help topic [Creating Progress Sales Orders](#).

If you use advanced shipping, there are separate processes for fulfilling progress sales orders and creating corresponding invoices. For more information about advanced shipping, read the help topic [Creating Progress Sales Orders](#).

If you do not use advanced shipping, you use a single process to fulfill and bill progress sales orders. Fulfilling a progress sales order automatically creates an invoice from the sales order.

#### To create an invoice from a progress sales order:

1. Go to Billing > Sales > Bill Sales Orders.
2. On the Bill Sales Order page, select the customer you want to create an invoice for.  
All open sales orders for that customer appear in a list on the page.
3. In the **Process** column, click **Invoice** next to the progress sales order you want to create an invoice for.
4. Under **Primary Information**, in the **Custom Form** field, select **Standard Progress Invoice**.
5. On the **Item** subtab, enter the percentage being billed in one of two ways:
  - Click the number of the line item you want to change.  
In the **Current %** field, enter the percentage of the sales order to be billed for, and click **Add**.
  - Click **Fill %**.  
Enter the percentage that all line items are to be billed for, and click **OK**.

6. Click **Save**.

When you have completed another portion of this project, repeat these steps to fulfill and invoice the sales order again.

To accept a customer payment for the invoice, go to Customers > > Accounts Receivable > Accept Customer Payments.

## Creating Invoices from Progress Sales Orders

Many service businesses use progress billing to bill for portions of projects as they are completed.

When using progress billing, a progress sales order is created and approved. When a portion of the project has been completed, the progress sales order is fulfilled, and a progress invoice is created for a percentage of the total cost.

Without advanced shipping, you use a single process to fulfill and bill progress sales orders. Fulfilling a progress sales order automatically creates an invoice from the sales order. Progress sales orders must be fulfilled individually.

If you use advanced shipping, there are separate processes for fulfilling progress sales orders and creating corresponding invoices. For more information about advanced shipping, read the help topic [Fulfilling Progress Sales Orders Using Advanced Shipping](#).

### To create an invoice from a progress sales order:

1. Go to Customers > Sales > Fulfill Orders.
2. Select the customer you want to create an invoice for.  
Open sales orders for that customer appear at the bottom of the page.
3. Click the date next to the progress sales order you want to create an invoice for.
4. Click **Bill**.
5. Under **Primary Information**, in the **Custom Form** field, select **Standard Progress Invoice**.
6. Click the number of the line item you want to change, in the **Current %** field, enter the percent of the sales order to be billed, and then click **Done**.
7. Click **Save**.

When you have completed another portion of this project, repeat the above steps to fulfill and invoice the sales order again. You can repeat these steps as many times as necessary to fulfill the entire sales order.

## Creating Invoices from Fulfilled Progress Sales Orders

Many service businesses use progress billing to bill for portions of projects as they are completed.

A progress sales order is a commitment to perform services. It has no accounting impact until services are performed and the order is turned into an invoice. With advanced shipping, your shipping and accounting departments use separate processes for fulfilling and billing sales orders.

When using progress billing, a progress sales order is created and approved. When a portion of the project has been completed, the progress sales order is fulfilled. Then a progress invoice is created for a percentage of the total cost. Progress sales orders must be billed individually.

First, fulfill part or all of a sales order, and then create an invoice for the percentage of rendered services. If you complete only part of an order, you can later fulfill and create an invoice for the rest of the order.

To fulfill a sales order, go to Customers > Sales > Fulfill Orders.

### To create an invoice from a fulfilled progress sales order:

1. Go to Customers > Sales > Bill Sales Orders.
2. Select the customer whose sales order you want to turn into an invoice.  
Open and fulfilled sales orders for that customer appear at the bottom of the page.  
If you have only partially fulfilled a sales order, you bill for only the parts of the order you have fulfilled.
3. Click the date next to the sales order you want to bill.
4. Click **Bill**.  
The sales order you have selected appears as an invoice or cash sale to save.
5. Under **Primary Information**, in the **Custom Form** field, select **Standard Progress Invoice**.
6. Click **Save**.

When you have completed another portion of the sales order, fulfill the sales order and repeat the above steps to bill the sales order again.

To accept a customer payment for an invoice, go to Customers > Accounts Receivable > Accept Customer Payments.

## Billing Costs to Customers

NetSuite enables you to bill costs back to customers for time, items, and expenses you incur while working on a project for them. For example, a locksmith can bill costs to customers in the following ways:

- **Bill Items to Customers** – Bill a customer for a safe that was ordered.
- **Bill Time to Customers** – Bill a customer for the time an employee spent installing a safe.
- **Bill Expenses to Customers** – Bill a customer for subcontractor fees incurred to install a safe in another state.

To bill costs to customers, you first need to enable the feature. An administrator can go to Setup > Company > Enable Features. On the **Transactions** subtab, check the **Bill Costs To Customers** box, and then click **Save**.

**Note:** For OneWorld accounts, you must also set up at least one tax schedule because Advanced Taxes is enabled in all OneWorld accounts.

**Note:** In accounts without OneWorld, you can bill costs to customers using items created before enabling the Advanced Taxes feature. If you enable Advanced Taxes after creating an item, you must assign a tax schedule to the item to bill a customer.

Bill your costs to customers in one of the following ways:

- **Enter the time, item or expense, and mark it as billable.**

When you enter time transactions, purchase orders, bills, checks or credit card payments for time, an item or expense, you can check the Billable box and assign a customer to be billed for the item or expense. Then, when you bill the customer, the expense appears on a Billable subtab of the invoice or cash sale.



**Important:** The Billable box has the following behavior:

- Billable field is disabled if project without customer is selected.
- Billable field is disabled if non-billable project task is selected.
- If project without billing type is selected then default value depends on accounting preference EXPENSES BILLABLE BY DEFAULT
- If project with billing type T&M or CB is selected then default value = T (it doesn't matter which value has accounting preference EXPENSES BILLABLE BY DEFAULT)
- If project with billing type FBI or FBM is selected then default value = F (it doesn't matter which value has accounting preference EXPENSES BILLABLE BY DEFAULT)



**Note:** If you enter the billable item on a purchase order, you must **bill the purchase order** before you can bill the customer for the item.

■ **Create an invoice or cash sale to bill the customer for the cost you entered.**

When you create a customer invoice or cash sale, the costs you have billed to customers show on the Billable Items, Billable Expenses and Billable Time subtabs. After you select the customer on the bill, click the appropriate subtab, and check the line items you want to bill the customer for.

Click **Mark All** or **Unmark All** to check or clear all items, expenses or time that show on the subtab.

For more information, read [Creating an Invoice or Cash Sales](#).

The items and expenses appear as normal line items on the transaction. The customer sees only a selling price that includes the markup spread over the expense items. However, you will see the markup as a separate entry when you view the invoice or cash sale in NetSuite.

## Preferences for Billing Costs

When you use the Bill Costs to Customers feature, set a preference for invoices to include memos you enter on time and expense records. To set this preference, go to Setup > Accounting > Preferences > Accounting Preferences > Time & Expenses. Check the Copy Time Memos to Invoices and Copy Expense Memos to Invoices boxes and then click Save.



**Note:** When you enable these preferences, the memos you have already entered on existing time and expense records show on invoices you create.

If you want to classify billable costs by class or department, you must enable the **Allow Per-Line Departments** and **Allow Per-Line Classes** preferences at Setup > Accounting > Preferences > Accounting Preferences > General under Classifications.

To save time, you can create invoices in bulk for billable expenses, time, and items. For more information, read [Invoicing Billable Customers](#).

## Tracking Class, Department, and Location for Billable Costs

If you track class and department on the line-item level, when you invoice a customer for billable costs, the classifications (class, department, and location) from the bill are maintained on the billable subtabs of the invoice regardless of the classifications set in the header of the invoice.

You can modify the classifications on the billable subtabs of the invoice.

 **Note:** If you group class, department, and location on invoices, the classifications on the billable subtabs are taken from the invoice header.

## Billing Items to Customers

Billing items back to customers enables you to purchase items and supplies for an order or project, and then bill the cost to the customer. For example, a locksmith can bill a customer for a special safe they installed.

To bill items to customers, you first need to enable the feature. An administrator can go to Setup > Company > Enable Features > Transactions, check the Bill Costs To Customers box, and click Save.

When the feature is enabled, you can bill an item to a customer. You can bill items to customers on purchase orders or other purchase transactions.

Income from billable items posts to the income account setup in your Chart of Accounts.

 **Note:** For OneWorld accounts, you must also set up at least one tax schedule because Advanced Taxes is enabled in all OneWorld accounts.

 **Warning:** In accounts without OneWorld, you can bill costs to customers using items created before enabling the Advanced Taxes feature. If you enable Advanced Taxes after creating an item, you must assign a tax schedule to the item to bill a customer.

### To bill items to customers using purchase orders:

1. Enter a purchase order for an item you want to bill to a customer.

When you enter a purchase order for a billable item, select the item and identify the customer or project, and then check the **Billable** box.

 **Note:** You must bill the purchase order before you can bill the customer for the billable item.

To enter a purchase order for a billable item, go to Transactions > Purchases/Vendors > Enter Purchase Orders.

For more information about purchasing billable items, read the help topic [Entering a Purchase Order](#).

2. Bill the purchase order that includes the item you want to bill to a customer.

The bill for the billable item must identify the customer or project and have the Billable box checked.

To bill a purchase order, go to Transactions > Purchases/Vendors > Bill Purchase Orders.

For more information, read the help topic [Billing a Purchase Order With Advanced Receiving](#).

After the purchase order is billed, this item can show on the Billable Items subtab of the customer bill.

3. Enter an invoice or cash sale to bill the customer for the billable item.

When you create an invoice, apply the appropriate billable item on the **Billable Items** subtab. Click **Mark All** or **Unmark All** to check or clear all items that show on the subtab.

- To create an invoice, go to Billing > Sales > Create Invoices.
- For more information about creating an invoice, read [Creating an Invoice](#).
- To create a cash sale, go to Billing > Sales > Enter Cash Sales.. For more information, see the help topic [Cash Sales](#).

For more information about creating a cash sale, read the help topic [Entering a Cash Sale](#).

4. If you track class, department, and location at the line-item level, you can change these settings for this billable item.

When you have created the invoice or cash sale, the customer is billed for the item.

### To bill items to customers using checks:

1. Enter a check to pay a vendor for the item you want to bill to a customer. For more information, read the help topic [Adding Items to a Check](#).
2. On the **Items** subtab of the check, select the item you want to bill to the customer.
3. Identify the customer or project and check the **Billable** box.
4. Complete additional fields on the check as necessary.
5. Click **Save**.

Now, this item can show on the Billable Items subtab of the customer bill.

### To bill items to customers using credit card transactions:

1. Enter a credit card transaction to pay a vendor for the item you want to bill to a customer. For more information, read the help topic [Entering Company Credit Card Charges](#).
2. On the **Items** subtab of the credit card transaction, select the item you want to bill to the customer.
3. Identify the customer or project and check the **Billable** box.
4. Complete additional fields on the credit card transaction as necessary.
5. Click **Save**.

Now, this item can show on the Billable Items subtab of the customer bill.

## Items Billable By Default

You can enable the Items Billable by Default preference. This preference lets you check the Billable box by default when you are purchasing items. After you select a customer on an item line of a purchase transaction, the Billable box is automatically checked.

The Items Billable by Default preference is enabled at Setup > Accounting > Preferences > Accounting Preferences > Time & Expenses. This preference is dependent on the Bill Costs to Customers feature.

When billing items back to customers, you can choose to include the date of service on all custom invoices except finance charge invoices. This is the date the bill for the items was entered.

## Billing Time to Customers

Billing time back to customers enables you to enter time spent on a project for customers, and then bill the cost to them. For example, a locksmith can bill a customer for the time an employee spent installing a safe.

### To bill time to customers:

1. Enable the Bill Costs to Customers and Time Tracking features:
  - a. Go to Setup > Company > Enable Features.
  - b. On the Transactions subtab, check the **Bill Costs To Customers** box.
  - c. On the **Employees** subtab, check the **Time Tracking** box, and then click **Save**.

**Note:** For OneWorld accounts, you must also set up at least one tax schedule because Advanced Taxes is enabled in all OneWorld accounts.

**Note:** In accounts without OneWorld, you can bill costs to customers using items created before enabling the Advanced Taxes feature. If you enable Advanced Taxes after creating an item, you must assign a tax schedule to the item to bill a customer.

2. (Optional) Set preferences to determine ways you track and process the hours that employees work. For more information, see the help topic [Setting Up Time Tracking Preferences](#).
3. Enter a time tracking transaction for time you want to bill.
  - a. Go to Transactions > Employees > Track Time.
  - b. When you enter a time tracking transaction for billable time, select the customer or project and service item, and ensure that the **Billable** box is checked.

Then, when you create an invoice, this time appears on the **Billable Time** subtab.

For more information about entering a time transaction, read the help topic [Entering a Time Transaction](#).
4. Create an invoice or cash sale that applies the billable time.
  - a. Go to Billing > Sales > Create Invoices.
  - b. When you create an invoice, apply the appropriate billable time on the **Billable Time** subtab. Click **Mark All** or **Unmark All** to check or clear all time that shows on the subtab.

**Note:** To generate deferred revenue for billable time, you must manually select revenue recognition schedules on the **Billable Time** subtab.

- For more information about creating an invoice, read [Creating an Invoice](#).
- c. To create a cash sale, go to Billing > Sales > Enter Cash Sales.. For more information, see the help topic [Cash Sales](#).
  5. If you track class, department, and location at the line-item level, you can change these settings for this billable time.

When you have created the invoice or cash sale, NetSuite bills the customer for the time.

When you bill time back to customers, you can also use billing classes. Billing classes set billable time rates for service items based on employees. For example, a law office can use billing classes to charge specific hourly rates for time billed by partners in the firm.

For more information about billing classes, read [Using Billing Classes](#).

When billing time to customers, you can choose to include the date of service on all custom invoices except finance charge invoices. If you combine time items on invoices, the date of service for all the time items you are billing is grouped under the last date the employee worked.

## Billing Expenses to Customers

Billing expenses back to customers enables you to bill customers for the costs you incur while working on a project for them. For example, a locksmith can bill a customer for subcontractor fees incurred to install a safe for a customer in another state.

To bill expenses back to customers, you first need to enable the feature. An administrator can go to Setup > Company > Enable Features. On the Transactions subtab, check the Bill Costs To Customers box, and then click Save.

When the feature is enabled, you can bill an expense to a customer. You can bill expenses to customers on purchase orders or other purchase transactions.

**Note:** For OneWorld accounts, you must also set up at least one tax schedule because Advanced Taxes is enabled in all OneWorld accounts.

**Note:** In accounts without OneWorld, you can bill costs to customers using items created before enabling the Advanced Taxes feature. If you enable Advanced Taxes after creating an item, you must assign a tax schedule to the item to bill a customer.

### To bill expenses to customers using purchase orders:

1. Enter a purchase order for an expense you want to bill to a customer.

When you enter a purchase order for a billable expense, select the expense and identify the customer or project, and then check the **Billable** box.

**Note:** You must bill the purchase order before you can bill the customer for the billable expense.

To enter a purchase order for a billable expense, go to Transactions > Purchases/Vendors > Enter Purchase Orders.

For more information about purchasing billable expenses, read the help topic [Entering a Purchase Order](#).

2. Bill the purchase order that includes the expense you want to bill to a customer.

The bill for the billable expense must identify the customer or project and have the **Billable** box checked.

To bill a purchase order, go to Transactions > Purchases/Vendors > Bill Purchase Orders.

For more information, read the help topic [Billing a Purchase Order With Advanced Receiving](#).

After the purchase order is billed, this expense can show on the Billable Expenses subtab of the customer bill.

3. Enter an invoice or cash sale to bill the customer for the billable expense .

When you create an invoice, apply the appropriate billable expense on the **Billable Expenses** subtab. Click **Mark All** or **Unmark All** to check or clear all expenses that show on the subtab.

- To create an invoice, go to Billing > Sales > Create Invoices.

For more information about creating an invoice, read [Creating an Invoice](#).

- To create a cash sale, go to Billing > Sales > Enter Cash Sales.. For more information, see the help topic [Cash Sales](#).  
For more information about creating a cash sale, read the help topic [Entering a Cash Sale](#).
- 4. If you track class, department, and location at the line-item level, you can change these settings for this billable item.

When you have created the invoice or cash sale, the customer is billed for the expense.

### To bill expenses to customers using checks:

1. Enter a check to pay a vendor for the item you want to bill to a customer. For more information, read the help topic [Adding Expenses to a Check](#).
2. On the **Expenses** subtab of the check, select the expense you want to bill to the customer.
3. Identify the customer or project and check the **Billable** box.
4. Complete additional fields on the check as necessary.
5. Click **Save**.

Now, this expense can show on the **Billable Items** subtab of the customer bill.

### To bill expenses to customers using credit card transactions:

1. Enter a credit card transaction to pay a vendor for the expense you want to bill to a customer. For more information, read the help topic [Entering Company Credit Card Charges](#).
2. On the **Expenses** subtab of the credit card transaction, select the expense you want to bill to the customer.
3. Identify the customer or project and check the **Billable** box.
4. Complete additional fields on the credit card transaction as necessary.
5. Click **Save**.

Now, this expense can show on the Billable Expenses subtab of the customer bill.

## Expenses Billable By Default

You can also enable the **Expenses Billable by Default** preference to check the Billable box by default when you are entering purchase orders. Then, when you select a customer on an item line of a purchase transaction, the Billable box is automatically checked.

The **Expenses Billable by Default** preference is enabled at Setup > Accounting > Preferences > Accounting Preferences > Time & Expenses subtab. This preference is dependent on the Bill Costs to Customers feature.

## Copy Expense Memos to Invoices

You can also set a preference for invoices to include memos entered on billable expenses. To set this preference, go to Setup > Accounting > Preferences > Accounting Preferences > Time & Expenses. Check the Copy Expense Memos to Invoices box and then click Save.



**Note:** When you enable this preferences, the memos you have already entered on existing expense records show on invoices you create.

When billing expenses back to customers, you can choose to include the date of service on all custom invoices except finance charge invoices. This is the date the bill for the expenses was entered.

# Customer Statements

A statement reviews a customer's account on a regular basis, usually monthly. Send statements to remind your customers to pay their invoices.

A statement lists your customer's transactions, including invoices, finance charges assessed, credit memos issued and payments received. It also contains an aging summary at the bottom that shows how long each amount has been due.

If you use the Multiple Currencies feature, statements include separate pages for each transaction currency for each customer. If you include balance information at the bottom of your invoices (mini statements), the balance amounts shown are in the same currency as the invoice currency. For more information, see the help topic [Customers and Multiple Currencies](#).

Click a link below for more information:

- [Generating Statements](#) - includes instructions on how to print, email or fax customer statements
- [Printing a Statement](#) – includes instructions for printing individual statements or bulk statements
- [Printing Mini Statements](#) – includes instructions for customizing invoices to print mini-statements and setting your PDF form to print mini-statements

## Generating Statements

You send statements to remind customers to pay their invoices. Most companies send statements one time a month.

You can choose to print, email or fax statements, in HTML or PDF format.

### To generate statements:

1. Go to Customers > Accounts Receivable > Generate Statements.
2. In the **Statement Date** field, accept or enter the date you want to appear on your statements.



**Note:** The system also uses this date to calculate aging.

3. (Optional) Enter a start date for your statements.

This is the date of the oldest transaction you want to appear on the statement.

If you do not enter a start date, the Balance Forward does not appear and all transactions in the customer's history appear on the statement.

4. (Optional) To filter the customer list, from the **Restrict By Group** list, select a customer group.



**Tip:** To view or edit any Customer Group record in the list, click the icon to open the selected record in a separate tab.

5. (Optional) Check the **Include Zero Balances** box to generate statements for customer accounts that meet these requirements:
  - have a zero balance as of the statement date
  - have accounts receivable transaction activity between the statement date and the start date
6. (Optional) If you want to include only open transactions, check the **Show Only Open Transactions** box.

If you choose to show only open transactions and do not include a start date, the statement includes all open transactions in your customer's history.

If you only show open transactions, the Charge and Payment columns on the statement will show the total transaction amount and will not reflect any payments made toward the invoice. The Balance column shows the remaining balance on each transaction.

The Statements in Queue field shows the number of statements that enter the print queue when you submit this page.



**Note:** The Show Only Open Transactions option is most useful for statements printed as of your current today date. If you select Show Only Open Transactions and are using a date other than today's date, you may have balance discrepancies.

7. (Optional) If you want the printed statements to show the consolidated balance information for customer-subcustomer hierarchies, check the **Consolidated Statement** box. Clear this box if you want to show the individual balance for each customer.

For more information, see the help topic [Consolidated Payments](#).

8. (Optional) Select the form you want to use to generate statements.

You can select the standard form, select a custom form, or let the system decide. For information about creating your own custom forms in NetSuite, see the help topic [Custom Forms](#).

9. Choose customers to receive statements. You can click the box at the top to select all customers in the list. Click the box again to clear the box for all customers.

Click **Customize** to customize the list to filter the customers listed or to add or remove columns.

10. Generate the statements:

- Click **Print** to print the statements.
- From the **Actions** list, select **Email** to email the statements.

To email the statements, each customer must have an email address entered in the Email field of the Email | Phone | Address section on their Customer record.

- From the **Actions** list, click **Fax** to fax the statements.



**Note:** To fax statements, an administrator must first set up fax service at Setup > Company > Setup Tasks > Company Information > Fax. Each customer must also have a fax number entered on their Customer record.



**Important:** When processing transactions, you must submit one page at a time. If you do not submit each page individually, information is not saved and can be lost when you switch between pages. To process multiple pages of information, always submit each page individually.

## Customize to Filter the Customer List

When you generate statements, you can filter the customer list by individual fields. This enables you to generate statements only for specific customers.

### To filter the customer list:

1. Go to Customers > Accounts Receivable > Generate Statements.
2. On the Print Statements page, click **Customize**.
3. To see additional columns in the customer list, click the **Additional Columns** subtab, then check the **Include** box beside the desired fields.
4. To see additional filters, click the **Additional Filters** subtab, then check the **Include** box beside the desired fields.

5. Click **Save**.

The additional columns you selected appear to the right of the standard (default) columns.

The additional filters you selected appear in the Filters area of the Print Statements page. When no filters are selected, this area remains hidden.

**Print Statements**

Print | Cancel | Actions ▾

STATEMENT DATE \*  
31.3.2021

START DATE  
28.2.2021

RESTRICT BY GROUP  
customer group 3

INCLUDE ZERO BALANCES

**FILTERS**

CATEGORY - All -	CONTACT
---------------------	---------

Customize

Now you can filter the customer list as needed.

You can also customize invoices to show a mini-statement at the bottom of the form. For more information, read [Printing Mini Statements](#).

## Printing a Statement

You can print a customer statement to send to a customer.

Your administrator can customize the layout of your printed forms using advanced PDF/HTML templates. For more information, see the help topic [Advanced PDF/HTML Templates](#).

Click a link below for more information about printing a statement:

- [Bulk Printing Statements](#)
- [Printing Individual Statements](#)

## Bulk Printing Statements

Follow this procedure to print many statements at one time.

### To bulk print statements:

1. Go to one of the following:
  - Customers > Accounts Receivable > Print Checks and Forms. On the Printing page, click **Statements**.
  - Customers > Accounts Receivable > Generate Statements.
2. In the **Statement Date** field, accept or enter the date you want to appear on your statements.



**Note:** The system also uses this date to calculate aging.

3. (Optional) Enter a start date for your statements.  
This is the date of the oldest transaction you want to appear on the statement.  
If you do not enter a start date, the Balance Forward does not appear and all transactions in the customer's history appear on the statement.
4. (Optional) If you want to generate statements for customers with no outstanding balances, check the **Include Zero Balances** box.
5. (Optional) If you want to include only open transactions, check the **Show Only Open Transactions** box.  
If you choose to show only open transactions and do not include a start date, the statement includes all open transactions in your customer's history.



**Note:** The Show Only Open Transaction option is most useful for statements printed as of your current today date. If you select Show Only Open Transactions and are using a date other than today's date, you may have balance discrepancies.

6. If you use the Consolidated Payments feature, check the **Consolidated Statements** box to print out statements showing the overall balance for the customer-subcustomer hierarchies customers are a part of. Clear this box to print a statement showing only the balance for this customer.  
For more information, see the help topic [Consolidated Payments](#).
7. Choose customers to receive statements. You can click the box at the top to select all customers in the list. Click the box again to clear the box for all customers.  
Click **Customize** to customize the list to filter the customers listed or to add or remove columns.
8. When you are finished, click **Print**.

## Printing Individual Statements

Follow this procedure to print a single statement.

### To print an individual statement:

1. Go to Customers > Accounts Receivable > Individual Statement.
2. Select the customer you want to generate and print a statement for.
3. In the **Statement Date** field, accept or enter the date you want to appear on the statement.
4. In the **Start Date** field, set the date of the earliest transactions you want to show on this statement. Leave this field blank if you want to show all transactions for this customer.
5. If you use the Consolidated Payments feature, check the **Consolidated Statement** box to print out a statement showing the overall balance for the customer-subcustomer hierarchy this customer is a part of. Clear this box to print a statement showing only the balance for this customer.  
For more information, see the help topic [Consolidated Payments](#).
6. Generate the statement:
  - Click **Print** to print the statement.
  - Click **Email** to email the statement.
 To email the statement, the customer must already have an email address entered in the customer's record.

- Click **Fax** to fax the statement.



**Note:** Each customer must also have a fax number entered on the customer's record.

## Printing Mini Statements

You can print a mini-statement at the bottom of invoices that show the customer's current balance and aging totals. This is the same information that shows at the bottom of regular customer statements.



**Note:** If you use the Consolidated Payments feature, mini-statements show the consolidated balance and aging amounts for individual customers instead of the consolidated balance. For more information, see the help topic [Consolidated Payments](#).

To print the mini-statement in HTML or PDF layout, you must first customize your invoice transaction form. If you choose to print in PDF Layout, you must also set your PDF form layout to do so.

### To customize invoices to print mini-statements:

1. Go to Customization > Forms > Transaction Forms (Administrator).
2. Click **Customize** next to the form you want to add the mini-statement to.
3. On the new Custom Transaction Form page, in the **Name** field, enter a name for the new custom form.
4. Check the **Form is Preferred** box.
5. Choose **Basic** to set the printing type.
6. Click the **Printing Fields** subtab.
7. On the Printing Fields subtab, click the **Footer** subtab, and check the **Print/Email** box for Aging:Current.
8. Click **Save**.

Now, invoices can print the mini-statement in HTML format. If you do not print statements in PDF format, you do not need to take additional steps.

To print mini-statements on invoices using PDF format, you must use a PDF transaction form layout that includes statement aging fields. You can select a standard layout or customize a standard layout.

### To set your PDF form to print mini-statements:

1. Go to Customization > Forms > Transaction Form PDF Layouts (administrator).
2. Select a PDF transaction form layout that includes statement aging fields:
  - To select a standard layout, check the **Preferred** box next to the layout you choose. There are four layouts that include statement aging fields:
    - Standard Aging Invoice Layout
    - Classic Aging Invoice Layout
    - Standard Aging/Disclaimer Invoice Layout
    - Classic Aging/Disclaimer Invoice Layout
  - To customize a layout, click **Customize** next to one of the four layouts that include statement aging fields. Then, enter your customization choices for the layout and check the **Layout is Preferred** box.

3. Click **Save** or **Submit**.

# SuiteBilling Overview

SuiteBilling is a collection of features you can use to create subscriptions for your customers. You can create subscriptions to sell services or non-inventory items over a period of time. SuiteBilling is not for the sale of physical inventory items.

You use SuiteBilling to:

- Create subscription plans, price books, subscriptions, and price plans
- Set up usage and rating, including tier minimums and maximums, included usage, and included an usage multiplier
- Renew an existing subscription
- Modify an existing or renewal subscription by using a change order
- Access billing accounts, charge records, and billing operations
- Integrate with [Revenue Recognition for SuiteBilling](#)

## SuiteBilling Terminology

The following terms are used when referring to SuiteBilling:

- **Billing Account:** Shows all billing-related information for a customer or subcustomer, including the billing schedule, bill-to address, ship-to address, and currency.
- **Billing Operation:** Automates the process of rating and invoicing charges.
- **Change Order:** Changes the status of a subscription or subscription line items. For example, a change order might change a subscription from **Pending Activation** to **Active** status.
- **Charge-Based Billing:** Create charges to bill customers by using the Charge-Based Billing feature. Examples include project milestones, travel expenses, and time entered.
- **Price Plan:** Sets the price for items. You enter amounts in price plans. You also select volume or tiered pricing. Price plans are defined in price books.
- **Price Book:** Defines the pricing for a collection of items or services that are sold as part of a subscription plan. A subscription plan can have multiple price books. Price plans are defined in a price book.
- **Ratable Event:** Certain tasks you perform on a subscription line item that affect NetSuite charge calculations. Examples of actions that cause ratable events include:
  - Activating, suspending, terminating, renewing and reactivating a line item
  - Modifying quantities, pricing, and rates on a line item
  - Saving a change order
- **Rating:** The process of creating charges for subscription line items. NetSuite calculates charges for a subscription using the information on its line items, such as the quantities, and pricing shown in the price plan and price book, the usage and effective date of the subscription, and the status of the subscription and its line items. You use a change order to set and change the status of subscription line items. For details, see [Managing Rating Runs](#).
- **Subscription:** Includes the items being sold and the price and quantity for each item. A subscription is specific to one customer.
- **Subscription Plan:** Includes all of the items that can be sold as part of a subscription. The subscription plan includes price book information and renewal information. A subscription plan is a template or possible bundle a company offers to its customers.

- **Usage:** Defines the charge for a subscription line item that can be consumed, such as money, time, software licenses, or internet data. For example, if you sell mobile phone service plans, your subscription may have a line item for a monthly fee. The monthly fee includes a specific amount of minutes or data in your customer's plan. When this included amount is exceeded, additional charges may apply. In NetSuite, you account for the usage amount included in a subscription when creating the subscription plan.

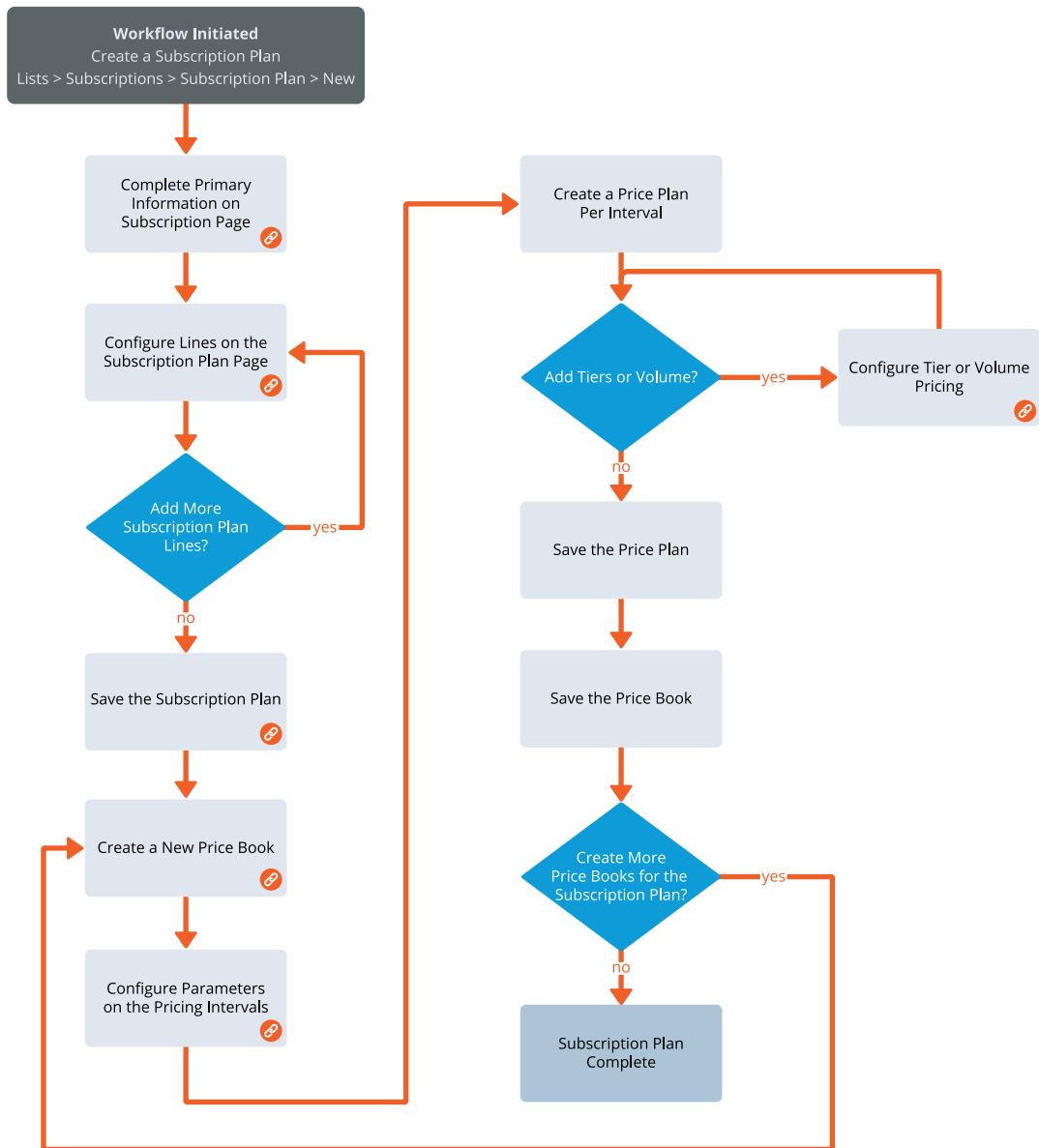
## SuiteBilling Process Diagrams

This section includes high-level overview diagrams of some of the most common SuiteBilling tasks as follows.

- [Subscription Plan Creation](#)
- [New Stand-Alone Subscription Creation](#)
- [Subscription Creation from a Sales Order](#)
- [Renew and Extend a Subscription from a New Subscription Record](#)
- [Renew a Subscription with a New Subscription Record](#)

### Subscription Plan Creation

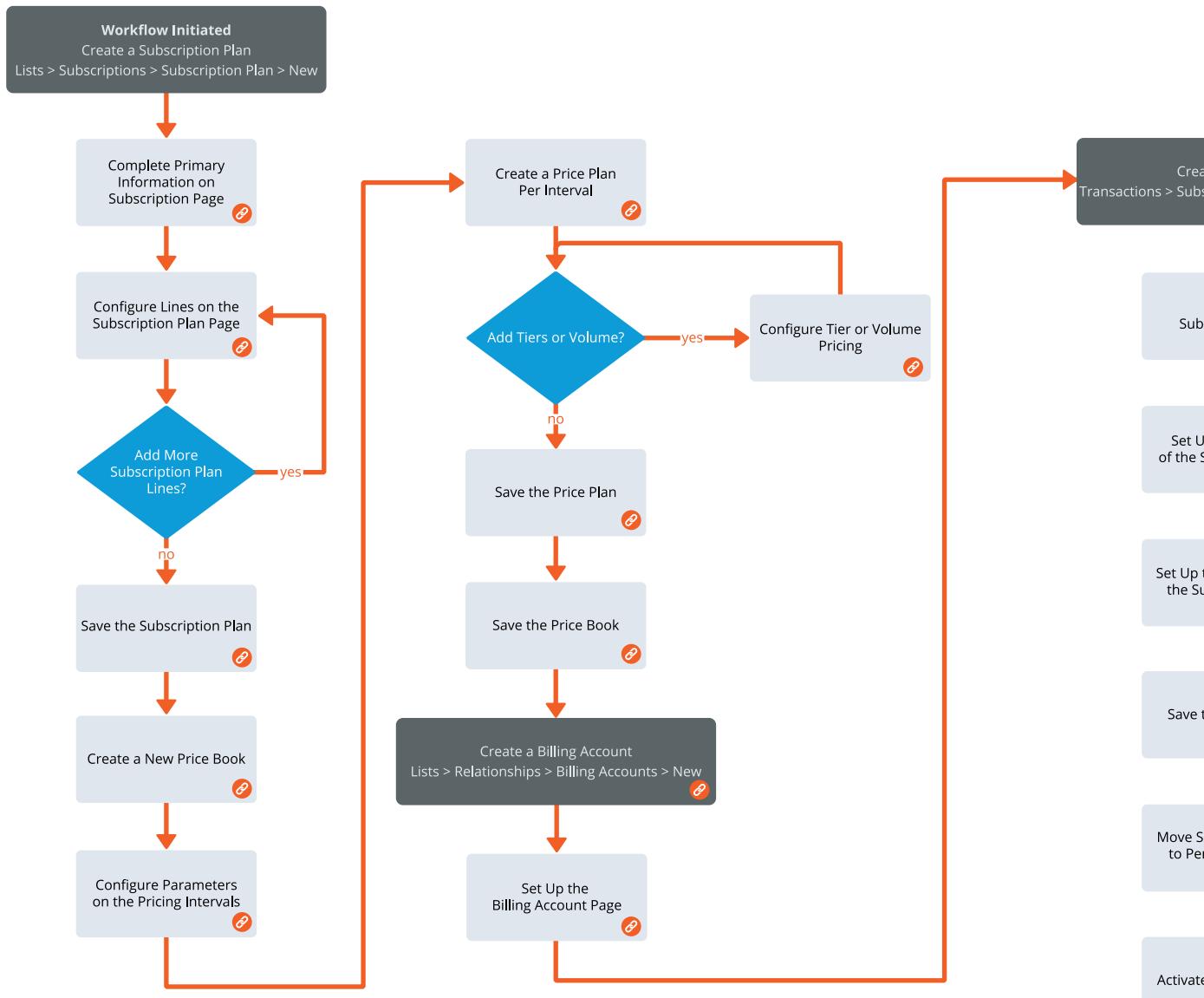
The following diagram shows how to create a new subscription plan. Click a link icon to open the instructions for a process in the diagram.



## New Stand-Alone Subscription Creation

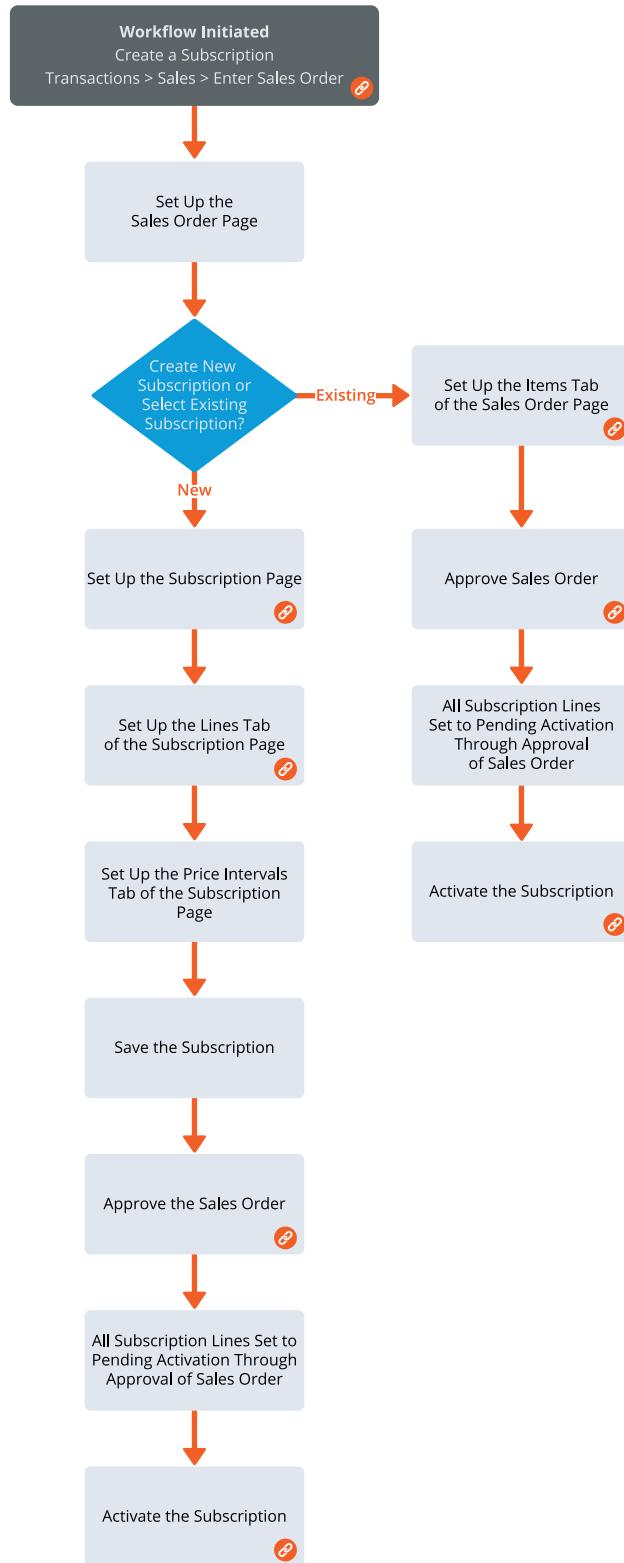
As shown in the following diagram, before you can create a subscription, you must create and set up its components. Subscription components include items, a price plan, a price book, a subscription plan, and a billing account. The following diagram shows how to create a new subscription. Click a link icon to open the instructions for a process in the diagram.

You can create a new, stand-alone subscription without using an existing transaction or subscription. For the instructions about creating a subscription without using an existing transaction or subscription, see [Creating a New Stand-Alone Subscription](#).



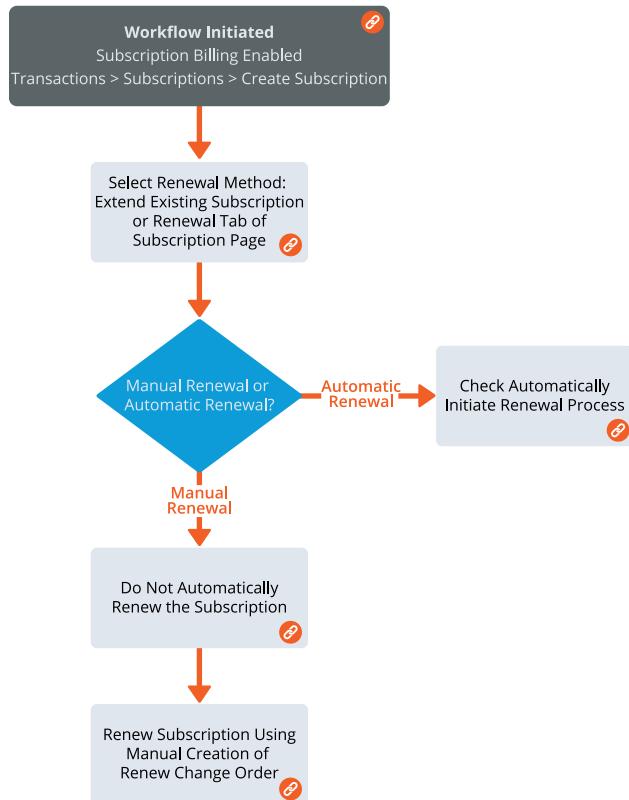
## Subscription Creation from a Sales Order

The following diagram shows how to create a subscription when using the Sales role. Use this workflow to create a new subscription from a new subscription record, a new subscription plan, and a new price books from a transaction. Click a link icon to open the instructions for a process in the diagram.



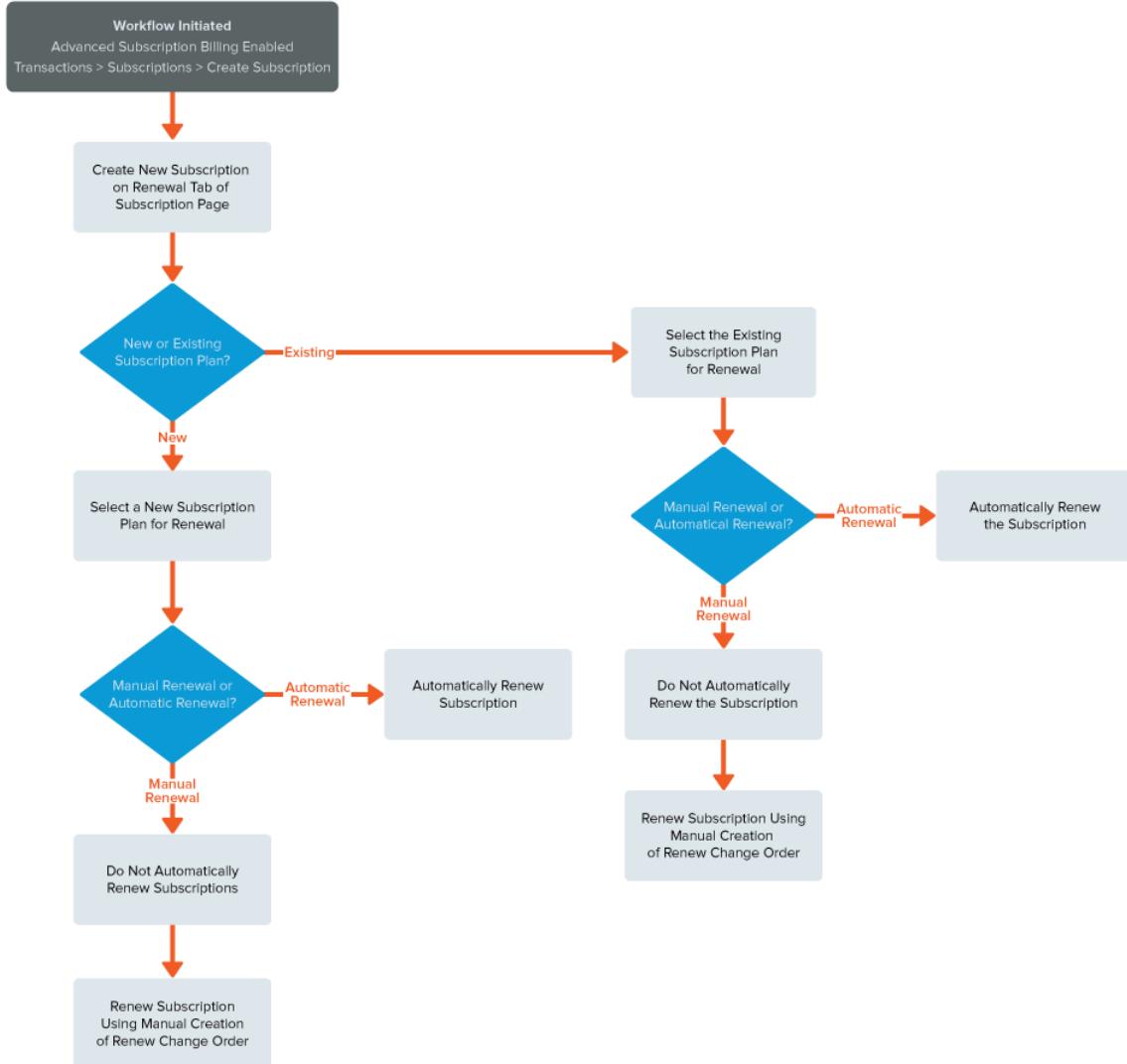
## Renew and Extend a Subscription from a New Subscription Record

The following diagram shows how to renew and extend a subscription by extending the date of an existing subscription. Click a link icon to open the instructions for a process in the diagram.



## Renew a Subscription with a New Subscription Record

The following diagram shows how to renew a subscription by creating a new subscription record, with either a new or existing subscription plan. Use this workflow to renew a subscription with a new subscription record. For the instructions for this process, see [Renewing a Subscription](#).



## SuiteBilling Roles and Permissions

The following table lists the SuiteBilling permissions and permission levels for each standard role. The Center Type column shows the center generally associated with the role. Navigation from Center lists the menu navigation for each role to the Subscriptions menu in that center. When you use the classic interface, you access subscription-related records from the Transactions menu, Subscriptions group.

Role	Center Type	Navigation from Center	Subscription Plan	Subscriptions	Price Book	Price Plan	Subscription Change Order	Usage
A/P Clerk	Accounting Center	Financial > Subscriptions	Full	None	Full	Full	None	Full
A/R Clerk	Accounting Center	Billing > Subscriptions	Full	Edit	Full	Full	Edit	Full
Accountant	Accounting Center	Billing > Subscriptions	Full	Edit	Full	Full	Edit	Full
Accountant (Reviewer)	Accounting Center	Billing > Subscriptions	View	View	View	View	View	View

Role	Center Type	Navigation from Center	Subscription Plan	Subscriptions	Price Book	Price Plan	Subscription Change Order	Usage
Advanced Partner	Advanced Partner Center	Client > Other Transactions	Full	Full	Full	Full	Full	Full
Bookkeeper	Accounting Center	Billing > Subscriptions	Full	Edit	Full	Full	Edit	Full
CEO	Executive Center	Sales > Marketing > Lists	Full	Full	Full	Full	Full	Full
CEO (Hands Off)	Executive Center	Sales > Marketing > Lists	View	View	View	View	View	View
CFO	Accounting Center	Billing > Subscriptions	Full	Full	Full	Full	Full	Full
Sales Administrator	Sales Center	Customers > Subscriptions	View	Full	Full	Full	Full	View
Sales Manager	Sales Center	Customers > Subscriptions	View	Full	Full	Full	Full	View
Sales Person	Sales Center	Customers > Subscriptions	View	Full	Full	Full	Full	View
Sales Vice President	Sales Center	Customers > Subscriptions	View	View	View	View	View	View
Support Administrator	Support Center	Cases > Other Lists	View	Full	View	View	Full	View
Support Manager	Support Center	Cases > Other Lists	View	Edit	View	View	Edit	View
Support Person	Support Center	Cases > Other Lists	View	Edit	View	View	Edit	View
System Administrator	System Administration Center	Lists > Relationships	Full	Full	Full	Full	Full	Full
Warehouse Manager	Shipping Center	Sales/Marketing > Lists	View	None	View	View	None	None
Consultant	Project Center	use global search	None	View	View	View	View	None
Marketing Administrator	Marketing Center	Customers > Subscriptions	View	Full	View	View	Full	None
Marketing Assistant	Marketing Center	Customers > Subscriptions	View	Full	View	View	Full	None
Marketing Manager	Marketing Center	Customers > Subscriptions	View	Full	View	View	Full	None
PM Manager	Support Center	Cases > Other Lists	None	Full	View	View	Full	None
Partner Center	Partner Center	Client > Other Transactions	None	Edit	View	View	Edit	None
Product Manager	Support Center	Cases > Other Lists	None	Full	View	View	Full	None
Resource Manager	Support Center	Cases > Other Lists	View	Full	View	View	Full	View
Store Manager	E-Commerce Management Center	Sales > Other Transactions	View	Edit	View	View	Edit	View
Customer Center	Customer Center	Customer Center Home Page	None	View	None	View	View	None



**Note:** The Navigation From Center column shows one navigation path, but more navigation options may be available. For example, the A/P clerk role can access subscriptions from both Financial > Subscriptions and from Customers > Subscriptions.

The standard roles available in your account may differ, depending upon the modules you purchase and the features you enable. You can see the roles available in your account at Setup > Users/Roles > Manage Roles. To review a role's details, click its link on the Manage Roles page.

For more information on roles and permissions, see the [Standard Roles Permissions Table](#).

## Default Permissions for Subscription Records

The default permission level is View. There are optional permission levels for Create and Edit. Roles with Full access permission can do the following actions:

- View
- Edit
- Create
- Delete

All roles can now view or edit the following records:

- Price book
- Price plan
- Subscription change order

New permissions in NetSuite do not change permissions to customized roles with subscriptions and subscription plan permissions.

The System Administrator role can also give users full access to the subscription record. Only the System Administrator role can grant access to and set up custom records and templates for SuiteBilling subscription processes. For instructions on how to grant permissions to subscription processes, see the help topic [Setting Permissions](#).

## SuiteBilling Change Orders

To edit an active subscription or its components, use a change order. There are specific change orders for each subscription record type.

Change orders change a subscription's or line item's status. For example, the activation change order changes the subscription from the **Pending Activation** status to the **Active** status. You also can use a change order to suspend, renew, or terminate a subscription, or to change prices, quantity, or discount.

The following rules apply to change orders:

- Change orders take effect on the date set in the Effective Date field.
- The change occurs at the beginning of the day. The one exception is the terminate change order, which takes effect at the end of the day.
- You can create change orders in any order, as long as it does not invalidate a change order that has a later effective date. For example, you cannot create a Reactivation change order for a subscription or line item that is already in the **Active** state. You cannot create a Modify Pricing change order on a subscription line item in the **Suspend** state. Likewise, you cannot terminate a subscription or line item that has a Suspend or Modify change order with a later effective date.
- You can have only one change order of any given type on the same date for the same subscription line item. For example, you cannot have more than one activation change order for the same subscription line item and date.

The following table includes change order types and their associated descriptions.

Change Order	Description
Activation	An activation change order begins a subscription. When you create a subscription, the initial status is <b>Draft</b> or <b>Not Included</b> . The <b>Not Included</b> status occurs when you have optional items in the subscription plan that are not included in the subscription. After you set any number of line

Change Order	Description
	<p>items to <b>Pending Activation</b>, submitting the activation change order activates the subscription or subscription line items. You can set the activation date to a past or future date.</p> <p>When you activate dependent line items, they must be activated at the same time. For example, a cell phone subscription contains an item for data usage. The amount of data included in the subscription is multiplied by the number of phones in the subscription. The included quantity multiplier is the item for the number of phones. In this case, both line items must be activated at the same time.</p> <p>NetSuite generates charges based on the activation date. You can delete activation change orders. If you delete an activation change order, all charges associated with the change order are deleted, and the subscription reverts to the <b>Pending Activation</b> status. The exception to this rule is that when there are other change orders associated with the subscription, you cannot delete the activation change order. For instructions on how to activate a subscription, see the help topic <a href="#">Activating a Subscription</a>.</p>
Modify Pricing	<p>A modify pricing change order changes the pricing tier or rate, the quantity of a subscription item, or applies a new discount to active line items on a subscription.</p> <p>The effective date of a modify pricing change order must be after its activation date. When you create and save a modify pricing change order, the charges automatically update to reflect the change order. You can also delete modify pricing change orders. Deleting the change order also deletes the changes to the charges that were applied as part of the change order. For instructions on how to create a modify pricing change order, see <a href="#">Modifying Pricing</a>. If you want to change to off-cycle pricing, which prorates pricing charges mid-cycle, see <a href="#">Off-Cycle Invoicing with Modify Pricing Change Orders</a>.</p>
Suspend	<p>A suspend change order temporarily suspends a subscription or subscription line item until it is reactivated. You can apply a suspend change order to any number of subscription line items. If you suspend a usage line item, you also must suspend the associated usage multiplier line item. You cannot suspend or terminate any line items or subscriptions with a status of <b>Suspend</b>. Subscription lines in the suspended state are not billed until reactivated. To make any changes to a suspended subscription or line item, you must first reactivate it. When a line item or subscription is in the <b>Suspend</b> status, you can terminate it using a terminate change order. For instructions, see <a href="#">Suspending a Subscription Line Item</a>.</p>
Reactivate	<p>Create a reactivate change order to reactivate a suspended subscription or subscription line item. The subscription or line item is reactivated on the date set as the effective date on the change order and changes the status to <b>Active</b>. For instructions, see <a href="#">Reactivating a Subscription Line Item</a>.</p>
Renew	<p>A renew change order renews a subscription when the subscription term ends. You cannot delete a renew change order, but you can make changes before the renewal start date. The renew change order lets you create a new subscription or extend the existing subscription. For more information on the different types of subscription renewals, see <a href="#">Renewing a Subscription</a>.</p>
Terminate	<p>Terminate change orders end a subscription line item at the end of the day on the effective date. To terminate a subscription line item, it must have a status of <b>Active</b>, <b>Pending Activation</b>, or <b>Suspend</b>. After saving a terminate change order, no other change orders are permitted after the effective date of the termination on the subscription line item. If you backdate a termination change order, however, you cannot edit it or delete the change order. You can still make changes prior to the termination effective date. For instructions, see <a href="#">Terminating a Subscription Line Item</a>.</p>

## Maximum Values in SuiteBilling

The maximum number you can enter in the Quantity column of several NetSuite records is 9,999,999,999.99999999. Quantities of one-billion or greater result in an error. This maximum applies to the following records:

- Pricing subtab of a Subscription record

- Items on a subscription Modify Pricing Change Order
- Sales Order
- Invoice
- Credit Memo
- Charge

## Decimal Precision in SuiteBilling

The Price, Quantity, and Discount fields in SuiteBilling display up to eight numerals to the right of the decimal. NetSuite rounds values with more than eight decimal places in these fields to the eighth decimal place. These fields appear on:

- Modify pricing change orders
- Renewal change orders
- Subscription change orders (activate, suspend, reactivate, terminate)
- **Price Book** subtab on subscription plans
- Price books
- Price plans
- Subscriptions
- Subscription Pricing subtab

Rounding to eight decimal places is mainly useful for both usage and high-volume products. NetSuite carries and uses eight decimal places for all subscription rating calculations until the final charge amount is calculated.

The charge amount and precision are based on your account currency, and the amount on the invoice appears in the account's currency. Charge-based billing may display a blended rate. To add or edit your account currencies, see [Currencies](#).

## Revenue Recognition for SuiteBilling

SuiteBilling integrates with Advanced Revenue Management (Essentials) and Advanced Revenue Management (Revenue Allocation) when you enable all three features. After setup, each subscription line generates a revenue element. A single revenue arrangement that corresponds with the subscription automatically includes the revenue elements from the subscription lines.

If you check the accounting preference **Create Revenue Elements for Subscription Revisions** box, subscription revisions also trigger modification elements and revenue arrangements. For more information, see [Setting the Optional Accounting Preference for Advanced Revenue Management](#).



**Note:** If you check the **Create Revenue Elements for Subscription Revisions** box, be sure to merge modification elements with the subscription revenue arrangements to get the correct accounting. For information, see the help topic [Combination and Modification of Performance Obligations](#).

All of the processes of Advanced Revenue Management (Essentials) and Advanced Revenue Management (Revenue Allocation) are available for SuiteBilling. For information about Advanced Revenue Management, see the help topic [Advanced Revenue Management \(Essentials\) and \(Revenue Allocation\)](#).

The following topics have detailed information about configuring advanced revenue recognition for SuiteBilling.

- Revenue Recognition Rules for SuiteBilling
- Calculations for Revenue Plans Created on Subscription Events
- Revenue Allocation for One-Time and Recurring Subscription Lines

After you configure Advanced Revenue Management for SuiteBilling, be sure to complete the additional month-end steps to recognize revenue in your month-end procedures. The basic month-end revenue recognition processes are as follows:

1. Create revenue recognition journal entries. You can schedule the revenue recognition journal entry creation process. For details, see the help topic [Revenue Recognition Journal Entries](#).
2. Reclassify deferred revenue. You can schedule the reclassification process. For details, see the help topic [Reclassification of Deferred Revenue](#).
3. Recalculate revenue forecast plans. See the help topic [Recalculating Revenue Forecast Plans](#).
4. Run and save the Deferred Revenue Waterfall Report. See the help topics [Run and Save the Deferred Revenue Waterfall Report](#) and the linked topics for the individual reports.

## Revenue Recognition Rules for SuiteBilling

**Note:** The information in this topic applies regardless of whether you check the **Create Revenue Elements for Subscription Revisions** accounting preference except where the accounting preference is clearly identified.

NetSuite provides the following default revenue recognition rules for SuiteBilling. You select revenue recognition rules on the item record. For more information, see [Configuring Subscription Items for Revenue Recognition](#) and [Creating Items for Subscription Plans](#).

The following table shows the recommended revenue recognition rules that correspond to the rate plans of the subscription line types:

Subscription Line Type	Revenue Recognition Rule
Recurring – Adjustable	Depends on the state of the accounting preference <b>Create Revenue Elements for Subscription Revisions</b> : <ul style="list-style-type: none"> <li>■ When clear, use Default Adjustable Recurring Fee</li> <li>■ When checked, use Default Fixed Recurring Fee</li> </ul>
Recurring – Fixed	Default Fixed Recurring Fee
One-Time	Default Fixed Recurring Fee
Usage	Default Usage

You can change the names of these default revenue recognition rules but not their settings. The default values for these fields are listed in the following table. For more information about the revenue recognition rules, see the help topic [Revenue Recognition Rules](#).

Revenue Recognition Rule	Recognition Method	Amount Source	Rev Rec Start Date Source	Rev Rec End Date Source
Default Adjustable Recurring Fee	Straight-line, prorate first & last period (period-rate)	Event-Percent based on amount	Subscription Event Start Date	Subscription Event End Date
Default Fixed Recurring Fee	Straight-line, prorate first & last period (period-rate)	Event-Percent based on amount	Revenue Element Start Date *	Revenue Element End Date *

Revenue Recognition Rule	Recognition Method	Amount Source	Rev Rec Start Date Source	Rev Rec End Date Source
Default Usage	Straight-line, prorate first & last period (period-rate)	Event-Percent based on amount	Event Date	Event Date

- \* To generate revenue plans using these values, you usually enter the **Start Date** and **End Date** in the revenue element. However, when the revenue element source is a subscription, these dates are automatically derived from the subscription lines.

None of the default revenue recognition rules for SuiteBilling use Event-Percent based on quantity as the amount source. This amount source is not suitable for SuiteBilling.

When the credit memo does not contain a linked subscription charge, the revenue recognition rule for the item record is ignored. Instead, Advanced Revenue Management (Essentials) uses the rule you select with the **Default Standard Revenue Recognition Rule** accounting preference to generate the revenue plan for the credit memo line. For information about accounting preferences for revenue recognition, see the help topic [Setting Advanced Revenue Management \(Essentials\) Preferences](#).

Subscription lines with usage do not create revenue plans until the bill run executes. Forecasting and reforecasting are not supported for subscription lines with usage. For information about reforecasting, see the help topic [Recalculating Revenue Forecast Plans](#).

## Calculations for Revenue Plans Created on Subscription Events



**Note:** This topic describes what happens when you clear the **Create Revenue Elements for Subscription Revisions** accounting preference. When you check the preference, the recommended value of the **Create Revenue Plans On** field, for all subscription items except usage items, is Revenue Arrangement Creation. For instructions on setting the preference and details about the checked preference, see [Setting the Optional Accounting Preference for Advanced Revenue Management](#).

The recommended value for the **Create Revenue Plans On** field on item records for Recurring – Adjustable and Usage subscription line types is Subscription Events. When a subscription event initiates creating a revenue recognition plan, the forecast revenue plan uses the full revenue amount. However, actual revenue plans are aggregated from a series of virtual plans that use the subscription event amount. Recognition methods for the virtual actual plans treat the charge period as the period.

You can use the virtual plans to set your own dates when the subscription quantity or pricing changes. The revenue recognition plan can pick up any changes in virtual plans from the period in which they occur, rather than recalculating the entire subscription term. Individual subscription charges, start dates, and end dates for virtual plans align with the lines on the Charges subtab of the Billing Account page.

Due to differences in dates and amounts, the forecast and actual revenue plans may differ from each other at the line level. The following examples show the calculations for actual and forecast revenue plans created on subscription events for two revenue recognition methods.

- [Straight-Line, by Even Periods Method](#)
- [Straight-Line, Prorate First & Last Period Method](#)

For information and examples of all recognition methods, see the help topic [Straight-Line Revenue Recognition Examples in Advanced Revenue Management \(Essentials\)](#). The examples share the following values:

- **Subscription term** – 6 months, April 9 – October 8
- **Monthly charge** – \$100
- **Total revenue amount** – \$600
- **Billing date** – the ninth day of each month, April – September

## Straight-Line, by Even Periods Method

This method divides the revenue amount evenly across all periods. Currency amounts are not prorated based on the number of days in any period. The virtual actual plans treat the charge period as the period.

The forecast plan has seven periods and a revenue amount of \$600. The last period includes the amount needed to correct rounding the previous periods.

The virtual plans have a revenue amount of \$100. The monthly charge amount is divided equally between the two periods shown as A and B in the following table. The actual revenue plan line is the sum of the calculations for:

- The days after and including the billing date (B)
- The days before the billing date (A)

Period	Billing Date	Charge	Forecast Plan	Virtual Plan A	Virtual Plan B	Actual Plan A + B =
Apr	Apr 9	100.00	85.71		50.00 = Charge a ÷ 2	50.00
May	May 9	100.00	85.71	50.00 = Charge a ÷ 2	50.00 = Charge b ÷ 2	100.00
Jun	Jun 9	100.00	85.71	50.00 = Charge b ÷ 2	50.00 = Charge c ÷ 2	100.00
Jul	Jul 9	100.00	85.71	50.00 = Charge c ÷ 2	50.00 = Charge d ÷ 2	100.00
Aug	Aug 9	100.00	85.71	50.00 = Charge d ÷ 2	50.00 = Charge e ÷ 2	100.00
Sep	Sep 9	100.00	85.71	50.00 = Charge e ÷ 2	50.00 = Charge f ÷ 2	100.00
Oct			85.74	50.00 = Charge f ÷ 2		50.00
Total		600.00	600.00			600.00

## Straight-Line, Prorate First & Last Period Method

This method recognizes revenue in equal amounts for all periods except the first and final periods, regardless of the number of days in those periods. Currency amounts are prorated for the first period and the final period based on the number of days in those periods, divided by the total number of days from the revenue recognition start and end dates, inclusive. The virtual actual plans treat the charge period as the period.

The forecast plan has 183 days and a revenue amount of \$600. The last full period includes the amount needed to correct rounding of the previous periods.

For the actual plan, each service period extends over parts of two months, so each is prorated by the number of days in the month. The actual revenue plan line is the sum of the calculations for the days:

- After and including the billing date (B)
- Before the billing date (A)

Period	Billing Date	Charge	Forecast Plan	Virtual Plan A	Virtual Plan B	Actual Plan A + B =
Apr	Apr 9	100.00	72.13 = $600 \div 183 \times 22$		73.33 = charge a ÷ $30 \times 22$	73.33
May	May 9	100.00	100.33 = $(600 - 72.13 + 26.23) \div 5$	26.67 = charge a - 73.33	74.19 = charge b ÷ $31 \times 23$	100.86
Jun	Jun 9	100.00	100.33 = $(600 - 72.13 + 26.23) \div 5$	25.81 = charge b - 74.19	73.33 = charge c ÷ $30 \times 22$	99.14
Jul	Jul 9	100.00	100.33 = $(600 - 72.13 + 26.23) \div 5$	26.67 = charge c - 73.33	74.19 = charge d ÷ $31 \times 23$	100.86
Aug	Aug 9	100.00	100.33 = $(600 - 72.13 + 26.23) \div 5$	25.81 = charge d - 74.19	74.19 = charge e ÷ $31 \times 23$	100.00
Sep	Sep 9	100.00	100.32 = $(600 - 72.13 + 26.23) \div 5$	25.81 = charge e - 74.19	73.33 = charge f ÷ $30 \times 22$	99.14
Oct			26.23 = $600 \div 183 \times 8$	26.67 = charge f - 73.33		26.67
Total		600.00	600.00			600.00

## Revenue Allocation for One-Time and Recurring Subscription Lines

**Note:** The information in this topic applies regardless of whether you check the **Create Revenue Elements for Subscription Revisions** accounting preference except where the accounting preference is clearly identified.

The Advanced Revenue Management (Revenue Allocation) feature must be enabled as a prerequisite for revenue allocation.

To allocate revenue for one-time and recurring subscription lines, revenue elements include a field called **Alternate Quantity**. The alternate quantity is the result of the quantity of the subscription line item

multiplied by the **recurrence frequency**, which is the number of times the customer is charged over the subscription term.

For example, consider two subscription lines. The line in example A has a term of 6 years with an annual recurrence frequency and a quantity of 2. Its alternate quantity is 12. The line in example B has a term of 3 years with a monthly recurrence frequency and a quantity of 5. Its alternate quantity is 180.

	Example A	Example B
Term	6 years	3 years
Recurrence Frequency	Annually	Monthly
Term x Recurrence Frequency	6 year term × 1 charge per year = 6	3 year term × 12 charges per year = 36
Quantity	2	5
Alternate Quantity	$2 \times 6 = 12$	$5 \times 36 = 180$

The fair value price record must include the appropriate values for Units Type and Units to support the allocation. When you include a value in Alternate Quantity and the other alternate fields, the {quantity} field in fair value formulas uses the alternate quantity. Units Type and Units are displayed on the fair value price record only when you enable the Multiple Units of Measure feature. For information, see the help topic [Multiple Units of Measure](#).

## Alternate Quantity Computation for Activation Revenue Arrangement When the Billing Schedule Starts Mid-Month

The following example shows how the Alternate Quantity is computed for the activation revenue arrangement when the billing schedule starts mid-month and proration is selected for subscription start date and subscription end date:

- Start Date 8/1/2019
- End Date 11/30/2019
- Recurrence Date 8/16/2019

With the recurrence start date preference on, the billing account invoice periods affect the subscription recurrence pattern.

The billing account affects the alternate quantity because the charge and rating period are offset from the start date, causing the recurrence start date to be 8/16/2019 for the subscription. This makes the first and last charge periods prorated, as shown in the following table:

Rating Periods	Start Date	End Date	Days	Daily Quantity
7/16/2019	8/15/2019	31	10	0.322581
8/16/2019	9/15/2019	31	10	0.322581
9/16/2019	10/15/2019	30	10	0.333333
10/16/2019	11/15/2019	31	10	0.322581
11/16/2019	12/15/2019	30	10	0.333333
Charge Periods				

Rating Periods	Start Date	End Date	Days	Daily Quantity
8/1/2019	8/15/2019	15	4.8387097	
8/16/2019	9/15/2019	31	10	
9/16/2019	10/15/2019	30	10	
10/16/2019	11/15/2019	31	10	
11/16/2019	11/30/2019	15	5	
			39.8387097	

# SuiteBilling Setup

This chapter describes the actions required to enable and set up SuiteBilling. SuiteBilling requires you to set preferences before you can enable its associated features. This chapter includes tasks that you usually need to perform only one time for your initial setup.

## Setting the SuiteBilling Preferences

**Note:** Before you can enable the SuiteBilling feature, you must contact your NetSuite sales or account representative to provision this feature for your NetSuite account.

You cannot enable the SuiteBilling features until these actions are completed:

- [Consolidating Projects on Sales Orders](#)
- [Disabling Employee Commissions](#)

You can also set [Setting Optional Preferences](#).

For high-level overview diagram of the workflows that include this procedure, see [Subscription Plan Creation](#), [New Stand-Alone Subscription Creation](#), and [Subscription Creation from a Sales Order](#).

### Consolidating Projects on Sales Orders

If you have the Projects feature enabled, before you can enable SuiteBilling, you need to set the Consolidate Projects on Sales Transactions preference.

#### To set the accounting preference to consolidate projects on sales orders:

1. Go to Setup > Accounting > Accounting Preferences.
2. Click the **Items/Transactions** subtab.
3. Check the **Consolidate Projects on Sales Transactions** box.
4. Click **Save**.

### Disabling Employee Commissions

Before you can enable SuiteBilling, you need to disable the Employee Commissions feature.

**Note:** When your account includes commissions data, that data is inaccessible, although still in the account, when Employee Commissions is disabled. If you want to refer to commissions data while using SuiteBilling, you can export the data from Reports > Commissions > Commission Overview Report for Employee commissions and Reports > Commissions > Partner Commission Overview Report for Partner commissions. To correctly set the time range of the report, set the **From** field to the employee start date and the **To** field to the current date. For additional instructions, see the help topic [Exporting a Report](#).

#### To disable the Employee Commissions feature:

1. Go to Setup > Company > Enable Features.
2. Click the **Employees** subtab.
3. In the Commissions section, clear the **Employee Commissions** box.
4. Click **Save**.

# Enabling SuiteBilling Features

SuiteBilling has a collection of features to enable.

Before you enable SuiteBilling features, be sure that:

- You are logged in to NetSuite using an Administrator role.
- You have set the SuiteBilling preferences as instructed in [Setting the SuiteBilling Preferences](#).

For high-level overview diagrams of the workflows that include this procedure, see [Subscription Plan Creation](#), [New Stand-Alone Subscription Creation](#), and [Subscription Creation from a Sales Order](#).

## To enable SuiteBilling features:

1. Go to Setup > Company > Enable Features.
2. Click the **Transactions** subtab.
3. In the Billing section, check the following boxes:
  - **Bill Costs to Customers**
  - **Advanced Billing**
  - **Charge-Based Billing**
  - **Billing Accounts**
  - **Billing Operations**
  - **Subscription Billing**
  - **Advanced Subscription Billing**

Optionally, you can check the following boxes:

- Add-on Items
- Time-Based Pricing, which you use to create multiple time-based pricing intervals. For instructions on how to use time-based pricing, see [Time-Based Pricing](#).

When Time-Based Pricing is enabled, you can use the Subscription Billing Enhanced UI SuiteApp. This SuiteApp has the same functionality as the SuiteBilling Time-Based Pricing feature, but with an improved user interface. For more information about the Subscription Billing Enhanced UI SuiteApp, see the help topic [Subscription Billing Enhanced UI](#).

4. On the Company tab in the Classifications section, you can optionally check any of the following boxes:
  - **Departments**
  - **Locations**
  - **Classes**
5. Click **Save**.

# Optional Preferences for SuiteBilling

SuiteBilling has several optional preferences.

## Setting Optional Preferences

You can streamline your subscription creation and management process with optional preferences.

## To set optional preferences for subscription creation and management:

1. Go to Setup > Accounting > Accounting Preferences.
2. On the **Classifications** subtab, check the box for each preference that you want to set:

- **Make Classes Mandatory**
- **Make Departments Mandatory**
- **Make Locations Mandatory**
- **Allow Per-Line Classes**
- **Allow Per-Line Departments**
- **Allow Per-Line Locations**

If you check the **Make Classes Mandatory**, **Make Departments Mandatory**, or **Make Locations Mandatory** preferences, these fields are required while you are creating the subscription or subscription line. If any of the Allow Per-Line preferences are checked, the line-level value is used in revenue reporting. If Allow Per-Line preferences are cleared, the class, department, and location values on the subscription line record come from the billing account. For more information about the **Class**, **Department**, and **Location**, see [Class, Department, and Location Fields on Subscription Lines](#).

3. Go to Setup > Accounting > Invoicing Preferences.
4. On the **Subscription Management** subtab:



**Note:** The **Subscription Management** subtab does not appear until after you enable SuiteBilling.

- a. Check the box for each preference that you want to set:
  - **When Sales Order Gets Approved, Auto-Change Subscription Status to 'Pending Activation'.** This will move subscription lines in Draft status to Pending Activation status if the subscription is saved on a sales order and the order gets approved.
  - **First Activation Change Order Creation Sets the Subscription Start Date.** With this preference enabled, the start date and activation date need to be the same. Otherwise, the start date changes to the line item activation date.
  - **Request Credit Memo for Off-Cycle Change Orders.** This preference automatically creates credit memos when doing off-cycle invoicing.
  - **Create Delta Charges for Changes to Invoiced Service Periods.** If this preference is not visible in your account, contact your NetSuite account representative or NetSuite Customer Support. There is no additional cost to activate Delta Charges.



**Note:** You cannot turn off Delta Charges after the first Delta charge is created.

To see how to set this preference, view the video:



[Enabling Delta Charges](#)

- b. To align subscription charges with the subscription start date, check the **Align Charge Amounts with Subscription** box.

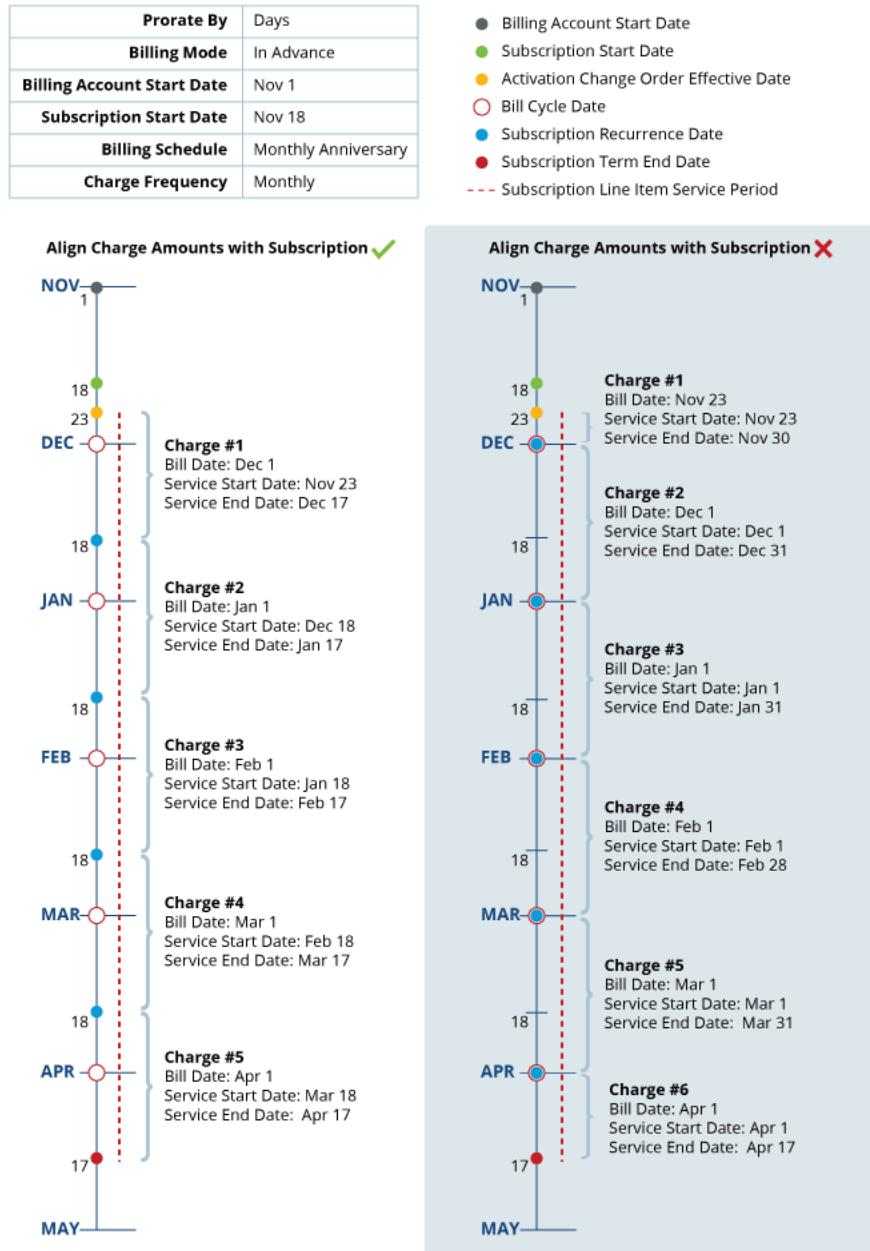
When you activate a subscription, its current preference setting dictates the subscription charge pattern.

When you check the **Align Charge Amounts with Subscription** preference, and then activate subscriptions:

- All lines within a single activated subscription have a charge pattern aligned with the subscription header start date instead of its bill dates.

- Multiple-year subscriptions with multiple charge frequencies, such as monthly, quarterly, and annual, recur based on the subscription start date for better alignment of charge periods.

The subscription charge patterns of the **Align Charge Amounts with Subscription** preference is shown in the following figure.



Existing subscriptions activated before setting this preference may display the original recurrence pattern.

If the **Align Charge Amounts with Subscription** preference is cleared, charges align with the billing schedule of the billing account.. When this preference is set, charges align the recurrence start date with the subscription start date. When the preference is set but you have not selected a billing account for the subscription, the recurrence start date is the

subscription start date. When the subscription has no billing but **Align Charge Amounts with Subscription** is cleared, the start date is not calculated until you save, when you must select a price book.

When you select a billing account for the subscription and click **Save**, the recurrence start date updates to align with the billing account recurrence pattern. This changes the ratable events that are generated in Draft mode, which may affect the Sales Order synchronization or other processes that depend on the ratable event amounts. When you select a billing account, rating runs.

If **Align Charge Amounts with Subscription** is cleared, the total contract value may change when you select a billing account on the subscription.

- c. To automatically extend the Estimated Revenue Recognition End Date for evergreen subscriptions, enter a value between 1 and 120 for the following preference:

#### **Months To Extend Estimated Revenue Recognition End Date**



**Tip:** Set this preference according to your billing schedule.

- If you bill evergreen subscriptions monthly, set this preference to 1 or greater.
- If you bill evergreen subscriptions quarterly, set this preference to 3 or greater.
- If you bill evergreen subscriptions annually, set this preference to 12 or greater.

- d. If you entered a value for the **Months To Extend Estimated Revenue Recognition End Date** preference, you also must enter a value between 1 and 120 for the following preference:

#### **Months Prior To Estimated Revenue Recognition End Date To Initiate Auto-Extend**

5. To ensure that all days are prorated to the same amount regardless of the number of days in the year, check the **Exclude February 29 When Prorating by Days or Months** preference.

To see how to set this preference, view the video:



[Setting the Leap Year Preference](#)

6. To remove a one-day charge from a subscription line item, check the **Default Terminations Effective at Start of Day** preference.
7. Click **Save**.

## Setting the Optional Accounting Preference for Advanced Revenue Management

To create individual revenue elements for each subscription change order, check the **Create Revenue Elements for Subscription Revisions** accounting preference. When you check the preference:

- All subscription revisions generate a separate revenue arrangement and corresponding revenue elements.
- All subscription revisions generate modification elements that account for the impact on revenue.

Check this preference to:

- Generate the modification element according to the ASC-606 accounting policies for subscription modifications when you have allocated revenue
- Prospectively merge the modification elements

You also can still retrospectively merge the modification element. For instructions, see the help topic, [Merging to Combine Revenue Arrangements](#). The Revenue Summary popup window for revenue arrangements shows merged elements. For more information, see the help topic [Revenue Summary Details](#).



**Note:** This accounting preference is not compatible with custom revenue recognition events.

### To set the optional accounting preference for Advanced Revenue Management:

1. Go to Setup > Accounting > Accounting Preferences.
2. Click the **General** subtab.
3. On the **Subscription Management** section, check the **Create Revenue Elements for Subscription Revisions** box.



**Warning:** After you set the **Create Revenue Elements for Subscription Revisions** preference, you cannot change it.

4. Click **Save**.

For more information, see [Revenue Recognition for SuiteBilling](#).

## Enabling the Multi-Subsidiary Customer Feature for SuiteBilling

If you have billing accounts across subsidiaries, you can take advantage of the Multi-Subsidiary Customer functionality. For more information, see the help topic [Assigning Subsidiaries to a Customer](#).

### To enable Multi-Subsidiary Customer for SuiteBilling:

1. Enable the SuiteTax feature. For details, see the help topic [Enabling the SuiteTax Feature](#).



**Note:** SuiteBilling does not require SuiteTax. However, if you want to use the Multi-Subsidiary Customer feature and SuiteBilling together, you need to enable SuiteTax. Nexus switching that is required to support multiple subsidiaries associated with one customer and the switching of Bill To addresses are best supported with SuiteTax.

2. Enable the Multi-Subsidiary Customer feature. For details, see the help topic [Enable the Multi-Subsidiary Customer Feature](#).

If the Multi-Subsidiary Customer feature is enabled, the billing account field is not automatically populated on the Subscription record.

## Creating Items for Subscription Plans

A subscription plan is made up of all of the items that define a subscription. These items include recurring services, usage, and one-time charges. Before you can create a subscription plan, you must create item records for the products and services you sell as part of subscriptions.

To learn how to create subscription items, view the video:



[Creating Subscription Items](#)

For high-level overview diagrams of the workflows that include this procedure, see [Subscription Plan Creation](#), [New Stand-Alone Subscription Creation](#), and [Subscription Creation from a Sales Order](#).

### To create an item for a subscription plan:

1. Go to Lists > Accounting > Items > New.
2. Subscription items must be either a non-inventory item or a service item for sale. Other item types are not available on subscription plans. Do one of the following:
  - To add a non-inventory item, under **Non-inventory Item**, click **For Sale** or **For Resale**.
  - To add a service item, under **Service Item**, click **For Sale** or **For Resale**.
3. In the **Item Name/Number** field, enter a value.
4. You can define the **Class**, **Department**, and **Location** fields on the Subscription Line record. For more information on class, department, and location, see [Class, Department, and Location Fields on Subscription Lines](#).
5. Complete any additional fields.
6. Click the **Accounting** subtab.
7. From the **Tax Schedule** list, select a tax schedule. To add a new tax schedule to the list, click the **Open** arrow.
8. Click the **Preferences** subtab.
9. Clear the **Can Be Fulfilled/Received** box.



**Note:** For non-inventory items, this box is checked by default and you must clear it. For service items, this box is cleared by default.

10. Click **Save**.

For more information about the item record and using items, see the help topic [Using Item Records](#).

To set up revenue recognition for the item, see [Configuring Subscription Items for Revenue Recognition](#). For more information about advanced revenue recognition, see [Revenue Recognition for SuiteBilling](#).

## Class, Department, and Location Fields on Subscription Lines

You can define the **Class**, **Department**, and **Location** fields on the Subscription and Subscription Line records in the Draft or Pending Activation status. NetSuite supplies the values of these fields automatically in the following order:

1. If the Item record has values in any of these fields, that field on the Subscription Line record shows the Item record's values.
2. If the Item record does not have a value for any of these fields, NetSuite checks the subscription plan, and the field on the Subscription Line record shows its subscription plan value, if any.
3. If a field does not have a value in the subscription plan, NetSuite checks the billing account, and the field on the Subscription Line record shows those values, if any.
4. If the Item record, subscription plan, and billing account do not have values in these fields, they are empty on the Subscription Line record.

If any of the **Allow Per-Line Classes**, **Allow Per-Line Departments**, or **Allow Per-Line Locations** preferences are checked, revenue reporting uses the line-level value. If the line-level preferences are cleared, the values for Class, Department, or Location of the subscription line come from the billing

account. For reporting processes, the values of the **Class**, **Department**, and **Location** fields from the subscription line are used for charges and revenue elements.

Values flow from the subscription record and subscription line to the charges and revenue processes, then to the invoice processes, and finally to recognizing revenue. Existing subscription lines update to the value set on the item or billing account, depending on whether the line-level preferences are set. Any existing revenue elements will not change. Charges may change if line-level preferences are cleared.

## Configuring Subscription Items for Revenue Recognition

**Note:** The information in this topic applies regardless of whether you check the **Create Revenue Elements for Subscription Revisions** accounting preference except where the accounting preference is clearly identified.

### To configure subscription items for revenue recognition:

1. To set up subscription items with different subscription line types, create an item record for each subscription type. For instructions, see [Creating Items for Subscription Plans](#).

 **Tip:** When the Advanced Revenue Management (Essentials) feature is enabled, the **Deferred Revenue Account** field on the **Accounting** subtab defaults to the system-generated Deferred Revenue account. If you want to use a different account, be sure to select it.

2. On the **Revenue Recognition/Amortization** subtab, in the Advanced Revenue Recognition group, set the other required fields for revenue recognition as follows:
  - a. Use the same value for the required Rev Rec Forecast Rule as for the Revenue Recognition Rule.
  - b. Accept the defaults for fields not included in the following table.
  - c. The recommended values for the subscription line types are in the following table.

Subscription Line Type	Revenue Recognition Rule	Create Revenue Plans On	Allocation Type
<b>Recurring – Adjustable</b> , when the <b>Create Revenue Elements for Subscription Revisions</b> box is cleared	Default Adjustable Recurring Fee	Subscription Events	Normal
<b>Recurring – Adjustable</b> , when the <b>Create Revenue Elements for Subscription Revisions</b> box is checked	Default Fixed Recurring Fee	Revenue Arrangement Creation	Normal
<b>Recurring – Fixed</b>	Default Fixed Recurring Fee	Revenue Arrangement Creation	Normal
<b>One Time</b>	Default Fixed Recurring Fee	Revenue Arrangement Creation	Normal
<b>Usage</b>	Default Usage	Subscription Events	Exclude*

\* For usage subscriptions, set the **Allocation Type** to Exclude to avoid potential errors when revenue arrangements are merged. Revenue Allocation is supported for recurring and one-time subscriptions. For more information, see [Creating Usage Records](#).

3. On the **Accounting** subtab of the item record, select a **Deferred Revenue Account**.

Create Revenue Plans On is not a required field because you can set its value in the revenue element. Do not use Revenue Arrangement Creation as the value for Create Revenue Plans On when the rule uses subscription events as the start and end date sources.

The Default Adjustable Recurring Fee rule is an example of a rule with these start and end date sources. Actual revenue recognition plans that are created on subscription events do not include start and end dates. For more information about plans created on subscription events, see [Calculations for Revenue Plans Created on Subscription Events](#).

## Creating Subscription Plans

When you create a subscription plan, you specify the items, such as recurring services, usage, and one-time charges, to include in the subscription.

To learn how to set up a subscription plan and price plan in NetSuite, view the video:



For high-level overview diagrams of the workflows that include this procedure, see [Subscription Plan Creation](#), [New Stand-Alone Subscription Creation](#), and [Subscription Creation from a Sales Order](#).

### To create a subscription plan:

1. Go to Lists > Subscriptions > Subscription Plans > New.
2. In the **Subscription Plan Name** field, enter a name.
3. From the **Initial Term** list, do one of the following tasks:
  - Select a term. Options include the following:
    - **Standard Term** – The subscription-level end date defines the end date for all of the items. This results in the alignment of end dates. NetSuite calculates the end date as the subscription start date plus the indicated subscription term. One-time subscription lines are the exception, because they are only active for one day.
    - **Custom Term** – You must set the subscription-level end date at the subscription level. This date defines the end date for all of the items, resulting in the alignment of end dates. One-time subscription lines are the exception, because they are only active for one day. The subscription-level end date for a custom term remains editable until any subscription line becomes active.
    - **Evergreen Term** – Revenue ends after three years unless you set optional invoicing preferences to automatically extend revenue recognition.
  - Create a new term by clicking the plus icon.
4. On the **Lines** subtab, do the following for each item you want to add:
  - a. In the **Item** column, select an item. If the item list does not include an item you need for this subscription plan, see [Creating Items for Subscription Plans](#).
  - b. In the **Required** column:
    - Check the box if you want the subscription to always include this item.
    - Clear the box if you want the customer to be able to include or exclude the item as an option at the time of sale.
  - c. In the **Type** column, select one of the following:
    - **One Time** – Items that incur a single charge, such as a setup or initialization fee.

- **Recurring** – Items that incur charges throughout the subscription, such as recurring service.
  - **Usage** – Items that incur charges based upon how much or how often the customer uses them, such as minutes or data.
- d. In the **Mode** column, select one of the following:
    - **In Advance** – Payment is due at the beginning of the charge period (for example, the first day of the month).
    - **In Arrears** – Payment is due at the end of the charge period (for example, the last day of the month).
  - e. In the **Include in Renewal Subscription** column, click the down arrow to display the available renewal options and select a different option if desired.
  - f. If required, for recurring items, check or clear the **Prorate Start Date** and **Prorate End Date** boxes. For details, see [Proration Options](#).
  - g. Click **Add**.
5. Repeat step 4 until you have added all items to the subscription plan.
  6. Complete any necessary additional fields, and then click **Save**.

When you save, NetSuite opens the **Price Books** subtab. After creating a subscription plan, the next task is to create a price book by creating a price plan for each item. See [Creating Price Books](#).

## Proration Options

Set the **Exclude February 29 When Prorating by Days or Months** preference in SuiteBilling to ensure that all days are prorated to the same amount regardless of the number of days in the year. When you enable this preference, NetSuite excludes February 29<sup>th</sup> when calculating prorated amounts in leap years. To locate the preference, go to Setup > Accounting > Invoicing Preferences and click the **Subscription Management** tab.

To see how to set this preference, view the video:



[Setting the Leap Year Preference](#)

On the subscription line, set the Prorate Start Date and Prorate End Date when you want to charge customers only for the portion of the charge period they use. All subscription items have the same end date as the subscription. If you leave the **Prorate Start Date** and **Prorate End Date** boxes cleared, NetSuite bills the entire charge period. In most cases, you would either check both boxes or clear both boxes.

On recurring subscription plan items:

- To set proration for the start date of an item, check the **Prorate Start Date** box.
- To set proration for the end date of an item, check the **Prorate End Date** box.

By default, if you set the Type field of a subscription plan item to Recurring, NetSuite checks the Prorate Start Date and Prorate End Date boxes.

This is what happens when you check or clear one or both of these boxes:

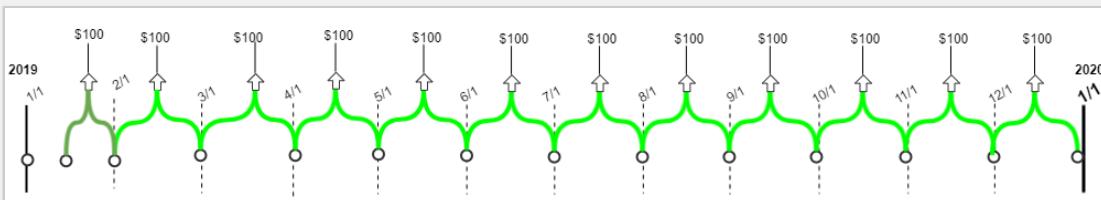
- If you check the **Prorate Start Date** box for an item, NetSuite bases the charge on the actual number of days used. If service begins after the start of the charge period, NetSuite charges the customer for the days used, not the entire charge period.
- If you clear the **Prorate Start Date** box for an item, the start date of the item is the first date in the charge period. Charges may begin earlier than the activation start date. For example, a customer

begins receiving service for a one-year subscription on December 15, and is charged for an entire year.

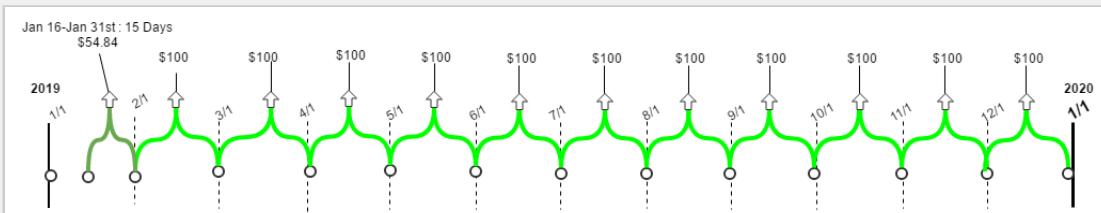
- If you check the **Prorate End Date** box for an item, NetSuite stops charges on the day the service ends. If service ends before the full charge period is complete, NetSuite bills the customer only for the days used.
- If you clear the **Prorate End Date** box for an item, the end date of the item is the last day in the charge period. Charges may extend past the subscription end date. For example, a customer cancels their phone subscription on December 15 but continues to be charged up to and including December 31.

**Example:** A subscription with a 12-month term and charge frequency set to monthly is activated on January 15. Each item represents one month of service with a charge to the customer of \$100 US.

If you clear the **Prorate Start Date** and **Prorate End Date** boxes, the customer is billed the full monthly cost for January.



If you check the **Prorate Start Date** and **Prorate End Date** boxes, the charge for the month of January is reduced to \$54.84. This reduced charge is based on the actual number of service days the customer received. If the line is terminated prior to 12/31/2019, the end charge is also not a full charge.



## Creating Price Books

After you save a subscription plan, you can create pricing for the items. A price book defines the pricing for a collection of items or services that are sold as part of a subscription plan. Every item on a subscription plan must be assigned to a price plan within the price book. A subscription plan can have one or multiple price books. You can use multiple price books to create different pricing options, such as multiple currencies, or monthly, quarterly, and annual pricing.

To learn how to set up a subscription plan and price book in NetSuite, view the video:



For high-level overview diagrams of the workflows that include this procedure, see [Subscription Plan Creation](#), [New Stand-Alone Subscription Creation](#), and [Subscription Creation from a Sales Order](#).

### To create a price book:

1. On the saved subscription plan record, click the **Price Books** subtab. This subtab opens by default after you save a subscription plan.

2. Click **New Price Book**.
3. On the price book record, in the **Name** field, enter a name for the price book.
4. From the **Currency** list, select the appropriate currency.
5. On the **Price Book Lines** subtab, in the **Edit** column, click the edit icon  for an item to make its price plan. For instructions on making price plans, see [Creating Price Books](#).

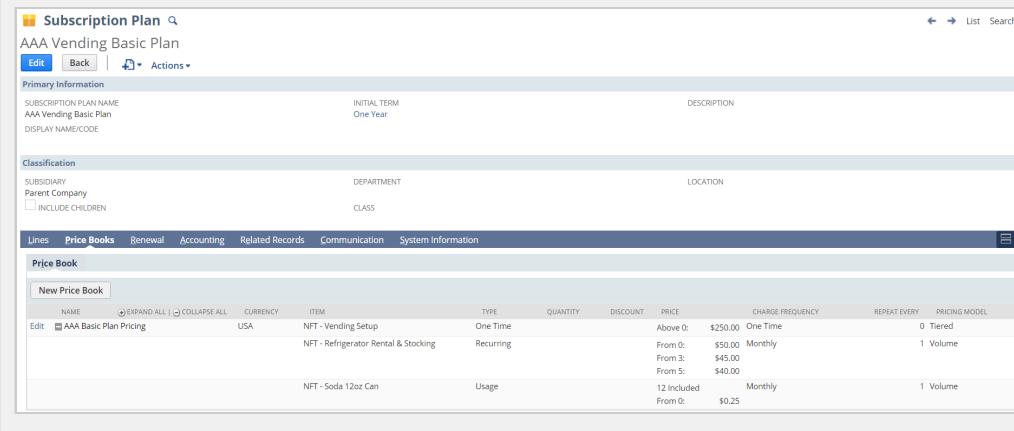


**Note:** You must complete a price plan for every item before you can save the price book. Complete the required fields on the price plan, and then click **Save** to return to the price book record.

6. From the **Interval** list (if showing), select the time period to set the duration of the interval. You can choose **Day**, **Week**, **Month**, or **Year**.
7. In the **Start On** field, enter a number for when the interval will begin.

For example, you set the **Interval** to **Month** and the **Start On** field to **1** (one). The interval runs from month 1 until the end of the term. If you create a subscription from this plan, NetSuite converts the relative dates to real dates based on the subscription start date. If you set the start date of the subscription to January 1 and the **Start On** field to 1, the interval will start on January 1.

**Example:** The following example price book, AAA Basic Plan Pricing, has currency in US dollars. The three items associated with the price book include a one-time item, a recurring item, and a usage item.



The screenshot shows the 'Subscription Plan' screen for 'AAA Vending Basic Plan'. The 'Price Book' subtab is selected. It displays a table with three rows of price book lines:

NAME	ITEM	TYPE	QUANTITY	DISCOUNT	PRICE	CHARGE FREQUENCY	REPEAT EVERY	PRICING MODEL
AAA Basic Plan Pricing	NFT - Vending Setup	One Time		Above 0:	\$250.00	One Time	0	Tiered
	NFT - Refrigerator Rental & Stocking	Recurring		From 0:	\$50.00	Monthly	1	Volume
				From 3:	\$45.00			
				From 5:	\$40.00			
	NFT - Soda 12oz Can	Usage	12 Included			Monthly	1	Volume
			From 0:		\$0.25			

8. (Optional) In the **Included Quantity Multiplier** column, enter a number. For more information, see [Creating Usage Records](#).
9. (Optional) If you want to enter a default discount, enter the discount in the **Discount** column. You can enter a discount as a percentage or as a flat rate discount. To enter a percentage discount, include the percentage sign (%) after the positive number you enter in the **Discount** field. Flat rate discounts do not need a notation. For details, see [Applying Discounts to Subscriptions](#).
10. From the **Charge Frequency** list, select **Weekly**, **Monthly**, or **Annually**.
11. Unless the item is a one-time item, select a number from the **Repeat Every** list for how often you want the charge to occur.

For example, if you want to charge monthly, select a **Monthly** from the **Charge Frequency** list and enter **1** in the **Repeat Every** field.

**Example:** The following example shows a basic price book with tiered pricing (line 1) and volume pricing (line 2), charge frequency, and recurrence. Notice that no discounts are included in the following price book example.

The screenshot displays two separate Price Book entries in the NetSuite interface.

**Price Book 1 (Top):**

- Primary Information:** NAME: PB 2, SUBSCRIPTION PLAN: Medical Records Management Plan 2.
- Price Book Lines:**

LINE NUMBER	REQUIRED	ITEM	LINE TYPE	INTERVAL *	START ON *	PRICE	EDIT	INCLUDED QUANTITY MULTIPLIER	DISCOUNT	CHARGE FREQUENCY	REPEAT EVERY *
1		Records Management	Recurring	Month	1	Above 0: \$32.00				Monthly	1
1		Records Management	Recurring	Month	4	Above 0: \$32.00 Above 1: \$34.00				Monthly	1
2		File Storage	Usage	Month	1	5 Included From 0: \$42.00				Monthly	1
2		File Storage	Usage	Month	4	5 Included From 0: \$42.00 From 6: \$44.00				Monthly	1

**Price Book 2 (Bottom):**

- Primary Information:** NAME: AAA Basic Plan Pricing, SUBSCRIPTION PLAN: AAA Vending Basic Plan.
- Price Book Lines:**

ITEM	SUBSCRIPTION LINE TYPE	PRICE	INCLUDED QUANTITY MULTIPLIER	DISCOUNT	CHARGE FREQUENCY	REPEAT EVERY
NFT - Vending Setup	One Time	Above 0: \$250.00			One Time	0
NFT - Refrigerator Rental & Stocking	Recurring	From 0: \$50.00 From 3: \$45.00 From 5: \$40.00			Monthly	1
NFT - Soda 12oz Can	Usage	12 Included From 0: \$0.25	NFT - Refrigerator Ren... Stocking - Recurring		Monthly	1

12. From the **Prorate By** column, select Month or Day.



**Tip:** Prorate By Month creates a rounded charge for the billing period. For example, a 12-month subscription costs \$12,000 that is billed monthly. If you select Prorate By Month, the charge for each month is rounded to \$1,000. However, if you select Prorate By Day for the same subscription, the monthly charge varies from \$920.55 to \$1,019.18 depending on the number of days in the month.

**Example:** When you enable proration for an item, NetSuite creates charges based on the number of days or months in the charge period. The default Prorate By setting for all recurring items is Day.

The screenshot shows the 'Price Book' creation screen. The 'NAME' field is populated with 'PB1'. The 'PRORATE BY' column for both items is highlighted with a red box. The table data is as follows:

LINE NUMBER	REQUIRED	ITEM	LINE TYPE	INTERVAL *	START ON *	INCLUDED QUANTITY MULTIPLIER	PRICE	EDIT	DISCOUNT	CHARGE FREQUENCY *	REPEAT EVERY *	PRORATE BY
1	Yes	Labor 1	Recurring	Month	1		Above 0: \$12,000.00			Annually	1	Month
2	Yes	Labor 2	Recurring	Month	1		Above 0: \$300.00			Monthly	3	Month

13. Repeat all of the previous steps for each price book item until all have pricing information.
14. Click **Save**.

## Setting Up Standard Pricing

If you have the System Administrator role, you can set up standard pricing that other users cannot modify.

### To set up standard pricing that other users cannot change:

1. Grant access permission to a subscription process with the following restrictions:
  - Limit the number of users who can edit the subscription pricing.
  - Reduce the permissions to edit a specific price plan, subscription change order, or price book.
 When you restrict access, users with roles other than System Administrator can change a subscription item discount or quantity but not the price plan unless they have price plan permissions. For instructions on how to grant roles the access permission to a subscription process, see the help topic [Setting Permissions](#).

## Copying a Price Book

You can copy a price book to save effort in filling out all the fields for a similar price book. For example, you might want to copy a price book when creating pricing books for different currencies.

### To copy a price book:

1. Open the price book record you want to copy.
2. From the **Actions** menu, select **Make Copy**.
3. Edit the copy with any needed changes.
4. Click **Save**.

## Deleting a Price Book

You can delete a price book that has not been used to create any subscriptions.

### To delete a price book:

1. Open the price book record you want to delete.
2. From the **Actions** menu, select **Delete**.

## Creating Price Plans

Every item in a subscription plan must have an associated price plan. You must create price plans for every pricing interval in a price book.

When creating a price plan, the quantity for each item must be greater than zero (0). When you modify the price plan, you cannot set the quantity to zero (0) or a negative number.



**Note:** Volume pricing charges all units at the price tier based on the quantity.

For high-level overview diagrams of the workflows that include this procedure, see [Subscription Plan Creation](#), [New Stand-Alone Subscription Creation](#), and [Subscription Creation from a Sales Order](#).

### To create a price plan:

1. From the **Price Books** subtab of the subscription plan record, select an existing price book record. If you have not already created a price book, follow the instructions in [Creating Price Books](#).
2. Configure the price plan:
  - a. Click the edit icon for an item. The price plan window opens.



**Note:** You also can click the edit icon to edit an existing price plan.

- b. From the **Type** list, select one of the following:
  - **Tiered** – Most commonly used for recurring items. Tiered pricing charges a specified rate for all items sold in the tier (for example, the first 10 items). When the quantity sold exceeds the upper limit for the tier, NetSuite charges the price specified for the next tier for the additional items.

#### Example:

A beverage subscription with tiered pricing includes the following pricing levels:

Quantity tier 1: For 0-10 items, the price is \$5.00

Quantity tier 2: For 11-20 items, the price is \$4.95

Quantity tier 3: For 21+ items, the price is \$4.90

If a customer orders 22 items, the charge is  $(10 \times \$5.00) + (10 \times \$4.95) + (2 \times \$4.90) = \$109.30$ .

- **Volume** – Most commonly used for usage pricing. For volume pricing, the total quantity sold defines the charge rate for all items in the order.

#### Example:

A beverage subscription with volume pricing includes the following pricing levels:

Volume tier 1: For 0-10 items, the price is \$5.00

Volume tier 2: For 11-20 items, the price is \$4.95

If the customer orders eight items, the per item charge for the order is \$5.00. The total charge is  $(8 \times \$5.00) = \$40$ . If the customer orders 22 items, the per item charge for the order drops to \$4.95. The total charge is  $(22 \times \$4.95) = \$108.90$ .

When you create a price plan, you can set minimum and maximum amount limits for pricing tiers. See [Price Plan Minimum and Maximum Amounts](#).

- c. From the **Pricing Option** list, select one of the following options:
  - **Rate** – Calculates the item amount as (quantity x rate).
  - **Fixed Amount** – The item's defined charge amount is not affected by its quantity.
- d. In the **Value** field, enter an amount to charge for the item.
- e. To add more pricing for additional quantities, click **Add**.
3. Repeat step 2 for each subsequent item in the price book. Every item in a subscription plan must have an associated price plan.
4. Click **Save** to submit the price plan. NetSuite returns to the Price Book page and displays the new price plan in the item columns.
5. From the **Charge Frequency** list, select **Weekly**, **Monthly**, or **Annually**.
6. Unless the item is a one-time item, from the **Repeat Every** list, enter a number for how often you want the charge to occur. For example, if you want to charge monthly, select **Monthly** from the **Charge Frequency** list and enter **1** in the **Repeat Every** field.
7. Click **Save** to save the price book.

## Price Plan Minimum and Maximum Amounts

You can set minimum and maximum amount limits for pricing tiers.

Consider the following pricing tier examples. The prices and discounts are structured in tiers based on the quantity purchased:

- 0–100 recurring licenses purchased cost \$3.00 each.
- 101–200 recurring licenses purchased cost \$2.00 each.
- 201+ recurring licenses purchased cost \$1.00 each.

You can enter a minimum amount of \$150. This forces customers to purchase at least 50 recurring licenses, even if the item has a quantity of only 1.

You may also have a discount in which after a certain price is reached, the cost for your customers does not increase. If you enter a maximum charge of \$500.00, the customer does not have to pay for additional licenses after they have spent \$500.

A volume tier can have a maximum price. But in most cases, a maximum price is needed only for the tier with the highest number range.

When a change order changes the price plan mid-period, NetSuite prorates the minimum and maximum amounts.

**Example:** A customer subscribes to your news website. The minimum charge is \$10. You submit a change order on April 13. For April 1–12, NetSuite prorates the minimum charge as days used so far divided by total days in the month, multiplied by the minimum charge:

$$4 = (12 \div 30) \times 10$$

For April 13–30 (18 days), NetSuite prorates the minimum as days remaining divided by total days in the month, multiplied by the minimum charge:

$$6 = (18 \div 30) \times 10$$

# Time-Based Pricing

Use the Time-Based Pricing feature to create pricing models that have planned pricing changes over time, also known as interval pricing models. Examples include free trials, ramp pricing, and introductory pricing. You use Time-Based Pricing to create intervals for price, quantity, or discount changes over time.

You can define how charges are calculated for time intervals during a subscription's term. For example, you offer a one-year subscription with an introductory pricing plan. A reduced price applies for the first three months and full price applies for the last nine months of the subscription's term. To set up this price plan, you add an interval for the item and set it to start on the fourth month without a discount.

For instructions on enabling Time-Based Pricing, see [Enabling SuiteBilling Features](#).

To learn how to create a subscription plan, price book, and price plans with Time-Based Pricing in NetSuite, view the video:



**Note:** If you use the SuiteBilling Enhanced UI SuiteApp (formerly called the Time-Based Pricing SuiteApp) with the Time-Based Pricing feature, see the help topic [Subscription Billing Enhanced UI](#).

## Creating a Price Plan with Intervals for Time-Based Pricing

You can apply a single price interval to all items on a price plan at once, or you can define different price intervals for each item.

You also can use the Time-Based Pricing feature to add a price interval to all items in the price book and subscription level.

Before creating a price interval, you must have a subscription plan, price book, and price plan. See [Creating Items for Subscription Plans](#), [Creating Subscription Plans](#), and [Creating Price Books](#).

The following example shows a subscription that includes two subscription line items, each with two price intervals. For each item, the new price takes effect in the 4th month.

LINE NUMBER	REQUIRED	ITEM	LINE TYPE	INTERVAL *	START ON *	PRICE	EDIT	INCLUDED QUANTITY MULTIPLIER	DISCOUNT	CHARGE FREQUENCY *	REPEAT EVERY *
1		Records Management	Recurring	Month	1	Above 0: \$200.00				Monthly	1
1		Records Management	Recurring	Month	4	Above 0: \$200.00 Above 10: \$225.00			Monthly	1	
2		File Storage	Usage	Month	1	50 Included From 0: \$350.00				Monthly	1
2		File Storage	Usage	Month	4	50 Included From 0: \$350.00 From 50: \$400.00				Monthly	1

To create a price interval, you must configure the price plan as described in the following procedures.

## Adding a Price Interval To All Items At Once

You can simultaneously add a price interval to all items in a subscription.

### To add a price interval to all items in a subscription:

- From the subscription plan on the **Price Book** subtab, click the edit icon  for an item. The price plan window opens.



**Note:** Items set up for one-time pricing cannot have more than one price interval.

- Click **Add Interval to All** to add a price interval to all items at once rather than creating separate price intervals for multiple individual items.
- Edit the price, discount, or quantity for the new price interval. For example, to create a price interval that starts after the subscription start date, change the **Start On** value.



**Note:** Price interval items that are located next to each other must be unique. If you have two identical adjacent price intervals, NetSuite combines the items from each into one price interval.

- Click **Save** to save the intervals in your price plan.
- When you have completed configuring the price interval for all items, click **Save** to save the price book. The **Interval** and **Start On** columns appear on the price book, as shown in the following image. For information on these columns of the price book, see steps 6 and 7 of [Creating Price Books](#).

## Adding a Price Interval To a Single Item

You can add a price interval to one subscription item at a time.

### To add a price interval to a single item in a subscription:

- Select the item with the price plan you want, and click **Make Copy**.



**Tip:** When you copy a price interval, you save the effort of completing all the fields that a new price interval requires. For example, if you have tiered pricing and do not want to recreate the tiers for each new price interval.

- Click the edit icon  for an item. The price plan window opens.
- Edit the price, discount, or quantity for the new price interval. For example, to create a price interval that starts after the subscription start date, change the **Start On** value.



**Note:** Price interval items that are located next to each other must be unique. If you have two identical adjacent price intervals, NetSuite combines the items from each into one price interval.

- Click **Save** to save the intervals in your price plan.
- Repeat steps 2–4 for each subsequent item in the price book.
- When you have completed configuring all intervals, click **Save** to save the price book. The **Interval** and **Start On** columns appear on the price book, as shown in the preceding image. For information on these columns of the price book, see steps 6 and 7 of [Creating Price Books](#).

## Deleting a Price Interval

You can delete a price interval when the subscription item is in the Draft status.

### To delete a price interval:

1. Select the price interval you want to delete.
2. Click **Remove**.

## Creating Billing Accounts

A billing account record shows all billing information, including the billing schedule, default payment terms, ship-to address, and currency, for a customer or sub-customer.

To enable billing accounts, you must first set the Consolidate Projects on Sales Transactions preference and enable the Charge-Based Billing and Billing Accounts features. For more information, see [Setting the SuiteBilling Preferences](#) and [Enabling SuiteBilling Features](#).

If you have multiple billing accounts for a customer, NetSuite uses the billing account marked as customer default when creating transactions. For example, when creating a new invoice or new subscription, NetSuite assigns the default customer billing account. On the billing account record, you can change the start date of a billing account only if it is not yet used in an active subscription.

A single company may choose to set up multiple billing accounts to accommodate different billing schedules. For example, they may want to bill recurring annual charges in advance and bill usage monthly, with a separate billing account for each.

**Use Case Example:** A coffee shop franchise purchases software using multiple billing accounts as follows:

- The licensing fees and maintenance are billed to the parent company annually on January 1st. This is the first billing account, with a fixed bill date of January 1st and the address of the parent company used as the ship to address and bill to address.
- The usage for the service is billed to the individual locations. For each location, there is a separate billing account with a monthly billing schedule. The bill to address and ship to address for each account correspond to the individual franchise location.

### To create a new billing account:

1. Go to Lists > Relationships > Billing Accounts > New.
2. If required, from the **Subsidiary** list, select a subsidiary or change the subsidiary from the primary subsidiary to a secondary subsidiary.
3. Indicate the customer for whom you are creating a billing account:
  - For an existing customer, select the customer name.
  - To create a new customer, click the plus icon.
4. From the **Currency** list, select the customer's currency.
5. From the **Billing Schedule** list, select a billing schedule.
6. Enter or select a **Start Date**.



**Note:** After you click **Save**, you will not be able to:

- remove the subsidiary from this customer
- delete billing accounts related to this customer and their secondary subsidiaries

7. Click **Save**.

The following screen is an example of a completed Billing Account record.

**Billing Account**

**Primary Information**

ACCOUNT NUMBER BA_1002	NAME BA_1002	CURRENCY USA
<input checked="" type="checkbox"/> CUSTOMER DEFAULT	ACCOUNT DESCRIPTION	<input type="checkbox"/> INACTIVE
CUSTOMER CAN Cust USD		

**Billing**

BILLING SCHEDULE Basic	START DATE 4/1/2020	NEXT BILL CYCLE DATE 4/26/2020
FREQUENCY Monthly	LAST BILL CYCLE DATE	LAST BILL DATE
<input checked="" type="checkbox"/> USE OFF-CYCLE INVOICE FOR ADVANCE CHARGES		

**Classifications**

SUBSIDIARY CAN Subsid	DEPARTMENT	LOCATION CAN ONLY LOCATION
CLASS CAN ONLY CLASS		

**Related Records** Address Preferences System Notes

Subscriptions Billing Schedule Off-Cycle Request Charges Transactions

You also can create a new billing account while you are working in several other records, such as the Customer record and the Subscription record. To create a new billing account from the Customer record, on the **Billing Account** subtab, click **-New-**. Follow the instructions above. To create a new billing account while creating a new subscriptions, see [Creating a New Stand-Alone Subscription](#).

## Viewing Records Related to the Billing Account

You can view a list of records related to the billing account, such as subscription records, charge records, and billing schedules.

### To view records related to the billing account:

1. Go to the Billing Account page.
2. Click the **Related Records** tab.

## Billing Addresses

The Bill-To Select list on the billing account determines the billing address for invoices, cash sales, and credit memos associated with the billing account. If you do not select a billing address, NetSuite uses the default billing address defined on the customer record. If needed, you can change the billing address on the billing account.

Customers with multiple subscriptions on the same billing account receive one single invoice based on the billing schedule defined by the billing account.

The Ship-to Select list on the billing account works the same as the Bill-To select. This determines the shipping address. If no shipping address is selected, NetSuite uses the default shipping address defined on the customer record.

## Billing Schedules

Billing accounts work with charge-based billing schedules. Charge-based billing schedules support two types of recurrence patterns:

- Fixed bill date billing schedules indicate that the recurring invoice is generated on a fixed-date basis. Billing is based on the selected bill date, frequency, recurrence frequency, and Repeat Every settings. Fixed date billing schedules work best when you want to invoice customers on a specific day of the month, or any other specific frequency.
- Anniversary bill date billing schedules indicate that the recurring invoice is generated on an anniversary date basis. Billing is based on the billing account start date, frequency, recurrence frequency, and Repeat Every settings. Use the anniversary bill date billing schedule to invoice the customer on the same date each year as the service start date.

If you define a billing schedule with a date that does not exist in every month, the billing schedule adjusts accordingly. For example, if you specify billing on the 31st, for months that have fewer days, NetSuite bills on the last day of that month. February 28th and April 30th are two examples.

## Changing the Billing Date for Future Invoices to a Billing Account

You must have edit permissions for billing accounts to change the next bill cycle date.

### To change the billing date for future invoices to a billing account:

1. Go to the saved a billing account with an associated billing schedule.
2. From the Next Bill Cycle Date list, change the date for the next bill cycle.

The list shows the possible dates for which you can advance the billing date up to a maximum of 10 billing cycles.

You can move the next bill date backward, but only to the next bill date after the creation cycle of the latest recurring invoice. For example, you prematurely generate a bill run and need to perform a bill run for the first bill cycle date. You can accomplish this by moving the next bill date backward.

## Creating Recurring Invoices for Subscriptions with Billing Accounts

You can create subscription invoices that automatically recur with a designated frequency.

Usage must be billed in arrears, but recurring and one-time charges can be billed in advance. For example, bill from the electric company is based on your usage and is billed in arrears. To generate revenue as early as possible, most SuiteBilling customers bill recurring charges in advance. For example,

cable and internet service companies bill their customers for the coming month at the beginning of the billing cycle. Charges begin the day the service is activated.

There are several ways to create a recurring invoice with a billing account.

### To create a recurring invoice with a billing account:

1. Go to Lists > Relationships > Billing Accounts.
2. Click **View** next to the billing account for which you want to create a recurring invoice.
3. Click the **Next Bill** button.



**Note:** You also can create a recurring invoice by clicking the **Next Bill** button from the **Billing Account** subtab of the customer record.

The Create Invoice page opens with the **Recurring Bill** box checked. The invoice date defaults to the next bill date on the billing account.

In addition, you can create a recurring invoice the following ways:

- Invoicing billable customers from the Invoice Customers page
- Using Billing Operations (Process Billing Operations and Schedule Billing Operations)
- Creating invoices from the Invoice page

For each customer, NetSuite reviews all billing accounts eligible for a bill run. NetSuite creates one recurring invoice per billing account per period when there are charges. To send an invoice to your customers, you can generate a statement of the bill run.

NetSuite uses the next bill cycle date on the billing account to determine the invoice date. When NetSuite creates the invoice, the Last Bill Cycle Date and the Last Bill Date change to the date of the invoice. The Next Bill Cycle Date advances to the next billing date. If you delete the invoice, the Last Bill Cycle Date and the Next Bill Cycle Date return to the date they were before you generated the invoice. Also, the Last Bill Date changes to the date of the previous bill.

## Using Billing Accounts with Off-Cycle Invoices

Use off-cycle invoices to bill for charges that require immediate (mid-cycle) billing. NetSuite uses the date on which you manually enter the invoice as the invoice date.

### To create an off-cycle invoice using a billing account:

1. Go to Transactions > Sales > Create Invoices > List.
2. Click the Edit link for the invoice for which you want to create an off-cycle invoice.
3. Clear the **Recurring Bill** box.

When you create the invoice, the Last Bill Date updates to the date of the invoice. When you delete an invoice, the Last Bill Date changes to the date of the previous bill.

## Renaming SuiteBilling Records

You can rename the following SuiteBilling records to better suit your specific business needs:

- Subscription

- Subscription Change Order
- Subscription Plan

When you rename a record, the new name replaces all instances of the original name in the NetSuite user interface. For example, after you change the name "subscription" to "contract," you navigate to Transactions > Contracts > Create Contracts instead of Transactions > Subscriptions > Create Subscriptions . The renamed record also appears in search links, search criteria, and search results. It does not appear in the Help Center search results or in help topics. The renamed record also appears in reports, KPIs, error messages, and alert messages.

 **Note:** If you change the name "subscription" to "contract," the feature name does not change at Setup > Company > Enable Features. The same is true with any information or error message on the Enable Features page.

If you rename the subscription record, NetSuite also renames the subscription item record and the subscription term record. For example, you change the name "subscription" to "contract." The subscription item record changes to contract item and the subscription term record changes to contract term. If you rename either the subscription change order record or the subscription plan record, it does not impact the name of any additional records.

For specific details and more information about renaming records in NetSuite, see the help topic [Renaming Records and Transactions](#). To learn more, view the video:



[Renaming Billing Records](#)

### To rename a SuiteBilling record:

1. Go to Setup > Company > Rename Records/Transactions.
2. Edit the names of the subscription billing records. For example, in the **Name for Subscription** field, you can change **Subscription** to **Contract**.
3. When you have finished renaming records, click **Save**.

### To rename a SuiteBilling record in multiple languages:

1. If you use different languages in your account, set your language preference:
  - a. Go to Home > Set Preferences.
  - b. In the Localization section, from the Search Sorting list, select your language preference.
2. Repeat the procedure [To rename a SuiteBilling record](#): for each language in your account.

# SuiteBilling Management

This chapter describes tasks that you may perform repeatedly to manage your subscription billing.

SuiteBilling subscriptions have an automated approval process that starts with the creation of each of the subscription's required parts:

- Subscription items in a subscription plan
- Pricing, including the price plan and price book
- Billing account

For instructions on how to create the prerequisite parts of a subscription, see:

- [Creating Items for Subscription Plans](#)
- [Creating Subscription Plans](#)
- [Creating Price Plans](#)
- [Creating Price Books](#)
- [Creating Billing Accounts](#)

After you have created all of the parts that make up a subscription, you can create a subscription.

To move the subscription through the approval process, you change its status using change orders. For high-level overview diagrams of creating and renewing subscriptions, see [SuiteBilling Process Diagrams](#). For more information on change orders, see [SuiteBilling Change Orders](#).

When you create a subscription, its original status is **Draft** or **Not Included**. In **Draft** status, the subscription has no revenue impact and no charge impacts. After approval, the subscription status changes to **Pending Activation**. When the subscription is activated, its status changes to **Active**. You also can manually change the status of individual subscription items.

As the subscription progresses through its lifecycle, you can make changes to the subscription or its components with change orders. If a customer decides to temporarily suspend their subscription, you can create a **Suspend** change order. When the customer is ready to restart their subscription, you can create a **Reactivate** change order.

You also use change orders to modify pricing.

You can create a subscription from a sales opportunity, estimate, or sales order. Create subscriptions based on:

- Customer
- Billing Account
- Subscription Plan
- Price Book
- Start Date
- Subscription Term
- Subscription Renewal

In addition, you can create a new stand-alone subscription without using an existing transaction or subscription.

## Creating a New Stand-Alone Subscription

A **stand-alone subscription** is a new subscription that you create on the Create Subscriptions page, rather than from another transaction such as a sales order.

You can activate a subscription only after you create and save the subscription's required components. For instructions on how to create the prerequisite parts of a subscription, see the following topics:

- [Creating Items for Subscription Plans](#)
- [Creating Subscription Plans](#)
- [Creating Price Books](#)
- [Creating Billing Accounts](#)

To see a high-level overview diagram of the process for creating a subscription plan, see [Subscription Plan Creation](#).

### To create a new stand-alone subscription:

1. Go to Transactions > Subscriptions > Create Subscriptions.
2. Select from the list or enter values for the following fields in any order:
  - Indicate the customer for whom you are creating the subscription:
    - From the **Customer** field, select an active customer from the list.
    - To create a new customer, click the plus icon next to the **Customer** field and select **-New-**.

You can select any active customer, regardless of whether they have a billing account. When you select a customer, the **Subsidiary** and **Currency** fields contain the values you specified for those fields for that customer. When you select values for the **Customer** and **Subsidiary** and leave **Currency** blank, the **Billing Account** field shows the billing accounts that are available for that customer.



**Note:** Depending on the order in which you select the values for the **Customer**, **Subsidiary**, **Currency**, and **Billing Account** fields, NetSuite fills the values of fields dependent on the Subscription and Billing Account records.

- You can set a subsidiary to use in a new billing account before creating it. If the customer has subsidiaries, indicate them as follows:
  - From the list in the **Subsidiary** field, select an active subsidiary. NetSuite populates the **Subsidiary** fields of the Subscription record with the field's values from the billing account.
  - You can change the subsidiary from the primary subsidiary to a secondary subsidiary. The subscription must be in Draft or Pending Activation status to change the subsidiary.
  - To create a new subsidiary, click the plus icon and select **-New-**.
- You can set a currency to use in a new billing account before creating it. Indicate the currency as follows:
  - From the list in the **Currency** field, select an active currency. NetSuite populates the **Currency** fields of the Subscription record with the field's values from the billing account.
  - You can change the currency when the subscription is in Draft or Pending Activation status.
  - To create a new currency, click the plus icon and select **-New-**.
- (Optional) You can create a subscription without a billing account until you activate the subscription. You can select a billing account from the list in the **Billing Account** field, or to create a new billing account, click the plus icon and select **-New-**. To create a new billing account, follow the instructions in [Creating Billing Accounts](#).

On the Subscription record, the **Customer**, **Subsidiary**, and **Currency** fields are required for creating a billing account. When you have selected a customer on the Subscription record, the **Billing Account** field contains the customer's default billing account. When all lines in the subscription are in Draft or Pending Activation status, you can change the billing account.

to another billing account for the same customer. When you select a billing account, the **Subsidiary** and **Currency** fields are read only and contain the values for those fields from the billing account.

The billing account is not required on default forms. Any automation or import templates work when you set a billing account value when you create the subscription.

You cannot activate a subscription until you select a billing account that matches the values of the **Customer**, **Currency**, and **Subsidiary** fields on the Subscription record.



**Note:** To ensure accurate data, select the **Billing Account**, **Customer**, or **Subsidiary** before selecting the subscription plan and price book

3. From the **Subscription Plan** list, select a subscription plan. The **Subscription Plan** field lists the subscription plans for the selected billing account. If you have not selected a billing account, you must select values for the **Currency** and **Subsidiary** fields to select a subscription plan.  
Only subscription plans with price books in the currency that corresponds to the selected price book currency are available to select.
4. If required, from the **Price Book** list, choose a different price book.
5. From the **Initial Term** list, select a term, or select **-New-** to define a different term.
6. In the **Start Date** field, enter or select a start date for the subscription. The default value is today's date.
7. You can define the **Class**, **Department**, and **Location** fields on the Subscription records in the Draft or Pending Activation status. These fields, located in the Plan and Term section, are useful for reporting. The values of these fields are automatically populated in the following order:
  1. If the selected subscription plan has values in any of the **Class**, **Department**, and **Location** fields, the field on the Subscription record shows its subscription plan value.
  2. If a field does not have a value in the subscription plan, NetSuite checks the billing account and the field on the Subscription record shows its billing account value, if any.
  3. If neither the subscription plan nor the billing account have values in these fields, they are empty on the Subscription record.

If the Subscription or Subscription Line record is in Draft status, you can change the values of the **Class**, **Department**, and **Location** fields. When the Subscription or Subscription Line record is in Pending Activation or Active status, you cannot change the values of the **Class**, **Department**, and **Location** fields.

The Lines tab of the Subscription record also has **Class**, **Department**, and **Location** columns.

8. If required, define one or more add-on items. See [Adding Line Items to an Existing Subscription](#).
9. If Revenue Recognition is enabled, set revenue recognition options. To enable Revenue Recognition, see the help topic [Enabling the Revenue Recognition Feature](#).



**Note:** After you click **Save**, you cannot change the subscription's customer, currency, or subsidiary. In this case, you must create a new subscription.

10. Click **Save**.

If a billing account that you expect to process during billing operations does not process, run billing operations a second time or schedule billing operations for a convenient time for your business. Reasons why a billing account is not processed during billing operations include:

- A charge is created after billing operations begins processing charges.
- A single billing operation can process up to 100,000 processing lines.

Running a subsequent billing operation may process any remaining invoices. For more information on Billing Operations, see [Billing Operations](#).

After you have created the subscription, you must individually activate each line item. For more information, see [Activating Subscription Line Items](#).

After creating the subscription, you can either associate it with a sales transaction or activate it.

## Editing a Draft Subscription

You can edit a Subscription record that is in Draft status.

### To edit a draft subscription:

1. Open the Subscription record that is in Draft status if it is not already open.
2. Click **Edit**.
3. Change the values of the field or fields.

**Note:** If you change the billing account while editing the subscription, the subscription plan, price plan, and term will not reset.

4. Click **Save**.

**Note:** After you save a draft subscription, you cannot change its customer, currency, or subsidiary. In this case, you must create a new subscription.

## Creating a Subscription from a Transaction

After setting up the billing account and subscription plan, you can create a subscription from an opportunity, estimate, or sales order.

- If you convert an opportunity to an estimate, the subscription carries over to the estimate.
- If you convert an estimate to a sales order, the subscription carries over to the sales order.

To create a subscription from a transaction, you can use either one of these two approaches:

- Add a saved subscription to the transaction. Use this approach if you intend to send the sales transaction to the customer.
- Create a new sales transaction. Use this approach if you intend to use the transaction to synchronize with sales or to move lines to Pending Activation.

### To create a subscription from a transaction:

1. On the transaction record, select the customer for which you want to create a subscription.

**Tip:** You can use the Quick Search portlet or Recent Records portlet to find the desired customer record.

2. Complete any required fields based on your transaction type.

- On the **Items** subtab, from the **Subscription** list, select the subscription you want to add, or select **New** to create a new subscription.

**Tip:** Select a subscription to add if you plan to send the sales transaction to the customer. Select New if you plan to use the transaction to synchronize with sales or move lines to Pending Activation. See [Updating a Subscription on a Transaction](#) and [Activating Subscription Line Items](#).



**Note:** Only draft subscriptions are available to add to sales transactions.

- If you select an existing subscription, click **Save**.
- If you select **-New-**:
  - Enter a **Subscription Name**.
  - From the **Billing Account** list, select the billing account you want to use.



**Note:** To ensure accurate data, select the **Billing Account**, **Customer**, or **Subsidiary** before selecting the subscription plan and price book

- From the **Subscription Plan** list, select the subscription plan you want to use or select **New** to create a new subscription plan.
- From the **Initial Term** list, select an initial term for the subscription.
- Enter or select a **Start Date**.

- Click **Add** to add the subscription to the transaction.
- Click **Save**.

If you enabled the [Setting Optional Preferences](#) preference, approve the sales order. NetSuite automatically moves all subscription lines to Pending Activation.

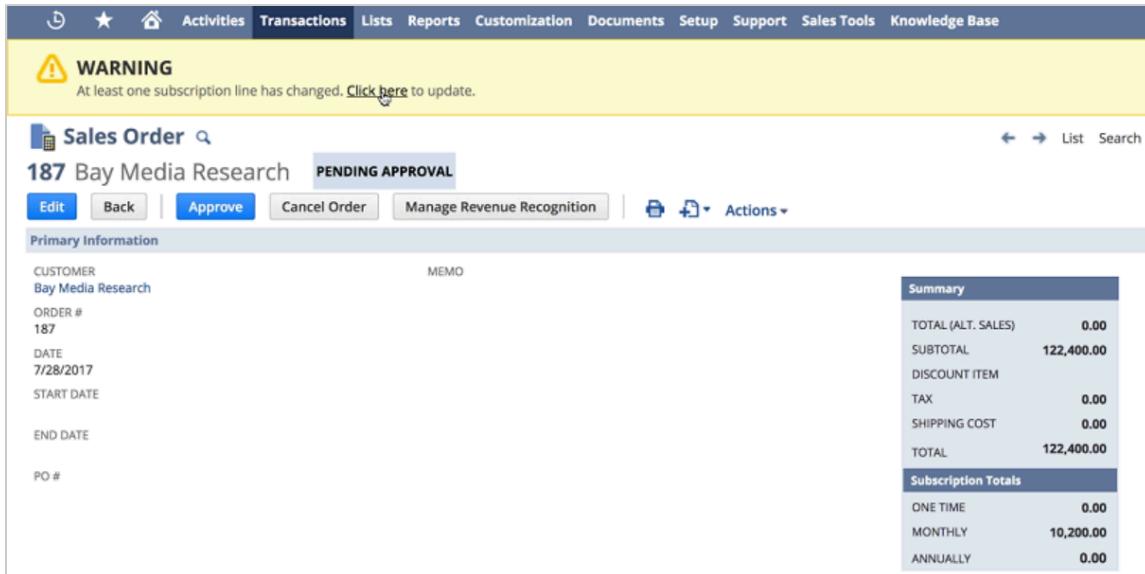
For information on creating a new subscription plan, see [Creating Subscription Plans](#).

## Updating a Subscription on a Transaction

You can update the associated sales transactions with any updates made to a subscription. Updating this information ensures that the transaction reflects the correct subscription information and totals.

Subscriptions must be in the **Draft** status. When a subscription enters the **Pending Activation** status, you can no longer update the associated sales transactions with any updates made to the subscription.

When a sales transaction needs updating, NetSuite displays a warning message in the top left corner of the screen.



### To update the subscription information in a sales transaction:

1. Go to the sales transaction associated with the modified subscription.
2. In the warning message, click the link to update the subscription information.
3. To save the updated sales transaction, click **Save**.

## Using Item Records

### Activating Subscription Line Items

After creating a new subscription or adding items to an existing subscription, you need to individually activate all items. To activate an item, it must include all information for the subscription, such as pricing, quantity, and so on. After an item with all the information relevant to a subscription is activated, it becomes a **line item**.

If a subscription line item is not activated, it is not included with the subscription when the subscription is activated. Required subscription line items must be activated to activate the subscription.

**Note:** Steps 1-3 in the following procedure pertain only to stand-alone subscriptions and Add-On Items added mid-subscription term.

### To activate subscription line items:

1. Go to Transactions > Subscriptions > Create Subscriptions > List.

- For the subscription you want to edit, click the subscription name or **Edit**.



**Tip:** To find the subscription, you can:

- Look in the **Recent Records** tab for recently completed subscriptions.
- Enter the subscription name or subscription ID into the Global Search field.

- On the **Lines** subtab of the Subscription page, for each subscription line item:
  - Click **Edit**.
  - Change the status to **Pending Activation**.
  - Click **Save**.
- Click **Activate**.
- On the Subscription Change Order – Activate page:
  - Edit the Effective Date if desired.
  - For each line item, check the **Apply** box.
  - Click **Save**.

## Adding Line Items to an Existing Subscription

Enable the Add-On Items feature if you want to add items to an existing subscription. To enable the Add-On Items feature, see [Enabling SuiteBilling Features](#).

You can add additional line items to a subscription at any time, even after the start of the subscription. On the **Lines** subtab of the subscription, in the Catalog Type column, these new line items are identified as Add-On. After you add an add-on item to the subscription, that item follows the same workflow as any subscription line item.

You can add multiple subscription line items at one time.

Add-on items are optional. You set the price of each add-on item as you create it. There is no predefined pricing. You can add any service or non-inventory item that is defined in the subscription catalog to a subscription. Once saved, an add-on item is treated the same as any subscription line item and can be moved to **Pending Activation**, then activated.

You can prorate recurring add-on items but not one-time or usage line items.

You can prorate charges by month or day.



**Note:** You can modify the Prorate By setting only when the recurring subscription item status is **Draft**.

**Example:** The following table shows the difference in charges with Proration By Day enabled versus Proration By Month for a 12-month contract:

	Jan.	Feb.	Mar.	Apr.	May	June	July	Aug.	Sept.	Oct.	Nov.	Dec.
Days in Month	31	28	31	30	31	30	31	31	30	31	30	31
Monthly Charge with Prorate By Day Enabled	1,019.18	920.55	1,019.18	1,019.18	1,019.18	1,019.18	1,019.18	1,019.18	1,019.18	1,019.18	1,019.18	1,019.18

	Jan.	Feb.	Mar.	Apr.	May	June	July	Aug.	Sept.	Oct.	Nov.	Dec.
Monthly Charge with Prorate By Month Enabled	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000

For more information and an example, see the help topic [Proration Options](#).

Included quantity multipliers are not permitted for usage-based add-on items. You cannot define recurring add-on items as an included quantity multiplier for any usage line items. For more information about pricing, see [Creating Price Books](#).

### To add a new line item to a subscription:

1. Go to Transactions > Subscriptions > Create Subscriptions > List.
2. For the subscription you want to edit, click the subscription name.
3. On the Subscription record, click **Edit**.

A new item with the Catalog Type set to Add-On appears at the bottom of the page.

Subscription Change Orders											
Lines		Pricing	Renewal	Subscription		Related Records	System Information				
INCLUDE	LINE NUMBER	ITEM*	SUBSCRIPTION LINE TYPE*	CATALOG TYPE*	BILLING MODE*	STATUS*	START DATE	RECURRANCE START DATE	END DATE	TERMIN DATE	
Yes	1	Initial Setup	One Time	Required	In Advance	Active	11/6/2018		11/6/2018		
Yes	2	Usage	Usage	Required	In Arrears	Active	11/6/2018	12/1/2018	11/5/2019		
Yes	3	Quarterly Recurring Subscription Service	Recurring	Required	In Advance	Active	11/6/2018	12/1/2018	11/5/2019		
<input checked="" type="checkbox"/>			Add-On		Draft						
<input type="button" value="✓ Add"/> <input type="button" value="✗ Cancel"/> <input type="button" value="Remove"/>											

4. On the **Lines** subtab, complete the information for the add-on item:
  - a. In the **Item** column, select the item you want to add.
  - b. In the **Subscription Line Type** column, select the appropriate line type: One Time, Recurring, or Usage.
  - c. Review the following fields and update the values if needed:
    - **Include** – The box is checked by default to include this item in the subscription.
    - **Billing Mode** – From the list, select the required billing mode: In Advance or In Arrears.
    - **Prorate Start Date** and **Prorate End Date** – Click in the field to display the check box and turn proration on or off as required.
    - **Include in Renewal Subscription** – Click in the field to display the check box. Check the box to include this item in any future renewals.
    - **Revenue Recognition Option** – Select one of the following options:
      - **One Time** – Select this option if available to recognize revenue one time at the beginning of the term (for example, for setup fees).
      - **Over subscription term** – Select this option to recognize revenue over the term of the subscription.
  - d. Click **Add**.
5. On the **Pricing** subtab, add pricing information for the new line item:
  - a. In the **Charge Frequency** column, for recurring or usage line items, select the appropriate charge frequency: **monthly**, **weekly**, or **annually**.
  - b. In the **Repeat Every** column, select the appropriate option.

**Tip:** For a monthly charge frequency, in the Charge Frequency column, select Monthly, and in the Repeat Every column, select 1. For a quarterly charge frequency, in the Charge Frequency column, select Monthly, and in the Repeat Every column, select 3.

The **Repeat Every** column on the subscription pricing line is an integer field. **Repeat Every** can be any positive integer, so you can create non-standard charge frequencies as needed. **Repeat Every** must be a positive integer between 1 and 120 for Weekly charge frequency, between 1 and 120 for Monthly charge frequency, or between 1 and 10 for Annual charge frequency.

- c. Click the edit icon.
- d. In the **Pricing Option** column, select **Fixed Amount** or **Rate**.
- e. In the **Value** column, enter the price.
- f. In the **Prorate By** column, if desired, change the default setting of Day to Month.

For example, a 12-month subscription costs \$12,000 and is billed monthly, but only 11 months are active. With the default setting of Prorate By Day, the monthly charge is \$1098.08. If you choose Prorate By Month instead, the charge for each month is rounded to \$1,000.

- g. Complete any additional optional fields.
- h. Click **Save**.

For more information on pricing, see [Creating Price Plans](#).

6. Repeat these steps for each additional add-on item.
7. When you finish adding items, click **Save**.

## Charges Based on a Fixed Amount

If you negotiate on the total amount over the subscription term, you can enter a total amount for a subscription line and a charge frequency to calculate recurring amounts, instead of creating a price plan and quantity for customers that A total amount for a subscription line can now be entered instead of a price plan

You set a total value for a subscription line in the **Total Interval Value** field. NetSuite generates the corresponding price plan when you save the subscription.

The screenshot shows the NetSuite Subscription creation interface. At the top, there are buttons for Save, Cancel, List, Search, Customize, and More. The Primary Information section includes fields for Subscription ID (To Be Generated), Subcription Name (To Be Generated), Customer (2 Continuum Software), Billing Account (2: 13 13), Status (Draft), Subsidiary (Parent Company), and Originating Sales Order. The Plan and Term section includes fields for Subscription Plan (SMB Konverge Management & Storage), Start Date (1/1/2021), End Date (12/31/2021), Next Renewal Start Date, Class (Licensing), Location (Main), Price Book (SMB USD Annual), and Initial Term (1 Year Term). The Billing Account section includes fields for Billing Schedule (Monthly The First Day of the Month), Start Date (1/1/2021), Last Bill Cycle Date (1/1/2021), Next Bill Cycle Date (2/1/2021), Last Bill Date (11/20/2020), and Location (Main). The Pricing tab is selected, showing a table with one row for a Set-up Fee. The row details are: Line Number 1, Item Set-up Fee, Line Type One Time, Start Date 1/1/2021, Interval Status Draft, Catalog Type Required, Included Quantity Multiplier 1, Quantity 1, Price Above 0: \$1,000.00, Edit, Discount, Total Interval Value 1000.00, Recurring Amount, Charge Frequency One Time, Repeat Every 0, and Prorate By.

LINE NUMBER	ITEM	LINE TYPE	START DATE	INTERVAL STATUS	CATALOG TYPE	INCLUDED QUANTITY MULTIPLIER	QUANTITY	PRICE	EDIT	DISCOUNT	TOTAL INTERVAL VALUE	RECURRING AMOUNT	CHARGE FREQUENCY*	REPEAT EVERY*	PRORATE BY
1	Set-up Fee	One Time	1/1/2021	Draft	Required	1	1	Above 0: \$1,000.00			1000.00		One Time	0	

The Total Interval Value takes into consideration the following parameters when calculating a price plan value:

- repeat every
- charge frequency
- prorate start
- prorate end
- discount
- subscription term

When the total interval value is populated, it clears the values in the price plan. Re-entering a value for the price plan clears the total interval value. Either the total interval value or the price plan can be set for a given subscription line.

You can enter a total interval value only when a subscription line is in Draft status. You can't use total interval values on usage items or evergreen terms. When a total interval value is entered, generated price plans are always single-tiered price plans.

## Modifying the Class, Department, or Location of a Subscription Line

To modify the Class, Department, or Location field of a subscription line, the line must be in the Draft status. If the subscription line has any change orders, void them. Manually change the subscription line status to Draft, and then change the Class, Department, or Location field.

To change a subscription line back to Draft status when a sales order is attached to the subscription, you should change the sales order from the Approved status to Draft status. If you are changing only one line, you do not have to void all change orders; you need to move only the line back to Draft status.

## Changing a Subscription Line's Status on a Subscription Record

When a subscription record is in **Edit** mode, you can change the subscription line status from **Draft** to **Pending Activation** or from **Pending Activation** to **Draft**. You can use Sales Orders to move subscription lines to **Pending Activation** to remove them from your subscription creation flow, reducing the number of records you must interact with.

## Removing an Add-On Item

After an add-on item is added to a subscription, you cannot delete it. Like other line items, the add-on item is part of the subscription. If you need to omit the add-on item from the subscription, change the add-on item's status to **Not Included**.

### To remove an add-on item from a subscription:

1. Go to Transactions > Subscriptions > Create Subscriptions > List.
2. For the subscription you want to edit, click the subscription name or **Edit**.

3. For the add-on item you want to remove, clear the box in the **Include** column.
4. Click **Save**.

## Activating a Subscription

To activate a subscription, you must first activate all its subscription line items. After all subscription line items are activated, the **Activate** button appears. After you activate each line item, an activation change order activates the subscription. For more information on change orders, see [SuiteBilling Change Orders](#).

Change orders are date-sensitive.

- NetSuite uses the effective date to create future charges.
- You can choose to have the subscription start date set when the first activation change order is created. For details, see [Optional Preferences for SuiteBilling](#).

You can create saved searches to make it easy to find subscriptions. You also can search for subscriptions by name or ID with Global Search.

### To activate a subscription:

1. Go to Transactions > Subscriptions > Create Subscriptions > List.
2. For the subscription you want to activate, click the subscription name to open the subscription in edit mode or click **Edit**.
3. On the **Lines** subtab, click **Edit** next to the line items you want to activate.
4. Change the **Status** field from **Draft** to **Pending Activation**.
5. Click **Save**.
6. Repeat these steps for each line item.
7. At the top of the page, click **Activate**.
8. On the Subscription Change Order – Activate page, check the **Apply** box for the items you want to activate. The status of every line item must be **Pending Activation**.
9. Click **Save**.

Subscription Change Order - Activate											List	Search			
Primary Information															
NUMBER	CO_1048	REQUESTER	Mark Wheel												
EFFECTIVE DATE	2/23/2017	MEMO													
<input checked="" type="checkbox"/> REQUEST OFF-CYCLE INVOICE FOR ADVANCE CHARGES															
Lines	Subscription	System Notes													
INCLUDE	LINE NUMBER	STATUS	NEW STATUS	ITEM	PRICE PLAN	QUANTITY	DISCOUNT	START DATE	END DATE						
Yes	1	Pending Activation	Active	Item_S2_R01_1485798867418	Above 0:	\$100.00	1	2/23/2017	2/23/2017						
Yes	2	Pending Activation	Active	Item_S2_R01_1485799022061	Above 0:	\$150.00	1	2/23/2017	2/23/2017						
	3	Pending Activation		Item_S2_R01_1485798996883	Above 0:	\$300.00	1	2/23/2017	2/23/2017						

## Creating Usage Records

**Usage** defines the charge for a subscription line item that can be consumed, such as money, time, software licenses, or internet data. Some subscription line items include an amount of usage. For example, if you sell mobile phone service plans, your subscription may have a line item for a monthly

fee. The monthly fee includes a specific amount of minutes or data in your customer's plan. When this included amount is exceeded, additional charges may apply. In NetSuite, you account for the usage amount included in a subscription when creating the subscription plan.

You also can multiply the amount of usage included in a subscription. For example, there are a total of five people in one household that are part of a cell phone plan. When you create the price book, you can multiply the included usage for the data item by five. This gives each person the same amount of included usage. For instructions, see [Creating Price Books](#).

One-time and recurring charges often bill in advance, whereas usage charges bill in arrears. Usage cannot be billed until the item has been used.

The Billing Specialist role without the accounting list permission can create usage with a usage record.

You cannot delete a usage charge after a rating run but you can void it. For more information, see [Voiding a Usage Record](#).

### To create a usage record:

1. Go to Transactions > Subscriptions > Create Usage.

The screenshot shows the 'Usage' creation screen in NetSuite. At the top, there are buttons for 'Save' (highlighted in blue), 'Cancel', and 'Reset'. Below this is a section titled 'Primary Information' with a rich text editor for 'MEMO'. To the right of the memo are dropdown menus for 'CUSTOMER', 'ITEM', 'SUBSCRIPTION PLAN', and 'SUBSCRIPTION'. Further to the right are fields for 'QUANTITY' (marked with a red asterisk) and 'USAGE DATE' (also marked with a red asterisk). At the bottom of the screen are three buttons: 'Save' (highlighted in blue), 'Cancel', and 'Reset'.

2. (Optional) From the **Customer** list, select a customer.
3. (Optional) From the **Item** list, select the item for which you want to associate usage.
4. (Optional) From the **Subscription Plan** list, select the subscription plan to which you want to add usage.
5. From the **Subscription** list, select a subscription.
6. From the **Subscription Line** list, select a subscription line item.
7. In the **Quantity** field, enter the quantity.
8. In the **Usage Date** field, enter or select a date.
9. Click **Save**.



**Note:** The usage record does not support user event scripts.

## Voiding a Usage Record

You can void any usage record in the Active or Suspended state. You can void a usage record at any time, even when it has already been rated or invoiced. If a usage record has been invoiced or revenue has been recognized, voiding it generates any required reversing charges or revenue. Voiding usage records eliminates the need for manually entering credits and corrections for erroneously rated usage.

You cannot delete a voided usage record. When a usage record is voided, it permanently becomes a view-only record.

If a usage is voided and another usage is created before rating or revenue updates, NetSuite uses the net quantity change to generate the charge amount.

You can create a termination change order even if a usage record exists after the termination effective date. NetSuite automatically voids usage records created after a termination change order.

### To void a usage record:

1. Go to Transactions > Subscriptions > Create Usage > List.
2. Click the number of the usage line you wish to void.



**Important:** The subscription line for the usage record must be in the Active or Suspended state. If the subscription line status is Terminated, Closed, Pending Activation or Draft, you cannot void the usage record.

3. On the Usage record, click **Void**.

The screenshot shows the 'Usage' transaction screen. At the top, there are buttons for 'Save', 'Cancel', and 'Void'. The 'Void' button is highlighted with a red box. Below the buttons is a section titled 'Primary Information' with a memo field and various input fields. To the right of the memo field, there is a table with customer and subscription details. Underneath this is a 'Charges' section with a table showing no records.

CUSTOMER	SUBSCRIPTION LINE
14 Intelliware Systems	Records Management Plus - 8/1/2020: 3
ITEM	QUANTITY
File Storage	1
SUBSCRIPTION PLAN	USAGE DATE
Records Management Plus	8/1/2020
SUBSCRIPTION	USAGE STATUS
Records Management Plus - 8/1/2020	Active

#	CHARGE ID	ITEM	CHARGE TYPE	SERVICE START DATE	SERVICE END DATE	DATE	AMOUNT	CURRENCY
No records to show.								

4. Click **Save**.

## Managing Rating Runs

**Rating** is the process of creating charges for subscription line items. NetSuite calculates charges for a subscription using the information on its line items, such as:

- The quantities, pricing, and usage shown in the price plan and price book
- The effective date of the subscription
- The status of the subscription and its line items

You use a change order to set and change the status of subscription line items.

Certain tasks you perform on a subscription line item that affect NetSuite charge calculations (for example, renewing a line item) are called **ratable events**. Examples of actions that cause ratable events include:

- Activating a line item
- Reactivating a line item
- Renewing a line item
- Suspending a line item
- Terminating a line item
- Modifying quantities and rates on a line item
- Saving a change order

## Automatic Rating Runs

When you save an action that causes a ratable event, NetSuite automatically starts calculating the charges specified in the subscription. This charge calculation process is called a **rating run**. You also can generate charges on demand, or you can schedule them with the current billing operations. You can run rating from the Billing Operation page.

When you save any change order, a ratable event occurs, which starts a rating run. Only one change order of a specific type can occur per day on a subscription line item. For example, you modify the price of a line item using a modify pricing change order, then realize you also need to add a discount. You must wait 24 hours to create another modify pricing change order for the same line.

The same is true for quantity changes. If you submit a change order that changes the quantity of a line item, you must wait at least 24 hours before submitting another.

Also, when you make changes to a draft subscription that alter pricing, an automatic rating run starts. This rating run does not generate charges but can determine things like the total subscription amount when synchronizing a draft subscription to a sales order. It is important to notice that the rating run is an asynchronous server process, so the most up-to-date result may not be available the instant the draft subscription is saved. To get the most up-to-date data, you can resynchronize the sales order.

**Use Case Example:** Consider a subscription line item, cell phone service, that starts on January 1, 2029 and ends on December 31, 2029.

1. You create a change order that changes the subscription start date to February 1, 2029. When you save the activation change order record, rating occurs, and NetSuite creates charges for February, March, and so on to the end of the subscription term, reflecting the date change.
2. On May 1, you create a change order to suspend the cell phone service subscription line item. When you save the change order, rating occurs. NetSuite starts a rating run that calculates the charges from May 1 to the end of the subscription term, and places the charges **On Hold**. If earlier charges have been invoiced, negative charges could occur.

## Manually Creating Subscription Charges with Billing Operations

You can manually start a rating run to create charges at any time. You might manually use billing operations because of a price change related to a subscription. Another example is a customer who decides to cancel their subscription early, and they are entitled to a partial refund. In this example, billing operations is used to create a credit memo. A subscription change order causes the need to manually create charges related to SuiteBilling. For more information on Billing Operations, see [Billing Operations](#).

## To manually calculate subscription charges with Billing Operations:

1. Go to Transactions > Billing > Process Billing Operations.
2. On the Billing Operation page, check the **Rating Run** box, then click **Run**.

**Billing Operation**

**Primary Information**

DESCRIPTION  
NEXT BILL RUN AS OF DATE \*  
7/5/2021

**Billing Operation Options**

RATING RUN  
 CREDIT MEMO RUN  
 BILL RUN  
 BILL RUN (OFF-CYCLE ONLY)  
 GROUP INVOICES  
 DO NOT APPLY GROUPING FOR BILL RUN

**Customer**

SUBSIDIARY \*  
Parent Company  
CUSTOMER/GROUP \*  
1 Kittyville Pies and Scones

ALL CUSTOMERS

**Billables**

CHARGES  
 TIME  
 EXPENSES  
 ITEMS  
 SALES ORDERS

SALES ORDER SAVED SEARCH

**Accounting**

CREDIT CARD APPROVED

**Communication**

TO BE PRINTED  
Respect Customer Preference  
TO BE EMAILED  
Respect Customer Preference  
TO BE FAXED  
Respect Customer Preference

## Creating Scheduled Subscription Charges Using Billing Operations

You can schedule a rating run to create subscription charges for a specific day and time, such as the charges for a subscription that is billed monthly. For more information on Billing Operations, see, [Billing Operations](#).

## To create scheduled subscription charges using Billing Operations:

1. Go to Transactions > Billing > Schedule Billing Operations.
2. Under Primary Information, enter a name and description for the bill run schedule. If you do not want to make the bill run schedule active right away, check the **Inactive** box.
3. Under Schedule, setup a schedule for the bill run.
4. Under Billing Operation Options, check the **Bill Run** box.
5. Under Billables, select the types of transactions for which you want to create bills.
6. Under Accounting, indicate whether the credit card is approved.
7. Under Communication, indicate whether bills should be printed, emailed, or faxed.
8. Click **Save**.

## Applying Discounts to Subscriptions

You can apply discounts at both the subscription plan and the subscription line item levels. You can apply discounts to one-time, usage, and recurring subscription line items. Discounts apply to all line types and all pricing options.

You can apply a discount to a subscription only when it is in the **Draft** state. After a subscription is activated, you cannot apply a discount. To apply a discount to a draft subscription, edit the discount field in-line. After a subscription is in the **Active** state, you can apply discounts only using a Modify Pricing Change Order.

Charges and invoices dynamically display the net charge amount after discount. Therefore, there are no associated discount line items on an invoice.

### To apply a discount to a subscription:

1. Go to Transactions > Subscriptions > Create Subscriptions > List.
2. For the draft subscription to which you want to add a discount, click the subscription name or **View**.
3. On the subscription record, click **Edit**.
4. For the subscription line item to which you want to add a discount, enter the discount:
  - You can enter a discount between 0 and 100.
  - To enter a percentage discount, include the percent sign (%), for example, 50%.
  - To enter a flat rate discount, do not include any notation, for example, 50.
  - Negative discounts and discounts above 100% are not permitted.
5. To add discount to an active subscription, see [Modifying Pricing](#).

## SuiteBilling Subscription Revisions

When you create, edit, or void a subscription change order, NetSuite generates a subscription revision with the individual contract modifications and the associated revenue impacts. When you save a subscription change order, all changes associated with it are stored in the subscription revision.

A subscription revision saves the history of changes made to the subscription. You can track contract and revenue changes that are made after activation to a specific change order. You also can track changed values over an entire contract term. For more information about subscription changes orders, see [SuiteBilling Change Orders](#).

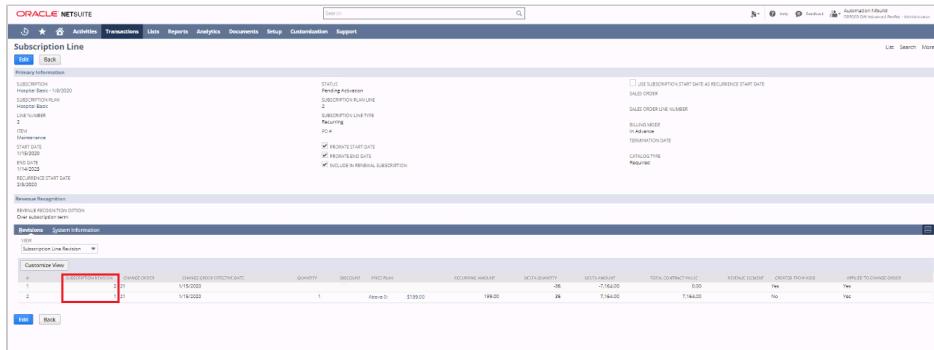
You can no longer delete a subscription change order, but you can void it. Voided change orders are visible but do not impact charge amounts, so you can track the revenue change history after change orders have been removed. For instructions, see [Voiding a Change Order](#).

## Viewing Subscription Revisions

You can see subscription revisions on the Subscription Line record or the Subscription Change Order record.

## To view a subscription revision:

- View a subscription revision in the following records:
  - The Subscription Line record, on the **Revisions** subtab, in the Subscription Revision column, as shown in the following figure:



- The Subscription Change Order record, on the **Lines** subtab, in the Subscription Line Revisions column, as shown in the following figure:



If you edit or delete a subscription that was activated before 20.1, NetSuite does not create subscription revisions. NetSuite creates new subscription revisions for a subscription that was activated before 20.1 only when you next save a subscription change order.

Subscription revisions contain the following data:

- Change Order Effective Date**
- Discount**
- Price Plan**
- Revenue Element** — For a Usage line, the subscription revision includes a link to its Revenue Element that resulted from a specific modification. For One-Time and Recurring lines, the subscription revision includes links to its Revenue Elements that resulted from specific modifications.
- Created From Void** — Subscription changes resulting from a voided change order.
- Applied to Change Order** — Indicates any line removed from a change order by clearing the **Apply** checkbox.
- Created By**
- End Date** — the line end date at the time the change order and revision are created.

Subscription revisions contain the following additional data for One-Time and Recurring line types:

- Quantity**
- Recurring Amount**
- Delta Quantity** — Delta values represent the total change from the subscription change order effective date to the subscription line end date. If the subscription term is Evergreen, the delta value is the Estimated Rev Rec End Date, or if Advance Revenue Recognition is not enabled, 3 years.

- **Delta Amount**
- **Total Contract Value** – The Total Contract Value is based on the entire contract term for the subscription line.

## Clearing Revisions

You can clear revisions only when no revenue has been recognized or merged. Clear Revisions does the following:

- Clears the change order history from the **Change Order** subtab
- Removes all history of revisions
- Removes all revenue arrangements and elements associated with the subscription

### To clear all of a subscription's revisions:

1. For the subscription you want to clear, click the subscription name or **Edit**.



**Tip:** To find the subscription, you can:

- Look in the **Recent Records** tab for recently viewed subscriptions.
- Enter the subscription name or subscription ID into the Global Search field.

2. From the Actions list, select Clear Revisions.

## Voiding a Change Order

As of NetSuite 2020.1, it is no longer possible to delete a subscription change order record. Instead, you can void subscription change orders.

You cannot edit a voided change order.

### To void a change order:

1. (Optional) If the subscription change order has revisions, you must clear the revisions before you can void the change order. See [Clearing Revisions](#).
2. If you have not already opened the change order record, open it. For information about change order types, see [SuiteBilling Change Orders](#). For the navigation path of a change order, refer to the procedure for the type of change order you want to locate. For example, for an activation change order, see the help topic [Activating a Subscription](#).
3. On the change order record, click **Void**.

Using SuiteScript, you can search for and view voided change orders and void a subscription change order. When the SuiteScript 1.0 nlapiDeleteRecord method or the SuiteScript 2.0 record.delete method run against a subscription change order, these methods void the record instead of deleting it. If you are using a subscription change order search in SuiteScript, use the new Status field to filter for active change orders to exclude voided change orders.

## Viewing a Voided Change Order

Voided records are included in subscription change order searches so that you can track revenue and contract changes.

### To view a voided change order:

1. Click a change order with the Voided status.
2. To see details, click the column heading.

## Change Order Type-Specific Revenue Impacts

The following are type-specific changes to subscription change orders:

- Activation and Reactivation change orders start revenue generation, and are considered the base element for reporting structures. Each line in the change order generates a revenue element starting on the change order effective date and ending on the subscription end date.
- In a Modify Pricing change order, a single line change could create multiple revenue elements for a single revision. This can occur when the line goes through future change orders such as quantity or price modifications. The start and end dates depend on the number of future dated change orders.
- Suspend and Termination change orders reverse all revenue beyond the effective date. Each modify, extend, and activation change order generates a revenue element.
- Renew-Extend change orders generate a revenue element for every active subscription line at the start date of the extension. The date of entry is the effective date of Renew-Extend change orders.

## Terminating a Subscription Line Item

Terminating a subscription is a line-level action. When you terminate a subscription line item, you can edit it on or before the termination date. After a subscription line is closed or terminated, the subscription line is locked from all edits. If you want to terminate a subscription using the Terminate change order, you must first terminate each subscription line item.

After a line item is terminated, the status of the line item changes to **Terminated** the day after the termination date. For example, if you terminate a subscription at 8:00 a.m. on July 1, the termination change order does not take effect until 11:59 p.m. If there is a daily charge or prorated charge for the subscription, the customer will be charged for July 1. Any future charges are canceled. If a subscription line item is terminated in error, you must enter a new subscription item. You cannot change the subscription line item status after it is set to **Terminated**.

### To terminate a subscription line item:

1. Go to Transactions > Subscriptions > Create Subscriptions > List.
2. Click the **Edit** link associated with the subscription for which you want to terminate line items.
3. From the **Actions** menu, select **Terminate**.
4. In the **Effective Date** field, enter or select a date for the termination.
5. For the line item you want to terminate, check the **Apply** box.
6. Click **Save**.

## Suspending a Subscription Line Item

Suspend subscription line items using the suspend change order. You can suspend subscription line items when a customer wants to temporarily stop receiving those items. If you need to suspend a subscription item, you must first activate it. For instructions about how to activate a subscription line item, see [Activating a Subscription](#).

To suspend a subscription line item, it must be in either **Active** or **Pending Activation**.

### To suspend a subscription line item:

1. On the subscription record, from the **Actions** menu, select **Suspend**.
2. In the **Effective Date** field, enter or select a date for the suspension.
3. In the **Requester** field, if necessary, change the requester.
4. In the **Modification Type** field, choose one of the following options:
  - **New or Churn** - The analytics revenue impact of the change order is classified as New or Churn.
  - **Upsell or Downsell** - The analytics revenue impact of the change order is classified as Upsell or Downsell.
  - **Ignored** - The change order has no analytics revenue impact.
5. Check the **Apply** box associated with the item you want to suspend. The **Apply** box for items that cannot be suspended is inactive.
6. Click **Save**.

After an item is suspended, you must use the reactivate change order to reactivate the item.

## Reactivating a Subscription Line Item

Reactivate suspended line items using the reactivate change order.

After an item is suspended, a customer may want to again receive the item that was suspended. To reactivate an item, it must have the **Suspended** status.

### To reactivate a subscription line item:

1. On the subscription record, from the **Actions** menu, select **Reactivate**.
2. In the **Effective Date** field, enter or select a date for the reactivation.
3. In the **Requester** field, if necessary, change the requester.
4. In the **Modification Type** field, choose one of the following options:
  - **New or Churn** - The analytics revenue impact of the change order is classified as New or Churn.
  - **Upsell or Downsell** - The analytics revenue impact of the change order is classified as Upsell or Downsell.
  - **Ignored** - The change order has no analytics revenue impact.
5. Check the **Apply** box associated with the line item you want to reactivate.
6. Click **Save**.

After submitting a reactivate change order, the subscription item reactivates on the effective date.

To revert a subscription line to Pending Activation status, you must void all its change orders. For instructions, see [Voiding a Change Order](#).

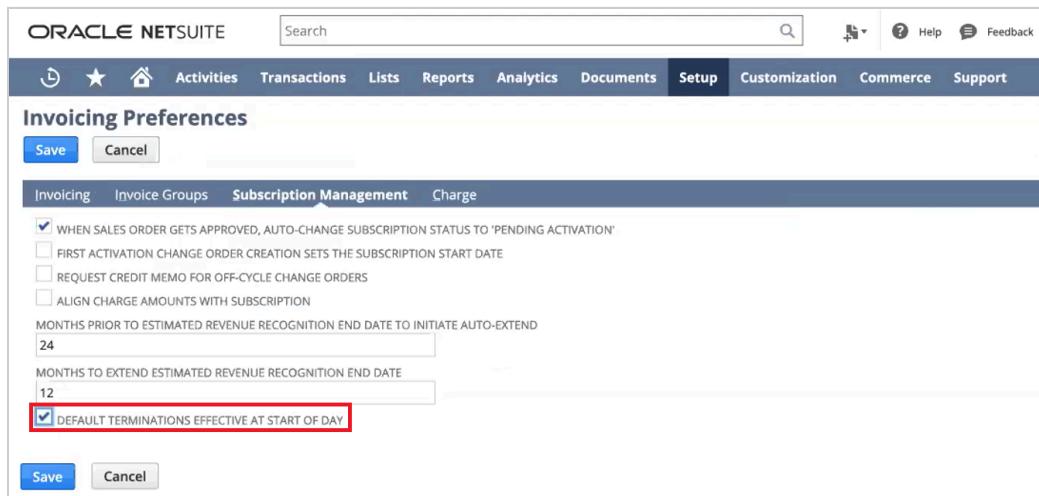
## Removing One-Day Charges

To remove a one-day charge caused by terminating a subscription line, you can create terminate change orders at the beginning of the day. Termination change orders can take effect at the start of day or end of day. Termination change orders can cancel Activation change orders.

To remove one-day charges for a single suspended day caused when you reactivate a subscription line, you can create a reactivation change orders on the item's suspension date. To reactivate all charges, you can create Reactivation change orders to take effect on the same day as Suspend change orders.

### To remove a one-day charge from a subscription line item:

1. Go to Setup > Accounting > Invoicing Preferences.
2. On the Subscription Management tab, click **Default Terminations Effective at Start of Day**.



3. Click **Save**.

The **Default Terminations Effective at Start of Day** preference determines the default behavior for when termination change orders will take effect, at the start or end of day. You can enable and disable this preference at any time. The setting at a given point in time determines the behavior of any termination change order created.

Removing one-day charges is useful when you create an Activation change order then realize you activated the subscription in error. You need to remove all of the charges from that subscription line. When you create a Termination change order with **Default Terminations Effective at Start of Day** set, and on the same date as the Activation change order, all charges for that subscription line are removed. For example, if you create a Termination change order for January 1 and **Default Terminations Effective at Start of Day** is checked, there is no charge for January 1. Termination change orders that are created on the same day as the activation change order, and are set to take place at start of day, cancel out the activation and remove all charges for the line. There is no single-day charge.

Likewise, you can create a Reactivation change order on the same day as a Suspend change order. Prior to NetSuite version 21.1, Reactivation change orders were not permitted to be created on the same day as a Suspend change order. A Suspend change order resulted in a single-day charge left in the On Hold status and unable to be billed. For example, you have a subscription that was created with two

lines. In Related records, you can see that invoices were sent January and February. One of the lines was suspended on 3/1/2021, so all charges from that date forward are on hold.

#	SUBSCRIPTION LINE	ITEM	CHARGE TYPE	SERVICE START DATE	SERVICE END DATE	BILL DATE	AMOUNT	CURRENCY	CHARGE STAGE
1	Mid Market Data Storage - 1/1/2021:1	Set-up Fee	One-Time	1/1/2021	1/1/2021	1/1/2021	\$2,000.00	USA	Processed
2	Mid Market Data Storage - 1/1/2021:2	Data Storage	Recurring Service	1/1/2021	1/31/2021	1/1/2021	\$500.00	USA	Processed
3	Mid Market Data Storage - 1/1/2021:2	Data Storage	Recurring Service	2/1/2021	2/28/2021	2/1/2021	\$500.00	USA	Processed
4	Mid Market Data Storage - 1/1/2021:2	Data Storage	Recurring Service	3/1/2021	3/31/2021	3/1/2021	\$500.00	USA	Hold
5	Mid Market Data Storage - 1/1/2021:2	Data Storage	Recurring Service	4/1/2021	4/30/2021	4/1/2021	\$500.00	USA	Hold
6	Mid Market Data Storage - 1/1/2021:2	Data Storage	Recurring Service	5/1/2021	5/31/2021	5/1/2021	\$500.00	USA	Hold
7	Mid Market Data Storage - 1/1/2021:2	Data Storage	Recurring Service	6/1/2021	6/30/2021	6/1/2021	\$500.00	USA	Hold
8	Mid Market Data Storage - 1/1/2021:2	Data Storage	Recurring Service	7/1/2021	7/31/2021	7/1/2021	\$500.00	USA	Hold
9	Mid Market Data Storage - 1/1/2021:2	Data Storage	Recurring Service	8/1/2021	8/31/2021	8/1/2021	\$500.00	USA	Hold
10	Mid Market Data Storage - 1/1/2021:2	Data Storage	Recurring Service	9/1/2021	9/30/2021	9/1/2021	\$500.00	USA	Hold
11	Mid Market Data Storage - 1/1/2021:2	Data Storage	Recurring Service	10/1/2021	10/31/2021	10/1/2021	\$500.00	USA	Hold
12	Mid Market Data Storage - 1/1/2021:2	Data Storage	Recurring Service	11/1/2021	11/30/2021	11/1/2021	\$500.00	USA	Hold
13	Mid Market Data Storage - 1/1/2021:2	Data Storage	Recurring Service	12/1/2021	12/31/2021	12/1/2021	\$500.00	USA	Hold

Creating a Reactivation change order to take effect on the same day as a Suspend change order results in the full charge amount returning to the Ready state.

### To remove a one-day charge for a single suspended day caused when you reactivate a subscription line:

- From the **Actions** menu, select **Reactivate**.
- On the Reactivate change order, set the **Effective Date** to 3/1/2021.
- For the line item you want to suspend, check the **Apply** box.
- Click **Save**.

APPLY	LINE NUMBER	ITEM	STATUS	NEW STATUS	QUANTITY	PRICE PLAN	DISCOUNT	RECURRING AMOUNT	START DATE	END DATE
<input type="checkbox"/>	1	Set-up Fee	Closed	<input type="button" value="Closed"/>	1	Above 0: \$2,000.00			1/1/2021	1/1/2021
<input checked="" type="checkbox"/>	2	Data Storage	Suspended	<input type="button" value="Active"/>	10	Above 0: \$50.00		500.00	1/1/2021	12/31/2021

When you go back to Related Records tab, all charges are back in the Ready status, as shown in the following image.

#	SUBSCRIPTION LINE	ITEM	CHARGE TYPE	SERVICE START DATE	SERVICE END DATE	BILL DATE	AMOUNT	CURRENCY	CHARGE STAGE
1	Mid Market Data Storage - 1/1/2021: 1	Set-up Fee	One-Time	1/1/2021	1/1/2021	1/1/2021	\$2,000.00	USA	Processed
2	Mid Market Data Storage - 1/1/2021: 2	Data Storage	Recurring Service	1/1/2021	1/31/2021	1/1/2021	\$500.00	USA	Processed
3	Mid Market Data Storage - 1/1/2021: 2	Data Storage	Recurring Service	2/1/2021	2/28/2021	2/1/2021	\$500.00	USA	Processed
4	Mid Market Data Storage - 1/1/2021: 2	Data Storage	Recurring Service	3/1/2021	3/31/2021	3/1/2021	\$500.00	USA	Ready
5	Mid Market Data Storage - 1/1/2021: 2	Data Storage	Recurring Service	4/1/2021	4/30/2021	4/1/2021	\$500.00	USA	Ready
6	Mid Market Data Storage - 1/1/2021: 2	Data Storage	Recurring Service	5/1/2021	5/31/2021	5/1/2021	\$500.00	USA	Ready
7	Mid Market Data Storage - 1/1/2021: 2	Data Storage	Recurring Service	6/1/2021	6/30/2021	6/1/2021	\$500.00	USA	Ready

This completely cancels out the suspension, allowing for the full charge amount to be billed. Suspend and reactivation change orders are still restricted to one change order of the same type per day for a subscription line. Therefore, you can't create a second suspend or reactivate change order for the same day.

## Modifying Pricing

Modify pricing using the modify pricing change order. Modify pricing change orders change the existing price for one or more active subscription items based on a quantity, rate, or discount change. For more information on change orders, see [SuiteBilling Change Orders](#).

You may need to modify the rate, quantity, or discount of an item during the subscription term. For example, a client calls three months into the subscription term and asks to add additional licenses because they have hired new employees. Or, the client needs to increase the discount during the term because one of their customers is unhappy.

### To modify pricing:

1. Go to Transactions > Subscriptions > Create Subscriptions > List.
2. For the subscription for which you want to modify pricing, click **View**.
3. Click **Modify Pricing**.
4. (Optional) If you want to use off-cycle pricing, check the **Use Off-Cycle Invoice for Advance Charges** box. For information about off-cycle invoicing, see [Off-Cycle Invoicing with Modify Pricing Change Orders](#).
5. In the **Effective Date** field, enter or select a date.
6. In the **Requester** field, if necessary, change the requester.
7. In the **Modification Type** field, choose one of the following options:
  - **New or Churn** - The analytics revenue impact of the change order is classified as New or Churn.
  - **Upsell or Downsell** - The analytics revenue impact of the change order is classified as Upsell or Downsell.
  - **Ignored** - The change order has no analytics revenue impact.
8. Click the item for which you want to modify pricing.
9. Click **Edit**.
10. On the **Sales/Pricing** subtab, change the pricing information.

11. Click **Save**.

After modifying pricing, you can return to the subscription by going to Transactions > Subscriptions > Create Subscriptions > List.

## Off-Cycle Invoicing with Modify Pricing Change Orders

The off-cycle modify pricing change order prorates pricing charges mid-cycle. An off-cycle pricing change order generates positive and negative charges. You need to process billing operations to generate the associated invoice and credit memo. You can then choose to manually apply the credit memo or apply it to a future date. For information on billing operations, see [Billing Operations](#).

## Closed Subscriptions

When the subscription term ends and the subscription is not automatically set to renew, the subscription closes. When a subscription closes, you can no longer create change orders for that subscription.

If a customer wants to renew a closed subscription, you can renew it by creating a new subscription, or by renewing and extending the existing subscription before it closes. For information on renewal options, see [Renewing a Subscription](#).

## Renewing a Subscription

There are different ways to specify whether and how to renew a subscription. You can do any of the following:

- Renew (or extend) an existing subscription. To view the process diagram, see [Renew and Extend a Subscription from a New Subscription Record](#).
- Renew by creating a new subscription transaction that uses the existing plan or a new plan. To view the process diagram, see [Renew a Subscription with a New Subscription Record](#).

You can manually create a renew change order during the subscription term.

You can automatically initiate the renewal process on the Subscription record. Under the **Renewal** subtab, you can view renewal options, including the term, method, and transaction type. From this same location, you can automatically initiate the renewal process before the subscription end date.

EDIT	VIEW	NAME	SALES ORDER	INITIAL TERM	START DATE	END DATE	STATUS	RENEWAL TRANSACTION
Edit	View	All Optional - 3/1/2017		One Year	3/1/2017	2/28/2018	Not Included	

For the Default Renewal Transaction Type field, you can choose Do Not Create Sales Transaction. If you choose this option, upon renewal, NetSuite creates a new subscription using the existing subscription plan. NetSuite does not associate the new subscription with a sales transaction.

If you want to renew a subscription before the subscription end date, you can extend the end date of the existing subscription. However, if the renewal requires a new subscription transaction, the subscription cannot be renewed using the extend existing subscription method.

A subscription that renews to a new Subscription record can have a different billing account from the original subscription. During subscription renewal, the newly created subscription has the billing account from the original subscription. However, you can change the billing account after you create the subscription while it is in the Draft or Pending Activation status.

Subscription renewals are considered either low-touch or high-touch:

- A low-touch renewal is one that automatically renews, or extends, when the initial subscription ends.
- A high-touch renewal requires more effort. Someone, for example, a sales representative, needs to contact the customer or manually renew the subscription.

#### Example 1: Low Touch Renewal at the End of the Term



In this example, the figure represents an existing magazine subscription that runs from January 1, 2021 to December 31, 2021. On January 1, 2022, the subscription automatically renews for another year using the extend existing subscription method with the same subscription plan and existing pricing.

#### Example 2: High Touch Renewal Before the End of the Term



In this example, an existing one-year subscription expires on December 31, 2021, but is not set to automatically renew. A sales person needs to contact the customer to negotiate another subscription. A new subscription is created using the create new subscription renewal method. The new subscription could have required changes to the subscription plan or a change in price, and is created with a **Draft** status. The associated sales order is created with **Pending Approval** status.

#### To renew a subscription automatically:

1. Go to Transactions > Subscriptions > Create Subscriptions > List.
2. For the subscription you want to renew, click **Edit**.
3. Click the **Renewal** subtab.
4. From the **Default Renewal Term** list, select a term for the renewal subscription.
5. From the **Default Renewal Method** list, select a default renewal method.
  - a. If you selected **Extend Existing Subscription**, go to step 6.
  - b. If you selected **Create New Subscription**:
    - i. From the **Default Renewal Subscription Plan** list, select a default subscription plan for the renewal subscription.

- ii. From the **Default Renewal Price Book** list, select a pricing option for the renewal subscription.
- iii. From the **Default Renewal Transaction Type** list, select one of the available options (**Do Not Create Sales Transaction Estimate, Opportunity, or Sales Order**).
- 6. To automatically start the renewal process, check the **Automatically Initiate Renewal Process** box and enter the appropriate number into the **Before Subscription End Date** box.



**Note:** The appropriate number to enter depends on your organization. It indicates how many days before the subscription ends that you want to contact the customer. For example, you might want to contact the customer to renegotiate the subscription terms.

- 7. Click **Save**.

### To renew a subscription manually:

1. Go to Transactions > Subscriptions > Create Subscriptions > List.
2. For the subscription you want to renew, click **Edit**.
3. From the Actions menu, click **Renew**.
4. From the **Default Renewal Term** list, select a term for the renewal subscription.
5. From the **Default Renewal Method** list, select a default renewal method.
  - a. If you selected **Extend Existing Subscription**, go to step 6.
  - b. If you selected **Create New Subscription**:
    - i. From the **Default Renewal Subscription Plan** list, select a default subscription plan for the renewal subscription.
    - ii. From the **Default Renewal Price Book** list, select a pricing option for the renewal subscription.
    - iii. From the **Default Renewal Transaction Type** list, select one of the available options (**Do Not Create Sales Transaction Estimate, Opportunity, or Sales Order**).
6. Click **Save**.

## Including Add-On Items in Subscription Renewals

When renewing a subscription manually using a renewal change order, you have the option to edit each line item before saving the draft subscription. Reasons you might want to edit lines include adding or removing items from the subscription. Perhaps a service is no longer available or has been replaced by a new service.

The following default rules apply to add-on items with the **Include In Renewal Subscription** box checked:

For these default renewal settings:	These rules apply:
Method – Extend Existing Subscription	All add-on items are included at the same price and quantity for the extended renewal period.
Method – Create New Subscription Subscription Plan – Same Subscription Plan	All add-on items are included in the draft renewal subscription with the same price and quantity.
Price Book – Keep Existing Pricing	

For these default renewal settings:	These rules apply:
Method – Create New Subscription Subscription Plan – Same Subscription Plan Price Book – a specific price book	All add-on items are included in the draft renewal subscription with the same quantity. Prices are defined by the specific price book.
Method – Create New Subscription Subscription Plan – (a different plan from the original subscription selected)	Add-on items from the original subscription are not included in the draft renewal subscription.  Only line items defined in the renewal plan are included on the renewal subscription.

## Adding Renewal Information to Subscription Plans

You can specify subscription renewal parameters when you create the subscription plan, or you can edit the plan later to add renewal information. The renewal options selected on the plan are used when creating all subscriptions from this plan. However, the plans can be edited when the draft subscription is created.

### To add renewal information to a subscription plan:

1. Go to Lists > Subscriptions > Subscription Plans.
2. For the subscription you want to renew, click **Edit**.
3. Click the **Renewal** subtab.
4. From the **Default Renewal Term** list, select a term for the renewal subscription.
5. From the **Default Renewal Method** list, select a default renewal method.
  - a. If you selected **Extend Existing Subscription**, go to step 6.
  - b. If you selected **Create New Subscription**:
    - i. From the **Default Renewal Subscription Plan** list, select a default subscription plan for the renewal subscription.
    - ii. From the **Default Renewal Transaction Type** list, select one of the available options (**Do Not Create Sales Transaction**, **Estimate**, **Opportunity**, or **Sales Order**).
6. To automatically start the renewal process, check the **Automatically Initiate Renewal Process** box and enter the appropriate number into the **Before Subscription End Date** box.
7. Click **Save**.

## Deleting a Subscription

Before you can delete a subscription, you must void all change order records and then remove them by clearing the subscription revisions.

### To delete a subscription:

1. Void all change order records. For instructions, see [Voiding a Change Order](#).
2. Use the Clear Revisions feature to remove subscription change orders. For instructions, see [Clearing Revisions](#).
3. For the subscription you want to delete, click the subscription name or **Edit**.



**Tip:** To find the subscription, you can:

- Look in the **Recent Records** tab for recently completed subscriptions.
- Enter the subscription name or subscription ID into the Global Search field.

4. From the Actions list, select **Delete**.

# SuiteBilling Reports

SuiteBilling offers several reports that let you monitor your subscriptions and customers:

- [Monthly Recurring Revenue Report](#)
- [Total Contract Value Report](#)
- [Billings To Date Report](#)
- [Churn By Customer Report](#)
- [Churn By Revenue Report](#)

To view the SuiteBilling reports, go to Reports > Subscriptions.

 **Tip:** If you don't see the Subscriptions option in the Reports menu, switch to an administrative role or contact your NetSuite administrator.

## Monthly Recurring Revenue Report

The Monthly Recurring Revenue report shows the sum of recurring line items for a specified month or range.

Each line of the The Monthly Recurring Revenue report includes the following columns:

- Customer
- Subscription Plan
- Subscription

The default date range is the previous rolling year.

### To view the Monthly Recurring Revenue report:

1. Go to Reports > Subscriptions > Montly Recurring Revenue.
2. If you want to customize this report, click [Report Customization](#).

The Monthly Recurring Revenue report shows the final rating value for a given month. This report does not include prorated amounts, even if proration is enabled for a line item.

Suspended subscriptions are reported the same as active subscriptions, but closed or terminated subscriptions are not reported. In addition, draft subscriptions are not currently reported.

The Monthly Recurring Revenue report assumes that subscriptions will be renewed. Any subscriptions that are not terminated or closed continue to report into the future.

For more information on reporting, see the help topic [Reporting Overview](#).

## Total Contract Value Report

The Total Contract Value report shows total bill charges for the full term of a subscription. To see charges up to the current time for a subscription, see the [Billings To Date Report](#).

The Total Contract Value report does not include usage charges because usage can vary each billing period.

Each line of the Total Contract Value report includes the following columns:

- Customer
- Subscription Plan
- Subscription

#### **To view the Total Contract Value report:**

1. Go to Reports > Subscriptions > Total Contract Value.
2. If you want to customize the TCV report, click [Report Customization](#).

The data is represented as the total sum of all charges for a given subscription. Credit memos are represented in the months in which they are created.

Future months beyond the current bill cycle show the total contract value based on any change orders or modifications to the subscription to date.

For more information on reporting, see the help topic [Reporting Overview](#).

## Billings To Date Report

The Billings to Date report shows total billed charges, up to the current cycle, for a subscription. To view charges for the full subscription term, see the [Total Contract Value Report](#).

Each line of the report includes the following columns:

- Customer
- Subscription Plan
- Subscription

#### **To view the Billings To Date report:**

1. Go to Reports > Subscriptions > Billings To Date.
2. If you want to customize the report, click [Report Customization](#).

The data is represented as a rolling transaction total from month to month. Credit memos are represented in the months in which they are created.

Future months beyond the current bill cycle show the total billings to date based on the last billing transaction for that subscription.

For more information on reporting, see the help topic [Reporting Overview](#).

## Churn By Customer Report

The Churn by Customer report shows how many customers are canceling or have subscriptions ending in a given month. Churn is calculated differently by different customers. As a result, the structure of the Churn by Customer report is different from most other NetSuite reports. It allows for the addition of fields.

#### **To view the Churn By Customer report:**

1. Go to Reports > Subscriptions > Churn by Customer.

- If you want to customize the report, click [Report Customization](#).

For more information on reporting, see the help topic [Reporting Overview](#).

The following table provides a list of columns and their definitions.

Column	Definition
Customers at beginning of month	The number of customers with active subscriptions at the beginning of the month.
Customers at end of month	The number of customers at the end of the month. This total is calculated by adding net new customers to the number of customers at the beginning of the month.
New Customers	Customers who activated their first new subscription at any point in the month.
Churned Customers	Customers whose subscriptions have all closed or terminated in the month.
Net New Customers	New customers minus churned customers. This can be a positive or negative number.
Customer Churn %	Churned customers divided by customers at the beginning of the month.

Notes about calculating churn and customers:

- Churn is included only in the month that the subscription ends.
- The default date selection is Last Rolling Year. This report only shows information for full months of data.
- Customers with a subscription in the first month of data collection are counted toward total customers.
- All customers in the first month of data collection are considered new customers.
- Customer churn % is calculated as: churned customers divided by new customers, then multiplied by 100 in the first month of data collection.
- Suspended subscriptions do not count toward churn.
- If the number of customers at the beginning of the month equal zero (0), the customer churn % cannot be calculated.

## Churn By Revenue Report

The Churn by Revenue report shows the reduction in monthly recurring revenue due to subscriptions being reduced in value, canceled, or ending.

### To view the Churn By Revenue report

- Go to Reports > Subscriptions > Churn by Revenue.
- If you want to customize the report, click [Report Customization](#).

For more information on reporting, see the help topic [Reporting Overview](#).

The following table provides a list of rows and their definitions.

Row	Definition
MRR at Beginning (of Month)	Monthly recurring revenue calculated on the last day of the previous month. This is the same amount reported for a prior month on the Monthly Recurring Revenue report.

Row	Definition
MRR at End (of Month)	Monthly recurring revenue calculated on the last day of a specific month. This is the same amount reported for a specific month in the Monthly Recurring Revenue Report.
New MRR	Monthly recurring revenue associated with new subscriptions started in a specific month. This is equal to the sum value of all subscriptions with a start date in a specific month.
Expansion MRR	Monthly recurring revenue associated with the upsell of existing subscriptions in a specific month. This is the sum value of all subscriptions that increased in value in a specific month.
Contraction MRR	Monthly recurring revenue associated with the downsell of existing subscriptions. This is the sum value of all subscriptions that decreased in value in a specific month.
Lost MRR	Overall recurring revenue lost in a specific month from canceled or closed subscriptions. This is the sum value of all monthly subscriptions that were canceled or closed in a specific month.
MRR Churn	The churn in recurring revenue from cancelled subscriptions and downgraded subscriptions.
New New MRR	This is the sum of new monthly recurring revenue plus expansion monthly recurring revenue minus lost monthly recurring revenue .
MRR Churn (%)	Monthly recurring revenue churn divided by monthly recurring revenue at the beginning of the month.

Notes about calculating churn and monthly recurring revenue:

- All MRR is considered new on the first month.
- Churn % cannot be calculated on the first month because all MRR is new.
- Revenue associated with suspended subscriptions does not count toward churn. Only revenue that has been lost or will be lost in the specific month contributed to MRR churn.
- If MRR at the beginning of the month equals zero (0), MRR churn cannot be calculated.

# Advanced Billing

- Advanced Billing Overview
- Enabling Advanced Billing
- Billing Schedules
- Creating Billing Schedules
- Creating Billing Schedules From an Estimate or Sales Order
- Applying Billing Schedules
- Discount and Markup Items and Billing Schedules
- Billing Customers Using Billing Schedules
- Invoicing Billable Customers
- Viewing Invoice Schedules
- Advanced Billing and Advanced Shipping

## Advanced Billing Overview

Advanced billing enables you to create billing schedules to invoice sales over a range of time or a contract term. For example, a service company can use advanced billing to manage recurring billing on sales such as a one-year membership billed monthly.

Billing schedules manage the billing process to track when to invoice customers and how much to bill.

When a billing schedule is assigned on a sale, NetSuite adds a bill to the billing queue at the intervals determined by the schedule. Then, you can go to Transactions > Sales > Bill Sales Orders to create the invoices in the billing queue.

The initial bill for each schedule is created on the date entered in the Start Date field of the order. If no start date is entered, the first invoice is created when the order is fulfilled or partially fulfilled.

For example, you can apply a billing schedule that defines the initial billing amount at 50% of the sale total with a recurrence frequency of monthly and a recurrence count of two. If the sales order totals \$1,000.00 and the Start date is January 15th, the sales order will be billed as below:

- Initial invoice for 50% of the sale = \$500.00 billed on January 15th  
The balance of the amount due is automatically split evenly between remaining bills.
- Second invoice for 25% of the sale = \$250.00 billed on February 15th
- Third invoice for 25% of the sale = \$250.00 billed on March 15th

When you view a sales order, you can click the Schedule link at the top of the page to open the schedule record.

For orders that have billings outstanding, you can create the remaining bills manually before NetSuite creates them from the billing schedule. You can create the next bill on the billing schedule, or you can bill all of the remaining unbilled portion of the order.

You can set a preference to Invoice in Advance of Fulfillment:

- Enable the preference to invoice your customers for the full amount prior to the complete fulfillment of an order. With this preference, you can invoice your customers in advance for an entire order that is shipped in parts.

- Disable the preference to limit the quantity fulfilled to the quantity remaining on the fulfillment.

Set the preference at Setup > Accounting > Preferences > Accounting Preferences > Order Management > Invoicing. For more information, read the help topics [Accounting Preferences](#) and [Billing Orders Using Advanced Shipping with Advanced Billing](#).

The Advanced Billing feature requires that you also enable the Sales Orders feature.

Billing schedules can also be used with the Project Management feature. For details, read the help topic [Project Billing](#).

## Enabling Advanced Billing

Follow these steps to enable Advanced Billing.

### To enable Advanced Billing:

1. Go to Setup > Company > Enable Features.
2. On the **Transactions** subtab, check the **Advanced Billing** box.
3. To use Advanced Billing, you must also enable the Sales Orders feature by checking the **Sales Orders** box.
4. Click **Save**.

To create a billing schedule, go to Lists > Accounting > Billing Schedules > New.



**Note:** If you later choose to turn off Advanced Shipping, you must first go to Transactions > Sales > Bill Sales Orders and process all sales orders that appear on the page.

## Billing Schedules

A billing schedule defines the schedule to bill a sale over a range of time or a contract term. For example, you may want to bill monthly for the sale of a one-year membership, or bill quarterly for a three-year service contract. Billing schedules manage the billing process by tracking when to invoice customers and how much to bill.

For example, you can create a billing schedule that defines the initial billing amount at 50% of the sale total with a recurrence frequency of monthly and a recurrence count of two. If you apply this billing schedule to a service contract sales order that totals \$1,000.00, the sales order will be billed as below:

- Initial invoice for 50% of the sale = \$500.00  
The balance of the amount due is automatically split evenly between remaining bills.
- Second invoice one month later for 25% of the sale = \$250.00
- Third invoice one month after the second for 25% of the sale = \$250.00

You can set up billing schedules to accommodate many types of sales. For example, some sales may require a set currency amount for the initial payment, while others require a percentage of the total. Sales may require weekly, monthly or quarterly recurrent invoices. You can set billing to recur monthly 12 times or quarterly 3 times, or even customize bill recurrence schedules to suit your needs.

When a billing schedule is created, you apply it to an entire sales order or select a different billing schedule for each line item on an order. You can also assign a default billing schedule to an item record so it automatically shows on a sales order when that item is selected.

You can also create a new billing schedule from an estimate or sales order. A billing schedule created from an estimate or sales order can be identified as Public or Private. Private billing schedules show in the list of billing schedules only on the transaction it is created from. Note that milestone billing schedules cannot be marked Public. If you manually click Next Bill to bill a sales order and there are any remaining lines that were part of the previous billing schedule, but not billed, NetSuite bills those remaining lines on the next billing date. Click Next Bill again to move forward to the next billing date on the billing schedule. To remove a line from a billing schedule, close the line on the sales order. If you delete the line, your billing schedule amounts could be different from expected.

When using a billing schedule, NetSuite expects fulfillable items to be fulfilled and completely billed on the first invoice before progressing to the next billing schedule invoice. If an item on an invoice is not fulfilled and not fully billed, the billing schedule does not advance to the next invoice.

When you view an estimate or sales order, you can click the Schedule link at the top of the page to open the schedule record. On a sales order, you can click the Schedule subtab to view the billing schedule for the order.

**i Note:** Editing a billing schedule that is already associated with an open sales order does not update existing instances of the billing schedule for the open order. To update existing instances, go to Transactions > Sales > Enter Sales Orders > List to open the sales order and re-save it. Upon saving, the billing schedules are updated with your changes.

Billing schedules affect your sales forecast report by dividing revenue across the billing schedule period. A sales order with a billing schedule is calculated into forecast schedules based on the initial and subsequent billing amounts and on the billing start date. Sales forecasts for a particular period include amounts set to be billed in that period.

## Creating Billing Schedules

Using the Advanced Billing feature, you can create billing schedules to apply to items and to sales. Billing schedules define the schedule of billing sales over a range of time or a contract term. They manage the billing process by tracking when to invoice customers and how much to bill. For example, you can bill monthly for the sale of a one-year membership or bill quarterly for a three-year service contract.

You create billing schedules by defining these characteristics of the schedule:

- **Initial Amount** – the amount of the first billing instance (This can be a currency amount or a percentage of the sale total if you use line-level billing schedules, but you can only use percentage initial amounts if you apply billing schedules on the transaction level.)
- **Type** – the type of billing schedule to use. If you select either the Time and Materials or the Charged-Based billing schedule type, the Initial Amount field does not display. If you are creating a billing schedule you want to apply to a transaction subtotal, this field must be set to Standard
- **Initial Payment Terms** – payment requirements applied to the first billing instance
- **Recurrence Frequency** – the frequency for billings to repeat, such as daily, weekly, quarterly, every four months, or monthly. If you are creating a billing schedule you want to apply to a transaction subtotal, this field must be set to Custom
- **Repeat Every** – how often recurring invoices are generated. The Repeat Every field is related to the Recurrence Frequency field. If Monthly is selected for the Recurrence Frequency, the Repeat Every field changes to Monthly.
- **Recurrence Count** – the number of bills that are created after the initial billing
- **In Arrears** – bills at the end of each period, instead of the beginning
- **Recurrence Payment Terms** – payment requirements applied to billings subsequent to the first

You can set up a billing schedule for each type of sale that incorporates the appropriate billing procedures. For example, you can create a billing schedule that defines the initial billing amount at 50% of the sale total with a recurrence frequency of monthly and a recurrence count of two. If you apply this billing schedule to a service contract sales order that totals \$1,000.00, the sales order will be billed as below:

- Initial invoice for 50% of the sale = \$500.00
- Second invoice one month later for 25% of the sale = \$250.00
- Third invoice one month after the second invoice for 25% of the sale = \$250.00



**Note:** The final installment of a billing schedule is calculated differently as compared to the rest of the installments. The final installment in a billing schedule is calculated as:

Final installment = Total amount - (sum of all other installments and initial amount)

If you want to apply a billing schedule to a subtotal, set the **Type** field to **Standard**, set the **Recurrence Frequency** field to **Custom**, and check the **Apply to Subtotal** box.

### To create a billing schedule:

1. Go to Lists > Accounting > Billing Schedules > New.
2. In the **Name** field, enter a name for this schedule.  
For example, you might name the schedule from the example above **50% down, 2 monthly**.
3. In the **Initial Amount** field, enter the amount that is billed on the first invoice created from the sales order. You can enter this amount as a currency amount or as a percentage. (This can be a currency amount or a percentage of the sale total if you use line-level billing schedules, but you can only use percentage initial amounts if you apply billing schedules on the transaction level.)
4. In the **Initial Payment Terms** field, select the terms to be used on the first invoice to be created from the sales order.

To add new payment terms, go to Setup > Accounting > Setup Tasks > Accounting Lists > New > Term > New > Term > New.

5. In the **Recurrence Frequency** field, select the frequency for bills to recur.  
Select **Custom** to create a new frequency or to create a billing schedule you want to apply to the subtotal of a transaction:

- Enter information in the fields in the middle of the page to define the frequency.
  1. In the **Count** field, enter the number of units of time to calculate a billing date.  
For example, enter **12** to bill twelve months from the initial bill.
  2. In the **Units** field, enter the units of time to calculate a billing date.  
For example, select **Months** to bill twelve months from the initial bill.
  3. Billing frequency defaults to be relative to the date of the initial invoice. If you prefer, the date of subsequent bills can be related to the date of the previous bill, not the date of the initial invoice, using the **Relative to Previous** box.

Check the **Relative to Previous** box if the frequency is relative to the most previous billing instead of relative to the transaction date.

On the first bill line, this box has no effect since the first bill is always relative to the initial bill. On subsequent bill lines, checking or clearing this box affects the Date of the bill.

For example, the initial invoice for a sales order is created on 01/01/2007:

The first line in the bill schedule has a Count and Units of 1 Month. This bill will be created on 02/01/2007.

The second line in the bill schedule has a Count and Units of 2 Months. If Relative to Previous is checked, this bill is scheduled for 2 months after the previous line, or 04/01/2007. If Relative to Previous is cleared, this bill is scheduled for 2 months after the initial bill, or 03/01/2007.

When you check the Relative to Previous box, the date of that billing appears in the Example field.

4. In the **Amount** field, enter the amount to be billed. This can be a currency amount or a percentage of the sale total if you use line-level billing schedules, but you can only use percentage initial amounts if you apply billing schedules on the transaction level.
5. If you selected a Recurrence Frequency other than custom, enter a Recurrence Count. This defines the number of invoices or cash sales generated after the initial sale. The remaining amount due for the sale (after the initial amount is deducted) is evenly divided between the recurring invoices. For example, the total to be billed for a sale is \$2,400.00. The billing schedule applied to the sale has an initial amount of 50%, a recurrence frequency of monthly, and a recurrence count of 12. The initial invoice total is \$1,200.00 and then a recurring invoice of \$100.00 is generated one time per month for 12 months to bill the remainder of the sale.
6. In Arrears will be checked by default.
  - When In Arrears is checked, billing occurs at the end of the recurrence period. From the example in step 6, if the initial invoice for \$1,200.00 is generated on January 1st, 2006, then the first recurring monthly invoice for \$100.00 is generated on February 1st, 2006.
  - When In Arrears is cleared, billing occurs at the beginning of the recurrence period. From the example in step 6, if the initial invoice for \$1,200.00 is generated on January 1st, 2006, then the first recurring monthly invoice for \$100.00 is also generated on January 1st, 2006. NetSuite combines these two amounts on one invoice, so the first invoice includes both the initial amount and the first recurring payment, for an invoice total of \$1,300.00.
7. In the **Recurrence Payment Terms** field, select the terms to be used on all invoices subsequent to the initial invoice.



**Note:** This field is greyed out when you select Custom in the Recurrence Frequency field.

8. In the **Recurrence Payment Terms** field, select the terms to be used on all invoices subsequent to the initial invoice.
9. If you do not want this schedule to be applied to items and new orders, deactivate the schedule by checking the **Schedule is Inactive** box.
10. Click **Save**.

It is possible to create a billing schedule that will bill more than 100% of the invoice amount.

For example, you could create a schedule for the initial amount of \$10,000.00 and then a custom frequency of 25% billed every three months. If you apply this schedule to an order totaling \$60,000.00, the invoices are created as below:

- 1st invoice = \$10,000.00
- 2nd invoice = \$12,500.00
- 3rd invoice = \$12,500.00
- 4th invoice = \$12,500.00
- 5th invoice = \$12,500.00
- **Total billed = \$60,000.00**

After the initial amount, invoices are created based on 25% of the total bill, not based on the remainder due after the initial amount. If your schedule uses both dollar amounts and percentages, or uses a custom frequency, verify that the total amount billed by the schedule is the amount intended.



**Note:** Billing schedules created using the above steps default to be Private and cannot be set to Public.

Private billing schedules show only in the transaction it is created from.

Public billing schedules show on any transaction.

To create a Public billing schedule, it must be created on the fly. Read [Creating Billing Schedules From an Estimate or Sales Order](#).

## Creating Billing Schedules From an Estimate or Sales Order

You can also create a new billing schedule at the time you enter an estimate or sales order.

### To create a billing schedule from an estimate or sales order:

1. Go to one of the following:
    - Transactions > Sales > Prepare Estimates
    - Transactions > Sales > Enter Sales Orders
  2. On the **Billing** subtab, select **New** in the **Billing Schedule** field.
  3. Follow the steps in [Creating Billing Schedules](#) to create the schedule.
  4. A billing schedule created from a transaction can be identified as Public or Private.
    - Private billing schedules show only in the transaction it is created from.
    - Public billing schedules show on any transaction.
- Billing schedules created from a transaction default to Private status.

To set the status to Public, check the **Public** box.



**Note:** Milestone billing schedules cannot be marked public.

5. Click **Save**.

If you create a private billing schedule from an estimate, that schedule is attached to items on the resulting sales order. When the sales order is created from the estimate, the private schedule can also be added to new lines you add to the sales order.

When created, private billing schedules can be changed to have a public status, but public schedules cannot be made private.

View the billing schedule list at Lists > Accounting > Billing Schedules, and choose to view **Public**, **Private** or **All** schedules in the top of the page in the **Type** field. Each private schedule shows a link to its associated transaction and a link to the billing schedule record.



**Note:** Editing a billing schedule that is already associated with open sales orders does not update existing instances of the billing schedule on those open orders.

To update existing instances, go to Transactions > Sales > Enter Sales Orders > List to open the sales order and re-save it. Upon saving, the billing schedules are updated with your changes.

Now, you can apply this billing schedule to sales or items to generate invoices. For more information on applying a billing schedule, read [Applying Billing Schedules](#).

## Applying Billing Schedules

The Advanced Billing feature enables you to create billing schedules that you apply to sales orders or to line items on sales orders. When you apply a billing schedule to an order, the order is billed in according to that schedule.

 **Note:** Unless you use the Apply to Subtotal option, billing schedules are applied to each line individually, not to the subtotal.

## Assigning a Default Billing Schedule to an Item

You can assign a billing schedule on an item record. Then, when the item is chosen on sales transactions, the billing schedule is applied by default. Default billing schedules can be changed on individual transactions.

 **Note:** The billing schedule shows on line items only when forms are customized to show schedules on lines. Read [Assigning a Billing Schedule to a Line Item](#).

You can assign billing schedules to the sale item records and resale item records listed below:

- Inventory (including matrix and serialized)
- Assembly (including serialized)
- Kit/Package
- Non-Inventory (for sale and resale)
- Other Charge (for sale and resale)
- Service (for sale and resale)

### To assign a billing schedule to a new item record:

1. Go to Lists > Accounting > Items > New > New.
2. Click the item type you want to create.
3. On the item record, enter information in the necessary fields.
4. On the **Sales / Pricing** subtab under Sales in the **Billing Schedule** field, select the billing schedule you want to associate with this item.
5. Click **Save**.

You can apply a billing schedule to an existing item by going to List > Accounting > Items and clicking Edit next to the item you want to change.

## Assigning a Billing Schedule to a Transaction

You can apply a billing schedule to an entire transaction.

## To apply a billing schedule to a new sales order:

1. Go to Transactions > Sales > Enter Sales Orders.
2. In the **Start Date** field, enter the date the first invoice is to be created.  
If this field is left blank, the first billing date is the date of the sales order.
3. Choose a customer in the **Customer** field.
4. On the **Billing** subtab, in the **Billing Schedule** field, choose a billing schedule to apply to all items on this order.  
To enter a new billing schedule, click **New**.
5. Fill in other fields as necessary.
6. Click **Save**.

You can also apply a billing schedule to an existing sales order by going to Transactions > Sales > Enter Sales Orders > List and clicking Edit next to the sales order you want to change.

## Assigning a Billing Schedule to an Invoice Subtotal

You can assign a billing schedule to an invoice subtotal. To create an invoice subtotal fixed amount billing schedule, the sales order Type must be set to Standard and the Recurrence Frequency field must be set to Custom. When you check the **Apply to Subtotal** box, enter a schedule of fixed currency amounts billed against the pre-tax, pre-discount invoice total.

For example, if you have an invoice that has three transactions with a subtotal of \$300. Transaction 1 totals \$50, transaction 2 totals \$100, and transaction 3 totals \$150 for a subtotal of \$300. The customer would like to pay the invoice in three installments of \$100; therefore, define three \$100 payments, totaling \$300.

## To apply a billing schedule to a transaction subtotal:

1. Go to Transactions > Sales Order > Enter Sales Order.
2. In the **Start Date** field, enter the date the first invoice is to be created.  
If this field is left blank, the first billing date is the date of the sales order.
3. Choose a customer in the **Customer** field.
4. On the **Billing** subtab, in the **Billing Schedule** field, choose a billing schedule created to apply to a transaction subtotal..  
To enter a new billing schedule, click **New**. To create and apply a subtotal billing schedule, the **Type** field must be set to **Standard**, the **Recurrence Frequency** field must be set to **Custom**, and you must check the **Apply to Subtotal** box.
5. Fill in other fields as necessary.
6. Click **Save**.

## Assigning a Billing Schedule to a Line Item

You can apply a billing schedule to an individual line item on a transaction. To do this, you must first customize the following transaction forms to show line item billing schedules:

- Sales orders
- Online order forms

## To customize your transaction forms for line-item billing schedules:

1. Go to Customization > Forms > Transaction Forms.
2. On the Custom Transaction Forms page, click **Customize** next to the transaction form you want to change.
3. In the **Name** field, enter a name for this form. For example, you might name it Billing Schedule Sales Order.  
This name appears in the Custom Form field on transactions.
4. Click the **Sublist Fields** subtab.
5. In the **Show** column, check the box next to **Billing Schedule** to show the billing schedule column when you view the transaction.
6. Click the **Printing Fields** subtab.
7. Click the **Columns** subtab under **Printing Fields**.
8. Check the **Print/Email** box next to **Billing Schedule** to show the billing schedule column when you print or email the transaction.
9. Click **Save**.

Now, when you choose this custom form in the Custom Form field on transactions, you can choose a billing schedule for each line item.

## To apply a billing schedule to a sales order line item:

1. Go to Transactions > Sales > Enter Sales Orders.
2. In the **Custom Form** field, select the sales order form you customized to show line-item billing schedules.
3. In the **Start Date** field, enter the date for the first invoice to be created.  
If this field is left blank, the first billing date is the date of the sales order.
4. Choose a customer in the **Customer** field.
5. On the **Items** subtab, in the **Items** column, choose an item.  
If the item you choose has a billing schedule assigned on the item record, that billing schedule autofills in the Bill. Sched. column. You can choose another billing schedule for that item.
6. Click **Add**.
7. Fill in other fields on the form as necessary.
8. Click **Save**.

## Discount and Markup Items and Billing Schedules

Billing schedules cannot be assigned to discount items or markup items, but these items can be included on orders with a billing schedule. Discounts and markups are applied to sales as follows:

- When you include a percentage discount item on a sales order with a billing schedule, the discount is applied to each billing instance.

For example, an item that costs \$100 is invoiced in two billings of \$50. The sales order shows a discount item for 10%. The first invoice total is \$45 (\$50 less 10% discount). The second invoice is also for \$45 (\$50 less 10% discount). The total billing amount for the order is \$90 (\$100 less 10% discount).

- When you include a currency amount discount item on a sales order with a billing schedule, the entire discount amount shows on the first invoice by default.

For example, an item that costs \$100 is invoiced in two billings of \$50 dollars. The sales order shows a discount item for \$20. The first invoice total is \$30 (\$50 less \$20 discount). The second invoice total is \$50. The total billing amount for the order is \$80 (\$100 less \$20 discount).

**Note:** This may cause the discount amount to be greater than the billing total on the first invoice. If this happens, you must change the discount amount in the line item on the first invoice. When you change the discount amount on the invoice, the next invoice includes the remainder of the discount amount.

## Billing Customers Using Billing Schedules

Using the Advanced Billing feature, orders that are associated with a billing schedule generate bills according to that schedule. After bills are scheduled, you can [Create Bills from the Billing Queue](#) or [Bill Manually](#).

### Create Bills from the Billing Queue

When a billing schedule is assigned on a sale or line, NetSuite adds a bill to the billing queue at the intervals determined by the schedule. Then, you can create the bills in the billing queue.

For more information on how schedules generate bills that are added to the queue, read [Billing Schedules](#).

**Note:** Use the Bill Sales Orders page to process orders with billing schedules; use the Invoice Billable Customers page to create invoices for billable expenses, items, and time.

#### To create bills from the queue:

- Go to Transactions > Sales > Invoice Sales Orders.
- Select a customer to filter the list to show only transactions associated with the customer.  
Select **All** to show all transactions and not filter the list.
- Select the period you want to post this transaction to.  
If a period is closed, you cannot post to that period.
- NetSuite inserts today's date as the date of this transaction. You can type or pick another date.
- In the **Next Bill On or Before** field, enter a date to filter the fulfillment list by the billing date on orders.
- Choose whether you want to print or email this form.
- Choose an Accounts Receivable account. When this bill is paid, the payment posts to this account.
- Check this box if the transactions you select below have already had credit card charges performed outside NetSuite, such as by a card-swipe terminal.

**Note:** If you check this box and submit this transaction, NetSuite automatically marks the Credit Card Approved box and sets the Account field on bills you process.

- Check the **Hide Unfulfilled Orders** box to filter the list to show only fulfilled orders.



**Note:** This field shows only if the Invoice in Advance of Fulfillment preference is enabled.



**Important:** If the Invoice in Advance of Fulfillment preference is enabled, Advanced Shipping is enabled, and this box is checked, then any sales order that includes at least one item that can be billed is included in the Sales Orders reminder and is displayed on the Sales Orders to Bill list.

- Click the **Select Order Number** field to scan in transaction bar codes.



**Note:** To scan in bar codes, you must select All in the entity field.

- Check the box in the **Invoice** column next to each order you want to bill.
- Click Submit.

Bills are created for the transactions indicated.

## Bill Manually

For orders that have billings outstanding, you can create the remaining invoices manually before NetSuite creates them from the billing schedule. You can create the next invoice on the billing schedule, or you can invoice all of the remaining unbilled portion of the order.

After an invoice is created, you can print, email or fax it to your customer.

### To create remaining invoices manually:

- Go to Transactions > Sales > Enter Sales Orders > List > List.
- On the **Sales Orders** page, click the date next to the order you want to invoice.
- On the sales order:
  - Click the **Next Bill** button to create the next invoice in the billing schedule. Then, click **Save**.
  - Click the **Bill Remaining** button to invoice the remainder of the order and ignore the billing schedule. Then, click **Save**.

### To send the invoice to the customer:

- Go to Transactions > Sales > Create Invoices > List.
- On the **Invoice** list, click the date next to the invoice.
- On the invoice:
  - Click the **Save & Print** button to print the invoice.
  - Click the **Save & Email** button to email the invoice.
  - Click the **Save & Fax** button to fax the invoice.

## Viewing Invoice Schedules

The schedule page shows a list of scheduled billings for a sales transaction. This includes cash sales and invoices generated from the sale. To view an invoice schedule, click the Billing subtab on the estimate or sales order, and then click the Schedule subtab.

If a billing schedule is not associated with the sale, then the schedule shows only a single billing instance for the transaction.

If one or more billing schedules are associated with the sale, then the schedule lists each invoice to be generated according to the associated schedules.

For example, a sales order shows the following information:

- Transaction Date = 11/1/05
- Start Date = 11/15/05
- Billing instance rate = monthly billing of \$100
- Billing instance quantity = total of 4

Then, the Invoice Schedule page for this sale shows four lines, one for each billing instance:

- \$100 to be invoiced on 11/15/05
- \$100 to be invoiced on 12/15/05
- \$100 to be invoiced on 1/15/06
- \$100 to be invoiced on 2/15/06

If a sales transaction contains some lines associated with a billing schedule and also some lines that are not, then the schedule page shows the following:

- one billing instance that combines all lines not associated with a billing schedule
- one billing instance for each invoice to be generated according to the associated schedules

For example, a sales order shows the following information:

- Transaction Date = 11/1/05
- Start Date = 11/15/05
- A line item sold for \$200 with NO associated billing schedule
- A line item associated with a billing schedule:
  - Billing instance rate = monthly billing of \$100
  - Billing instance quantity = total of 4

Then, the invoice schedule for this sale shows five lines, one for each billing instance:

- \$200 to be invoiced on 11/1/05
- \$100 to be invoiced on 11/15/05
- \$100 to be invoiced on 12/15/05
- \$100 to be invoiced on 1/15/06
- \$100 to be invoiced on 2/15/06

# Advanced Billing and Advanced Shipping

When you use the Advanced Billing feature and the Advanced Shipping feature together, order processing is based on whether each item on an order is able to be fulfilled or received.

Items are able to be fulfilled or received based on the status set on the item record. Some item types have a permanent status that marks them fulfilled or received. Other item types permit you to set the status for always fulfilling and receiving them or never doing so. You can set item statuses as follows:

- **Always Fulfillable/Receivable**
  - Inventory
  - Assembly/Bill of Materials
- **Never Fulfillable/Receivable**
  - Group
  - Description
  - Discount
  - Markup
- **Allows Changes to Fulfillable/Receivable Status**
  - Kit/Package
  - Non-Inventory
  - Download
  - Gift Certificate
  - Other Charge for Sale or Resale
  - Service Item for Sale or Resale

To set the fulfillable/receivable status for an item, go to List > Accounting > Items, and click **Edit** next to a Gift Certificate, Other Charge or Service item record. On the **Preferences** subtab, check or clear the **Can be Fulfilled/Received** box, and click **Save**.

Once you set the fulfillable/receivable status for items, when you fulfill sales orders or bill sales orders, NetSuite processes the appropriate items based on the following:

- If the order includes unbilled items
- If the order includes associated billing schedules
- If the order includes fulfillable items
- Setting for the **Invoice in Advance of Fulfillment** preference

## Billing Orders Using Advanced Shipping with Advanced Billing

When you use the Advanced Shipping feature in addition to Advanced Billing, you must set the fulfillable/receivable status for items. Then, when you bill sales orders, NetSuite bills the appropriate items based on the following:

- If the order includes unbilled items
- If the order includes associated billing schedules

- If the order includes fulfillable items
- Setting for the **Invoice in Advance of Fulfillment** preference

When you view an open sales order that has an attached billing schedule, you can use the Next Bill and Bill Remaining buttons to generate invoices as follows:

- Click **Next Bill** to generate an invoice that bills the next cycle.
- Click **Bill Remaining** to ignore the schedule and issue a single invoice for the total outstanding amount.

You can also bill sales orders by going to Transactions > Sales > Bill Sales Orders. On the Bill Sales Orders page, you can filter the list of orders by date with the Next Bill On or Before field. This filter applies only to sales orders with associated billing schedules.

Bill sales orders by going to Bill sales orders by going to Transactions > Sales > Bill Sales Orders.. For more information on billing orders, read [Billing Customers Using Billing Schedules](#).

# Electronic Invoicing



**Note:** The Electronic Invoicing SuiteApp provides the ability to generate and send e-documents as well as receive and convert them into transactions records. Inbound and outbound electronic invoicing processes support multiple transaction records. For more information about supported transactions, see [Transactions and Processes Supported by the Electronic Invoicing SuiteApp](#).

To understand how to set up and use the electronic invoicing features, read the following topics:

- [Electronic Invoicing Overview](#)
  - [Outbound Electronic Invoicing](#)
  - [Inbound Electronic Invoicing](#)
  - [Transactions and Processes Supported by the Electronic Invoicing SuiteApp](#)
  - [Customizations Supported by the Electronic Invoicing SuiteApp](#)
  - [E-Document Audit Trail and Statuses](#)
  - [Electronic Invoicing Permissions and Access Levels](#)
  - [Electronic Invoicing Limitations and Best Practices](#)
- [Electronic Invoicing Administrator Guide](#)
  - [Installing and Setting Up Electronic Invoicing](#)
  - [Prerequisites for Using Electronic Invoicing](#)
  - [Installing the Electronic Invoicing SuiteApp](#)
    - [Considerations When Setting Up Electronic Invoicing](#)
    - [Electronic Invoicing Setup Tasks](#)
  - [Granting Access Permission to the E-Documents Portlet](#)
  - [Creating E-Document Packages](#)
  - [Creating E-Document Templates](#)
    - [XPath and Regex Examples for E-Document Templates](#)
    - [PEPPOL Template](#)
      - [ANZ PEPPOL Template: PEPPOL Template for Australia and New Zealand](#)
      - [Vendor Bill](#)
    - [Understanding Inbound E-Document Templates in JSON Format](#)
    - [Understanding XSD in Inbound E-Document Templates](#)
    - [Understanding XSD in Outbound E-Document Templates](#)
    - [Creating a Digital Signature Plug-in Implementation for E-Documents](#)
    - [Creating an Outbound Validation Plug-in Implementation for E-Documents](#)
    - [Creating a Custom Plug-in Implementation for E-Document Custom Data Source](#)
  - [Editing E-Document Templates](#)
  - [Creating E-Document Sending Methods](#)
  - [Selecting a Designated E-Document Sender](#)
  - [Setting Up Custom Roles to Send E-Documents](#)
  - [Customizing Roles to Restrict E-Document Generation or Sending](#)
  - [Deploying the Bulk Generation Script for E-Documents](#)

- Deploying the Script for Scheduled Sending of E-Documents
- Updating E-Document Certification Statuses
- Electronic Invoicing Inbound Email Capture
  - Setting Up Inbound Email Capture
  - Enabling Inbound Email Capture Plug-in
- Using SOAP Web Services for Inbound Processing
- Setting Up Custom Roles that can Convert Inbound E-Documents
- Inbound Validation Plug-ins
  - Creating a Custom Plug-in for Inbound E-Document Validation
- Deploying Automatic Bulk Conversion Script for Inbound E-Documents
- Electronic Invoicing User Guide
  - Overview of Outbound E-Document Process
  - Outbound E-Document Statuses
  - Assigning E-Document Packages to Customer or Vendor Records
  - Defining E-Document Email Recipients
  - Selecting E-Document Packages, Templates and Sending Methods on Transactions
  - Enabling PDF File Reference Generation
  - Generating and Regenerating E-Documents
    - Generating E-Documents for Single Transactions
    - Regenerating E-Documents for Single Transactions
    - Generating and Regenerating E-Documents in Bulk
  - Sending and Resending E-Documents
    - Sending the E-Document of a Single Transaction
    - Resending the E-Document of a Single Transaction
    - Resending E-Documents in Bulk
  - Overview of Inbound E-Document Processing
  - Inbound E-Document Statuses
  - Receiving Inbound E-Documents by Email Capture
  - Receiving E-Document XML Files from Web Service
  - Uploading Received XML Files as Inbound E-Documents
  - Converting Inbound E-Documents into Transaction Records
    - Converting an E-Document into Vendor Bill Linked to Purchase Order
    - Prerequisites and Conditions for Conversion
  - Common Scenarios in Vendor Bill Conversion
  - Converting Individual Inbound E-Documents into Vendor Bills
  - Converting Failed Inbound E-Documents
  - Compatibility of Approval Workflows with Vendor Bill Conversion
  - Canceling Inbound E-Documents
- Electronic Invoicing Errors

- [Electronic Invoicing Common Errors](#)
- [Outbound E-Document Generation Errors](#)
- [Outbound E-Document Sending Errors](#)
- [Inbound E-Document Conversion Errors](#)

## Electronic Invoicing Overview

The Electronic Invoicing SuiteApp enables you to create and use electronic documents (e-documents) in XML or JSON, for your business. The use of XML or JSON e-documents has become a standard for exchanging business information in many countries. E-documents can help you comply with legal requirements in your country, or to automate your order-to-cash and purchase-to-payment processes. By using e-documents, your company, customers, vendors, tax agencies or government regulatory bodies, can exchange information about business transactions electronically using standard data format.

The Electronic Invoicing SuiteApp supports outbound and inbound processing of e-documents. Outbound e-document processing involves generating XML or JSON e-documents from supported NetSuite transactions, and then sending the XML or JSON e-documents to your customers, vendors and tax agencies. On the other hand, inbound e-document processing involves receiving XML e-documents from your vendors and then converting the XML e-documents into NetSuite transaction records. Both inbound and outbound e-documents use templates that you can create. Sample templates for inbound and outbound e-documents are included in the SuiteApp.



**Important:** The Electronic Invoicing SuiteApp provides a framework for automating e-documents processing. It does not include native support for any country-specific requirements or e-document standards. But you can create custom country-specific e-document templates and packages using the Electronic Invoicing SuiteApp.

The Electronic Invoicing SuiteApp enables you to track received, converted, generated and sent e-documents through an audit trail on the **E-Document** subtab of a transaction record.

The SuiteApp is available free of charge when used for a single country. To use e-documents across multiple countries, contact your NetSuite account representative to purchase a license. For more information, see [Electronic Invoicing SuiteApp Availability and License Client](#).

The Electronic Invoicing SuiteApp also supports multiple languages that the NetSuite user interface can be displayed in. For more information, see the help topics [Configuring Multiple Languages](#) and [Choosing a Language for Your NetSuite User Interface](#).

## Outbound Electronic Invoicing

The Electronic Invoicing SuiteApp enables you to generate outbound XML or JSON e-documents from the following NetSuite transactions:

- Bill
- Credit Memos
- Cash Sales
- Cash Refunds
- Customer Payments
- Estimates

- Invoices
- Item Fulfillment
- Purchase Orders
- Registered Custom Transaction Types
- Returns
- Transfer Order
- Vendor Credit or Bill Credit

Generated e-documents can then be sent to your customers, vendors or tax agency, individually or in batches. The default sending method of the outbound electronic invoicing is through email. You can also create custom sending methods like SOAP web services or automate the sending of e-documents by deploying scripts. The electronic invoicing outbound process also supports e-document certification, by sending e-documents to a certification authority and then receiving the certified e-documents.

In general, the electronic invoicing outbound process is composed of the following steps:

1. The administrator creates and sets up outbound e-document templates and sending methods:
  - a. The outbound e-document template must contain the XML or JSON code and specify the supported transactions and the recipient subsidiaries. For more information, see [Creating E-Document Templates](#) and [Multi-subsidiary Support in the Outbound Process](#).
  - b. The sending method must specify the custom plug-in implementation, the supported transactions, the recipient subsidiaries, and if it will be used for certification. For more information, see [Creating E-Document Sending Methods](#), [E-Document Certification in the Outbound Process](#), and [Multi-subsidiary Support in the Outbound Process](#).
2. The administrator creates an e-document package, which specifies an e-document template and sending method. See [Creating E-Document Packages](#). Users can assign the e-document package to customers or vendors. The transactions of that customer or vendor will apply the e-document template and sending method of the e-document package. For more information, see [Assigning E-Document Packages to Customer or Vendor Records](#).
 

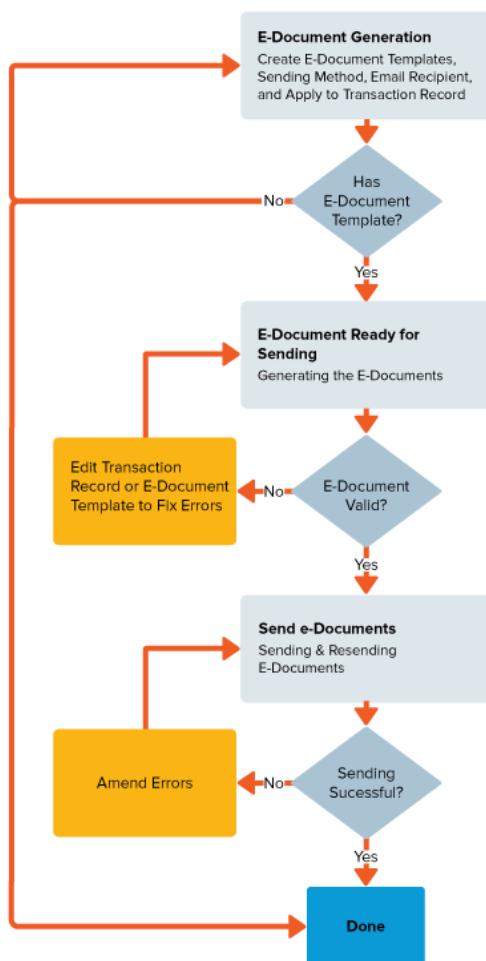
Or, users can create or edit a transaction record, then select an e-document template and an e-document sending method. For more information, see [Transactions and Processes Supported by the Electronic Invoicing SuiteApp](#) and [Selecting E-Document Packages, Templates and Sending Methods on Transactions](#).
3. Users can generate the outbound e-documents of transactions with an associated e-document template and sending method. Outbound e-documents can be generated from the following transaction types:
  - Bill
  - Credit Memos
  - Cash Sales
  - Cash Refunds
  - Customer Payments
  - Estimates
  - Invoices
  - Item Fulfillment
  - Purchase Orders
  - Returns
  - Registered Custom Transaction Types

- Transfer Order
- Vendor Credit or Bill Credit

See [Generating and Regenerating E-Documents](#).

4. Users can send generated e-documents to your customers, vendors or tax agency. For more information, see [Sending and Resending E-Documents](#).
  - a. If you need to certify the generated e-document, specify a certification sending method. Then, send the e-documents to the certification authority or tax agency. See [E-Document Certification in the Outbound Process](#).
5. Users can perform all the e-document processes by clicking the single **Process E-Document** button in a transaction , if the e-document template and sending method is configured. See [Processing E-Documents Automatically for Individual Transactions](#)

The following diagram shows the general process flow of outbound electronic invoicing in NetSuite.



For more information about the process of generating and sending outbound e-documents, read the following topics:

- [Overview of Outbound E-Document Process](#)
- [Selecting E-Document Packages, Templates and Sending Methods on Transactions](#)
- [Generating and Regenerating E-Documents](#)

- Sending and Resending E-Documents

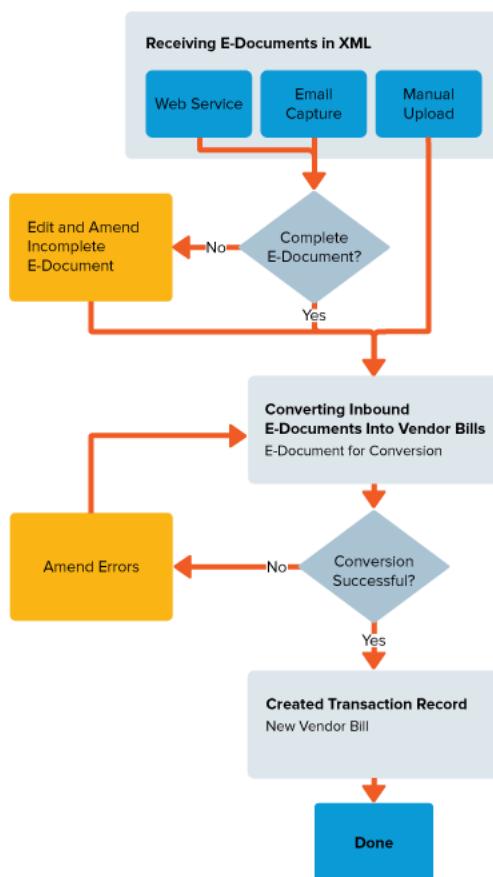
## Inbound Electronic Invoicing

You can receive inbound e-documents from your vendors or government regulatory bodies. Inbound e-documents are received in NetSuite through email or SOAP web services, which are the supported inbound channels. You can also receive e-documents from portable file storage media or devices from your vendors. Then, you can manually upload to NetSuite the received XML e-documents, which you will subsequently convert into NetSuite transactions.

An inbound e-document template must be created to convert the received e-documents into supported NetSuite transactions. A sample inbound e-document template is included with the SuiteApp.

For example, a vendor whom you initially sent a purchase order may send you an XML e-document invoice, which you receive and then convert into a vendor bill record in NetSuite.

The following diagram shows the general process flow of inbound electronic invoicing in NetSuite.



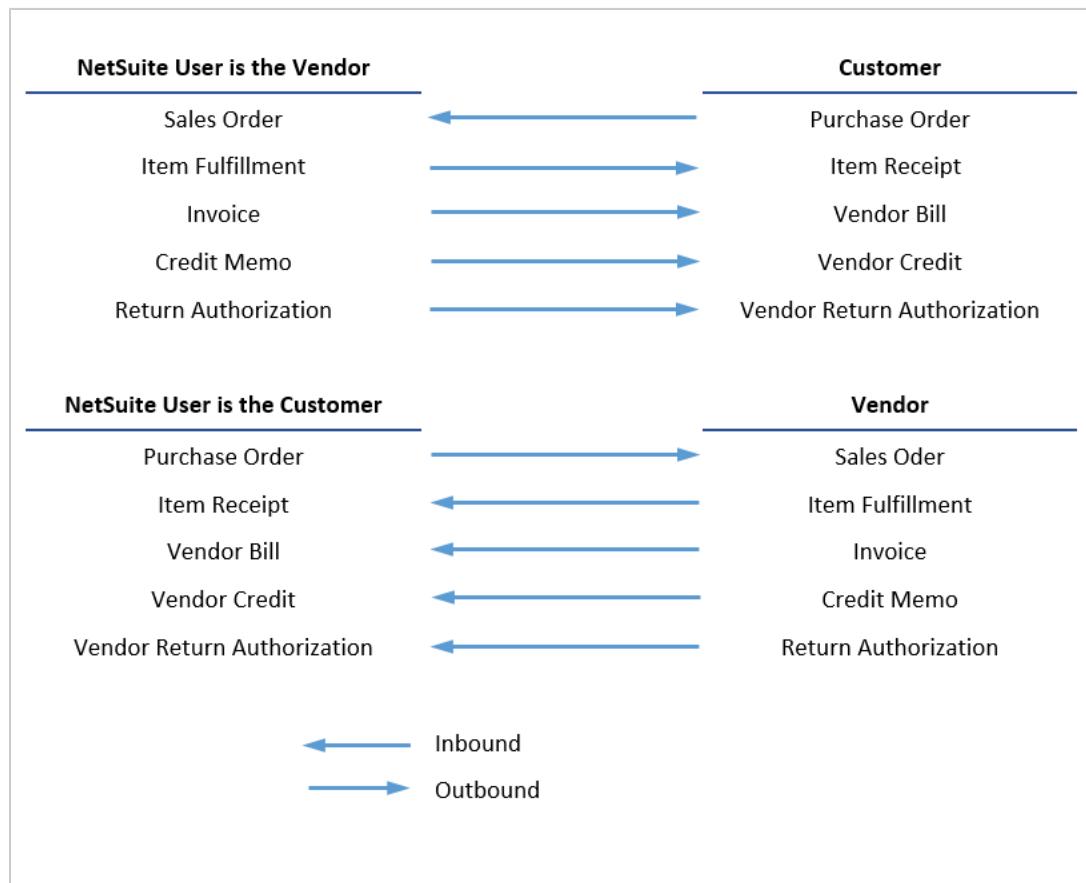
For more information about the process of receiving and converting inbound e-documents, read the following topics:

- Overview of Inbound E-Dокумент Processing
- Receiving Inbound E-Documents by Email Capture
- Uploading Received XML Files as Inbound E-Documents
- Converting Inbound E-Documents into Transaction Records

## Transactions and Processes Supported by the Electronic Invoicing SuiteApp

Inbound and outbound transaction types are better understood in the perspective of the NetSuite user, who can either be the vendor or the customer in the conduct of a transaction.

The following diagram illustrates scenarios where transaction types are transformed into other types in the perspective of the NetSuite user along the outbound and inbound e-document process.



Based on the diagram, the NetSuite user can either be a vendor or a customer, who can generate and send XML or JSON outbound e-documents or receive inbound XML e-documents using the Electronic Invoicing SuiteApp.

As the vendor (the party selling items or providing services), the NetSuite user can generate the XML or JSON e-document of an invoice, based on an initial purchase order sent by a customer. Likewise, the NetSuite user can generate the XML or JSON e-document of an item fulfillment, credit memo, or return authorization. Generated outbound XML or JSON e-documents can then be sent through email or web service to a customer or tax authority.

Outbound e-document generation for vendor bill is a special case, as it supports self-billing. This is useful if you want to charge your company in advance for goods or services acquired from a vendor, and then send the generated e-document of the bill to your vendor along with the payment.

As the customer (the party buying items or acquiring services), the NetSuite user can receive inbound XML e-documents from a vendor or tax authority through email or web service. After the inbound XML e-document is received in NetSuite, it can be converted into its corresponding transaction record. A vendor

can send to your company the e-document of an invoice, which you can receive and convert into a vendor bill. Currently, only vendor bill is supported for inbound processing.

The following table lists the transaction types currently supported by outbound and inbound electronic invoicing.

Outbound	Inbound
Bill (for self-billing)	Bill
Credit Memo	
Cash Refund	
Cash Sale	
Customer Payment	
Invoice	
Item fulfillment	
Purchase Order	
Return Authorization	
Registered Custom Transaction Types	
Transfer Order	
Vendor Credit or Bill Credit	

## Electronic Documents Dashboard SuiteApp Portlet

The Electronic Documents Dashboard SuiteApp Portlet displays the number of e-documents that are pending a process. Clicking the number under a process opens a results page with a list of e-documents pending a process, or the page for performing an e-document process.

For outbound e-document processing, the left column displays the number of:

- Outbound E-Documents for Generation
- Outbound E-Documents for Sending, or Outbound E-Documents for Certification



**Note:** If a certification sending method is specified, Outbound E-Documents for Certification is displayed instead of Outbound E-Documents for Sending.

- Certified E-documents for Sending is only displayed if at least one certification sending method is specified.
- Outbound E-Documents with Errors

A Send Failed Outbound E-Documents link is included on the outbound column to allow searching and resending of outbound e-documents that failed initial sending. For more information, see [Sending and Resending E-Documents](#).

For inbound e-document processing, the right column displays the number of:

- Inbound E-Documents for Conversion
- Convert Failed Inbound E-Documents

- Incomplete Inbound E-Documents

The inbound column also has a link for manually uploading inbound e-documents. For more information, see [Uploading Received XML Files as Inbound E-Documents](#).

E-document settings for administrator tasks can be accessed from the menu, Setup > E-Documents.

For more information about the e-documents portlet, see [Granting Access Permission to the E-Documents Portlet](#) and [Displaying the E-Documents Portlet on the Home Page](#).

## Customizations Supported by the Electronic Invoicing SuiteApp

The Electronic Invoicing SuiteApp is designed to support the following customizations:

- Custom sending methods – See [Creating E-Document Sending Methods](#).
- Scheduling of outbound e-document generation – See [Deploying the Bulk Generation Script for E-Documents](#).
- Scheduling of outbound e-document sending - See [Deploying the Script for Scheduled Sending of E-Documents](#)
- Scheduling of inbound e-document conversion - See [Deploying Automatic Bulk Conversion Script for Inbound E-Documents](#)
- Customizing the preferences for processing e-documents automatically – See [Processing E-Documents Automatically for Individual Transactions](#)

## E-Document Audit Trail and Statuses

The **E-Document Audit Trail** subtab logs all changes to an e-document. The Audit Trail is available on the **E-Document** subtab of a transaction record. The e-document status describes the current processing state of a transaction or e-document. E-document statuses are indicated in the **E-Document Status** field on the **E-Document** subtab. For more information about e-document statuses, see [Outbound E-Document Statuses](#) and [Inbound E-Document Statuses](#).

To understand e-document processing, including inbound and outbound process flows, see the following topics:

- [Overview of Outbound E-Document Process](#)
- [Overview of Inbound E-Document Processing](#)

## Electronic Invoicing Permissions and Access Levels

The following tables show the permissions and access levels for features provided by the Electronic Invoicing SuiteApp:

### Administrator and Setup Tasks

Tasks or Functionality	Role and Permission
<a href="#">Creating E-Document Packages</a>	<p>By default, only the Administrator can create, edit, or delete e-document packages.</p> <p>All other roles can only view e-document packages.</p>

Tasks or Functionality	Role and Permission
Creating E-Document Templates	By default, only the Administrator can create, edit, or delete e-document template records.  All other roles can only view e-document template records.
Creating E-Document Sending Methods  Creating a Custom Plug-in Implementation for Sending E-Documents	By default, only the Administrator can create, edit, or delete e-document sending methods.  All other roles can only view e-document sending methods.
Selecting a Designated E-Document Sender	By default, only the Administrator can select a designated e-document sender.
Assigning E-Document Packages to Customer or Vendor Records	Roles with access to customer and vendor and contact records can assign e-document packages to customer and vendor records.
Creating a Custom Plug-in for Inbound E-Document Validation	By default, only the Administrator can create a custom plug-in for inbound e-document validation.
Processing E-Documents Automatically for Individual Transactions	By default, only the Administrator can select the required <b>E-Document Automation Type</b> from the Automatic E-Invoicing tab on the E-Document Preferences page for an individual transaction.

### Outbound Processing Permissions and Access Levels

Tasks or Functionality	Role and Permission
Defining E-Document Email Recipients	Roles with <b>Edit</b> or <b>Full</b> access to customer and vendor contact records can define e-document email recipients on customer or vendor records.
Setting Up Custom Roles to Send E-Documents  Customizing Roles to Restrict E-Document Generation or Sending	The Administrator can restrict the default permission of roles to generate or send outbound e-documents.
Generating E-Documents for Single Transactions  Regenerating E-Documents for Single Transactions	Roles with minimum permission to view supported transaction records, can generate and regenerate e-documents for single transactions.  An administrator can grant custom roles the permission to generate e-document for sending to customers and vendors.
Deploying the Bulk Generation Script for E-Documents	Only the Administrator role can configure the script used to generate and regenerate e-documents in bulk.
Sending the E-Document of a Single Transaction  Resending the E-Document of a Single Transaction	Roles with minimum permission to view supported transaction records, can send and resend e-documents for single transactions.
Setting Up Custom Roles to Send E-Documents  Resending E-Documents in Bulk	The following roles can send and resend e-documents in bulk:  <ul style="list-style-type: none"> <li>■ A/P Clerk</li> <li>■ A/R Clerk</li> <li>■ Accountant</li> <li>■ Administrator</li> <li>■ Bookkeeper</li> <li>■ CFO</li> </ul>

Tasks or Functionality	Role and Permission
	<p>An administrator can grant custom roles the permission to send bulk e-documents.</p> <p>An administrator can grant custom roles the permission to send e-documents to customers/vendors.</p>
<p>Template and Sending Method Auto-selection For more information about Template and Sending Method Auto-selection, see <a href="#">Multi-subsidiary Support in the Outbound Process</a></p>	<p>Roles with <b>Edit</b> or <b>Full</b> access to customer and vendor records can have access to template and sending method auto-selection feature.</p>

### Inbound Processing Permissions and Access Levels

Tasks or Functionality	Role and Permission
<a href="#">Setting Up Custom Roles that can Convert Inbound E-Documents</a>	Only the Administrator role can set the permission to perform conversion of inbound e-documents
<a href="#">Uploading Received XML Files as Inbound E-Documents</a>	The following roles can upload inbound e-documents manually: <ul style="list-style-type: none"> <li>■ A/P Clerk</li> <li>■ A/R Clerk</li> <li>■ Accountant</li> <li>■ Bookkeeper</li> <li>■ CFO</li> <li>■ Administrator</li> <li>■ Custom Roles (with permission)</li> </ul>
<a href="#">Converting Individual Inbound E-Documents into Vendor Bills</a>  <a href="#">Converting an Inbound E-Document Without a Purchase Order Number</a>	The following roles can convert inbound e-documents into transaction records: <ul style="list-style-type: none"> <li>■ A/P Clerk</li> <li>■ A/R Clerk</li> <li>■ Accountant</li> <li>■ CFO</li> <li>■ Administrator</li> <li>■ Custom Roles (with permission)</li> </ul>
<a href="#">Converting Failed Inbound E-Documents</a>  <a href="#">Deploying Automatic Bulk Conversion Script for Inbound E-Documents</a>	The following roles can convert inbound e-documents into transaction records in bulk: <ul style="list-style-type: none"> <li>■ A/P Clerk</li> <li>■ A/R Clerk</li> <li>■ Accountant</li> <li>■ Bookkeeper</li> <li>■ Buyer</li> <li>■ CFO</li> <li>■ Administrator</li> <li>■ Custom Roles (with permission)</li> </ul>
<a href="#">Canceling Inbound E-Documents</a>	The following roles can cancel inbound e-documents: <ul style="list-style-type: none"> <li>■ A/P Clerk</li> <li>■ A/R Clerk</li> </ul>

Tasks or Functionality	Role and Permission
	<ul style="list-style-type: none"> <li>■ Accountant</li> <li>■ Bookkeeper</li> <li>■ Buyer</li> <li>■ CFO</li> <li>■ Administrator</li> <li>■ Custom Roles (with permission)</li> </ul>
Step 4 of Prerequisites for Using Electronic Invoicing	Only the Administrator role can designate an employee or group of employees who will receive the email notification upon completion of batch conversion.

### Vendor Credit or Bill Credit Permissions and Access Levels

Descriptions	Required Minimum Permissions	Access Levels
For Custom roles that can only create Vendor Credit or Bill Credit transactions but do not perform any electronic invoicing tasks.	E-document sending method > Custom Record subtab	View
	Perform search > List subtab	View
For Custom roles that can view or edit Vendor Credit or Bill Credit transactions but do not perform any electronic invoicing tasks.	E-document sending method > Custom Record subtab	View
	E-document templates > Custom Record subtab	View
	Perform search > List subtab	View
For custom Roles that can create, edit or view Vendor Credit or Bill Credit transactions and also perform the following electronic invoicing tasks: <ul style="list-style-type: none"> <li>■ Allow Manual generation of E-documents</li> <li>■ Allow sending of E-documents for certification</li> <li>■ Allow sending of E-documents</li> </ul>	E-document sending method > Custom Records subtab	View
	E-document templates > Custom Records subtab	View
	E-document audit trail > Custom Records subtab	Full/Edit
	Setup Company > Setup subtab	View
	Vendor record > List subtab	View
	Perform Search > List subtab	View
	Find transaction > Transaction subtab	View

## Electronic Invoicing SuiteApp Availability and License Client

The Electronic Invoicing SuiteApp enables you to use and manage e-documents for various transaction types. The SuiteApp has two major components, inbound and outbound e-document processing. Outbound e-document processing enables you to generate and send e-documents to other parties such as customers, vendors, or tax agencies. Inbound e-document processing enables you to receive e-documents from vendors and then convert them into bills. For more information, see [Transactions and Processes Supported by the Electronic Invoicing SuiteApp](#).

The Electronic Invoicing SuiteApp can be installed for free in any NetSuite account.

For free use without a paid license in OneWorld accounts, you must select a country in the **E-Document Country for Free Use** field on the Company Information page from Setup > Setup Tasks > Company > Company Information. You must ensure that the country you select is:

- The same country where you will send e-documents to
- The same country of the vendor's subsidiary

However, if you want to send e-documents to multiple countries and convert e-documents received from other countries, you must purchase a license from your NetSuite account manager.

The NetSuite SuiteApps License Client (Bundle ID: 116144) must be installed before you install the Electronic Invoicing SuiteApp (Bundle ID: 436209).

The NetSuite SuiteApps License Client controls an account's access to the electronic invoicing features. Upon installation, the NetSuite SuiteApps License Client communicates with the NetSuite SuiteApps License Server to obtain active license information. If you have an active license, the E-Document Country for Free Use can be left blank.

You cannot use key features in inbound and outbound e-document processing such as sending e-documents to multiple countries and converting received e-documents from other countries if:

- The NetSuite SuiteApps License Client is not installed, or
- The NetSuite SuiteApps License Client is installed but your license is expired in OneWorld accounts.



**Note:** It takes a maximum of 30 minutes to update license status of Electronic Invoicing SuiteApp after license refresh.

For more information, see the help topic [NetSuite SuiteApps License Client](#).

## Electronic Invoicing Limitations and Best Practices

Read the following topics to know the current limitations of the Electronic Invoicing SuiteApp, and to be guided by best practices for setting up and using electronic invoicing features.

### Electronic Invoicing Limitations

#### General limitations of the Electronic Invoicing SuiteApp:

- Only FreeMarker is supported for outbound e-document templates.
- Only the following transaction types are currently supported:
  - Bill
  - Cash sales
  - Cash refunds
  - Credit memos
  - Customer payment
  - Estimates
  - Invoices
  - Item Fulfillment
  - Purchase orders
  - Returns
  - Registered Custom Transaction Types

- Transfer Order
- Vendor Credit or Bill Credit
- Mass download of e-documents is not supported.
- Intercompany transfer order is not supported.
- Creating the following records using the create new icon is not supported.
  - E-Document Template (on all transaction records)
  - E-Document Package (on vendor/customer records, transfer order transaction records and basic and journal style custom transaction types)
  - E-Document Sending Method (on all transaction records)
  - E-Document Email Recipient (on transfer order transactions and basic and journal style custom transaction types)

#### **Limitations in outbound e-document processing:**

The system can send an e-document by email to a maximum of 10 recipients for each customer.



**Note:** The system counts each contact added as a recipient. If you add the same contact multiple times, each instance is considered one recipient.

#### **Limitations in inbound e-document processing:**

- Only one XML file and one PDF file reference can be processed per email received.
- Inbound e-documents can only be uploaded manually, one at a time.
- Currently, only vendor bills can be created or converted from inbound e-documents.

## **Electronic Invoicing Best Practices**

Perform the following best practices to prevent errors and performance issues:

- Do not create or add MR script deployments. Maintain the default single script deployments of the Electronic Invoicing SuiteApp, which are: Generate E-Document Content MR, Automatic Send E-Document MR, and Convert Inbound E-Document MR.
- When assigning an e-document package to a customer or vendor, make sure that the customer or vendor has contact records if the e-document package uses an email sending method. To prevent validation errors caused by missing e-document email recipients, it is that you create the contact records first before assigning the e-document package to the customer or vendor.
- When using CSV import to assign e-document packages to customers or vendors, be sure to include email recipients for e-document packages that use an email sending method. The system will encounter an error when sending an e-document that has no email recipient.
- When sending e-documents in bulk, NetSuite that you first make sure that custom roles have the required permissions and access to use the bulk sending feature. See [Setting Up Custom Roles to Send E-Documents](#).
- When sending e-documents in bulk, NetSuite that you use the filters to limit the number of e-documents to process at a time.
- When setting up sending method custom plug-ins:
  - There is no need to include loading of customer and invoice records as well as vendor and purchase order records to retrieve data within your custom plug-in. This information is already provided by the Electronic Invoicing SuiteApp.
  - There is no need to load recipients within your custom plug-in . This information is already provided by the Electronic Invoicing SuiteApp.

- Avoid loading e-document recipients within your custom plug-in for sending email. When an e-document package with an email sending channel is assigned to a customer or vendor, the Electronic Invoicing SuiteApp automatically performs a validation check to make sure customer or vendor records have recipients with valid email addresses. If your custom plug-in loads e-document recipients, those recipients will not be included in the validation.
- Be aware of SuiteScript 2.0 governance and time limits. Suitelet limits apply to individual sending. Map function limits (Map/Reduce script type) apply to bulk sending. See the following topics:
  - [SuiteScript 2.x Suitelet Script Type](#)
  - [Map/Reduce Governance](#)
- Test your e-document sending plug-ins thoroughly before using them on live data.
- Avoid data leaks when sending e-documents by email or sending to third party SOAP web services. Data leaks can be avoided by making sure codes are properly reviewed. Also make sure you are not sending more information than what is required.
- Use appropriate and specific error messages in your scripts. Error messages should inform users when a problem occurs, help users understand why the problem occurred, and an action that can fix the problem.

## Electronic Invoicing Administrator Guide

Only the Administrator role can install the Electronic Invoicing SuiteApp and set up the custom records and templates required for generating and sending e-documents. The administrator must also deploy the script for generating e-documents in bulk.

The following topics are intended for administrators:

### General Setup Tasks

- [Installing the Electronic Invoicing SuiteApp](#)
- [Prerequisites for Using Electronic Invoicing](#)
- [Considerations When Setting Up Electronic Invoicing](#)
- [Electronic Invoicing Setup Tasks](#)
- [Defining the E-Dокумент Country for Free Use](#)
- [Advanced PDF/HTML Template](#)
- [Granting Access Permission to the E-Documents Portlet](#)
- [Understanding E-Documents and E-Document Packages](#)
- [Creating E-Document Packages](#)
- [Multi-subsidiary Support in the Outbound Process](#)
- [Creating E-Document Templates](#)
- [Editing E-Document Templates](#)

### Outbound E-Document Processing Setup Tasks

- [E-Document Certification in the Outbound Process](#)
- [Creating E-Document Sending Methods](#)
- [Setting Up an Email Sending Method for E-Documents](#)
- [Creating Custom Methods for Sending E-Documents](#)
- [Selecting a Designated E-Document Sender](#)
- [Setting Up Custom Roles to Send E-Documents](#)

- Customizing Roles to Restrict E-Document Generation or Sending
- Deploying the Bulk Generation Script for E-Documents
- Deploying the Script for Scheduled Sending of E-Documents
- Updating E-Document Certification Statuses
- Processing E-Documents Automatically for Individual Transactions

### **Inbound E-Document Processing Setup Tasks**

- Electronic Invoicing Inbound Email Capture
- Using SOAP Web Services for Inbound Processing
- Setting Up Custom Roles that can Convert Inbound E-Documents
- Inbound Validation Plug-ins
- Deploying Automatic Bulk Conversion Script for Inbound E-Documents

See also the following topics:

- [Electronic Invoicing Overview](#)
  - [Understanding E-Documents and E-Document Packages](#)
  - [Electronic Invoicing Permissions and Access Levels](#)
  - [Electronic Invoicing Limitations and Best Practices](#)
- [Electronic Invoicing Errors](#)
  - [Electronic Invoicing Error Codes](#)
  - [Outbound E-Document Generation Errors](#)
  - [Outbound E-Document Sending Errors](#)

## **Installing and Setting Up Electronic Invoicing**

Read the following topics to understand how to install and set up Electronic Invoicing:

- [Prerequisites for Using Electronic Invoicing](#)
- [Installing the Electronic Invoicing SuiteApp](#)
- [Considerations When Setting Up Electronic Invoicing](#)
- [Electronic Invoicing Setup Tasks](#)
- [Electronic Invoicing Limitations and Best Practices](#)

## **Prerequisites for Using Electronic Invoicing**

Before installing the Electronic Invoicing SuiteApp, you must complete the following steps:

1. Install the NetSuite SuiteApps License Client (Bundle ID: 116144). For more information, see the help topic [NetSuite SuiteApps License Client](#).

The NetSuite SuiteApps License Client (Bundle ID: 116144) must be installed before you install the Electronic Invoicing SuiteApp (Bundle ID: 436209). To install, see the help topic [Installing a Bundle](#).

The NetSuite SuiteApps License Client controls an account's access to the Electronic Invoicing features. Upon installation, the NetSuite SuiteApps License Client communicates with the NetSuite SuiteApps License Server to obtain active license information.

If the account has an active license, you can use Electronic Invoicing features for multiple countries. If the account does not have an active license, you can still use the Electronic Invoicing

SuiteApp for free, but your account can generate and send e-documents to only one country, and convert into vendor bills received e-documents from the same country. See [Defining the E-Document Country for Free Use](#) to set up your account for free use of the Electronic Invoicing SuiteApp.

If the NetSuite SuiteApps License Client is not installed, your OneWorld accounts can neither generate and send e-documents nor convert received e-documents into vendor bills.

If the NetSuite SuiteApps License Client is installed, your single instance accounts can generate, certify or send all outbound e-documents and convert inbound e-documents into bills.

2. Make sure the following features are enabled in your account:

- Custom Records
- Custom Transactions.
- Advanced PDF/HTML Templates
- Client SuiteScript
- Server SuiteScript
- Web Services

To enable features, see the help topic [Enabling Features](#).

3. Make sure the Company Information page has a return email address.



**Important:** If the **Return Email Address** field is blank, the system will encounter script errors when accessing Electronic Invoicing features.

- a. Go to Setup > Company > Company Information.
- b. In the **Return Email Address** field, enter a valid email address for the company.
- c. Click **Save**.



**Note:** If you have a OneWorld account , you must set up the Return Email per subsidiary that uses the Electronic Invoicing SuiteApp.

4. Define the Recipient of E-Dокумент Notifications on the Company Information page.
  - a. Go to Setup > Company > Setup Tasks > Company Information.
  - b. In the **Recipient of E-Dокумент Notifications** field, enter the email of the user whom you want to receive notifications about e-document processing.
  - c. Click **Save**.

If no notification recipient is defined, the system will send notifications about e-document processes to all active administrators.

## Installing the Electronic Invoicing SuiteApp

1. Make sure all prerequisites are met. See [Prerequisites for Using Electronic Invoicing](#).
2. Go to Customization > SuiteBundler > Search & Install Bundles.
3. In the Keywords box, enter the bundle ID or name:
  - Bundle ID: 436209
  - Bundle Name: Electronic Invoicing
4. Click **Search**.
5. Click the link for the Electronic Invoicing SuiteApp.
6. On the Bundle Details page, click **Install**.

The Electronic Invoicing SuiteApp is a managed SuiteApp. When improvements or new features are added to the SuiteApp, your account is automatically updated.

But in sandbox, the update is not automatic. Your administrator must manually update the SuiteApp to get the latest version.

For information about purchasing a license for the Electronic Invoicing SuiteApp, contact your NetSuite account representative.

## Considerations When Setting Up Electronic Invoicing

Before you set up an account to use the Electronic Invoicing SuiteApp, it is that you consider the following questions to help you decide on the settings to apply:

### For outbound e-document processing:

- To whom should I send my e-documents?
- What e-document templates do I need to use?
- How do I want to send my e-documents to stakeholders?
- Do I want to generate outbound e-documents one by one or in bulk?

### For inbound e-document processing:

- From whom should I receive e-documents?
- How do I want to receive the inbound e-documents from vendors or other parties?
- Do I want to convert inbound e-documents into vendor bills, one by one or in bulk?

Be sure to also read [Electronic Invoicing Limitations and Best Practices](#).

## Electronic Invoicing Setup Tasks

After installing the Electronic Invoicing SuiteApp, an administrator must accomplish some common setup tasks. As the Electronic Invoicing SuiteApp is composed of two major components, the administrator must also complete specific setup tasks for outbound e-document processing and inbound e-document processing. Some inbound and outbound settings have a common record.

### Common Setup Tasks for Outbound and Inbound E-Document Processing:

1. Install the Electronic Invoicing SuiteApp.
2. Define the country for free use (required only if you choose not to purchase a license). See [Defining the E-Document Country for Free Use](#).
3. Create an e-document package and assign it to customer records (for outbound) or vendor records (for inbound). An e-document package record contains inbound and outbound e-document templates, outbound e-document sending methods, and inbound validation plug-in. See [Creating E-Document Packages](#).
4. Create e-document templates. An e-document template record contains templates for both outbound and inbound processing. See [Creating E-Document Templates](#).

### Setup Tasks for Outbound Processing:

1. Create e-document sending methods. See [Creating E-Document Sending Methods](#).
2. (Optional) Select a designated e-document sender. See [Selecting a Designated E-Document Sender](#).

### Setup Tasks for Inbound Processing:

1. (Optional) Create an inbound e-document validation plug-in. See [Inbound Validation Plug-ins](#).
2. (Optional) Create an XSD file for automatic template selection. See the help topic [Understanding XSD in Inbound E-Document Templates](#).
3. Enable the Email Capture Plug-in. See [Electronic Invoicing Inbound Email Capture](#).
4. (Optional) Use SOAP web services to receive inbound e-documents. See [Using SOAP Web Services for Inbound Processing](#)
5. Update item records.
6. Update vendor records.
7. Define the schedule for automatic bulk conversion. See [Deploying Automatic Bulk Conversion Script for Inbound E-Documents](#).

## Defining the E-Document Country for Free Use

The Electronic Invoicing SuiteApp can be used for free if you intend to send e-documents to only one country.

To set up your account for free use of the Electronic Invoicing SuiteApp, you must go to the Company Information page and define the country to which you will be sending e-documents.

This is the first step in setting up your account so that you can generate e-documents and send them to customers or vendors. You can send e-documents only to those customers whose country billing addresses match the e-document country for free use.

For Single Instance accounts that have NetSuite SuiteApp License Client installed, it is not required to define the country. The country selected in the **E-Document Country for Free Use** field is not applicable for Single Instance accounts. Single Instance accounts have single business in only the country where their business operates.

**Note:** For single instance accounts that have NetSuite SuiteApps License Client installed, you need not define the country. The **E-Document Country for Free Use** field is disabled and populates the company's country because these accounts have a single business in only the country where their business operates. The Electronic Invoicing SuiteApp can be used without any license for this scenario.

## Defining E-document Country for Free Use

You can follow the following steps to select a country for free use.

### To define the e-document country for free use:

1. Go to Setup > Company > Company Information.
2. In the **E-Document Country for Free Use** field, select the country to which you want to send e-documents. Only countries with subsidiaries are available in the list. This setting applies to all subsidiaries of the parent company.

**Note:** If you have a license for multi-country use of the Electronic Invoicing SuiteApp, there is no requirement to select a free country. You can use it for all countries without any restriction. Hence, the field will be unavailable.

3. Click **Save**.

After defining the e-document country for free use, you can create e-document packages. See [Creating E-Document Packages](#).



**Note:** If the license is inactive in One World Account, E-document Free Country Use field will be available. However, as it gets active, the field will be unavailable. In single instance accounts, the E-document Free Country field is always disabled irrespective of license status.

## Advanced PDF/HTML Template

You can create a new [Advanced PDF/HTML Templates](#) or customize an existing Advanced PDF/HTML template. Along with it, you can also add an element for displaying the QR Code in a generated E-Document. From the code snippet below, refer to "custbody\_qr\_string" which is the field ID that holds the QR string such as base 64 string value after successful generation of QR. You can use the free marker syntax with "if condition" to render only if the value is present in that field.

To create an Advanced PDF/HTML template, go to Customization>Forms>Advanced PDF/HTML Templates.

```

1 <if record.custbody_qr_string?has_content>
2   <table style="width: 100%;">
3     <tr>
4       <td align="center">
5         <barcode codetype="qrcode" showtext="false" height="150" width="150" value="${record.custbody_qr_string}" />
6       </td>
7     </tr>
8   </table>
9 </if>
```

### Setting Advanced PDF/HTML Template

1. Go to Setup > Company > Subsidiaries for OW account, and Setup > Company > Company Information for SI account.
2. The field "Advanced PDF/HTML Template" helps choose a template from the list of advanced PDF/HTML templates. You can select one of the advanced PDF/HTML templates to generate the E-Document in the particular format.

If no template is chosen in subsidiary record/company information, the generated e-document will take the template selected on the "Print Template" field in transaction form. In case of standard forms, it takes the default print template.



**Note:** The selection of the template in subsidiary record/company information page is only applicable for invoice and credit memo.

## Granting Access Permission to the E-Documents Portlet

Roles that have default permissions to access and perform e-document processes, will be able to add and display the e-documents portlet on the Home page. These roles include:

- AR Clerk
- AP Clerk
- Accountant
- Administrator
- Bookkeeper
- CFO
- Custom Role for Accounting Center.

### To grant roles the access permission to the e-documents portlet:

1. Go to Customizations > Scripting > Script Deployments.
2. On the Script Deployments page, click the Filters plus sign (+) to display the filters. In the **Type** filter, select **Portlet** from the dropdown list.
3. Click the Edit link of the E-Document Dashboard PT script.
4. On the script deployment page of the portlet, go to the **Audience** subtab. In the **Roles** dropdown list, select the roles that you want to grant permission to access the e-documents portlet.
5. Click **Save**.

## Understanding E-Documents and E-Document Packages

The following topics describe e-documents and e-document packages according to how they are used by NetSuite and the Electronic Invoicing SuiteApp.

### E-Document

An e-document is a NetSuite transaction represented as an XML or JSON document generated according to a specified standard.

E-documents are transmitted to interested external parties, such as customers, vendors, or tax agencies. E-documents can be downloaded or shared through email, or transmitted through a web service or other custom method.

Some countries legally recognize these XML or JSON files as official documents, whereas other countries recognize the printed transactions as the officially accepted documents.

For information about currently supported transactions, see [Electronic Invoicing Limitations and Best Practices](#).

### E-Document Package

An e-document package defines a set of formats used for various transactions to ensure clear communication between all stakeholders of a specific business process, such as vendors communicating with customers, and vendors communicating with tax authorities.

For example, the PEPPOL e-document package defines all possible transactions between vendors and government agencies (sales orders, invoices, credit memos, return authorizations, and others).

An e-document package specifies the following:

- the file format and template
- the communication medium for transmitting the e-documents
- the process necessary for ensuring authenticity and privacy of the information

An e-document package is usually defined by a tax authority or an industry governing body. Some e-document packages are defined by independent entities such as [PEPPOL](#) and [ISO](#).

## Creating E-Document Packages

An administrator must first create e-document packages so that users can assign them to customer and vendor records.

An e-document package defines the e-document templates and sending methods to be used for the customer's or vendor's transactions.

The Electronic Invoicing SuiteApp includes a default e-document package record to which you can associate e-document templates and sending methods. The default e-document package cannot be edited or deleted.

### To create an e-document package:

1. Go to Setup > E-Documents > E-Document Package > New.
2. In the **Name** field, enter a name for the e-document package.
3. In the **Inbound Validation Plugin** field, select a validation plug-in.  
for more information about the validation plug-in, see [Inbound Validation Plug-ins](#).
4. (Optional) In the **Description** field, enter text that describes this e-document package.
5. Click **Save**.

You can now assign e-document templates and sending methods to the e-document package. For more information, see [Creating E-Document Templates](#) and [Creating E-Document Sending Methods](#).

## Multi-subsidiary Support in the Outbound Process

If you are using OneWorld accounts and have multi-subsidiary customer enabled, outbound e-documents templates and sending methods can be associated with subsidiaries as preferred or necessary. To do this, the administrator must specify subsidiaries in the **Subsidiary** field on an outbound e-document template or sending method record. The transactions of the selected subsidiaries will use the outbound template and sending method. For more information, see [Creating E-Document Templates](#) and [Creating an E-Document Sending Method Record](#).

If only one outbound template or one sending method is associated with a customer or vendor, you can set automatic selection of that template and sending method for the transactions of the customer or vendor. To do this, edit a customer or vendor record and go to the E-Document subtab, then check the **Template and Sending Method Auto-selection** box and save the record. The template and sending method will be automatically selected as the default value of the E-Document Template or E-Document Sending Method field on the transaction records of the customer or vendor.

## Creating E-Document Templates

In outbound processing, the e-document template maps what data in a NetSuite transaction record will populate which elements in the XML or JSON file that will be generated and sent to customers or vendors.

In inbound processing, the opposite is implemented. The e-document template maps what elements in the received XML file will populate which data fields in the NetSuite transaction record that will be created from the XML file.

Both outbound and inbound e-document templates can be created on the E-Document Templates record. In this record, you can define the name of the e-document template, the e-document package that the template will be applied to, the applicable transaction types for the template, and the template content.

The template content is different for an outbound and inbound transaction. An outbound e-document template should have content in XML or JSON freemarker template, whereas an inbound e-document template will have content as JSON objects for the field mapping.

An e-document template can be used for both outbound and inbound e-documents, given that the right transaction types are selected and the template content for both outbound and inbound are defined. You can create or customize e-document templates for each country you do business in or for certain industries, according to specified standards.

Use FreeMarker to create the template content for both outbound and inbound e-documents. For more information about scriptable templates and FreeMarker, see the following topics:

- [Scriptable Templates](#)
- [FreeMarker Data Model](#)
- [FreeMarker Syntax](#)

Or, you can customize the sample outbound and inbound e-document templates included with the Electronic Invoicing SuiteApp.

The sample outbound template is based on the Brazil NF-e standard. It can be downloaded from the File Cabinet in the Sample Templates folder of the SuiteApp in Documents > File Cabinet > SuiteBundles > Bundle 436209. The following table lists the sample e-document templates and details.

Filename	Description
customer_payment_sample.xml	sample outbound e-document template with tags for customer payment
inbound_template_expenses_items_sample.txt	sample inbound e-document template with tags for items and expenses
inbound_template_sample.txt	sample inbound e-document template with tags for items only
NFe_sample.xml	sample outbound e-document template for Brazil
itemfulfillment_sample.xml	sample outbound e-document template with tags for customer and company information, includes PO number and items
vendorbill_sample.xml	sample outbound e-document template with tags for vendor and company information, includes items and expenses

### To create an e-document template:

1. Go to Setup > E-Documents > E-Document Templates > New.
2. In the **Name** field, enter a name for the template.
3. (Optional) In the **Description** field, enter text that describes this template.
4. In the **E-Document Package** field, select the e-document package that this template will be associated with.

For more information, see [Creating E-Document Packages](#).



**Note:** If the template is for outbound, the e-document package that this template will be included in, must be assigned to corresponding customers or vendors.

5. (Optional) Select a plug-in implementation in the **Digital Signature Plugin Implementation** field. The selected plug-in enables you to include a Digital Signature to the e-documents that will be generated using the template. For more information, see [Creating a Digital Signature Plug-in Implementation for E-Documents](#).
6. (Optional) Select a plug-in implementation in the **Outbound Validation Plug-in Implementation** field. The selected plug-in implementation enables you to validate the outbound e-document generation process. For more information, see [Creating an Outbound Validation Plug-in Implementation for E-Documents](#).
7. (Optional) Select a plug-in implementation in the **Custom Data Source Plugin Implementation** field. The selected plug-in enables you to include a custom data source in the template for adding more field values to e-documents that will be generated using the template. For more information, see [Creating a Custom Plug-in Implementation for E-Document Custom Data Source](#).

8. In the **Transaction Type** field, select one or more transaction types for which this template will be used. To select multiple transaction types, press and hold the **Ctrl** key while selecting the transaction types.



**Note:** Make sure that you select the right transaction type that match the template you are creating. If you are creating an outbound template, you must select transaction types that are applicable to outbound processing. See [Transactions and Processes Supported by the Electronic Invoicing SuiteApp](#) to know which transaction types you can select.

Selecting a transaction type for either outbound or inbound will make its corresponding Template Content field required. For example, if you select the outbound transaction type Invoice, the **Template for Outbound E-Documents** field will be required. On the other hand, if you select Bill, which is an inbound transaction type, the JSON **Field Mapping for Inbound E-Documents** field will be required.

The selected transaction types cannot be modified after the template has been used in a transaction. You must remove the e-document template from the transaction before you can modify this field.

9. (For an outbound template) In the **Subsidiary** field, select the subsidiaries that you want to associate with this template. To select multiple subsidiaries, press and hold the **Ctrl** key while selecting the subsidiaries.

If only this template is associated with a subsidiary, the supported transactions of that subsidiary will display this template on the E-Document Template field on the E-Document subtab. For more information, see [Multi-subsidiary Support in the Outbound Process](#).

10. In the **Restrict Editing of Transactions with E-document Status** field, select any of the following e-document statuses:

- Sent
- Sending
- Ready for Sending
- Certification in Progress

To select multiple e-document statuses, press and hold the **Ctrl** key while selecting the statuses. Transactions with the selected e-document status will be locked for editing if this template is associated with them. Those transactions will display a banner message indicating that the transaction cannot be edited. Editing is only locked on the user interface, transactions can still be edited through script.

11. Under the Template Content section, do the following:

- For outbound processing, select XML or JSON format from the **Content Type** field of the e-document for which the template is generated.
- If you selected transaction types for outbound processing, enter the XML or JSON content of the outbound e-document template in the **Template for Outbound E-Documents** field.
- If you selected transaction types for inbound processing, enter JSON content in the **Field Mapping for Inbound E-Documents** field. For more information about the inbound e-document template in JSON format, see [Understanding Inbound E-Document Templates in JSON Format](#).

If the template record you are creating is for both outbound and inbound transaction types, make sure that you enter content in both Template for Outbound E-Documents, and Field Mapping for Inbound E-Documents fields. Otherwise, error messages will be displayed when you save the template record.

12. Create an XSD file to enable the system to automatically assign the right template to a received XML file after validating required tags and attributes in the XML document. Then, upload the XSD file you created to the File Cabinet. For more information and a sample of the XSD file, refer to [Understanding XSD in Inbound E-Document Templates](#).
13. Return to the E-Documents Templates page, and under Template Content, select the XSD file you created from the **Inbound XSD File** dropdown field.
14. Create an XSD file to enable the system to validate required tags and attributes in the generated XML e-document. Then, upload the XSD file you created to the File Cabinet. For more information and a sample of the XSD file, refer to [Understanding XSD in Outbound E-Document Templates](#).
15. Return to the E-Documents Templates page, and under Template Content, select the XSD file you created from the **Outbound XSD File** dropdown field.
16. Enter the folder ID or folder path of a File Cabinet in the **XSD Folder** field. The folder contains the XSD files imported by the outbound XSD file.
17. (Optional) Add path and regex validation expressions to use for validating the template.  
There is no need to add node() at the end to get the value.  
See [XPath and Regex Examples for E-Document Templates](#).
18. Click **Save**.

After creating the e-document template, you can include it in an e-document package, which is then assigned to a customer or vendor. See [Creating E-Document Packages](#)

## XPath and Regex Examples for E-Document Templates

The examples provided in the table can be used to validate the following sample template:

```

1 <document>
2 <buyer>Abuyer</buyer>
3 <amount>100</amount>
4 <items>
5 <item><id>1</id><name>Mouse</name></item>
6 <item><id>2</id><name>Keyboard</name></item>
7 <item><id>3.0</id><name>Monitor</name></item>
8 <count>3.0</count>
9 </items>
10 </document>
```

The following table contains sample xpath and regex validation expressions:

Xpath	Regex validation expression	Description	Values for validation	Validation result
"document/buyer"	/^.+\$/	The value cannot be blank	Abuyer	Pass
"document/amount"	/^\d+\.\d{0,4}\$/	The value is a decimal with up to 4 decimal places	100	Fail
"document/items/item/id"	/^\d+\$/	The value is a whole number	Each value will be validated	
			1	Pass
			2	Pass
			3.0	Fail
"document/items/count"	/^\d+({\d{0,2}}){1}\$/	The value is a whole number or a decimal	3.0	Pass

Xpath	Regex validation expression	Description	Values for validation	Validation result
		number with up to 2 decimal places		

For more information about xpath and regex validations, go to the w3schools website:

- [w3schools xpath tutorial](#)
- [w3schools regex tutorial](#)

## PEPPOL Template

Electronic Invoicing SuiteApp supports PEPPOL standard templates. PEPPOL is an extremely secure international network that enables your company to exchange business critical electronic documents with users who have registered as a part of the PEPPOL network. The use of PEPPOL templates is governed by a multi-lateral agreement structure owned and maintained by OpenPeppol. PEPPOL enables trading partners to exchange standards based electronic documents over the PEPPOL network.

NetSuite suggests you to use the PEPPOL templates along with the Custom Data Source Plug-in implementation to generate the e-document templates correctly. Also, you can customize the template to get the generated e-document template in the required form.

The following table has the names, location of the templates and custom source data plug-in implementations for using the PEPPOL standard templates:

Names	Template Name	Template Location	Plug-in Implementation Name
Invoice Template	Invoice_Peppol_Template_Master.txt	Bundle 436209/Sample Templates	
Generic Invoice Template	Invoice_Peppol_Template_Generic.txt	Bundle 436209/Sample Templates	
Credit Memo Template	CreditMemo_Peppol_Template_Master.txt	Bundle 116079/Sample Templates	
Generic Credit Memo Template	CreditMemo_Peppol_Template_Generic.txt	Bundle 116079/Sample Templates	
Custom Data Source Plugin	pl_custom_data_source_peppol.js	Bundle 436209/src/comp/pl	PEPPOL Outbound Custom Data Source



**Note:** To use these templates, you must perform the following actions:

- Create an e-document template by adding the content of the template in the **Templates for Outbound E-Documents** field, in the Create E-Document Templates page.
- Select the PEPPOL Outbound Custom Data Source plug-in implementation from the **Custom Data Source Plug-in Implementation** drop down list.

The following table gives an overview of all the elements that are used in the and Generic PEPPOL templates for Invoice and Credit Memo transaction types . The elements in the following table use hard

coded values or values from the transactions and its related to Subsidiary, Customer and other records. The remaining elements are sourced from custom data source plug-in implementation.

SL No.	ID	Field ID	Technical Name of Field	Transaction Type	NetSuite Field Mapping Logic Used for Templates	NetSuite Field Mapping Logic Used for Generic Templates
1	ubl:Invoice			Invoice	This is the root element of the invoice's PEPPOL template. It has a default value.	This is the root element of the invoice's PEPPOL template. It has a default value.
2	ubl:CreditNote			Credit Memo	This is the root element of the credit memo's PEPPOL template. It has a default value.	This is the root element of credit memo's PEPPOL template. It has a default value.
3	cbc:CustomizationID			Both	This element identifies the specifications of the rules for semantic content, cardinalities, and business to which the data in the instance document conforms to. This element has a default value.	This element identifies the specifications of a set of rules for semantic content, cardinalities, and business to which the data in the instance document conforms. This element has a default value.
4	cbc:ProcessID			Both	This element identifies the context of the business process related to the transaction and lets the buyer process the invoice accurately. This element has a default value.	This element identifies the context of the business process related to the transaction and lets the buyer process the invoice accurately. This element has a default value.
5	cbc:ID	transaction.tranid	Invoice Number or Credit Number	Both	This element displays a unique number that identifies if the transaction is an Invoice or Credit Memo. This value is mapped with the entry number of the transaction.	This element displays a unique number that identifies if the transaction is an Invoice or Credit Memo. This value is mapped with the <b>entry number</b> of the transaction.
6	cbc:IssueDate	transaction.trandate	Date	Both	This element displays the issue date value present in the <b>Date</b> field of the transaction. The date format is YYYY-MM-DD in a template.	This element displays the issue date value present in the <b>Date</b> field of the transaction. The date format is YYYY-MM-DD in a template.
7	cbc:DueDate	transaction.duedate	Due Date	Invoice	This element displays the payment due date value present in the <b>Due Date</b> field of an Invoice. The date format is YYYY-MM-DD in a template.	This element displays the payment due date value present in the <b>Due Date</b> field of an Invoice. The date format is YYYY-MM-DD in a template.
8	cbc:InvoiceTypeCode		Custom Field	Invoice	<p>This element displays a code specifying the invoice type. This has a default assigned value of 380.</p> <p>You must map the value with the required custom field in the SuiteApp.</p> <p>The following list has information about the Invoice list and the Invoice Type Codes (UNCL1001):</p> <ul style="list-style-type: none"> <li>■ Commercial Invoice – 380</li> <li>■ Consignment Invoice – 395</li> <li>■ Debit Note for Financial Adjustments – 84</li> <li>■ Debit Note for Goods or Services – 80</li> <li>■ Debit Note – 383</li> <li>■ Factored Invoice – 393</li> <li>■ Forwarder's Invoice – 623</li> <li>■ Insurer's Invoice – 575</li> <li>■ Metered Services Invoice – 82</li> <li>■ Prepayment Invoice – 386</li> </ul> <p>You must map the value with the required custom field in the SuiteApp.</p>	<p>This element displays a code specifying the invoice type. This has a default assigned value of 380.</p> <p>You must map the value with the required custom field in the SuiteApp.</p> <p>The following list has information about the Invoice list and the Invoice Type Codes (UNCL1001):</p> <ul style="list-style-type: none"> <li>■ Commercial Invoice – 380</li> <li>■ Consignment Invoice – 395</li> <li>■ Debit Note for Financial Adjustments – 84</li> <li>■ Debit Note for Goods or Services – 80</li> <li>■ Debit Note – 383</li> <li>■ Factored Invoice – 393</li> <li>■ Forwarder's Invoice – 623</li> <li>■ Insurer's Invoice – 575</li> <li>■ Metered Services Invoice – 82</li> <li>■ Prepayment Invoice – 386</li> </ul>
9	cbc:CreditNoteTypeCode		Custom Field	Credit Memo	<p>This element displays a code specifying the credit memo type. This has a default assigned value of 381.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>The following list has the information about Credit Memo list and the Credit Memo Type Codes (UNCL1001):</p> <ul style="list-style-type: none"> <li>■ Credit Note for Foods or Services – 81</li> <li>■ Credit Note for Financial Adjustments – 83</li> </ul>	<p>This element displays a code specifying the credit memo type. This has a default assigned value of 381.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>The following list has the information about Credit Memo list and the Credit Memo Type Codes (UNCL1001):</p> <ul style="list-style-type: none"> <li>■ Credit Note for Foods or Services – 81</li> <li>■ Credit Note for Financial Adjustments – 83</li> <li>■ Credit Note – 381</li> </ul>

					<ul style="list-style-type: none"> <li>■ Credit Note – 381</li> <li>■ Factored Credit Note – 396</li> <li>■ Forwarder's Credit Note – 532</li> </ul> <p>You can map the value with the required custom field in the SuiteApp.</p>	<ul style="list-style-type: none"> <li>■ Factored Credit Note – 396</li> <li>■ Forwarder's Credit Note – 532</li> </ul>
10	cbc:Note	transaction.memo	Memo or Custom Field	Both	<p>This element displays a note with information about any change in the invoice or credit memo. This element is displayed only if there is a value in the <b>Memo</b> field of a transaction.</p> <p>In some SuiteApps, this value can be a combination of the value from the <b>Memo</b> field and the value from the custom field on a Subsidiary record.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element displays a note with information about any change in the invoice or credit memo. The note is retrieved from the value in the <b>Memo</b> field of a transaction.</p> <p>In some SuiteApps, this value can be a combination of the value from the <b>Memo</b> field and the value from the custom field on a subsidiary record.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
11	cbc:Tax Point Date	transaction.taxpointdate	Tax Point Date	Both	<p>This element displays the <b>Tax Point Date</b> field value on the Tax Details tab of a transaction. This field value must not be the same as the value of the <b>Date</b> field. It is available only in accounts with SuiteTax enabled. This element is displayed only if there is a value in the <b>Tax Point Date</b> field of the Transaction. The Tax Point Date format is YYYY-MM-DD in a template.</p> <p>The Tax Point Date is the date when the VAT becomes accountable for the seller and buyer. This field is required and acceptable only if this date is different from the transaction's issue date, according to the VAT directive.</p>	<p>This element displays the <b>Tax Point Date</b> field value on the Tax Details tab of a transaction. This field value must not be the same as the value of the <b>Date</b> field. It is available only in accounts with SuiteTax enabled. This element is displayed only if there is a value in the <b>Tax Point Date</b> field of the Transaction. The Tax Point Date format is YYYY-MM-DD in a template.</p> <p>The Tax Point Date is the date when the VAT becomes accountable for the seller and buyer. This field is required and acceptable only if this date is different from the transaction's issue date, according to the VAT directive.</p>
12	cbc:DocumentCurrencyCode	transaction.currency.symbol	Currency	Both	<p>This element displays the ISO Code of the currency used in the transaction, if Multi-Currency feature is enabled in an account.</p> <p>The value is not displayed if the Multi-Currency feature is not enabled in an account.</p>	<p>This element displays the ISO Code of the currency used in the transaction, if Multi-Currency feature is enabled in an account.</p> <p>The value is not displayed if the Multi-Currency feature is not enabled in an account.</p>
13	cbc:TaxCurrencyCode	custom.subCurrencyISOCode Refer Custom Data Source (CDS) Plug-in Implementation	Currency — In Subsidiary Record	Both	<p>This element displays the ISO Code used by the subsidiary of the transaction if the currency is not same for transaction and the subsidiary. The value of this element is retrieved from the custom data source plug-in used in the template.</p> <p>TaxCurrencyCode is used for VAT accounting and reporting, as accepted or required in the country of the seller.</p>	<p>This element displays the ISO code of the Currency used by the subsidiary in a transaction in One World (OW)Accounts.</p> <p>In Single Instance (SI) accounts the value of the <b>Currency</b> field in Company Information Page is used. The ISO code is displayed when the value of the currency and the subsidiary are not the same for the transaction and Multi Currency feature is enabled in the account. The value of this element is retrieved from the custom data source plug-in used in the template.</p> <p>TaxCurrencyCode is used for VAT accounting and reporting purposes as accepted or required in the country of the Seller.</p>
14	cbc:AccountingCost		Custom Field	Both	<p>This element displays the details of the booking data related to the buyer's financial accounts.</p> <p>You must map this field value with the required custom field in the SuiteApp.</p>	The template does not support this element.
15	cbc:BuyerReference	transaction.otherrefnum	PO #	Both	<p>This element displays the identifier or code assigned by the buyer for internal routing. The <b>PO#</b> field is used to identify the buyer and the order details. This element is displayed only if the <b>PO#</b> field has a value in a transaction. The value can be the name of the person ordering the products, employee number, or an identification code for a buyer, department, or group.</p> <p>You can map the value of this field with the required custom field in the SuiteApp.</p>	<p>This element displays the identifier or code assigned by the buyer for internal routing. The <b>PO#</b> field is used to identify the buyer and the order details. This element is displayed only if the <b>PO#</b> field has a value in a transaction. The value can be the name of the person ordering the products, employee number, or an identification code for a buyer, department, or group.</p> <p>You can map the value of this field with the required custom field in the SuiteApp.</p>
16	cac:InvoicePeriod		Invoice		<p>This is a parent element containing a group of business terms that provide information about the invoice period, also known as the</p>	<p>This is a parent element containing a group of business terms that provide information about the invoice period, also known as the delivery</p>

					delivery period. If you use this group, you must also use the invoice period start date, end date, or both.  This parent element is not displayed if any of the child elements do not have a value.	period. If you use this group, you must also use the invoice period start date, end date, or both.  This element is displayed only if the value of start date, end date, or both is available.
17	cbc:Start Date	transaction.startdate	Start Date	Invoice	This element displays the start date of an invoice period for a transaction. This element is displayed along with a value on the generated e-document only if the value is entered in the <b>Start Date</b> field of the transaction.  You can map the value with the required custom field in the SuiteApp.	This element displays the start date of an invoice period for a transaction. This element is displayed along with a value on the generated e-document only if the value is entered in the <b>Start Date</b> field of the transaction.
18	cbc:End Date	transaction.enddate	End Date	Invoice	This element displays the end date of an invoice period for a transaction. This element is displayed along with a value on the generated E-Document only if the value is entered in the <b>End Date</b> field of the transaction.  You can map the value with the required custom field in the SuiteApp.	This element displays the end date of an invoice period for a transaction. This element is displayed along with a value on the generated E-Document only if the value is entered in the <b>End Date</b> field of the transaction.
19	cbc:Des criptionCode		Custom Field	Both	This element displays the code of the date when VAT becomes accountable for seller and buyer.  For example: <ul style="list-style-type: none"><li>■ 3 - Invoice document issue date, time</li><li>■ 35 - Delivery date, time</li><li>■ 42 - Paid to date</li></ul> You can map the value with the required custom field in your account.	The template does not support this element.
20	cac:OrderReference			Both	This is a parent element with details of the order and sales order references.	This is a parent element with details of the order and sales order references.
21	cbc:ID	transaction.otherrefnum customer.account number customer.entityid	PO# Account Number Customer ID	Both	This element displays one the following identifiers for a referenced purchase order allotted by the buyer: <ul style="list-style-type: none"><li>■ PO#</li><li>■ Account Number – The account number is displayed if the PO# value is missing.</li><li>■ Customer ID – The customer ID is displayed if the account number is missing.</li></ul> The element represents an identifier for a referenced purchase order issued by the Buyer.	This element displays one the following identifiers for a referenced purchase order allotted by the buyer: <ul style="list-style-type: none"><li>■ PO#</li><li>■ Account Number – The account number is displayed if the PO# value is missing.</li><li>■ Customer ID – The customer ID is displayed if the account number is missing.</li></ul> The element represents an identifier for a referenced purchase order issued by the Buyer.
22	cbc:SalesOrderID	transaction.createdfrom	Created From	Invoice	This element displays an identifier for a referenced sales order allotted by the seller. It has the entry number of the sales order transaction only if an invoice is created from a sales order. The entry number is retrieved from the <b>Created From</b> field in the invoice. If the invoice is not created from a sales order, then the element is not printed.	This element displays an identifier for a referenced sales order allotted by the seller. It has the entry number of the sales order transaction only if an invoice is created from a sales order. The entry number is retrieved from the <b>Created From</b> field in the invoice. If the invoice is not created from a sales order, then the element is not printed.
23	cac:BillingReference			Credit Memo	This is a parent element containing a group of business terms with information about previous invoices. In the template, the element has no value and has child elements that can have values if required.	This is a parent element containing a group of business terms with information about previous invoices. In the template, the element has no value and has child elements that can have values if required.
24	cac:InvoiceDocumentReference			Credit Memo	This parent element has no value in the template but contains child elements identifying an invoice.	This parent element has no value in the template but contains child elements identifying an invoice. This element has no value in the template.
25	cbc:ID	transaction.createdfrom	Created From	Credit Memo	This element displays a value identifying an invoice that was previously sent by the seller. If a credit memo has an invoice applied, then the value of the invoice entry is displayed in the <b>Created From</b> field.	This element displays a value identifying an invoice that was previously sent by the seller. If a credit memo has an invoice applied, then the value of the invoice entry is displayed in the <b>Created From</b> field.

26	cbc:Issue Date	transaction.create.dfrom.trandate	Date – Invoice's <b>Date</b> field	Credit Memo	This element displays the date when the last invoice was issued. If a credit memo has an invoice applied to it, then the value of the invoice entry is displayed in the <b>Created From</b> field. The <b>Date</b> field value represents the issue date of that invoice.	This element displays the date when the last invoice was issued. If a credit memo has an invoice applied to it, then the value of the invoice entry is displayed in the <b>Created From</b> field. The <b>Date</b> field value represents the issue date of that invoice.
27	cac:DespatchDocumen tReference			Both	This parent element provides reference to the dispatch advice. This element has no value, but it has child elements with values associated to it in the template.	The template does not support this element.
28	cbc:ID		Custom Field	Both	This element refers to an identifier of a referenced dispatch advice.  You can map the value of this key with the preferred custom field in your account.	The template does not support this element.
29	cac:ReceiptDocumen tReference			Both	This parent element has reference to the receipt advice. This element has no value, but it has child elements with values associated to it in the template.	The template does not support this element.
30	cbc:ID		Custom Field	Both	This element displays an identifier of a referenced receiving advice.  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
31	cac:OriginatorDocumen tReference			Both	This parent element contains a tender or lot reference details. This element has no value, but it has child elements with values associated to it in the template.	The template does not support this element.
32	cbc:ID		Custom Field	Both	This element displays the call for tender or lot of the invoice ID.  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
33	cac:ContractDocumen tReference			Both	This parent element provides details of any existing contract associated with a transaction. This element has no value, but it has child elements with values associated to it in the template.	The template does not support this element.
34	cbc:ID		Custom Field	Both	This element has a value that identifies a contract.  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
35	cac:AdditionalDocumen tReference			Both	This parent element has no value, but it has child elements with values associated to it in the template. It has information about group of business terms and details, with additional supporting documents having terms of the invoice. The supporting documents include: <ul style="list-style-type: none"><li>■ Reference to a document number known to the receiver</li><li>■ External documents referenced by a URL</li><li>■ Embedded documents</li><li>■ Base64 encoded documents like time report</li></ul>	The template does not support this element.
36	cbc:ID		Project	Both	This element displays the identifier of an object based on the invoice provided by the seller or a value used for identifying a supporting document.  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
37	@sche meID		Custom value from list	Both	This element must have a value displaying the identifier of the identification scheme of an invoiced object. It is used inside the	The template does not support this element.

					cbc:ID element. The value of this element must be from the <a href="#">list</a> .  You can map the value with the required custom field in the SuiteApp.	
38	cbc:DocumentTypeCode		Custom Field	Both	This element displays the code value 50 if there is a project specified in the header of the credit memo.  If the credit memo has an applied invoice or it does not display any content if conditions are not met, then the code value is 130 instead of 50.  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
39	cbc:DocumentDescription		Custom Field	Both	This element has a value that describes supporting documents such as time sheets and usage reports.  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
40	cac:Attachment			Both	This parent element has information about attached documents with the transaction. The element has no value, but it has child elements with values associated to it in the template.	The template does not support this element.
41	cbc:EmbeddedDocumentBinaryObject		Custom Field	Both	This element has a value referencing any attached document embedded as a binary object (Base64) or sent together with the invoice.  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
42	@mimeCode		Custom Value	Both	This is a value used with the element cbc:EmbeddedDocumentBinaryObject that represents the mime code of an attached document. For example: text/csv  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
43	@fileName		Custom Value	Both	This is a value used with the element cbc:EmbeddedDocumentBinaryObject that represents the file name of an attached document.  For example: abc.csv  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
44	cac:ExternalReference			Both	This is a parent element providing details of the external document's reference such as its location. The element has no value, but it has child elements with values associated to it in the template.  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
45	cbc:URI		Custom Field	Both	The value of this element represents the Uniform Resource Locator (URL) that identifies the location of the external document. This locates the resource and its primary access mechanism, such as http:// or ftp://.  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
46	cac:ProjectReference			Invoice	This parent element has information about any project associated with an invoice. This element has no value, but it has child elements with values associated to it in the template. This element is not displayed if its child element do not have any value.  You can map the value with the required custom field in the SuiteApp.	This parent element has information about any project associated with an invoice. This element has no value, but it has child elements with value associated to it in the template. This element is not displayed, if the <b>Project</b> field has no value.  You can map the value with the required custom field in the SuiteApp.
47	cbc:ID	transaction.job	Project	Invoice	This element indicates the ID of the project associated with an invoice. If the <b>Project</b> field has	This element indicates the ID of the project associated with an invoice. If the <b>Project</b> field has

					field has no value in the transaction, the element is not printed.	no value in the transaction, the element is not printed.	
48	cac:AccountingSupplierParty			Both	This is a parent element that provides information about the seller using various child elements. This element has no value, but it has child elements with values associated to it in the template.	This is a parent element that provides information about the seller using various child elements. This element has no value, but it has child elements with values associated to it in the template.	
49	cac:Party			Both	This is a parent element and a child element of cac:AccountingSupplierParty, having information about the seller like electronic address. This element has no value, but it has child elements with values associated to it in the template.	This is a parent element and a child element of cac:AccountingSupplierParty, having information about the seller like electronic address. This element has no value, but it has child elements with values associated to it in the template.	
50	cbc:EndpointID	Vat Registration No Subsidiary Tax Reg Number	Both	<p>This element identifies the seller's electronic address to which the application level response to the invoice can be delivered.</p> <p>The element uses the custom data source plug-in to get one of the following values:</p> <ul style="list-style-type: none"> <li>■ Value of the <b>VAT Registration No</b> field from the subsidiary in legacy tax accounts or</li> <li>■ Value of the <b>Subsidiary Tax Reg Number</b> field from the transaction for SuiteTax accounts.</li> </ul> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element identifies the seller's electronic address to which the application level response to the invoice can be delivered.</p> <p>The element uses the Custom Data Source plugin to get one of the following values:</p> <ul style="list-style-type: none"> <li>■ Value of the <b>VAT Registration No</b> field from the subsidiary in legacy tax accounts or</li> <li>■ Value of the <b>Subsidiary Tax Reg Number</b> field from the transaction for SuiteTax accounts.</li> </ul> <p>You can map the value with the required custom field in the SuiteApp.</p>		
51	@schemeID	Custom Value	Both	<p>This element is used with the element cbc:EndpointID and identifies the scheme ID of the seller's electronic address.</p> <p>You can find the scheme ID value from <a href="#">Scheme ID List</a>.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element is used with the element cbc:EndpointID and identifies the scheme ID of the seller's electronic address.</p> <p>You can find the scheme ID value from <a href="#">Scheme ID List</a>.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>		
52	cac:PartyIdentification		Both	This parent element identifies the seller or the seller's unique banking reference ID provided by the bank. This element has no value, but it has child elements with values associated to it in the template.	The template does not support this element.		
53	cbc:ID	Custom Field	Both	<p>This parent element identifies the seller or the seller's unique banking reference ID provided by the bank. ICD code list is used for the seller's ID. SEPA is the code used for SEPA bank's assigned creditor reference. For the buyer to identify a supplier automatically, the following identifiers are available:</p> <ul style="list-style-type: none"> <li>■ BT-29 – Seller identifier</li> <li>■ BT-30 – Seller legal registration identifier</li> <li>■ BT-31 – Seller VAT identifier</li> </ul> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.		
54	@schemeID	Custom Value	Both	This element is used with the element cbc:ID to identify the scheme of the seller Identifier. For example, for a bank assigned credit identifier BT-90, the seller ID must be SEPA.	The template does not support this element.		
55	cac:PartyName		Both	This parent element displays the seller's information such as name and address. This element has no value, but it has child elements with values associated to it in the template. This element is not displayed if its child elements do not have values.	This parent element displays the seller's information such as name and address. This element has no value in the template, but it has child elements with values associated with them. This element is not displayed if the <b>Addressee</b> field value of the subsidiary's main address or <b>Name</b> field value of the subsidiary is not present.		
56	cbc:Name	transaction.subsidiary.name custom.addressee	Address (Subsidiary's main address's address Name (Subsidiary name)	Both	This element displays the business name, another name by which the Seller is known, other than the Seller name. It contains a value in the <b>Addressee</b> field of the Subsidiary's main address. If the field has no value, then this element prints the Subsidiary name without the full hierarchy. This	This element displays the business name, another name by which the Seller is known, other than the Seller name. It contains a value in the <b>Addressee</b> field of the Subsidiary's main address retrieved from custom data source plug-in. If the <b>Addressee</b> field has no value, then this element prints the Subsidiary name without the	

					element prints the name of the subsidiary without a OneWorld account.	full hierarchy. In Single Instance accounts, the <b>Name</b> of the subsidiary to which the transaction is mapped is printed.  Example: Parent Company, Arizona Company etc
57	cac:PostalAddress			Both	This is a parent element with information about the seller's address. Required fields of the address must be filled out to comply with the legal requirements. This element has no value, but it has child elements that have values associated to it in the template.	This is a parent element with information about the seller's address. Required fields of the address must be filled out to comply with the legal requirements. This element has no value, but it has child elements that have values associated to it in the template.
58	cbc:StreetName	transaction.subsidiary.address1	Address 1 (Subsidiary's main address's address line 1)	Both	This element displays the main address line of the seller's address. The element has the value of the <b>Address 1</b> field of the subsidiary's main address. This element is displayed only if a value is entered in the <b>Address 1</b> field on the Subsidiary record; otherwise, the element is not displayed.	This element displays the main address line of the seller's address. The element has the value of the <b>Address 1</b> field of the subsidiary's main address. This element is displayed only if a value is entered in the <b>Address 1</b> field on the Subsidiary record; otherwise, the element is not displayed.
59	cbc:AdditionalStreetName	transaction.subsidiary.address2	Address 2 (Subsidiary's main address's address line 2)	Both	This element displays another address line with more details about the main address. This element has the value of the <b>Address 2</b> field in the subsidiary's main address. This element is displayed only if a value is entered in the <b>Address 2</b> field on the Subsidiary record; otherwise, the element is not displayed.	This element displays another address line with more details about the main address. This element has the value of the <b>Address 2</b> field in the subsidiary's main address. This element is displayed only if a value is entered in the <b>Address 2</b> field on the Subsidiary record; otherwise, the element is not displayed.
60	cbc:CityName	transaction.subsidiary.city	City (Subsidiary's main address's city)	Both	This element displays the common name of the city, town, or village of the seller's location. The element's value is displayed in the <b>City</b> field of the subsidiary's main address.	This element displays the common name of the city, town, or village of the seller's location. The element's value is displayed in the <b>City</b> field of the subsidiary's main address.
61	cbc:PostalZone	transaction.subsidiary.zip	ZIP(Subsidiary's main address's zip)	Both	This element identifies a group of properties with addresses based on the relevant postal service. This element is displayed only if there is a value in the <b>Zip</b> field of the subsidiary's main address.	This element identifies a group of properties with addresses based on the relevant postal service. This element is displayed only if there is a value in the <b>Zip</b> field of the subsidiary's main address.
62	cbc:CountrySubentity	Transaction.subsidiary.state	State (Subsidiary's main address's state)	Both	This element identifies the subdivision of the country in the subsidiary's main address. The element is displayed only if there is a value in the <b>State</b> field of the subsidiary's main address.	This element identifies the subdivision of the country in the subsidiary's main address. The element is displayed only if there is a value in the <b>State</b> field of the subsidiary's main address.
63	cac:AddressLine			Both	This parent element has information about the additional address line in the subsidiary address. This element has no value in the template.	This parent element has information about the additional address line in the subsidiary address. This element has no value in the template.
64	cbc:Line	transaction.subsidiary.address3	ADDRESS 3 (Subsidiary's main address's address line 3)	Both	This element has information about an additional address line with more details about the main address line. This element has the value of the <b>Address 3</b> field in the subsidiary's main address. It is displayed only if the <b>Address 3</b> field has a value.	This element has information about an additional address line with more details about the main address line. This element has the value of the <b>Address 3</b> field in the subsidiary's main address. It is displayed only if the <b>Address 3</b> field has a value.
65	cac:Country			Both	This parent element has information about the country associated with the subsidiary's address. This element has no value in the template.	This parent element has information about the country associated with the subsidiary's address. This element has no value in the template.
66	cbc:IdentificationCode	In OW accounts: Country of transaction's subsidiary.  In SI account: Country name in Company Information page.	custom.subPrimaryCountry (Refer CDS)  custom.countryInCompanyInfoCode (Refer CDS)	Both	This parent element displays the ISO Code of a transaction's country in the subsidiary's main address in OneWorld accounts. Without a OneWorld account, the element displays the ISO Code of the <b>Country</b> field on the Company Information page. The values are displayed using the custom data source plug-in implemented in the template.	This element displays the following information: <ul style="list-style-type: none"><li>Value of ISO code of the transaction's subsidiary's <b>Country</b> in OW accounts.</li><li>Value of the ISO code of the <b>Country</b> field in the Company Information page for SI accounts.</li></ul> The values are displayed using the custom data source plug-in implemented in the template.
67	cac:PartyTaxScheme			Both	This parent element with its child element displays the seller's VAT identifier or tax registration details using different transaction fields. This element has no value in the template, but it has child elements that have values associated with them.	This parent element with its child element displays the seller's VAT identifier or tax registration details using different transaction fields. This element has no value in the template, but it has child elements that have values associated with them.

68	cbc:CompanyID	custom.subVatRegNo (Refer CDS)	For Legacy Tax: Employee Identification Number(EIN) on transaction's Subsidiary. For SuiteTax: Subsidiary Tax Reg Number	Both	<p>This element displays the seller's VAT identification number or local identification for tax purpose or reference. This enables a seller to indicate their registered tax status.</p> <p>The element displays the value in the <b>Employee Identification Number</b> field of a subsidiary in a transaction in the generated e-document. The value is displayed using the custom data source plug-in implemented in the template.</p>	<p>This element displays the seller's VAT identification number or local identification for tax purpose or reference. This enables a seller to indicate their registered tax status.</p> <p>The element displays the following information about generated e-document:</p> <ul style="list-style-type: none"> <li>■ The value of the transaction's subsidiary's <b>Employee Identification Number</b> field in Legacy Tax accounts</li> <li>■ The value of the transaction's <b>Subsidiary Tax Reg Number</b> field in Suite Tax accounts.</li> </ul> <p>The values are displayed using the custom data source plug-in implemented in the template.</p>
69	cac:TaxScheme				<p>This parent element has a child element cac:PartyTaxScheme that displays the seller's VAT identification or tax registration details using various transaction fields. This element has no value, but it has child elements that have values associated to it in the template.</p>	<p>This parent element has a child element cac:PartyTaxScheme that displays the seller's VAT identification or tax registration details using various transaction fields. This element has no value, but it has child elements that have values associated to it in the template.</p>
70	cbc:ID		Fixed Value: VAT	Both	<p>This element is displayed if the seller is a VAT identifier or TAX Identifier. The default value is VAT.</p>	<p>This element is displayed if the seller is a VAT identifier or TAX Identifier. The default value is VAT.</p>
71	cac:PartyLegalEntity				<p>This parent element has information about the seller's legal details like registration name. This element has no value, but it has child elements that have values associated to it in the template.</p>	<p>This parent element has information about the seller's legal details like registration name. This element has no value, but it has child elements that have values associated to it in the template.</p>
72	cbc:RegistrationName	transaction.subsidiary.legalname	Legal Name	Both	<p>This element displays the seller's registered name as per the following conventions:</p> <ul style="list-style-type: none"> <li>■ National Registry of Legal Entities</li> <li>■ Taxable person</li> <li>■ Trading as a person or persons</li> </ul> <p>The value of the element is retrieved from the subsidiary's <b>Legal Name</b> field in a transaction. This element is not displayed if the field has no value.</p>	<p>This element displays the seller's registered name as per the following conventions:</p> <ul style="list-style-type: none"> <li>■ National Registry of Legal Entities</li> <li>■ Taxable person</li> <li>■ Trading as a person or persons</li> </ul> <p>The value of the element is retrieved from the subsidiary's <b>Legal Name</b> field in a transaction. This element is not displayed if the field has no value.</p>
73	cbc:CompanyID		Custom Field	Both	<p>This element is used as an identifier allotted by an official registrar. It identifies the seller as legal entity or person. This element does not have any value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element is used as an identifier allotted by an official registrar. It identifies the seller as legal entity or person. This element does not have any value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
74	@schemeID		Custom Value	Both	<p>This element is used with the element cbc:CompanyID to identify the scheme identifier of a seller's legal registration. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element is used with the element cbc:CompanyID to identify the scheme identifier of a seller's legal registration. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
75	cbc:CompanyLegalForm			Both	<p>This parent element has details about additional legal information related to the seller. This element has no value in the template.</p>	<p>The template does not support this element.</p>
76	cac:Contact			Both	<p>This is a parent element and a child of cbc:Party with information about a group of business terms providing contact information of the seller. This element has no value in the template, but it has child elements that have values associated with them. This parent element is not displayed if the child elements do not have values.</p>	<p>This is a parent element and a child of cbc:Party with information about a group of business terms providing contact information of the seller. This element has no value, but it has child elements that have values associated to it in the template. This element is displayed if the transaction has a sales representative.</p>
77	cbc:Name	transaction.salesrep	Sales Rep	Both	<p>This element provides contact details for a legal entity or representative and displays the <b>Sales Rep</b> field value in a transaction. The element is displayed only if there is a value in the <b>Sales Rep</b> field of the transaction.</p>	<p>This element provides contact details for a legal entity or representative and displays the <b>Sales Rep</b> field value in a transaction. The element and the value is not displayed only if there is no value in the <b>Sales Rep</b> field of the transaction.</p>

78	cbc:Telephone	transaction.salesrep.phone	Phone	Both	<p>This element displays the phone number of the contact person. It uses the phone number of the sales representative selected in the transaction as the value. This element is displayed only if there is a value in the sales representative's <b>Phone</b> field in the employee record.</p>	<p>This element displays the phone number of the contact person. It uses the phone number of the sales representative selected in the transaction as the value. This element and value is not displayed if there is a value in the sales representative's <b>Phone</b> field in the employee record.</p>
79	cbc:Email	transaction.salesrep.email	Email	Both	<p>This element displays the email address of the contact person and uses the email address of the sales representative selected in the transaction as the value. This element is displayed only if there is value in the sales representative's <b>Email</b> field in the employee record.</p>	<p>This element displays the email address of the contact person and uses the email address of the sales representative selected in the transaction as the value. This element is displayed only if there is value in the sales representative's <b>Email</b> field in the employee record.</p>
80	cac:AccountingCustomerParty				<p>This parent element contains a group of business terms, with information about the buyer. This element has no value, but it has child elements that have values associated to it in the template.</p>	<p>This parent element contains a group of business terms, with information about the buyer. This element has no value, but it has child elements that have values associated to it in the template.</p>
81	cac:Party			Both	<p>This is a parent element and a child of cac:AccountingCustomerParty, providing information about the buyer. This element has no value, but it has child elements with values associated in the template.</p>	<p>This is a parent element and a child of cac:AccountingCustomerParty, providing information about the buyer. This element has no value, but it has child elements with values associated in the template.</p>
82	cbc:EndpointID	For Legacy Tax: customer.vatregnumber in customer record For SuiteTax: customer.defaulttaxreg in customer record	In Legacy Tax Account: <b>Tax Reg Number</b> field in the customer record  In SuiteTax account: <b>Default Tax Reg</b> field in the customer record	Both	<p>This element displays a value identifying a buyer's electronic address to deliver the invoice.</p> <p>This element is displayed along with the field value based on the following criteria:</p> <ul style="list-style-type: none"> <li>■ A customer record has a value in the <b>Tax Reg Number</b> field in Legacy Tax accounts.</li> <li>■ A customer record has a value in the <b>Default Tax Reg</b> field in SuiteTax accounts.</li> </ul> <p>Example: 987654321</p>	<p>This element displays a value identifying a buyer's electronic address to deliver the invoice.</p> <p>This element is displayed along with the field value based on the following criteria:</p> <ul style="list-style-type: none"> <li>■ A customer record has a value in the <b>Tax Reg Number</b> field in Legacy Tax accounts.</li> <li>■ A customer record has a value in the <b>Default Tax Reg</b> field in SuiteTax accounts.</li> </ul> <p>Example: 987654321</p>
83	@schemeID		Custom Value	Both	<p>This element is used with the element cbc:EndpointID and has no value in the template. This element identifies the scheme identifier of a buyer's electronic address.</p> <p>Example: 0192</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element is used with the element cbc:EndpointID and has no value in the template. This element identifies the scheme identifier of a buyer's electronic address.</p> <p>Example: 0192</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
84	cac:PartyIdentification			Both	<p>This is a parent element providing information about the buyer identifier. This element has no value, but it has child elements that have values associated with them.</p>	The template does not support this element.
85	cbc:ID	customer.account number	Account (Customer's transaction)	Both	<p>This element populates the customer identifier. The value of <b>Account</b> field on the transaction's customer record is used as value for this element.</p>	The template does not support this element.
86	@schemeID		Custom Value	Both	<p>This element is used with the element cbc:ID to display or populate the scheme ID of the buyer identifier.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
87	cac:PartyName				<p>This parent element displays the buyer's details. This element has no value, but it has child elements that have values associated with it in the template.</p>	<p>This parent element displays the buyer's details. This element has no value, but it has child elements that have values associated with it in the template.</p>
88	cbc:Name	customer.companyname	Company Name	Both	<p>This element displays another name that the buyer is known, other than the Business name. The value in the <b>Company Name</b> field on a customer's transaction is displayed in the generated e-document for this element.</p>	<p>This element displays another name that the buyer is known, other than the Business name. The value in the <b>Company Name</b> field on a customer's transaction is displayed in the generated e-document for this element.</p>

89	cac:PostalAddress			Both	This parent element has a group of business terms providing information about the buyer's postal address. This element has no value, but it has child elements that have values associated with it in the template.	This parent element has a group of business terms providing information about the buyer's postal address. This element has no value, but it has child elements that have values associated with it in the template.
90	cbc:StreetName	transaction.billaddr1	Address 1 (Transaction's billing address)	Both	This element displays the main address line of an address. The element retrieves the value of the <b>Address 1</b> field from the billing address in a transaction. This element is displayed only if the <b>Address 1</b> field has a value.	This element displays the main address line of an address. The element retrieves the value of the <b>Address 1</b> field from the billing address in a transaction. This element is displayed only if the <b>Address 1</b> field has a value.
91	cbc:AdditionalStreetName	transaction.billaddr2	Address 2 (Transaction's billing address)	Both	This element displays an additional address line to have more details on the main address. This element retrieves the value of the <b>Address2</b> field from the billing address in a transaction. This element is displayed only if the <b>Address2</b> field has a value.	This element displays an additional address line to have more details on the main address. This element retrieves the value of the <b>Address2</b> field from the billing address in a transaction. This element is displayed only if the <b>Address2</b> field has a value.
92	cbc:CityName	transaction.billcity	City (Transaction's billing address)	Both	This element displays the common name of the city, town, or village of the buyer's location. This element is displayed only if the <b>City</b> field of the transaction's billing address has a value.	This element displays the common name of the city, town, or village of the buyer's location. This element is displayed only if the <b>City</b> field of the transaction's billing address has a value.
93	cbc:PostalZone	transaction.billzip	Zip (Transaction's billing address)	Both	This element identifies a group of properties with address based on the relevant postal service. <b>Zip</b> field of the transaction's billing address is used as value for this element. This element is not displayed if this field has no value.	This element identifies a group of properties with address based on the relevant postal service. <b>Zip</b> field of the transaction's billing address is used as value for this element. This element is not displayed if this field has no value.
94	cbc:CountrySubentity	transaction.billstate	State (Transaction's billing address)	Both	This element identifies the subdivision of the country. This element is displayed only if the <b>State</b> field of the transaction's billing address has a value.	This element identifies the subdivision of the country. This element is displayed only if the <b>State</b> field of the transaction's billing address has a value.
95	cac:AddressLine				This parent element shows information about the additional address line in a transaction's billing address. This element has no value, but it has child elements that have values associated with them. This parent element is displayed only if its child elements have values.	This parent element shows information about the additional address line in a transaction's billing address. This element is not displayed if the <b>Address 3</b> field of transaction's billing address has no value.
96	cbc:Line	transaction.billaddr3	Address 3 (Transaction's billing address)	Both	This element displays an additional address line to have more details on the main address line. This element retrieves the value of the <b>Address 3</b> field from the billing address in a transaction. This element is only displayed if the <b>Address 3</b> field has a value.	This element displays an additional address line to have more details on the main address line. This element retrieves the value of the <b>Address 3</b> field from the billing address in a transaction. This element is only displayed if the <b>Address 3</b> field has a value.
97	cac:Country				This parent element provides information about the billing address's country in a transaction. This element has no value, but it has child elements that have values associated with it in the template.	This parent element provides information about the billing address's country in a transaction. This element has no value, but it has child elements that have values associated with it in the template.
98	cbc:IdentificationCode	custom.billCountryISOCode	Country (Transaction's billing address)	Both	This element displays a code identifying the billing country of a transaction. The element displays the ISO code of the country.  The value of this element is displayed using the custom data source plug-in implemented in the template.	This element displays a code identifying the billing country of a transaction. The element displays the ISO code of the country.  The value of this element is displayed using the custom data source plug-in implemented in the template.
99	cac:PartyTaxScheme				This parent element has information about the party VAT identifier. This element has no value, but it has child elements that have values associated with it in the template.	This parent element has information about the party VAT identifier. This element has no value in the template.
100	cbc:CompanyID	For Legacy Tax: customer.vatregnumber  For SuiteTax: customer.defaulttaxreg	In Legacy Tax Account: Tax Reg Number  In SuiteTax account: Default Tax Reg	Both	This element displays the buyer's VAT identification number. This element is displayed with the field value in a customer record for the following accounts:  ■ <b>Default Tax Reg Number</b> in Suite Tax account. ■ <b>Tax Reg Number</b> field in Legacy Tax account.  You can map the value with the required custom field in the SuiteApp.	This element displays the buyer's VAT identification number. This element is displayed with the field value in a customer record for the following accounts:  ■ <b>Default Tax Reg Number</b> in Suite Tax account. ■ <b>Tax Reg Number</b> field in Legacy Tax account.

101	cac:TaxScheme				This parent element provides information about the tax scheme used by the customer. This element has no value, but it has child elements that have values associated to it in the template.	This parent element provides information about the tax scheme used by the customer. This element has no value, but it has various child elements that have values associated with it in the template.
102	cbc:ID		Custom Field	Both	This element provides the code of the tax scheme being used. This element has a fixed value VAT.	This element provides the code of the tax scheme being used. This element has a fixed value VAT.
103	cac:PartyLegalEntity				This parent element provides legal information about the buyer. This element has no value, but it has child elements that have values associated to it in the template.	This parent element provides legal information about the buyer. This element has no value, but it has child elements that have values associated with it in the template.
104	cbc:RegistrationName	transaction.billaddress customer.companyname	Addressee Company Name		This element displays the buyer's full name. If the <b>Addressee</b> field has a value in the Billing Address of a transaction, then the full name is displayed along with the element. If not, then the element displays the customer's company name.	This element displays the buyer's full name. If the <b>Addressee</b> field has a value in the Billing Address of a transaction, then the full name is displayed along with the element. If not, then the element displays the customer's company name.
105	cbc:CompanyID		Custom Field	Both	This element displays an identifier of the buyer as a legal entity or person. This identifier is issued by an official registrar. This element has no value in the template.  You can map the value with the required custom field in the SuiteApp.	This element displays an identifier of the buyer as a legal entity or person. This identifier is issued by an official registrar. This element has no value in the template.  You can map the value with the required custom field in the SuiteApp.
106	@schemeID		Custom Value	Both	This element displays the scheme ID of the buyer's legal registration identifier. This element has no value associated with it in the template.  You can map the value with the required custom field in the SuiteApp.	This element has the point of contact of the buyer's legal entity or person. This element has no value associated with it in the template.  You can map the value with the required custom field in the SuiteApp.
107	cac:Contact				This is parent element has multiple child elements and provides contact information relevant to the buyer. This element has no value associated with it in the template.  This element is displayed only if the transaction's customer has a phone number or an email address.	This parent element has multiple child elements and provides contact information relevant to the buyer. This element does not have a value in the template. This element is displayed only if the transaction's customer has a phone number or an email address.
108	cbc:Name		Custom Field	Both	This element has the point of contact of the buyer's legal entity or person. This element has no value associated with it in the template.  You can map the value with the required custom field in the SuiteApp.	This element displays the e-mail address of the buyer's contact. This element is displayed in the generated e-document only if the customer's <b>E-mail Address</b> field has a value.
109	cbc:Telephone	customer.phone	Phone	Both	This element displays the phone number of the buyer's contact. This element is displayed in the generated e-document only if the customer's <b>Phone Number</b> field has a value.	The template does not support this element.
110	cbc:Email	customer.email	Email	Both	This element displays the e-mail address of the buyer's contact. This element is displayed in the generated e-document only if the customer's <b>E-mail Address</b> field has a value.	The template does not support this element.
111	cac:PayeeParty				This parent element has multiple child elements, with information about the payee. This element is used when the payee is different from the seller. This element has no value in the template.	The template does not support this element.
112	cac:PartyIdentification			Both	This is a parent element and a child of cac:PayeeParty, which has child elements used for identifying the payee party. This element has no value in the template.	The template does not support this element.
113	cbc:ID		Custom Field	Both	This element is used to identify both the payee and the unique banking reference identifier of the payee assigned by the payee's bank. For identifying the payee, you can use the ICD code list. You can use SEPA code for identifying the SEPA bank assigned creditor reference. For example, FR932874294.	The template does not support this element.

					You can map the value with the required custom field in the SuiteApp.	
114	@schemeID		Custom Value	Both	<p>This element used with the cbc:ID element identifies the payee's scheme ID. This has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
115	cac:PartyName				<p>This is a parent element and a child of cac:PayeeParty element. This element provides information about the payee name. This element has no value in the template.</p>	The template does not support this element.
116	cbc:Name		Custom Field	Both	<p>This element displays the name of the payee. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
117	cac:PartyLegalEntity				<p>This parent element has legal information about the payee. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
118	cbc:CompanyID		Custom Field	Both	<p>This element identifies the payee's legal registration. The identifier is issued by an official registrar and identifies the payee as a legal entity or person. For example, FR932874294.</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
119	@schemeID		Custom Value	Both	<p>This element is used with cbc:CompanyID and has the scheme ID of the payee's legal registration identifier.</p> <p>For example: 0002</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
120	cac:TaxRepresentativeParty			Both	<p>This parent element has information about the seller's tax representative. This element has no value, but it has child elements that have values associated to it in the template.</p>	The template does not support this element.
121	cac:PartyName				<p>This is a parent element and a child element of cac:TaxRepresentativeParty. This element displays the name of the seller's tax representative. This element has no value, but it has child elements that have values associated to it in the template.</p>	The template does not support this element.
122	cbc:Name		Custom Field	Both	<p>This element displays the full name of the seller's tax representative. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
123	cac:PostalAddress			Both	<p>This is a parent element and a child element of cac:TaxRepresentativeParty. The element has no value, but it has child elements with information about the postal address of the tax representative in template. The required address fields must be filled out to comply with legal requirements.</p>	The template does not support this element.
124	cbc:StreetName		Custom Field	Both	<p>This element displays the main address line of the seller's tax representative. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
125	cbc:AddressLine		Custom Field	Both	<p>This element displays the value of an additional address line of the main address</p>	The template does not support this element.

	iStreetName			<p>of a seller's tax representative. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	
126	cbc:CityName		Custom Field	<p>This element displays the common name of the city, town, or village of the tax representative's location. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
127	cbc:PostalZone		Custom Field	<p>This element identifies an addressable group of properties according to the relevant postal service of the seller's tax representative. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
128	cbc:CountrySubentity		Custom Field	<p>This element has information about a country's subdivision of the seller's tax representative. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
129	cac:AddressLine			<p>This element is a parent element and a child of cac:PostalAddress, with information about the additional address line of the seller's tax representative main address. This element has clarifications on the addresses. It has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
130	cbc:Line		Custom Field	<p>This element has information about the additional address line of a main address. It gives information about the seller's tax representative's main address. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
131	cac:Country			<p>This element is a parent element and child of cac:PostalAddress, with information about the country of the address. This element has no value, but it has child elements that have values associated to it in the template.</p>	The template does not support this element.
132	cbc:IdentificationCode		Custom Field	<p>This element displays the ISO code of the seller's tax representative country. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
133	cac:PartyTaxScheme			<p>This is a parent element with information about the VAT party identifier. This element has no value, but it has child elements with values associated with it in the template.</p>	The template does not support this element.
134	cbc:ComplianceID		Custom Field	<p>This element represents the VAT identifier of the seller's tax representative. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
135	cac:TaxScheme			<p>This parent element displays the tax scheme used by the tax representative. This element has no value in the template.</p>	The template does not support this element.
136	cbc:ID		Custom Field	<p>This element displays the name of the tax scheme and has a required value VAT, added in the template.</p>	The template does not support this element.
137	cac:Delivery			<p>This parent element has information about the delivery time and location of the invoiced goods and services. This element has no value in the template, but it has child elements with</p>	<p>This parent element has information about the delivery time and location of the invoiced goods and services. This element has no value in the template, but it has various child elements with</p>

					elements with information about the delivery address in the template.	information about the delivery address in the template.
138	cbc:ActualDeliveryDate	transaction.trandate	Date	Both	<p>This element displays the date when the supply of goods and services is completed. This element is displayed along with the value on the generated e-document only if the value is provided in the <b>Date</b> field of the transaction. This date must be in YYYY-MM-DD format.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element displays the date when supply of goods and services is completed. This element is displayed along with the value on the generated e-document only if the value is provided in the <b>Date</b> field of the transaction. This date must be in YYYY-MM-DD format.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
139	cac:DeliveryLocation			Both	<p>This element has information about the location where goods and services were delivered. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element has information about the location to which the goods and services are delivered. This element has no value in the template. This element is not printed if the shipping country is not available in the transaction.</p>
140	cbc:ID		Custom Field	Both	<p>This element displays the identifier for the delivery location of the goods and services. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
141	@schemeID		Custom Value	Both	<p>This element is a part of cbc:ID element and displays the scheme ID of the <b>Deliver to Location</b> field. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this value.
142	cac:Address			Both	This parent element has information about the address where invoiced goods or services are delivered. This element has no value in the template, but it has child elements that have values associated with it.	This parent element has information about the address to which the invoiced goods or services are delivered. This element has no value in the template, but it has various child elements that have values associated with it.
143	cbc:StreetName	transaction.shipaddr1	Address 1 (Transaction's address line 1)	Both	This element displays the main address line in the transaction's shipping address. This element has the value of <b>Address 1</b> field in the transaction's shipping address. This element is not displayed if the <b>Address 1</b> field has no value in the transaction.	This element displays the main address line in the transaction's shipping address. This element has the value of <b>Address 1</b> field in the transaction's shipping address. The element and its value are not displayed if the <b>Address 1</b> field has no value in the transaction.
144	cbc:AdditionalStreetName	transaction.shipaddr2	Address 2 (Transaction's address line 2)	Both	This element displays the main address line in the transaction's shipping address. This element has the value of the <b>Address 2</b> field in the transaction's shipping address. The element and its value are not displayed if the <b>Address 2</b> field has no value.	This element displays the main address line in the transaction's shipping address. This element has the value of the <b>Address 2</b> field in the transaction's shipping address. The element and its value are not displayed if the <b>Address 2</b> field has no value.
145	cbc:CityName	transaction.shipcity	City (Transaction's shipping address's city)	Both	This element displays the common name of the city, town, or village of the delivery location in the shipping address. This element has the value of the <b>City</b> field in the transaction's shipping address. The element and its value are not displayed if the <b>City</b> field has no value.	This element displays the common name of the city, town or village of the delivery location in the shipping address. This element has the value of the <b>City</b> field in the transaction's shipping address. The element and its value are not displayed if the <b>City</b> field has no value.
146	cbc:PostalZone	transaction.shipzip	Zip (Transaction's shipping address's zip)	Both	This element displays an identifier for a group of properties based on their postal service. This element has the value of the <b>Zip</b> field in the transaction's shipping address. The element and its value are not displayed if the <b>Zip</b> field has no value.	This element displays an identifier for a group of properties based on their postal service. This element has the value of the <b>Zip</b> field in the transaction's shipping address. The element and its value are not displayed if the <b>Zip</b> field has no value.
147	cbc:CountrySubentity	transaction.shipstate	State (Transaction's shipping address's state)	Both	This element displays the subdivision of a country in the shipping address. This element has the value of the <b>State</b> field in the transaction's shipping address. The element and its value are not displayed if the <b>State</b> field has no value.	This element displays the subdivision of a country in the shipping address. This element has the value of the <b>State</b> field in the transaction's shipping address. The element and its value are not displayed if the <b>State</b> field has no value.
148	cac:AddressLine			Both	This parent element displays additional information about the shipping address. This element has no value but has child elements with values in the template. This parent element is not displayed if its child elements do not have values in the template.	This parent element displays additional information about the shipping address. This element has no value in the template but has a child element with value in <b>Address 3</b> field of the transaction's shipping address. This element is not displayed if there is no value in <b>Address 3</b> .

149	cbc:Line	transaction.addr3	Address 3 ( Transaction's address line 3)	Both	<p>This is a parent element and a child of cac:AddressLine, with information about the additional line of an address, supporting the primary address. This element contains value of <b>Address 3</b> field of the transaction's shipping address in the template. This element is not displayed if the <b>Address 3</b> field has no value.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element is the child of cac:AddressLine with information about additional address line to support the main line. This element contains value of <b>Address 3</b> field of the transaction's shipping address in the template. This element is not displayed if there is no value in <b>Address 3</b> field.</p>
150	cac:Country			Both	This parent element displays the country of the shipping address. This element has no value, but it has child elements with values in the template.	This parent element displays the country of the shipping address. This element has no value, but it has child element with value of the <b>Country</b> field in the template. This element is not printed if there is no value in the <b>Country</b> field.
151	cbc:Id entific ationC ode	transaction. shipcountry	Country (T ransaction 's shipping addressee's country)	Both	This element displays the ISO code of the country in the transaction's shipping address. It displays the value of the <b>Country</b> field in the template.	This element displays the ISO code of the country in transaction's shipping address. It displays the value of the <b>Country</b> field in the template.
152	cac:Deli veryP arty			Both	This parent element displays the details of the delivery party. This element has no value but has a child element with values associated to it in the template.	This parent element displays the details of the delivery party. This element has no value but has a child element with values associated to it in the template.
153	cac:Pa rtyNa me			Both	This is a parent element and a child of cac:DeliveryParty, with information about the delivery party's name. This element has no value but has a child elements with values associated to it in the template.	This is a parent element and a child of cac:DeliveryParty with information of the delivery party's name. This element has no value but has a child element with values associated to it in the template.
154	cbc:N ame	transaction.shipa ddressee	Addressee ( Transaction 's shipping address's country)	Both	This element displays the name of the customer or party whom the goods and services are to be delivered. This element displays the value of the <b>Addressee</b> field in the transaction's shipping address.	This element displays the name of the customer or party to which the goods and services are to be delivered. This element displays the value of the <b>Addressee</b> field in a transaction's shipping address.
155	cac:Pa yment Means			Both	<p>This parent element provides a group of business terms by using different child elements to give payment information about an invoice or refunds in a credit memo. This element has no value, but it has various child elements that have values associated with it in the template.</p>	<p>This parent element provides a group of business terms by using different child elements to give information about the payments or refunds. This element has no value, but it has various child elements that have values associated with it in the template.</p>
156	cbc:Pa yment Means Code		Custom Field	Both	<p>This element displays the payment type code for a payment made or to be made. This element has no value in the template.</p> <p>Example: 30</p> <p>The following list has the details of the Payment Types and its associated code:</p> <ul style="list-style-type: none"> <li>■ Credit Transfer – 30</li> <li>■ Direct Debit – 49</li> <li>■ Opostgiro – 50</li> <li>■ Credit Card – 54</li> <li>■ Debit Card – 55</li> <li>■ Bankgiro – 56</li> <li>■ Sepa Credit Transfer – 58</li> <li>■ Sepa Direct Debit –</li> <li>■ Online Payment Service – 68</li> <li>■ Finland – 93, 94, 95.</li> <li>■ Instrument not defined – 1</li> <li>■ National or Regional Clearing – 9</li> <li>■ Cash – 10</li> </ul> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element displays the payment type code for a payment made or to be made. This element has no value in the template.</p> <p>Example Value: 30</p> <p>Example: 30</p> <p>The following list has the details of the Payment Types and its associated code:</p> <ul style="list-style-type: none"> <li>■ Credit Transfer – 30</li> <li>■ Direct Debit – 49</li> <li>■ Opostgiro – 50</li> <li>■ Credit Card – 54</li> <li>■ Debit Card – 55</li> <li>■ Bankgiro – 56</li> <li>■ Sepa Credit Transfer – 58</li> <li>■ Sepa Direct Debit –</li> <li>■ Online Payment Service – 68</li> <li>■ Finland – 93, 94, 95.</li> <li>■ Instrument not defined – 1</li> <li>■ National or Regional Clearing – 9</li> <li>■ Cash – 10</li> </ul> <p>You can map the value with the required custom field in the SuiteApp.</p>
157	@na me		Custom Value	Both	This element displays the name of the mode of payment. This element has no value in the template.	This element displays the name of the payment method used or to be used to make a payment. This element has no value in the template.

					Example: Credit Transfer  You can map the value with the required custom field in the SuiteApp.	Example: Credit Transfer  You can map the value with the required custom field in the SuiteApp.
158	cbc:PaymentID		Custom Field	Both	<p>This element displays a text value for associating the payment and the invoice issued by the seller. This is used for creditors' critical reconciliation information. The value helps the seller assign an incoming payment to the relevant payment process.</p> <p>This element has no value in the template. A custom field with a payment reference can be used for this element.</p>	<p>This element displays a text value used to establish a link between the payment and the invoice issued by the seller. It is used for creditor's critical reconciliation information. The value helps the seller to assign an incoming payment to the relevant payment process. This element has no value in the template. A custom field with payment reference can be used for this element.</p> <p>Example: 432948234234234</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
159	cac:CardAccount				<p>This parent element represents a group of business terms with information about the card used to make a payment against an issued invoice.</p> <p>If a buyer made a payment using a card such as a credit or debit card, then the information about the Production account Number (PAN) is mentioned in the invoice.</p> <p>This element has no value in the template, but it has various child elements that have values associated with it.</p> <p>The element is displayed only if there are any payments made using cards or if the refund method in the Credit Memo is a card.</p>	<p>This parent element represents a group of business terms with information about the card used to make a payment against an issued invoice.</p> <p>If a buyer made a payment using a card such as credit or debit card, then the information about the Production account Number (PAN) is mentioned in the invoice.</p> <p>This element has no value, but it has various child elements that have values associated with it in the template.</p> <p>This element is printed only if payments are made using card in Invoice or if the refund method used in Credit Memo is Card.</p>
160	cbc:PrimaryAccountNumberID	payment.cardNumber (Refer CDS)		Both	<p>This element represents the Production account Number (PAN) of the card used for payment. As per card payment security standards, an invoice must not indicate the full PAN of the card.</p> <p>Example: 1234</p> <p>The last 4 to 6 digits of the PAN (BT-87) are indicated if payment card information (BG-18) is provided in the invoice.</p> <p>This element displays the last 4 digits of any cards used for payment and uses custom data source plug-in to display this information in the template.</p>	<p>This element represents the Production account Number of the card used to make a payment or refund. In accordance with the card payments security standards, an invoice must not have a full PAN of the card.</p> <p>Example: 1234</p> <p>The last 4 to 6 digits of the payment card PAN number (BT-87) is present if payment card information (BG-18) is provided in the invoice.</p> <p>This element displays the last 4 digits of the card(s) used for payments or refunds. It uses a custom data source plug-in to display this information in the template.</p>
161	cbc:NetworkID	payment.paymentMethodName (Refer CDS)		Both	<p>This element displays the card network identifier such as VISA, American Express, Card.</p> <p>Example: VISA</p> <p>The payment method name is displayed in this element using a custom data source plug-in.</p>	<p>This element displays the card network identifier such as VISA, American Express, Card.</p> <p>Example: VISA</p> <p>The payment method name or the refund method is displayed in this element using custom data source plug-in.</p>
162	cbc:HolderName	payment.cardHolderName (Refer CDS)		Both	<p>This element displays the name of the holder of the card.</p> <p>Example: John Doe</p> <p>This element uses a custom data source plug-in to display the name of the cardholder.</p>	<p>This element displays the cardholder's name.</p> <p>Example: John Doe</p> <p>This element displays the card holder name by using the custom data source plug-in.</p>
163	cac:PayeeFinancialAccount			Both	<p>This parent element displays a group of business terms specifying the credit transfer payments.</p> <p>This element has no value, but it has various child elements with values associated with it in the template.</p>	<p>This parent element displays a group of business terms specifying the credit transfer payments. This element has no value, but it has various child elements that have values associated with it in the template.</p>
164	cbc:ID		Custom Field	Both	<p>This element identifies a payment account. It displays a unique ID of the financial payment account for a payment service provider to which the payment is made, such as IBAN or BBAN.</p> <p>Example: NO99991122222</p>	<p>This element identifies a payment account. It displays a unique ID of the financial payment account for a payment service provider to which the payment is made, such as IBAN or BBAN.</p> <p>Example: NO99991122222</p> <p>This element has no value in the template.</p>

					This element has no value in the template. You can map the value with the required custom field in the SuiteApp.	You can map the value with the required custom field in the SuiteApp.
165	cbc:Name		Custom Field	Both	<p>This element displays the payment account name of the service provider to which the payment is made.</p> <p>Example: Payment Account</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
166	cac:FinancialInstitutionBranch			Both	<p>This parent element displays the details of a payment service provider. This element has no value, but it has various child elements with values associated to it in the template.</p>	<p>This parent element displays the details of a payment service provider. This element has no value, but it has various child elements that have values associated with it in the template.</p>
167	cbc:ID		Custom Field	Both	<p>This element displays an ID for payment service provider where a payment account is located such as BIC or a national clearing code. Identification scheme identifier is not used in this element.</p> <p>Example: 9999</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element displays an ID for payment service provider where a payment account is located such as BIC or a national clearing code. Identification scheme identifier is not used in this element.</p> <p>Example: 9999</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
168	cac:PaymentMandate			Both	<p>This parent element displays a group of business terms for direct debit transactions. This element has no value, but it has various child elements with values associated to it in the template.</p>	The template does not support this element.
169	cbc:ID		Custom Field	Both	<p>This element displays a unique identifier assigned by the payee for referencing a direct debit mandate. This is used to notify the buyer of a SEPA direct debit in advance.</p> <p>Example: 123456</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
170	cac:PayerFinancialAccount			Both	<p>This parent element has no value, but it has various child elements with values associated to it in the template. It has information about the payer's financial account.</p>	The template does not support this element.
171	cbc:ID		Custom Field	Both	<p>This element displays an ID with information about the account to be debited by direct debit.</p> <p>Example: 12345676543</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
172	cac:PaymentTerms		Invoice		<p>This parent element has information about the payment terms applied to the due amount. This parent element has no value, but it has a child element with value associated with it in the template.</p>	<p>This parent element has information about the payment terms applied on the due amount. This element has no value, but it has a child element that has values associated with it in the template.</p>
173	cbc:Note		Terms	Invoice	<p>This parent element displays text describing the payment terms that apply to the amount due for payments. It can also describe applicable penalties.</p> <p>If an amount is due for payment (BT-115), then the payment due date (BT-9) or the payment terms (BT-20) is indicated. This element displays a value in the <b>Terms</b> field on a transaction.</p> <p>Example: Net within 30 days</p>	<p>This parent element displays a description text of the payment terms that apply on the due amount of the payment. It can also include a description of applicable penalties.</p> <p>If an amount is due for payment (BT-115), then the payment due date (BT-9) or the payment terms (BT-20) is present. This element displays a value in the <b>Terms</b> field for a transaction.</p> <p>Example: Net within 30 days</p>

174	cac:AllowanceCharge			Both	<p>This parent element displays a group of business terms with information about applicable allowances of a complete invoice. This element also displays a group of business terms with information about charges and taxes excluding VAT for a complete invoice.</p> <p>This element has no value and is displayed in the generated e-document only if there is a Header Discount or Shipping Cost associated with the transaction.</p>	<p>This parent element displays a group of business terms with information applicable allowances to a complete invoice. This element also displays a group of business terms with information about charges and taxes excluding VAT for a complete invoice.</p> <p>This element has no value and is displayed in the generated e-document only if it has a Header Discount or Shipping Cost associated with the transaction.</p>
175	cbc:ChargeIndicator	True or False	Fixed value	Both	<p>This element displays the charge type used for a transaction. The generated e-document displays the value true when informing about charges and the value false when informing about allowances.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>■ True for shipping and handling.</li> <li>■ False for header discounts.</li> </ul>	<p>This element displays the charge type used for a transaction. The generated e-document displays the value true when informing about charges and the value false when informing about allowances.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>■ True for shipping and handling.</li> <li>■ False for header discounts.</li> </ul>
176	cbc:AllowanceReasonCode		Fixed value	Both	<p>This element displays the document level allowance code or the allowance charge reason code. You can use the UNCL5189 code from the code list for allowance of a subset. For charges, you can use the UNCL7161 code from the code list.</p> <p>The document level allowance reason code and document level allowance reason display the same reason.</p> <p>Example: 95 for Discount SAA for Shipping and Handling</p> <p>This element does not have a value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element displays the document level allowance code or the allowance charge reason code. You can use the UNCL5189 code from a code list for allowance of a subset. For charges, you can use the UNCL7161 code from the code list.</p> <p>The document level allowance reason code and document level allowance reason display the same reason.</p> <p>Example: 95 for Discount SAA for Shipping and Handling</p>
177	cbc:AllowanceReason		Fixed value	Both	<p>This element displays document level allowance reason or charge reason in text format. The document level allowance reason code and the document level allowance reason, display the same allowance reason.</p> <p>This element displays the following values:</p> <ul style="list-style-type: none"> <li>■ Shipping and handling for charges</li> <li>■ Discount for allowances</li> </ul>	<p>This element displays document level allowance reason or charge reason in text format. The document level allowance reason code and the document level allowance reason display the same allowance reason.</p> <p>This element displays the following values:</p> <ul style="list-style-type: none"> <li>■ Shipping and handling for charges</li> <li>■ Discount for allowances</li> </ul>
178	cbc:MultiplierFactorNumeric	transaction.discountRate	Rate (Header discount)	Both	<p>This element displays the document level allowance or charge percentage.</p> <p>It is displayed only for header discounts and uses the value of the <b>Rate</b> field.</p> <p>The percentage is used with the document level allowance base amount to calculate the document level or charge amount. The value must be 20.</p> <p>Example: 20</p> <p>This element has no value associated with Shipping and Handling. It displays the discount rate in the generated e-document retrieved from a transaction.</p>	<p>This element displays the document level allowance or charge percentage. This element is displayed only for header discounts and uses the value of the <b>Rate</b> field in a transaction.</p> <p>The percentage is used with the document level allowance base amount to calculate the document level or charge amount. For 20 percent the value must be 20.</p> <p>Example: 20</p>
179	cbc:Amount	transaction.shippingcost+transaction.handlingcost (For shipping and handling) transaction.discountTotal (for discount)	Discount Item	Both	<p>This element displays the document level allowance or charge amount without VAT. This value must be rounded off to a maximum of 2 decimal.</p> <p>This element uses the following values from the Summary section of a transaction in the template:</p> <ul style="list-style-type: none"> <li>■ The sum of shipping and handling costs for charges</li> </ul>	<p>This element displays the document level allowance or charge amount without VAT. The value must be rounded maximum to 2 decimals.</p> <p>This element uses the following values from the Summary section of a transaction in the template:</p> <ul style="list-style-type: none"> <li>■ The sum of shipping and handling costs for charges</li> <li>■ The value of Discount Item field for allowances.</li> </ul>

					<ul style="list-style-type: none"> <li>■ The value of Discount Item field for allowances.</li> </ul> <p>Example: 200</p>	Example: 200
180	@curr encyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with the cbc:Amount element, to display the ISO code of a transaction's currency.</p>	<p>This element is used with the cbc:Amount element, to display the ISO code of a transaction's currency.</p>
181	cbc:BaseAmount	transaction.subtotal	Subtotal	Both	<p>This element displays the base amount along with the document level allowance or charge percentage. It calculates the document level allowance or charge amount. The value is rounded off to 2 decimal places in the template.</p> <p>A value is provided in the <b>Subtotal</b> field in the Summary section of a transaction</p> <p>Example value: 200</p>	<p>This element displays the base amount along with the document level allowance or charge percentage. It calculates the document level allowance or charge amount. The value is rounded off to 2 decimal places in the template.</p> <p>A value is provided in the <b>Subtotal</b> field present in the summary section of a transaction. This element is displayed only if the header discount is available.</p> <p>Example value: 200</p>
182	@curr encyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:BaseAmount element, to display the currency's ISO code used by a transaction.</p>	<p>This element is used with cbc:BaseAmount element to display the currency's ISO code used by a transaction.</p>
183	cac:TaxCategory			Both	<p>This parent element displays the unique tax category and its details. This element has no value, but it has child elements with values associated to it in the template.</p>	<p>This parent element displays the unique tax categories and their details. This element has no value, but it has a child element that has values associated with it in the template.</p>
184	cbc:ID		Custom Field	Both	<p>This element displays a code to identify the VAT category applied to document level allowance or charge.</p> <p>Example: Value: S</p> <p>In EMEA, the following codes are used:</p> <ul style="list-style-type: none"> <li>■ AE - Vat Reverse Charge – Specifies the standard VAT rate reversed from the invoice.</li> <li>■ E - Exempt from Tax – Specifies that taxes are not applicable</li> <li>■ S – Standard Rate – Specifies the standard rate</li> <li>■ Z – Zero Rated Goods – Specifies that the goods are at zero rate</li> <li>■ G - Free Export Item, VAT Not Charged – Specifies that the item is free export and taxes are not charged</li> <li>■ K – VAT exempt for EEA intra-community supply of goods and services – A tax category code that identifies of the item is VAT exempted due to intra community supply in the European Economic Area</li> </ul> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element displays a code to identify the VAT category applied to document level allowance or charge.</p> <p>Example: Value: S</p> <p>In EMEA, the following codes are used:</p> <ul style="list-style-type: none"> <li>■ AE - Vat Reverse Charge – Specifies the standard VAT rate reversed from the invoice.</li> <li>■ E - Exempt from Tax – Specifies that taxes are not applicable</li> <li>■ S – Standard Rate – Specifies the standard rate</li> <li>■ Z – Zero Rated Goods – Specifies that the goods are at zero rate</li> <li>■ G - Free Export Item, VAT Not Charged – Specifies that the item is free export and taxes are not charged</li> <li>■ K – VAT exempt for EEA intra-community supply of goods and services – A tax category code that identifies of the item is VAT exempted due to intra community supply in the European Economic Area</li> </ul> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
185	cbc:Percent	For LegacyTax account: <ul style="list-style-type: none"> <li>■ transaction.shippingtax1rate</li> <li>■ taxCodeDet.taxRate</li> </ul> For SuiteTax account: <ul style="list-style-type: none"> <li>■ custom.shippingTax.RatesOfSuiteTax</li> <li>■ suiteTaxCodeDet.taxRate</li> </ul>	Shipping Tax Rate	Both	<p>This element displays the document level allowance or VAT charge rate. The VAT rate is displayed as the percentage that applies to the document level allowance or charge.</p> <p>The categories include the following:</p> <ul style="list-style-type: none"> <li>■ Tax rates of the unique tax codes when displaying header discounts.</li> <li>■ Tax rates of shipping when displaying shipping and handling charges.</li> </ul> <p>It uses Custom Data Source plug-in to display values.</p> <p>Example: 25</p> <p>This element displays the value in the <b>Shipping Tax Rate</b> field. This element is not displayed if the field has no value.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element displays the document level allowance or VAT charge rate. The VAT rate is displayed as the percentage that applies to the document level allowance or charge.</p> <p>The categories include the following:</p> <ul style="list-style-type: none"> <li>■ Tax rates of the unique tax codes when displaying header discounts.</li> <li>■ Tax rates of shipping when displaying shipping and handling charges.</li> </ul> <p>It uses custom data source plug-in to display the values.</p> <p>Example: 25</p>

186	cac:TaxScheme			Both	<p>This parent element displays the details of the tax scheme used in a transaction. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This parent element displays the details of the tax scheme used in a transaction. This element has no value in the template.</p>
187	cbc:ID		Fixed Value	Both	<p>This element displays the name of the tax scheme used in a transaction. A fixed VAT value is provided in the template for this element.</p>	<p>This element displays the name of the tax scheme used in a transaction. A fixed value VAT is provided in the template for this element.</p>
188	cac:Tax Total			Both	<p>This parent element displays the total tax details including the amounts for a transaction. When a tax currency code is provided, two instances of tax total must be present and only one instance with tax subtotal must be used. This element does not have a value in the template.</p> <p>This element has no value in the template.</p>	<p>This parent element displays the total tax details including the amounts for a transaction. When a tax currency code is provided, two instances of tax total must be present and only one instance with tax subtotal must be used.</p> <p>This element has no value in the template.</p>
189	cbc:TaxAmount	transaction.taxtotal	Tax	Both	<p>This element displays the total VAT amount for an invoice or the total VAT amount in the accounting currency as required by the seller.</p> <p>This value is rounded off to a maximum of 2 decimal places in the template.</p> <p>This element prints the value of the <b>Tax Total</b> field from the Summary section of the invoice.</p>	<p>This element displays the total VAT amount for an invoice or the total VAT amount present in the accounting currency as required by the seller. This value is rounded off to maximum 2 decimals in template.</p> <p>This element prints the value of the <b>Tax Total</b> field from the Summary section of the invoice.</p>
190	@curr encyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:TaxAmount element, to display the ISO code of the currency used by the transaction.</p>	<p>This element is used with cbc:TaxAmount element to display the ISO code of the currency used by the transaction.</p>
191	cac:TaxSubtotal			Both	<p>This parent element displays a group of business terms with information about VAT break down based on different categories, rates and exemption reasons.</p> <p>This element has no value, but it has child elements with values associated with it.</p>	<p>This parent element displays a group of business terms with information about VAT break down on basis of different categories, rates, and exemption reasons.</p> <p>This element has no value, but it has various child elements with values associated to it in the template.</p>
192	cbc:TaxableAmount	taxDetails.taxableAmount (refer CDS)	Amount of all items calculated per unique tax category	Both	<p>This element displays all the taxable amount for each item specific to a unique VAT category code. The element uses custom data source plug-in to get the values.</p> <p>It also adds the shipping cost to the taxable amount as required. This value must be rounded off to 2 decimal places.</p>	<p>This element displays all the taxable amounts for each item specific to a unique VAT category code or tax code. It uses custom data source plug-in to get the values. The element also adds the shipping cost wherever required in the taxable amount. This value must be rounded off to 2 decimals.</p>
193	@curr encyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:TaxableAmount element to display the ISO code of the transaction's currency.</p>	<p>This element is used with cbc:TaxableAmount element to display the ISO code of the transaction's currency.</p>
194	cbc:TaxAmount ((Using CDS))		Tax amount of all items calculated per unique tax category	Both	<p>This element displays the total VAT amount for each item specific to a unique VAT category code. The element uses custom data source plug-in to get the values.</p> <p>The sum of all VAT amounts of the Unique VAT category is equal to <b>Total Tax</b> field in the Summary section. The value is rounded off to a maximum of 2 decimal places in the template.</p>	<p>This element displays the total VAT or Tax amount for each item specific to a unique VAT category code or tax code. It uses custom data source plug-in to get the values. The value is rounded off to a maximum of two decimals in the template.</p>
195	@curr encyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:TaxAmount to display the ISO code of the transaction's currency.</p>	<p>This element is used with cbc:TaxAmount to display the ISO code of the transaction's currency.</p>
196	cac:TaxCategory			Both	<p>This parent element displays the VAT category details. This element has no value, but it has child elements with values associated with it.</p>	<p>This parent element displays the VAT category details. This element has no value, but it has various child elements that have values associated with it in the template.</p>
197	cbc:ID		Custom Field	Both	<p>This element displays a code to identify the VAT category.</p> <p>Example:</p> <p>Value: S</p>	<p>This element displays a code to identify the VAT category.</p> <p>Example:</p> <p>Value: S</p>

					<p>In EMEA, the following codes are used:</p> <ul style="list-style-type: none"> <li>■ AE - Vat Reverse Charge - Specifies the standard VAT rate reversed from th</li> <li>■ E - Exempt From Tax - Specifies that taxes are not applicable</li> <li>■ S - Standard Rate - Specifies the standard rate.</li> <li>■ Z - Zero Rated Goods - Specifies that the goods are at zero rate</li> <li>■ G - Free Export Item, VAT Not Charged - Specifies that the item is free to export and taxes are not charged.</li> <li>■ K - VAT Exempt for EEA intra-community Supply of Goods and Services - A tax category code that identifies of the item is VAT exempted due to intra community supply in the European Economic Area.</li> </ul> <p>This element has no value in the template. You can map the value with the required custom field in the SuiteApp.</p>	<p>In EMEA, the following codes are used:</p> <ul style="list-style-type: none"> <li>■ AE - Vat Reverse Charge - Specifies the standard VAT rate reversed from th</li> <li>■ E - Exempt From Tax - Specifies that taxes are not applicable</li> <li>■ S - Standard Rate - Specifies the standard rate.</li> <li>■ Z - Zero Rated Goods - Specifies that the goods are at zero rate</li> <li>■ G - Free Export Item, VAT Not Charged - Specifies that the item is free to export and taxes are not charged.</li> <li>■ K - VAT Exempt for EEA intra-community Supply of Goods and Services - A tax category code that identifies of the item is VAT exempted due to intra community supply in the European Economic Area.</li> </ul> <p>This element has no value in the template. You can map the value with the required custom field in the SuiteApp.</p>
198	cbc:Percent	In LegacyTax account: <ul style="list-style-type: none"> <li>■ taxDetails. taxRate</li> <li>■ custom. handlingTax Rate</li> <li>■ custom. shippingTax Rate</li> </ul> In SuiteTax account: <ul style="list-style-type: none"> <li>■ taxDetails. taxRates</li> <li>■ custom. handlingTax Rate</li> <li>■ custom. shippingTax Rate</li> </ul>	Tax Rate	Both	<p>This element displays the VAT category rate for each unique tax code used by items for shipping. This element is not displayed if a VAT category has no rate.</p> <p>The VAT rate is displayed as percentage for the applicable VAT category.</p> <p>This uses custom data source plug-in to retrieve the values. In SuiteTax accounts, if the tax type has multiple tax rates, then rates are also displayed for single tax type.</p> <p>Example: 25</p>	<p>This element displays the VAT category rate for each unique tax code used in items and shipping. The VAT rate is displayed as a percentage that applies to the applicable VAT category. It uses custom data source plug-in to get the values.</p> <p>In SuiteTax accounts, if a tax type has multiple tax rates, then multiple tax rates are displayed for a single tax type.</p> <p>Example: 25</p>
199	cbc:TaxExemptionReasonCode		Custom Field	Both	<p>This element displays a VAT exemption reason code. A coded statement indicates the reason for exempting the VAT amount.</p> <p>This element has no value in the template. You can map the value with the required custom field in the SuiteApp.</p>	<p>This element displays a VAT exemption reason code. A coded statement is provided explaining the reason for exempting the VAT amount.</p> <p>This element has no value in the template. You can map the value with the required custom field in the SuiteApp.</p>
200	cbc:TaxExemptionReason		Custom Field	Both	<p>This element displays the VAT exemption reason text.</p> <p>Example: Exempt</p> <p>This element has no value in the template. You can map the value with the required custom field in the SuiteApp.</p>	<p>This element displays the VAT exemption reason text. A statement in text is provided explaining the reason for exempting the VAT amount.</p> <p>Example: Exempt</p> <p>This element has no value in the template. You can map the value with the required custom field in the SuiteApp.</p>
201	cac:TaxScheme				<p>This parent element displays the tax scheme. This element has no value, but it has child elements with values associated with it in the template.</p>	<p>This parent element displays tax scheme. This element has no value, but it has various child elements that have values associated with it in the template.</p>
202	cbc:ID		Fixed Value	Both	<p>This element displays the name of the tax scheme used in a transaction. A fixed value VAT is assigned to this element.</p>	<p>This element displays the name of the tax scheme used in a transaction. A fixed value VAT is assigned in a template.</p>
203	cac:LegalMonetaryTotal			Both	<p>This parent element displays the document totals. This element has no values, but it has child elements that provide monetary totals for an invoice or a credit memo in a template.</p>	<p>This parent element displays the document totals. This element has no value, but it has various child elements that have values associated with it in the template.</p>

204	cbc:LineExtensionAmount	transaction.subtotal	Subtotal	Both	<p>This element displays the sum of all the invoice line net amounts in an invoice. The value is rounded off to a maximum of 2 decimal places in the template.</p> <p>This element displays the <b>Subtotal</b> field value from the transaction's Summary tab.</p>	<p>This element displays the sum of all the invoice line net amounts in an invoice. This value is rounded off to maximum 2 decimals in template.</p> <p>This element displays the <b>Subtotal</b> field value from the transaction's Summary tab.</p>
205	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	This element is used with cbc:LineExtensionAmount element to display the ISO code of a transaction's currency.	This element is used with cbc:LineExtensionAmount element to display the ISO code of a transaction's currency.
206	cbc:TaxExclusiveAmount	transaction.altshippingcost + transaction.althandlingcost + transaction.subtotal - transaction.discounttotal	Subtotal + Allowances - Discount	Both	<p>This element displays the total amount of an invoice without VAT. The value of this element must be rounded off to two decimals. In a template, you must add the transaction's amount in the Subtotal field in the Summary tab with the handling and shipping charges to get a value.</p> <p>Example: 3600.0</p> <p>You can add extra allowances and change the template to accommodate this element.</p>	<p>This element displays the total amount of an invoice without VAT. The value of this element must be rounded off to two decimals. In a template, you must add the transaction's amount in the Subtotal field in the Summary tab with the handling and shipping charges and subtract the total discount to get a value.</p> <p>Example: 3600.0</p> <p>You can add extra allowances and change the template to accommodate this element.</p>
207	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	This element is used with cbc:TaxExclusiveAmount to display the ISO code of the transaction's currency.	This element is used with cbc:TaxExclusiveAmount to display the ISO code of the transaction's currency.
208	cbc:TaxInclusiveAmount	transaction.total	Total	Both	This element displays the total amount of a transaction with VAT. The value is displayed in the <b>Total</b> field on a transaction's Summary tab. This value is rounded off to 2 decimal places in the template.	This element displays the total amount of transactions with VAT. The value is displayed in the <b>Total</b> field on a transaction's Summary tab. This value is rounded off to two decimals in template.
209	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	This element is used with cbc:TaxInclusiveAmount element to display the ISO code of the transaction's currency.	This element is used with cbc:TaxInclusiveAmount element to display the ISO code of the transaction's currency.
210	cbc:AllowanceTotalAmount	transaction.discounttotal	Discount Item	Both	<p>This element displays the sum of allowances at the document level. The value of this element is rounded off to 2 decimal places.</p> <p>Example: 200.0</p>	<p>This element displays the sum of allowances at the document level. The value of this element is rounded off to two decimals.</p> <p>Example: 200.0</p>
211	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	This element is used with cbc:AllowanceTotalAmount element to display the ISO code for the transaction's currency.	This element is used with cbc:AllowanceTotalAmount element to display the ISO code for the transaction's currency.
212	cbc:ChargeTotalAmount	transaction.altshippingcost	Shipping Cost	Both	<p>This element displays the shipping cost charge at the document level in the transaction. The value is rounded off to 2 decimal places in the template.</p> <p>Example: 10.0</p>	<p>This element displays the sum of charges at document level in the transaction. The value is rounded to two decimals in the template. It displays the total cost of shipping.</p> <p>Example: 123.23</p>
213	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	The element is used with cbc:ChargeTotalAmount to display the ISO code of the transaction's currency.	The element is used with cbc:ChargeTotalAmount to display the ISO code of the transaction's currency.
214	cbc:PrepaidAmount	transaction.amountpaid	Payments	Both	<p>This element displays the sum of the amount paid in advance, for a transaction. The value is rounded off to a maximum 2 decimal places in the template.</p> <p>This value displays the sum of all the payments made for a transaction.</p>	<p>This element displays the sum of the amount paid in advance for a transaction. The value is rounded off by maximum of two decimals in the template.</p> <p>This value displays the sum of all the payments made for a transaction.</p> <p>Example: 1000.0</p>
215	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	This element is used with cbc:PrepaidAmount element to display the ISO code of the transaction's currency.	This element is used with cbc:PrepaidAmount element to display the ISO code of the transaction's currency.
216	cbc:PayableRoundingAmount		Custom Field	Both	<p>This element displays the rounding amount. The amount is added to the invoice total to round off the amount to be paid. The value is rounded off by maximum two decimals in template.</p> <p>Example: 0.0</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.

217	@curr encyID	transaction. currency.symbol	Transaction' s Currency	Both	This element is used with cbc:PayableRoundingAmount element to display the ISO code of the transaction's currency.	This element is used with cbc:PayableRoundingAmount element to display the ISO code of the transaction's currency.
218	cbc:Pa yableA mount	For Invoice: transaction.amou ntremainingtotal box  For Credit Memo: transaction.amoun tremaining	Amount Due	Both	This element displays the amount due for payment and the outstanding amount that is requested to be paid. The value is rounded off to a maximum 2 decimal places in the template. The <b>Amount Due</b> field value in the Summary tab is used for the transaction.  Example: 3500.0	This element displays the amount due for payment and the outstanding amount requested to be paid. The value is rounded off by maximum two decimals in the template. The <b>Amount Due</b> field value in the Summary tab is used for the transaction.  Example: 3500.12
219	@curr encyID	transaction. currency.symbol	Transaction' s Currency	Both	This element is used with cbc:PayableAmount element to display the ISO code of the transaction's currency.	This element is used with cbc:PayableAmount element to display the ISO code of the transaction's currency.
220	cac:In voice Line			Both	This parent element displays the line item components. It also has a group of business terms providing information about individual transaction lines. This element has no value, but it has child elements with values in the template.	This parent element displays the line item components on Invoice. It also has a group of business terms providing information about individual Invoice lines. This element does not have value, but it has child elements with values in template. This element is used only in Invoice transaction type.
221	cac:Cr editNo teLine			Cred it Me mo	This parent element displays the line item components on a Credit Memo. It also has a group of business terms providing information about individual invoice lines for the credit memo. This element has no value, but it has child elements with values associated to it in the template. This element is used only for the Credit Memo transaction type.	This parent element displays the line item components for the Credit Memo transaction type. It also has a group of business terms providing information about individual Invoice lines for Credit Memo transactions.  This element has no value, but it has various child elements that have values associated with it in the template. This element is used only in Credit Memo transaction type.
222	cbc:ID	item.line	Line Sequen ce Number	Both	This element displays the transaction line ID for each item. It has a unique identifier for an individual line in the transaction. The line sequence number of a line item is used in the template.  Example: 12	This element displays the transaction line ID for each item. It has a unique identifier for individual line within the transaction. The line sequence number of a line item is used in the template.  Example: 12
223	cbc: Note	transaction.memo	Memo	Both	This element is used in the invoice line note. It is a text note with unstructured information relevant to the invoice line. The header level <b>Memo</b> field is used as value in the template.  Example: New article number 12345	This element is used in invoice line note. It is a text note giving unstructured information relevant to the invoice line. The header level <b>Memo</b> field is used as value in the template.  Example: New article number 12345
224	cbc:Inv oicedQ uantity	item.quantity	Quantity	Both	This element displays the value of invoiced quantity of each item in the transaction line. The <b>Quantity</b> field on the line item is used as value in the template.  Example: 100  The element value with a decimal separator (.) or a whole number is valid. Group separator or more than one decimal separator is invalid.	This element displays the value of invoiced quantity of each item in the invoice line. The value of the <b>Quantity</b> field in the line item is used in the template. This element is not printed if the <b>Quantity</b> field has no value in the item. This element is used only in Invoice transaction type.  Example: 100  The element value with a decimal separator (.) or a whole number is valid. Group separator or more than one decimal separator is invalid.
225	cbc:C redite dQua ntity	item.quantity	Quantity	Cred it Me mo	This element displays the value of invoiced quantity of each item in the credit note line. The <b>Quantity</b> field on the line item is used as value in the template. If the <b>Quantity</b> field for the item has no value, this element will not be printed. This element is used only for the Credit Memo transaction type.  Example: 100  The element value with a decimal separator (.) or a whole number is valid. Group separator or more than one decimal separator is invalid.	This element displays the value of the credited quantity of each item in the credit note line. The value of the <b>Quantity</b> field in the line item is used in the template. This element is not printed if the Quantity field has no value in the item. This element is used only in the Credit Memo transaction type.  Example: 100  The element value with a decimal separator (.) or a whole number is valid. Group separator or more than one decimal separator is invalid.
226	@unit Code		Fixed Value	Both	This element is used with cbc:InvoicedQuantity element, and displays the invoiced quantity for unit of measure. This unit of measure applies to the invoiced	This element is used with cbc:InvoicedQuantity or cbc:CreditedQuantity element and displays the invoiced or credited quantity for the unit of measure. This unit of measure applies to

					<p>quantity. An alphanumeric code consisting of 2 or 3 letters can be used.</p> <p>Example: C62</p> <p>This element has a fixed value "NAR" in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>the invoiced quantity. An alphanumeric code consisting of 2 or 3 letters can be used here.</p> <p>Example: C62</p> <p>This element has a fixed value NAR in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
227	cbc:LineExtensionAmount	item.amount	Amount	Both	<p>This element displays the total amount of the invoice line. The total amount includes the line level allowances, charges, and other relevant taxes but excludes VAT. The value is rounded off to a maximum of 2 decimal places. The amount of each element is used as the value in the template.</p> <p>This element is not printed if the <b>Amount</b> field has no value for an item.</p> <p>Example: 2145.00</p>	<p>This element displays the total amount of the invoice line. The total amount excludes VAT but includes the line level allowances, charges, and other relevant taxes. The value is rounded off to maximum 2 decimals. The amount of each item is used as a value in the <b>Amount</b> field in the template. This element is not displayed if the Amount field has no value in an item.</p> <p>Example: 2145.00</p>
228	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:LineExtensionAmount element to display the ISO code of the transaction's currency.</p>	<p>This element is used with cbc:LineExtensionAmount element to display the ISO code of the transaction's currency.</p>
229	cbc:AccountingCost		Custom Field	Both	<p>This element displays the accounting reference of the invoice line buyer. A text value specifies the location of where to book relevant data in the buyer's financial accounts.</p> <p>Example: 1287:65464</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element displays the accounting reference of the invoice line buyer. A text value specifies the location on where to book relevant data in the buyer's financial accounts.</p> <p>Example: 1287:65464</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
230	cac:InvoicePeriod		Invoice	Invoice	<p>This is parent element displays a group of business terms providing information about the invoice line period. This element has no value in the template.</p>	The template does not support this element.
231	cbc:StartDate		Custom Field	Invoice	<p>This element displays the invoice line period start date. The format of the invoice line start date is YYYY-MM-DD.</p> <p>Example: 2017-10-05</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
232	cbc:EndDate		Custom Field	Invoice	<p>This element displays the invoice line period end date. The format of the invoice line end date is YYYY-MM-DD.</p> <p>Example: 2017-10-15</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
233	cac:OrderLineReference			Both	<p>This parent element displays the reference of the order line. This element has no value, but it has child elements with values associated to it in the template.</p>	The template does not support this element.
234	cbc:LineID		Custom Field	Both	<p>This element displays an object ID of a seller's invoice line.</p> <p>Example: AB12345</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
235	cac:DocumentReference			Both	<p>This parent element displays the details of the line object identifier. This element has no value, but it has child elements with values associated to it in the template.</p>	The template does not support this element.
236	cbc:ID		Custom Field	Both	<p>This element displays the seller's invoice line object identifier.</p>	The template does not support this element.

					Example: AB12345  This element does not have a value in the template.  You can map the value with the required custom field in the SuiteApp.	
237	@sche meID		Custom Value	Both	This element is used with cbc:LineID element to display the ID of the scheme identifier of an invoice line object.  Example: ABZ  This element does not have a value in the template.  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
238	cbc:Do cume ntType Code		Fixed value 130 for Invoice  Fixed Value 130 if Credit Memo is created from Invoice	Both	This element displays the document type code.  Code "130" indicates the reference for an invoice object. This code is not used for other documents. In the Invoice template, the code value is always 130. If a credit memo is created from an invoice, the code is always 130 in the credit memo template.  Default Value: 130  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
239	cac:Al lowan ceCha rge			Both	This parent element displays a group of business terms with information about allowances or charges of an individual invoice. This element has no value, but it has child elements with values associated with it in the template.	The template does not support this element.
240	cbc:Ch argeIn dicator		If line item is of type Discount, display "false"  For other type of items, display "true"	Both	This element displays "true" or "false" value for the item type in a transaction. If the item type is discount, the value is "false". For item types other than discount, the value is "true" in the generated e-document.  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
241	cbc:Al lowan ceCha rgeRe asonC ode		If line item is of type Discount, the value 95 is displayed  For other type of items, display no value	Both	This element displays the line level allowance or charge reason code. If the line item type is Discount, the value of the code is 95; otherwise, the element will not have a value in the template.  Example: 95  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
242	cbc:Al lowan ceCha rgeRe ason		If item type is Discount then display "Discount"  For other type of items, the element has no value	Both	This element displays the line level allowance or charge reason in text. If the line item type is Discount, the value of the field is <b>Discount</b> in template; otherwise, the element is not displayed.  Example: Discount  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
243	cbc:M ultiplie rFacto rNum eric	item.rate	If item type is Discount , the values of Unit Price or Rate are displayed.  For other type of items, the	Both	This element displays the line level allowance or charge percentage depending on the line level allowance base amount. It calculates the line level allowance or charge amount only for discount item type. For discount item type, the unit price or rate value of the item is displayed. If item type is not discount, the element is not available in the generated e-document.	The template does not support this element.

			element has no value. Unit Price (Item) or Rate (Item).	For discount item type, the unit price or rate value of the item is displayed. If unit price or rate is in percentage, the value will be in units.  Example: 20% will be 20  If unit price or rate is not percentage, the value will be in decimals.  Example: 20 will be 20.00  You can map the value with the required custom field in the SuiteApp.	
244	cbc:Amount	item.amount	If item type is Discount the Amount value is displayed.  For other item types, the element has no value. Amount	Both  This element displays the line level allowance or charge amount without VAT. The value of this element is used from the value of <b>Amount</b> field on a transaction's item. The value is rounded off to maximum two decimals.  For Discount item type, the value is displayed in <b>Amount</b> field of a line item. For other item types, this element has no value.  Example: 200  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
245	@curr encyID	transaction.currency.symbol	Transaction's Currency	Both  This element is used with cbc:Amount element to display the ISO code of the transaction's currency.	The template does not support this element.
246	cbc:BaseAmount		Custom Field	Both  This element displays the line level allowance or charge base amount. The base amount is used according to the line level allowance or charge percentage. This is to calculate the line level allowance or charge amount. The value is rounded off to maximum two decimals.  Example: 1000	The template does not support this element.
247	@curr encyID	transaction.currency.symbol	Transaction's Currency	Both  This element is used with cbc:BaseAmount element to display the ISO code of the transaction's currency.	The template does not support this element.
248	cac:Item			Both  This parent element displays a group of business terms with information about invoiced goods and services for all the item types except Description type items. This element has no value, but it has child elements with values associated with it in the template.	This parent element displays a group of business terms with information about invoiced goods and services. The element has no value but has various child elements that have values associated with it in the template. This element is not displayed for the Description item type.
249	cbc:Description	item.description	Description	Both  This element displays the description and features of each item in a transaction. The value of the <b>Description</b> field for items is displayed.  Example: Long description of an item	This element displays the description and features for each item in a transaction. The value of the <b>Description</b> field of items is displayed.  Example: Long description of an item
250	cbc:Name	item.name	Item Name	Both  This element displays the item name entered in the <b>Item Name</b> field in a transaction.  Example: Item name	This element displays the item name.  Example: Item name
251	cac:BuyerIdentification			Both  This parent element displays the buyer's item identification and has no value in the template.	The template does not support this element.
252	cbc:ID		Custom Field	Both  This element displays an identifier assigned by the buyer for an item.  Example: 123455  This element has no value in the template.  You can map the value with the required custom field in the SuiteApp.	The template does not support this element.
253	cac:SellerIdentification			Both  This parent element displays the information about seller's identification. This element has no value, but it has child elements with values associated with it in the template.	This parent element displays the information about seller's identification.  This element has no value in the template.

	ntifica tion					
254	cbc:ID	item.item	Item Name or Item Number	Both	<p>This element displays an identifier assigned by the seller for an item. The element displays the values of the <b>Item Name</b> or <b>Item Number</b> fields or both. The value is displayed only for items without discount and description in the generated e-document.</p> <p>Example: 9873242</p>	<p>This element displays an identifier assigned by the seller for an item that has the value of the <b>Item Name</b> or <b>Number</b> or both the fields. This is only for items that do not have discount and description in the generated e-document.</p> <p>Example: 9873242</p>
255	cac:Standard ItemId identification			Both	<p>This parent element has standard item identification details. This element has no value, but it has child elements with values associated with it in the template.</p> <p>This Parent element is displayed only if the item has UpcCode.</p>	<p>This element has the standard item identification details. The element has no value but has various child elements that have values associated with it in the template. This element is not displayed if the item does not have a UPC code.</p>
256	cbc:ID	item1.itemUpcCode (Refer CDS)	UPC Code	Both	<p>This element displays the standard item identifier based on a registered scheme. It displays the value of the <b>UPC Code</b> field on the Item Record</p> <p>Example: 10986700</p> <p>This element uses custom data source plugin to source the value of an item's UPC code in a template.</p>	<p>This element displays a standard item identifier based on a registered scheme.</p> <p>Example: 10986700</p> <p>This element uses a custom data source plug-in to source the value of an item's <b>UPC code</b> field in a template.</p>
257	@schemeID		Fixed value	Both	<p>This element is used with cbc:ID element, and displays the scheme ID for the item standard identifier. This element has a default value 0160.</p> <p>Example: 0160</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element is used with cbc:ID element and displays the scheme ID for the item standard identifier.</p> <p>Example: 0160</p> <p>This element has a fixed value 0160 in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
258	cac:OriginCountry			Both	<p>This parent element is used to display an item's origin country. This element has no value, but it has child elements with values associated with it in the template.</p> <p>This parent element is displayed only if the item has an origin country.</p>	<p>This parent country is used to display the item's origin country. This is a parent element that has no value. It has child elements with values.</p>
259	cbc:Id entificationCode	item1.itemCountry	Manufacturer Country	Both	<p>This element displays the ISO code of an item's country of origin. This element displays the value of the <b>Manufacturer Country</b> field in the item record. This element is not displayed if the field has no value.</p> <p>Example: CN</p> <p>This element uses custom data source plugin implementation to source the value of an item's manufacturer country.</p>	<p>This element displays the ISO code of an item's origin country. It is not printed if the item has no value in the <b>Manufacturer Country</b> field.</p> <p>Example: CN</p> <p>This element uses custom data source plug-in to source the value of an item's manufacturer country.</p>
260	cac:CommodityClassification			Both	<p>This parent element displays the commodity classification code. This is required when members from same country buy from each other, and the traders need statistical information in the invoice.</p> <p>It is that you use the Item Classification Identifier (BT-158) with the code HS, as an identifier list for this purpose.</p> <p>This element has no value, but has child elements with values associated with it in the template.</p> <p>This element is not printed if there is no value for UPC Code and Manufacturing country in the item.</p>	<p>This parent element displays commodity classification code when members from the same country buy commodities from each other, and the traders need statistical information in the invoice.</p> <p>NetSuite you use the Item Classification Identifier (BT-158) with a code HS as an identifier list for this purpose.</p> <p>The element has no value but has various child elements that have values associated with it in the template. This element is not printed if the <b>UPC Code</b> and <b>Manufacturing Country</b> fields have no values in the item.</p>
261	cbc:ItemClassificationCode	item1.itemUpcCode	UPC Code	Both	<p>This element displays an item classification identifier code to classify items based on its type or nature.</p> <p>Example: 9873242</p>	<p>This element displays an item classification identifier code to classify items based on its type or nature.</p> <p>Example: 9873242</p>

					This element uses custom data source plug-in to source the value of an item's UPC code.	This element uses custom data source plug-in to source the value of an item's UPC code.
262	@listID		Fixed Value	Both	<p>SME: This element is used with <b>cbc:ItemClassificationCode</b> element and displays a scheme ID for item classification identifier. It has a fixed value HS.</p> <p>Example: STI</p> <p>You can map the value with the required custom value in the SuiteApp</p>	<p>This element is used with <b>cbc:ItemClassificationCode</b> element and displays a scheme ID for item classification identifier. It has a fixed value HS in the template.</p> <p>Example: STI</p> <p>You can map the value with the required custom value in the SuiteApp.</p>
263	@listVersionID		Custom Value	Both	<p>This element is used with <b>cbc:ItemClassificationCode</b> element and displays scheme ID for the item classification identifier. This element has no value in the template.</p>	<p>This element is used with <b>cbc:ItemClassificationCode</b> element and displays scheme ID for the item classification identifier.</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom value in the SuiteApp.</p>
264	cac:ClassifiedTaxCategory			Both	<p>This parent element displays a group of business terms with information about the VAT applicable for invoiced goods and services in an invoice line.</p> <p>This element has no value, but it has child elements with values associated with it in the template.</p> <p>SuiteTax accounts can have multiple tax rates to display this element and its child element multiple times, except for the Discount item type.</p>	<p>This parent element displays a group of business terms with information about the VAT applicable for invoiced goods and services in an invoice.</p> <p>This element has no value, but it has various child elements that have values associated with it in the template. The SuiteTax accounts can have multiple tax rates that display this element and its child element multiple times, except for the Discount item type.</p>
265	cbc:ID		Custom Field	Both	<p>This element displays the VAT category code for an invoiced item.</p> <p>Example: S</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	<p>This element displays the VAT category code for an invoiced item.</p> <p>Example: S</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
266	cbc:Percent	For Legacy Tax Account: item.taxrate1  For SuiteTax Account: <ul style="list-style-type: none"> <li>▪ custom.uniqueItems</li> <li>▪ WithMultiple TaxRatesIn STArr (Refer CDS)</li> </ul>	Tax Rate	Both	<p>This element displays the VAT rate percentage applied to the invoiced item. It displays the tax rate value of an item. This element is not displayed if the <b>Tax Rate</b> field in an invoice has no value.</p> <p>Example: 25</p>	<p>This element displays the VAT rate in the percentage applied to the invoiced item. It displays the value of tax rate of the item.</p> <p>Example: 25</p>
267	cac:TaxScheme			Both	<p>This element displays information about tax scheme.</p> <p>This parent element has no value, but it has child elements with values associated with it in the template.</p>	<p>This element displays information about tax scheme.</p> <p>This element has no value, but it has various child elements that have values associated with it in the template.</p>
268	cbc:ID		Fixed Value	Both	<p>This element displays an ID of the tax scheme used in an invoice. A fixed value VAT is added in the template.</p>	<p>This element displays an ID of the tax scheme used in an invoice. A fixed value VAT is added in the template.</p>
269	cac:AdditionalProperty			Both	<p>This parent element displays a group of business terms with information about item attributes of invoiced goods and services.</p> <p>This element has no value, but it has various child elements that have values associated to it in the template.</p>	The template does not support this element.
270	cbc:Name		Custom Field	Both	<p>This element displays the name of the item attribute or the property of the item.</p> <p>Example: Color</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.

271	cbc:Value		Custom Field	Both	<p>This element displays a value for the item attribute.</p> <p>Example: Black</p> <p>This element has no value in the template. You can map the value with the required custom field in the SuiteApp.</p>	The template does not support this element.
272	cac:Price			Both	<p>This parent element displays the price details. It has child elements with information about the price applied for the invoiced goods and services in an invoice.</p> <p>This element has no value, but it has various child elements with values in the template.</p>	<p>This parent element displays the price details and has child elements with information about the price applied for the invoiced goods and services in the invoice.</p> <p>This element has no value, but it has various child elements that have values associated with it in the template.</p>
273	cbc:PriceAmount	Item.rate	Unit Price or rate	Both	<p>This element displays the price of an item excluding VAT and item price discount. The item net price must be equal to the item gross price, which is less than the item price discount if both the prices are provided. The item price cannot be negative.</p> <p>The unit price or rate of an item is used as a value and must be rounded off to 2 decimal places.</p> <p>Example: 10.8955 is rounded to 11</p>	<p>This element displays the price of an item excluding VAT and item price discount. The item net price must be equal to the item gross price, which is less than the item price discount if both prices are provided. The item price cannot be negative. The unit price or rate of an item is used as a value and must be rounded to two decimal points.</p> <p>Example: 10.8955 is rounded to 11</p>
274	@curr encyID	transaction.currency.symbol	Transaction's Currency	Both	This element is used with cbc:PriceAmount to display the ISO code of the transaction's currency.	This element is used with cbc:PriceAmount to display the ISO code of the transaction's currency.
275	cbc:BaseQuantity	item.quantity	Quantity	Both	<p>This element displays quantity of item units to which the price applies. The value is displayed in the item's <b>Quantity</b> field.</p> <p>Example: 1</p> <p>The element value with a decimal separator (.) or a whole number is valid. Group separator or more than one decimal separator is invalid.</p>	The template does not support this element.
276	@unitCode		Custom Value	Both	This element displays the unit of measure code applicable to the item price base quantity. Also, the unit of measure must be same as the unit code of the invoiced or credited quantity. The unit of measure value has a 3-digit alphanumeric code. This element has no value in the template.	The template does not support this element.
277	cac:AllowanceCharge			Both	This element displays allowance details of an invoice. This element has no value, but it has child elements with values in the template.	The template does not support this element.
278	cbc:ChargeIndicator		Fixed Value		This element indicates the charge. A charge on price level is not valid. This element has a fixed value false.	The template does not support this element.
279	cbc:Amount	item.amount	Amount		This element is used to calculate the item price discount. The total discount is subtracted from the gross item price to calculate the net item price. This element displays the value if only if the item type is discount.	The template does not support this element.
280	@curr encyID	transaction.currency.symbol	Transaction's Currency	Both	This element is used with cbc:Amount element to display the ISO code of the transaction's currency.	The template does not support this element.
281	cbc:BaseAmount	item.amount	Item Amount	Both	<p>This parent element displays the gross item price. Item amount is used as value for this element. It is the unit price excluding VAT before subtracting the item price discount.</p> <p>The item price discount value must not be negative.</p> <p>The formula for calculating gross item price is, quantity multiplied by unit, and price divided by rate per Item. Alternatively, the item amount can also be used.</p>	The template does not support this element.

				Example: 123.5	
282	@currEncyID	transaction.currency.symbol	Transaction's Currency	Both	This element is used with cbc:BaseAmount element to display the ISO code of the transaction's currency.  The template does not support this element.

## ANZ PEPPOL Template: PEPPOL Template for Australia and New Zealand

The following table gives an overview of all the elements which are being used in the generic PEPPOL templates for transaction types- invoice, credit memo and vendor bill. These templates are specifically designed for Australia and New Zealand Localization. Some of the elements below are using hard coded values, some are using the values from the transaction and its related records like subsidiary, customer, and some are being derived from Custom Data Source Plug-in.

Users are suggested to use these templates along with the Custom Data Source plug-in to generate the e-document template for invoice and credit memo or converting inbound e-documents to vendor bill correctly. Also, they can add their own customization in the template to get the generated e-document template in their desired form.

The names and the location of the templates, Custom Data Source in Electronic Invoicing SuiteApp is below:

### 1. ANZ Generic Invoice Template

Template Name: ANZ\_Invoice\_Peppol\_Template\_Generic.txt

Template Location: Bundle 436209/Sample Templates

### 2. ANZ Generic Credit Memo Template

Template Name: ANZ\_Creditmemo\_Peppol\_Template\_Generic.txt

Template Location: Bundle 436209/Sample Templates

### 3. ANZ Generic Vendor Bill Template

Template Name: ANZ\_VendorBill\_Inbound\_Template\_Generic.txt

Template Location: Bundle 436209/Sample Templates

### 4. Custom Data Source Plug-in (Invoice and Credit Memo)

File Name: pl\_custom\_data\_source\_anz\_peppol.js

Location: Bundle 436209/src/comp/pl

Plug-in Implementation name: ANZ PEPPOL Custom Data Source Plug-in

### 5. Custom Data Source Plugin (Vendor Bill)

File Name: pl\_custom\_data\_source\_anz\_inbound\_peppol.js

Location: Bundle 436209/src/comp/pl

Plug-in Implementation name: ANZ PEPPOL Inbound CDS

## Details of the Invoice and Credit Memo Outbound Templates



**Note:** To use the above templates, paste the content into "TEMPLATE FOR OUTBOUND E-DOCUMENTS" field of the new E-Document Template creation page.

In the template, while you create, select "ANZ PEPPOL Custom Data Source Plug-in" from the Custom Data Source Plug-in Implementation drop down.

### Limitation:

In the ANZ PEPPOL Template, we currently do not support the population of "ID" under "TaxCategory" and "TaxExemptionReasonCode" values.

The ANZ PEPPOL Template mappings are according to this below table:

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
1	UBL:Invoice			Invoice	This is the root element of the invoice's PEPPOL template. It has a default value.
2	UBL:Credit Note			Credit Memo	This is the root element of the credit memo's PEPPOL template. It has a default value.
3	cbc:CustodianID			Both	This element identifies the specifications of the rules for semantic content, cardinalities, and business to which the data in the instance document conforms to. This element has a default value.
4	cbc:ProfileID			Both	This element identifies the context of the business process related to the transaction and lets the buyer process the invoice accurately. This element has a default value.
5	cbc:ID	transaction.tranid	Invoice Number or Credit Number	Both	This element displays a unique number that identifies if the transaction is an Invoice or Credit Memo. This value is mapped with the entry number of the transaction.
6	cbc:Issue Date	transaction.trandate	Date	Both	This element displays the issue date value present in the <b>Date</b> field of the transaction. The date format is YYYY-MM-DD in a template.
7	cbc:Due Date	transaction.duedate	Due Date	Invoice	This element displays the payment due date value present in the <b>Due Date</b> field of an Invoice. The date format is YYYY-MM-DD in a template.
8	cbc:InvoiceTypeCode		Custom Field	Invoice	<p>This element displays a code specifying the invoice type. This has a default assigned value of 380.</p> <p>You must map the value with the required custom field in the SuiteApp.</p> <p>The following list has information about the Invoice list and the Invoice Type Codes (UNCL1001):</p> <ul style="list-style-type: none"> <li>■ Commercial Invoice – 380</li> <li>■ Consignment Invoice – 395</li> <li>■ Debit Note for Financial Adjustments – 84</li> <li>■ Debit Note for Goods or Services – 80</li> <li>■ Debit Note – 383</li> <li>■ Factored Invoice – 393</li> <li>■ Forwarder's Invoice – 623</li> <li>■ Insurer's Invoice – 575</li> <li>■ Metered Services Invoice – 82</li> <li>■ Prepayment Invoice – 386</li> </ul>
9	cbc:Credit Note Type Code		Custom Field	Credit Memo	<p>This element displays a code specifying the credit memo type. This has a default assigned value of 381.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>The following list has the information about Credit Memo list and the Credit Memo Type Codes (UNCL1001):</p> <ul style="list-style-type: none"> <li>■ Credit Note for Goods or Services – 81</li> <li>■ Credit Note for Financial Adjustments – 83</li> <li>■ Credit Note – 381</li> <li>■ Factored Credit Note – 396</li> <li>■ Forwarder's Credit Note – 532</li> </ul>
10	cbc:Note	transaction.memo	Memo or Custom Field	Both	<p>This element displays a note with information about any change in the invoice or credit memo. The note is retrieved from the value in the <b>Memo</b> field of the transaction.</p> <p>In some SuiteApps, this value can be a combination of the value from the <b>Memo</b> field and the value from the custom field on a subsidiary record.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
11	cbc:Tax Point Date	transaction.taxpointdate	Tax Point Date	Both	This element displays the <b>Tax Point Date</b> field value on the Tax Details tab of a transaction. It is available only in accounts with SuiteTax enabled.

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
					<p>This value represents the date when the GST becomes accountable for the Seller and for the Buyer in so far as that date can be determined and differs from the date of issue of the invoice, according to the GST directive. This element is required if the Value added tax point date is different from the Invoice issue date.</p> <p>The format in which it be displayed is "YYYY-MM-DD".</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
12	cbc:DocumentCurrencyCode	transaction.currency.symbol	Currency (Under Subsidiary Record)	Both	<p>This element displays the ISO Code of the currency used in the transaction, if Multi-Currency feature is enabled in an account.</p> <p>The value is not displayed if the Multi-Currency feature is not enabled in an account.</p>
13	cbc:TaxCurrencyCode	custom.subCurrencyISOCode Refer Custom Data Source (CDS) Plug-in Implementation	Currency — In Subsidiary Record	Both	<p>This element displays the ISO Code used by the subsidiary of the transaction if the currency is not same for transaction and the subsidiary. The value of this element is retrieved from the custom data source plug-in used in the template.</p>
14	cbc:AccountingCost		Custom Field	Both	<p>This element displays the details of the booking data related to the buyer's financial accounts.</p> <p>You must map this field value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
15	cbc:BuyerReference	transaction.otherrefnum	PO #	Both	<p>This element displays the identifier or code assigned by the buyer for internal routing. The <b>PO#</b> field is used to identify the buyer and the order details. The value can be the name of the person ordering the products, employee number, or an identification code for a buyer, department, or group.</p> <p>You can map the value of this field with the required custom field in the SuiteApp.</p>
16	cac:InvoicePeriod			Invoice	<p>This is a parent element containing a group of business terms that provide information about the invoice period, also known as the delivery period. If you use this group, you must also use the invoice period start date, end date, or both.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
17	cbc:StartDate	transaction.startdate	Start Date	Invoice	<p>This element displays the start date of an invoice period for a transaction. This element is displayed along with a value on the generated e-document only if the value is entered in the <b>Start Date</b> field of the transaction.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
18	cbc:EndDate	transaction.enddate	End Date	Invoice	<p>This element displays the end date of an invoice period for a transaction. This element is displayed along with a value on the generated E-Document only if the value is entered in the <b>End Date</b> field of the transaction.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
19	cbc:DescriptionCode		Custom Field	Both	<p>This element displays the code of the date when VAT becomes accountable for seller and buyer.</p> <p>For example:</p> <ul style="list-style-type: none"> <li>■ 3- Invoice document issue date, time</li> <li>■ 35- Delivery date, time</li> <li>■ 432- Paid to date</li> </ul> <p>You can map the value with the required custom field in your account.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
20	cac:OrderReference			Both	<p>This is a parent element with details of the order and sales order references.</p>
21	cbc:ID	transaction.otherrefnum customer.accountnumber customer.entityid	1. PO# If 1. is absent, then Account's value on customer record. If 2. is absent, then Customer	Both	<p>This element displays one the following identifiers for a referenced purchase order allotted by the buyer:</p> <ul style="list-style-type: none"> <li>■ PO#</li> <li>■ Account Number – The account number is displayed if the PO# value is missing.</li> <li>■ Customer ID – The customer ID is displayed if the account number is missing.</li> </ul>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
			ID's value on customer record.		The element represents an identifier for a referenced purchase order issued by the Buyer.
22	cbc:SalesOrderID	transaction.createdfrom	Created From	Invoice	This element displays an identifier for a referenced sales order allotted by the seller. It has the entry number of the sales order transaction only if an invoice is created from a sales order. The entry number is retrieved from the <b>Created From</b> field in the invoice. If the invoice is not created from a sales order, then the element is not printed.
23	cac:BillingReference			Credit Memo	<p>This is a parent element which contains a group of business terms providing information about invoices to which the credit memo is applied to. The element with all its child elements is created separately for each invoice the credit memo is applied to. This does not contain any value but has child elements. In the template, we can find the element only with no value bound to it.</p> <p>If the credit memo is not applied to any invoice, the element with all its child elements will not be displayed.</p>
24	cac:InvoiceDocumentReference			Credit Memo	This parent element has no value in the template but contains child elements identifying the invoice. In this template, element is not provided with any value.
25	cbc:ID	invoice.invoiceId Refer CDS		Credit Memo	<p>This key displays a value identifying an invoice that was previously sent by the seller, to which the credit memo is applied to. The list of invoices the credit memo is applied to is displayed in the Apply subtab and this key refers to the Reference Number for the particular invoice</p> <p>The value of this field is retrieved from the Custom Data Source Plug-in (CDS) used in the template.</p>
26	cbc:IssueDate	invoice.invoiceDate Refer CDS		Credit Memo	<p>This element is used to show the Issue Date for each invoice to which the credit memo is applied to. The list of invoices the credit memo is applied to is displayed in the Apply Subtab and this key refers to the transaction date for the particular invoice. The format in which it be displayed is "YYYY-MM-DD".</p> <p>The value of this field is retrieved from the Custom Data Source Plug-in (CDS) used in the template.</p>
27	cac:DespatchDocumentReference			Both	<p>This parent element provides reference to the dispatch advice. This element has no value, but it has child elements with values associated to it in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
28	cbc:ID		Custom Field	Both	<p>This element refers to an identifier of a referenced dispatch advice.</p> <p>You can map the value of this key with the preferred custom field in your account.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
29	cac:ReceiptDocumentReference			Both	<p>This parent element has reference to the receipt advice. This element has no value, but it has child elements with values associated to it in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
30	cbc:ID		Custom Field	Both	<p>This element displays an identifier of a referenced receiving advice.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
31	cac:OriginatorDocumentReference			Both	<p>This parent element contains a tender or lot reference details. This element has no value, but it has child elements with values associated to it in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
32	cbc:ID		Custom Field	Both	<p>This element displays the call for tender or lot of the invoice ID.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
33	cac:ContractDocumentReference			Both	<p>This parent element provides details of any existing contract associated with a transaction. This element has no value, but it has child elements with values associated to it in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
34	cbc:ID		Custom Field	Both	<p>This element has a value that identifies a contract.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
35	cac:AdditionalDocumentReference			Credit Memo	<p>This parent element has no value, but it has child elements with values associated to it in the template. It has information about group of business terms and details, with additional supporting documents having terms of the invoice. The supporting documents include:</p> <ul style="list-style-type: none"> <li>■ Reference to a document number known to the receiver</li> <li>■ External documents referenced by a URL</li> <li>■ Embedded documents</li> <li>■ Base64 encoded documents like time report</li> </ul> <p>Basically, the project reference details are being used in this parent's child element.</p> <p>By default, the element along with all its child elements will only be displayed when the header project is specified.</p>
36	cbc:ID	transaction.job	Project	Credit Memo	<p>This element displays the identifier of an object based on the invoice provided by the seller or a value used for identifying a supporting document.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>By default, the value for this key is used for the identification of the header project if a header project is specified else element is not displayed.</p>
37	@schemaID		Custom value from list	Credit Memo	<p>This element must have a value displaying the identifier of the identification scheme of an invoiced object. It is used inside the cbc:ID element. The value of this element must be from the <a href="#">list</a>.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
38	cbc:DocumentTypeCode		Custom Field	Credit Memo	<p>This element displays the code value as 50 if there is a project specified in the header of the credit memo.</p> <p>If the credit memo has an applied invoice or it does not display any content if conditions are not met, then the code value is 130 instead of 50.</p> <p>Another reason to use the code 130 is when the credit memo does not display any content if the conditions are not met.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>By default, a value of 50 is provided if a header project is specified else the element is not displayed.</p>
39	cbc:DocumentDescription		Custom Field	Both	<p>This element has a value that describes supporting documents such as time sheets and usage reports.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
40	cac:Attachment			Credit Memo	<p>This parent element has information about attached documents with the transaction. The element has no value, but it has child elements with values associated to it in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
41	cbc:EmbeddedDocumentBinaryObject		Custom Field	Credit Memo	<p>This element has a value referencing any attached document embedded as a binary object (Base64) or sent together with the invoice.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
42	@mime Code		Custom Value	Credit Memo	<p>This is a value used with the element cbc:EmbeddedDocumentBinaryObject that represents the mime code of an attached document. For example: text/csv</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
43	@file Name		Custom Value	Credit Memo	<p>This is a value used with the element cbc:EmbeddedDocumentBinaryObject that represents the file name of an attached document.</p> <p>For example: abc.csv</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE</p>
44	cac:ExternalReference			Both	<p>This is a parent element providing details of the external document's reference such as its location. The element has no value, but it has child elements with values associated to it in the template.</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
					You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE
45	cbc:URI		Custom Field	Both	The value of this element represents the Uniform Resource Locator (URL) that identifies the location of the external document. This URL locates the resource and its primary access mechanism, such as http:// or ftp://.  You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE
46	cac:ProjectReference			Invoice	This parent element has information about any project associated with an invoice. This element has no value, but it has child elements with values associated to it in the template. This element is not displayed if its child element do not have any value.  You can map the value with the required custom field in the SuiteApp.
47	cbc:ID	transaction.job	Project	Invoice	This element indicates the ID of the project associated with an invoice. If the <b>Project</b> field has no value in the transaction, the element is not printed.
48	cac:AccountingSupplierParty			Both	This is a parent element that provides information about the seller using various child elements. This element has no value, but it has child elements with values associated to it in the template.
49	cac:Party			Both	This is a parent element and a child element of cac:AccountingSupplierParty, having information about the seller like electronic address. This element has no value, but it has child elements with values associated to it in the template.
50	cbc:EndpointID	By Default: custom.subTaxRegNo Refer CDS	Custom Field can be used.  OR  By Default:  For Legacy: GST REGISTRATION NO/ ABN (subsidiary record)  For SuiteTax: SUBSIDIARY TAX REG. NUMBER  (Transaction record under taxdetails tab)	Both	This element identifies the seller's electronic address to which the application level response to the invoice can be delivered.  The element uses the custom data source plug-in to get one of the following values:  In OW Account: <ul style="list-style-type: none"><li>■ For Legacy: using subsidiary record's federalidnumber.</li><li>■ For SuiteTax Account: using transaction's subsidiarytaxregnum</li></ul> This element is populated only if value exists.  In SI Account: This element will not be displayed.  You can map the value with the required custom field in the SuiteApp.
51	@schemeID	By Default: custom.schemeID Refer CDS	Custom Value	Both	This element is used with the element cbc:EndpointID and identifies the scheme ID of the seller's electronic address.  You can find the scheme ID value from <a href="#">Scheme ID List</a> .  Possible values for scheme ID in case of ANZ PEPPOL Template are:  New Zealand: 0088  Australia: 0151  The value of this field is retrieved from the Custom Data Source Plug in (CDS) used in the template only for OW Accounts.
52	cac:PartyIdentification			Both	This parent element identifies the seller or the seller's unique banking reference ID provided by the bank. This element has no value, but it has child elements with values associated to it in the template.
53	cbc:ID	By Default: custom.subTaxRegNo Refer CDS	Custom Field can be used.  OR  By Default:  For Legacy: GST REGISTRATION NO/ ABN (subsidiary record)  For SuiteTax: SUBSIDIARY TAX REG. NUMBER	Both	This parent element identifies the seller or the seller's unique banking reference ID provided by the bank. ICD code list is used for the seller's ID. SEPA is the code used for SEPA bank's assigned creditor reference. For the buyer to identify a supplier automatically, the following identifiers are available: <ul style="list-style-type: none"><li>■ BT-29 – Seller identifier</li><li>■ BT-30 – Seller legal registration identifier</li><li>■ BT-31 – Seller VAT identifier</li></ul> The element uses the custom data source plug-in to get one of the following values:  In OW Account: <ul style="list-style-type: none"><li>■ For Legacy: using subsidiary record's federalidnumber.</li><li>■ For SuiteTax Account: using transaction's subsidiarytaxregnum</li></ul>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
			(Transaction record under taxdetails tab)		<p>This element is populated only if value exists.</p> <p>In SI Account: This element will not be displayed.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
54	@schemaID		Custom Value	Both	<p>This element is used with the element cbc:ID to identify the scheme of the seller Identifier.</p> <p>For example, for a bank assigned credit identifier BT-90, the seller ID must be SEPA.</p>
55	cac:PartyName			Both	<p>This parent element displays the seller's information such as name and address. This element has no value, but it has child elements with values associated to it in the template. This element is not displayed if its child elements do not have values.</p>
56	cbc:Name	transaction.subsidiary.name	Addressee (Subsidiary's main address's addressee name)  If 1. is absent, use subsidiary name.	Both	<p>This element displays the business name, another name by which the Seller is known, other than the Seller name. It contains a value in the <b>Addressee</b> field of the Subsidiary's main address.</p> <p>Example: Parent Company, Arizona Company etc</p>
57	cac:PostalAddress			Both	<p>This is a parent element with information about the seller's address. Required fields of the address must be filled out to comply with the legal requirements. This element has no value, but it has child elements that have values associated to it in the template.</p>
58	cbc:StreetName	transaction.subsidiary.address1	Address 1 (Subsidiary's main address's address line 1)	Both	<p>This element displays the main address line of the seller's address. The element has the value of the <b>Address 1</b> field of the subsidiary's main address. This element is displayed only if a value is entered in the <b>Address 1</b> field on the Subsidiary record; otherwise, the element is not displayed.</p>
59	cbc:AdditionalStreetName	transaction.subsidiary.address2	Address 2 (Subsidiary's main address's address line 2)	Both	<p>This element displays another address line with more details about the main address. This element has the value of the <b>Address 2</b> field in the subsidiary's main address. This element is displayed only if a value is entered in the <b>Address 2</b> field on the Subsidiary record; otherwise, the element is not displayed.</p>
60	cbc:CityName	transaction.subsidiary.city	City (Subsidiary's main address's city)	Both	<p>This element displays the common name of the city, town, or village of the seller's location. The element's value is displayed in the <b>City</b> field of the subsidiary's main address.</p> <p>If the field value in subsidiary is blank, the element is not displayed.</p>
61	cbc:PostalZone	transaction.subsidiary.zip	ZIP(Subsidiary's main address's zip)	Both	<p>This element identifies a group of properties with addresses based on the relevant postal service. This element is displayed only if there is a value in the <b>Zip</b> field of the subsidiary's main address.</p> <p>If the field value in subsidiary is blank, the element is not displayed.</p>
62	cbc:CountrySubentity	custom.mainAddressStateRefer CDS	State (Subsidiary's main address's state)	Both	<p>This element identifies the subdivision of the country in the subsidiary's main address. The element is displayed only if there is a value in the <b>State</b> field of the subsidiary's main address.</p> <p>If the field value in subsidiary is blank, the element is not displayed.</p> <p>The state in Address field is displayed with codes for Australian States. The code is converted to the State Name when displayed in the template.</p> <p>The mapping for state code and state name are given below:</p> <ul style="list-style-type: none"> <li>■ "ACT": "Australian Capital Territory", "NSW": "New South Wales"</li> <li>■ "NT": "Northern Territory"</li> <li>■ "QLD": "Queensland"</li> <li>■ "SA": "South Australia"</li> <li>■ "TAS": "Tasmania"</li> <li>■ "VIC": "Victoria"</li> <li>■ "WA": "Western Australia"</li> </ul>
63	cac:AddressLine			Both	<p>This parent element has information about the additional address line in the subsidiary address. This element has no value in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
64	cbc:Line		Custom field	Both	<p>This element has information about an additional address line with more details about the main address line. No value has been provided for the element in this template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
65	cac:Country			Both	This parent element has information about the country associated with the subsidiary's address. This element has no value in the template.
66	cbc:IdentificationCode	Subsidiary country If not, then display the company information country.	custom.subPrimaryCountry (Refer CDS) custom.countryInCompanyInfoCode (Refer CDS)	Both	This parent element displays the ISO Code of the nexus country. It uses the <b>Custom Data Source plug-in</b> implemented in the template to display the values.  In OW account, the subsidiary's country name is populated.  In SI account, the company information country is populated.  If the values are not present, the element is not populated.
67	cac:PartyTaxScheme			Both	This parent element with its child element displays the seller's VAT identifier or tax registration details using different transaction fields. This element has no value in the template, but it has child elements that have values associated with them.
68	cbc:CompanyID	custom.subVatRegNo (Refer CDS)  For SuiteTax: EMPLOYEE ID ENTIFICATION NUMBER (Transaction's subsidiary)	For Legacy Tax: GST REGISTRATION NO. (Transaction's subsidiary)	Both	This element displays the seller's GST identification number or local identification for tax purpose or reference. This enables a seller to indicate their registered tax status.  The element displays the value in the <b>Employee Identification Number</b> field of a subsidiary in a transaction in the generated e-document. The value is displayed using the Custom Data Source plug-in implemented in the template.
69	cac:TaxScheme			Both	This parent element has a child element cac:PartyTaxScheme that displays the seller's GST identification or tax registration details using various transaction fields. This element has no value, but it has child elements that have values associated with it in the template.
70	cbc:ID		Fixed Value: GST	Both	This element is displayed if the seller is a GST identifier or TAX Identifier. The default value is GST.
71	cac:PartyLegalEntity			Both	This parent element has information about the seller's legal details like registration name. This element has no value, but it has child elements that have values associated with it in the template.
72	cbc:RegistrationName	transaction. subsidiary. legalname	Legal Name	Both	This element displays the seller's registered name as per the following conventions: <ul style="list-style-type: none"><li>■ National Registry of Legal Entities</li><li>■ Taxable person</li><li>■ Trading as a person or persons</li></ul> The value of the element is retrieved from the subsidiary's <b>Legal Name</b> field in a transaction.
73	cbc:CompanyID	Custom Field can be used.  OR  By Default:  For Legacy: GST REGISTRATION NO/ ABN (subsidiary record)  For SuiteTax: SUBSIDIARY TAX REG. NUMBER (Transaction record under taxdetails tab)	By Default: custom.subTaxRegNo Refer CDS	Both	This element is used as an identifier allotted by an official registrar. It identifies the seller as legal entity or person.  The value of this field is retrieved from the Custom Data Source Plug-in (CDS) used in the template.  In OW Account:  For Legacy: using subsidiary record's federalidnumber For SuiteTax Account: using transaction's subsidiarytaxregnum  This element is populated only if value exists.  In SI Account:  This element will not be displayed.  You can map the value with the required custom field in the SuiteApp.
74	@schemaID		Custom Value	Both	This element is used with the element cbc:CompanyID to identify the scheme identifier of a seller's legal registration. This element has no value in the template.  You can map the value with the required custom field in the SuiteApp.
75	cbc:CompanyLegalForm			Both	This parent element has details about additional legal information related to the seller. This element has no value in the template.  This element is not supported in the ANZ PEPPOL TEMPLATE
76	cac:Contact			Both	This is a parent element and a child of cbc:CompanyLegalForm with information on a group of business terms providing contact information of the seller. No value has

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
					<p>been provided to this element, but it has child elements which have values associated to them.</p> <p>This element will only be displayed if Sales Rep of the transaction has either phone number or email address.</p>
77	cbc:Name	transaction.salesrep	Sales Rep	Both	This element provides contact details for a legal entity or representative and displays the <b>Sales Rep</b> field value in a transaction. The element is displayed only if there is a value in the <b>Sales Rep</b> field of the transaction.
78	cbc:Telephone	transaction.salesrep.phone	Phone	Both	This element displays the phone number of the contact person. It uses the phone number of the sales representative selected in the transaction as the value. This element is displayed only if there is a value in the sales representative's <b>Phone</b> field in the employee record.
79	cbc:ElectronicMail	transaction.salesrep.email	Email	Both	This element displays the email address of the contact person and uses the email address of the sales representative selected in the transaction as the value. This element is displayed only if there is value in the sales representative's <b>Email</b> field in the employee record.
80	cac:AccountingCustomerParty			Both	This parent element contains a group of business terms, with information about the buyer. This element has no value, but it has child elements that have values associated to it in the template.
81	cac:Party			Both	This is a parent element and a child of cac:AccountingCustomerParty, providing information about the buyer. This element has no value, but it has child elements with values associated in the template.
82	cbc:EndpointID	For Legacy Tax: customer.vatregnumber in customer record  For SuiteTax: customer.defaulttaxreg in customer record	Custom Field or  In Legacy Tax Account: <b>Tax Reg Number</b> field in the customer record  In SuiteTax account: <b>Default Tax Reg</b> field in the customer record	Both	<p>This element displays a value identifying a buyer's electronic address to deliver the invoice.</p> <p>For Legacy: using customer record's vatregnumber</p> <p>For SuiteTax Account: using customer record's defaulttaxreg</p> <p>Example: 987654321</p>
83	@schemaID		Custom Value	Both	<p>This element is used with the element cbc:EndpointID and has no value in the template. This element identifies the scheme identifier of a buyer's electronic address.</p> <p>Example: 0192</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
84	cac:PartyIdentification			Both	This is a parent element providing information about the buyer identifier. This element has no value, but it has child elements that have values associated with them.
85	cbc:ID	customer.accountnumber	Account (Customer's transaction)	Both	This element populates the customer identifier. The value of <b>Account</b> field on the transaction's customer record is used as value for this element.
86	@schemaID		Custom Value	Both	<p>This element is used with the element cbc:ID to display or populate the scheme ID of the buyer identifier.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
87	cac:PartyName			Both	This parent element displays the buyer's details. This element has no value, but it has child elements that have values associated with it in the template.
88	cbc:Name	customer.companyname	Company Name	Both	This element displays another name that the buyer is known, other than the Business name. The value in the <b>Company Name</b> field on a customer's transaction is displayed in the generated e-document for this element.
89	cac:PostalAddress			Both	This parent element has a group of business terms providing information about the buyer's postal address. This element has no value, but it has child elements that have values associated with it in the template.
90	cbc:StreetName	transaction.billaddr1	Address 1 (Transaction's billing address)	Both	This element displays the main address line of an address. The element retrieves the value of the <b>Address 1</b> field from the billing address in a transaction. This element is displayed only if the <b>Address 1</b> field has a value.
91	cbc:AdditionalStreetName	transaction.billaddr2	Address 2 (Transaction's billing address)	Both	This element displays an additional address line to have more details on the main address. This element retrieves the value of the <b>Address2</b> field from the billing address in a transaction.

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
92	cbc:City Name	transaction.billcity	City (Transaction's billing address)	Both	This element displays the common name of the city, town, or village of the buyer's location. This element is displayed only if the <b>City</b> field of the transaction's billing address has a value.
93	cbc: Postal Zone	transaction.billzip	Zip (Transaction's billing address)	Both	This element identifies a group of properties with address based on the relevant postal service. <b>Zip</b> field of the transaction's billing address is used as value for this element. This element is not displayed if this field has no value.
94	cbc:CountrySubentity	transaction.billstate	State (Transaction's billing address)	Both	This element identifies the subdivision of the country. This element is displayed only if the <b>State</b> field of the transaction's billing address has a value.
95	cac:AddressLine			Both	This parent element shows information about the additional address line in a transaction's billing address. This element has no value. This element is not supported in the ANZ PEPPOL TEMPLATE.
96	cbc:Line	transaction.billaddr3	Address 3 (Transaction's billing address)	Both	This element displays an additional address line to have more details on the main address line. This element does not contain any value. This element is not supported in the ANZ PEPPOL TEMPLATE.
97	cac:Country				This parent element provides information about the billing address's country in a transaction. This element has no value, but it has child elements that have values associated with it in the template.
98	cbc:IdentificationCode	custom.billCountryISOCode	Country (Transaction's billing address)	Both	This element displays a code identifying the billing country of a transaction. The element displays the ISO code of the country. The value of this element is displayed using the Custom Data Source plug-in implemented in the template.
99	cac:PartyTaxScheme				This parent element has information about the party GST identifier. This element has no value, but it has child elements that have values associated to it in the template.
100	cbc:CompanyID	For Legacy Tax: customer.vatregnumber  For SuiteTax: customer.defaulttaxreg	Custom Field  By default: In Legacy Tax Account: Tax Reg Number  In SuiteTax account: Default Tax Reg	Both	This element displays the buyer's GST identification number. The field value of transaction's Customer Tax Reg. Number in Suite Tax is used as value for this element. For Legacy Tax, you can use the <b>Tax Reg Number</b> field in a customer record as value for this element. You can map the value with the required custom field in the SuiteApp.
101	cac:TaxScheme			Both	This parent element provides information about the tax scheme used by the customer. This element has no value, but it has child elements that have values associated to it in the template.
102	cbc:ID		Custom Field	Both	This element provides the code of the tax scheme being used. This element has a fixed value GST.
103	cac:PartyLegalEntity			Both	This parent element provides legal information about the buyer. This element has no value, but it has child elements that have values associated to it in the template.
104	cbc:RegistrationName	transaction.billaddress  customer.companynamne	Addressee  Company Name	Both	This element displays the buyer's full name. If the <b>Addressee</b> field has a value in the Billing Address of a transaction, then the full name is displayed along with the element. If not, then the element displays the customer's company name.
105	cbc:CompanyID	For Legacy Tax: customer.vatregnumber  For SuiteTax: customer.defaulttaxreg	Custom Field  By default: In Legacy Tax Account: Tax Reg Number  In SuiteTax account: Default Tax Reg	Both	This element displays an identifier of the buyer as a legal entity or person. This identifier is issued by an official registrar. This element has no value in the template. For Legacy: Using customer record's vatregnumber For SuiteTax Account: using customer record's defaulttaxreg You can map the value with the required custom field in the SuiteApp.
106	@schemeid		Custom Value	Both	This element displays the scheme ID of the buyer's legal registration identifier. This element has no value associated with it in the template.

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
					You can map the value with the required custom field in the SuiteApp.
107	cac:Contact				This is parent element has multiple child elements and provides contact information relevant to the buyer. This element has no value associated with it in the template.
108	cbc:Name	custom.primary Contact Refer CDS	Customer Primary Contact NAME (customer record)	Both	<p>This element has the point of contact of the buyer's legal entity or person.</p> <p>The value of this field is retrieved from the Custom Data Source Plug-in (CDS) used in the template.</p> <p>Use Primary Contact Name of Customer Record if present, else populate the value with UNDEFINED.</p>
109	cbc:Telephone	custom.primary Contact Refer CDS	Customer Primary Contact PHONE (customer record)	Both	<p>This element displays the phone number of the buyer's contact. This element is displayed in the generated e-document only if the customer's <b>Phone Number</b> field has a value.</p> <p>The value of this field is retrieved from the Custom Data Source Plug-in (CDS) used in the template.</p>
110	cbc:ElectronicMail	custom.primary Contact Refer CDS	Customer Primary Contact EMAIL (customer record)	Both	<p>This element displays the e-mail address of the buyer's contact. This element is displayed in the generated e-document only if the customer's <b>E-mail Address</b> field has a value.</p> <p>The value of this field is retrieved from the Custom Data Source Plug-in (CDS) used in the template.</p>
111	cac:Payee Party			Both	<p>This parent element has multiple child elements, with information about the payee. This element is used when the payee is different from the seller. This element has no value in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
112	cac:PartyIdentification			Both	<p>This is a parent element and a child of cac:PayeeParty, which has child elements used for identifying the payee party. This element has no value in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
113	cbc:ID		Custom Field	Both	<p>This element is used to identify both the payee and the unique banking reference identifier of the payee assigned by the payee's bank. For identifying the payee, you can use the ICD code list. You can use SEPA code for identifying the SEPA bank assigned creditor reference. For example, FR932874294.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
114	@schemaID		Custom Value	Both	<p>This element used with the cbc:ID element identifies the payee's scheme ID. This has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
115	cac:PartyName				<p>This is a parent element and a child of cac:PayeeParty element. This element provides information about the payee name. This element has no value in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
116	cbc:Name		Custom Field	Both	<p>This element displays the name of the payee. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
117	cac:PartyLegalEntity			Both	<p>This parent element has legal information about the payee. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
118	cbc:CompanyID		Custom Field	Both	<p>This element identifies the payee's legal registration. The identifier is issued by an official registrar and identifies the payee as a legal entity or person. For example, FR932874294.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
119	@schemaID		Custom Value	Both	<p>This element is used with cbc:CompanyID and has the scheme ID of the payee's legal registration identifier.</p> <p>For example: 0002</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
					You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE.
120	cac:TaxRepresentativeParty			Both	This parent element has information about the seller's tax representative. This element has no value, but it has child elements that have values associated to it in the template. This element is not supported in the ANZ PEPPOL TEMPLATE.
121	cac:PartyName			Both	This is a parent element and a child element of cac:TaxRepresentativeParty. This element displays the name of the seller's tax representative. This element has no value, but it has child elements that have values associated to it in the template. This element is not supported in the ANZ PEPPOL TEMPLATE.
122	cbc:Name		Custom Field	Both	This element displays the full name of the seller's tax representative. This element has no value in the template. You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE.
123	cac:PostalAddress			Both	This is a parent element and a child element of cac:TaxRepresentativeParty. The element has no value, but it has child elements with information about the postal address of the tax representative in template. The required address fields must be filled out to comply with legal requirements. This element is not supported in the ANZ PEPPOL TEMPLATE.
124	cbc:StreetName		Custom Field	Both	This element displays the main address line of the seller's tax representative. This element has no value in the template. You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE.
125	cbc:AdditionalStreetName		Custom Field	Both	This element displays the value of an additional address line of the main address of a seller's tax representative. This element has no value in the template. You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE.
126	cbc:CityName		Custom Field	Both	This element displays the common name of the city, town, or village of the tax representative's location. This element has no value in the template. You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE.
127	cbc:PostalZone		Custom Field	Both	This element identifies an addressable group of properties according to the relevant postal service of the seller's tax representative. This element has no value in the template. You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE.
128	cbc:CountrySubentity		Custom Field	Both	This element has information about a country's subdivision of the seller's tax representative. This element has no value in the template. You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE.
129	cac:AddressLine			Both	This element is a parent element and a child of cac:PostalAddress, with information about the additional address line of the seller's tax representative main address. This element has clarifications on the addresses. It has no value in the template. You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE.
130	cbc:Line		Custom Field	Both	This element has information about the additional address line of a main address. It gives information about the seller's tax representative's main address. This element has no value in the template. You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE.
131	cac:Country			Both	This element is a parent element and child of cac:PostalAddress, with information about the country of the address. This element has no value, but it has child elements that have values associated to it in the template. This element is not supported in the ANZ PEPPOL TEMPLATE.

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
132	cbc:IdentificationCode		Custom Field	Both	<p>This element displays the ISO code of the seller's tax representative country. This element has no a value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
133	cac:PartyTaxScheme			Both	<p>This is a parent element with information about the GST party identifier. This element has no value, but it has child elements with values associated with it in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
134	cbc:CompanyID		Custom Field	Both	<p>This element represents the GST identifier of the seller's tax representative. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
135	cac:TaxScheme			Both	<p>This parent element displays the tax scheme used by the tax representative. This element has no value in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
136	cbc:ID		Custom Field	Both	<p>This element displays the name of the tax scheme and has a required value GST, added in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
137	cac:Delivery				<p>This parent element has information about the delivery time and location of the invoiced goods and services. This element has no value in the template, but it has child elements with information about the delivery address in the template.</p>
138	cbc:ActualDeliveryDate	transaction.trandate	Custom Field OR By Default: DATE	Both	<p>This element displays the date when the supply of goods and services is completed. This element is displayed along with the value on the generated e-document only if the value is provided in the <b>Date</b> field of the transaction. This date must be in YYYY-MM-DD format. It is populated with transaction date.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
139	cac:DeliveryLocation			Both	<p>This element has information about the location where goods and services were delivered. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
140	cbc:ID		Custom Field	Both	<p>This element displays the identifier for the delivery location of the goods and services. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
141	@schemeID		Custom Value	Both	<p>This element is a part of cbc:ID element and displays the scheme ID of the <b>Deliver to Location</b> field. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
142	cac:Address			Both	<p>This parent element has information about the address where invoiced goods or services are delivered. This element has no value in the template, but it has child elements that have values associated with it.</p>
143	cbc:StreetName	transaction.shipaddr1	Address 1 (Transaction's address line 1)	Both	<p>This element displays the main address line in the transaction's shipping address. This element has the value of <b>Address 1</b> field in the transaction's shipping address. This element is not displayed if the <b>Address 1</b> field has no value in the transaction.</p>
144	cbc:AdditionalStreetName	transaction.shipaddr2	Address 2 (Transaction's address line 2)	Both	<p>This element displays the main address line in the transaction's shipping address. This element has the value of the <b>Address 2</b> field in the transaction's shipping address. The element and its value are not displayed if the <b>Address 2</b> field has no value.</p>
145	cbc:CityName	transaction.shipcity	City (Transaction's shipping address's city)	Both	<p>This element displays the common name of the city, town, or village of the delivery location in the shipping address. This element has the value of the <b>City</b> field in the transaction's shipping address. The element and its value are not displayed if the <b>City</b> field has no value.</p>
146	cbc:PostalZone	transaction.shipzip	Zip (Transaction's shipping address's zip)	Both	<p>This element displays an identifier for a group of properties based on their postal service. This element has the value of the <b>Zip</b> field in the transaction's shipping address. The element and its value are not displayed if the <b>Zip</b> field has no value.</p>
147	cbc:CountrySubentity	transaction.shipstate	State (Transaction's shipping address's state)	Both	<p>This element displays the subdivision of a country in the shipping address. This element has the value of the <b>State</b> field in the transaction's shipping address. The element and its value are not displayed if the <b>State</b> field has no value.</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
148	cac:AddressLine			Both	<p>This parent element displays additional information about the shipping address. This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
149	cbc:Line		Custom Field	Both	<p>This is a parent element and a child of cac:AddressLine, with information about the additional line of an address, supporting the primary address. This element has no value in the template. You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
150	cac:Country			Both	<p>This parent element displays the country of the shipping address. This element has no value, but it has child elements with values in the template.</p>
151	cbc:IdentificationCode	transaction.shippingcountry	Country (Transaction's shipping addressee's country)	Both	<p>This element displays the ISO code of the country in the transaction's shipping address. It displays the value of the <b>Country</b> field in the template.</p>
152	cac:DeliveryParty			Both	<p>This parent element displays the details of the delivery party. This element has no value but has a child element with values associated to it in the template.</p>
153	cac:PartyName			Both	<p>This is a parent element and a child of cac:DeliveryParty, with information about the delivery party's name. This element has no value but has a child elements with values associated to it in the template.</p>
154	cbc:Name	transaction.shippingaddressee	Addressee (Transaction's shipping address's country)	Both	<p>This element displays the name of the customer or party whom the goods and services are to be delivered. This element displays the value of the <b>Addressee</b> field in the transaction's shipping address.</p>
155	cac:PaymentMeans			Invoice	<p>This parent element provides a group of business terms by using different child elements to give payment information. This element has no value, but it has various child elements that have values associated with it in the template. This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
156	cbc:PaymentMeansCode		Custom Field	Invoice	<p>This element displays the payment type code for a payment made or to be made. This element has no value in the template. Example: 30 The following list has the details of the Payment Types and its associated code:</p> <ul style="list-style-type: none"> <li>■ Credit Transfer – 30</li> <li>■ Direct Debit – 49</li> <li>■ Opostgiro – 50</li> <li>■ Credit Card – 54</li> <li>■ Debit Card – 55</li> <li>■ Bankgiro – 56</li> <li>■ Sepa Credit Transfer – 58</li> <li>■ Sepa Direct Debit – 59</li> <li>■ Online Payment Service – 68</li> <li>■ Finland – 93, 94, 95</li> <li>■ Instrument not defined – 1</li> <li>■ National or Regional Clearing – 9</li> <li>■ Cash – 10</li> </ul> <p>By Default: ZZZ is populated in the Invoice Template You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
157	@name		Custom Value	Invoice	<p>This element displays cbc:PaymentMeansCode , the name of the mode of payment. This element has no value in the template. Example: Credit Transfer By Default: NETSUITE is populated in Invoice Template You can map the value with the required custom field in the SuiteApp. This element is not supported in the ANZ PEPPOL TEMPLATE.</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
158	cbc:PaymentID		Custom Field	Invoice	<p>This element displays a text value for associating the payment and the invoice issued by the seller. This is used for creditors' critical reconciliation information. The value helps the seller assign an incoming payment to the relevant payment process.</p> <p>This element has no value in the template. A custom field with a payment reference can be used for this element.</p> <p>Example value: 432948234234234</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
159	cac:Card Account			Credit Memo	<p>This parent element represents a group of business terms with information about the card used to make a payment against an issued invoice.</p> <p>If a buyer made a payment using a card such as a credit or debit card, then the information about the Primary Account Number (PAN) is mentioned in the invoice.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
160	cbc:PrimaryAccountNumberID		Custom Field	Credit Memo	<p>This element is used to represent the Payment card primary account number. The Primary Account Number (PAN) of the card used for payment .In accordance with card payments security standards, an invoice should never include a full card primary account number.</p> <p>Example: 1234</p> <p>The last 4 to 6 digits of the PAN (BT-87) are indicated if payment card information (BG-18) is provided in the invoice.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
161	cbc:NetworkID		Custom field	Credit Memo	<p>This element displays the card network identifier such as VISA, American Express, Card.</p> <p>Example: VISA</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
162	cbc:HolderName		Custom Field	Credit Memo	<p>This element displays the name of the holder of the card.</p> <p>Example: John Doe</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
163	cac:PayeeFinancialAccount			Invoice	<p>This parent element displays a group of business terms specifying the credit transfer payments.</p> <p>This element has no value, but it has various child elements with values associated to it in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
164	cbc:ID		Custom Field	Invoice	<p>This element identifies a payment account.</p> <p>It displays a unique ID of the financial payment account for a payment service provider to which the payment is made, such as IBAN or BBAN.</p> <p>Example: NO99991122222</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
165	cbc:Name		Custom Field	Invoice	<p>This element displays the payment account name of the service provider to which the payment is made.</p> <p>Example: Payment Account</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
166	cac:FinancialInst			Both	<p>This parent element displays the details of a payment service provider. This element has no value, but it has various child elements with values associated to it in the template.</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
	itutionBranch				This element is not supported in the ANZ PEPPOL TEMPLATE.
167	cbc:ID		Custom Field	Both	<p>This element displays an ID for payment service provider where a payment account is located such as BIC or a national clearing code. Identification scheme identifier is not used in this element.</p> <p>Example: 9999</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
168	cac:PaymentMandate			Both	<p>This parent element displays a group of business terms for direct debit transactions. This element has no value, but it has various child elements with values associated to it in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
169	cbc:ID		Custom Field	Both	<p>This element displays a unique identifier assigned by the payee for referencing a direct debit mandate. This is used to notify the buyer of a SEPA direct debit in advance.</p> <p>Example: 123456</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
170	cac:PayerFinancialAccount			Both	<p>This parent element has no value, but it has various child elements with values associated to it in the template. It has information about the payer's financial account.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
171	cbc:ID		Custom Field	Both	<p>This element displays an ID with information about the account to be debited by direct debit.</p> <p>Example: 12345676543</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
172	cac:PaymentTerms			Invoice	<p>This parent element has information about the payment terms applied to the due amount. This parent element has no value, but it has a child element with value associated with it in the template.</p>
173	cbc:Note	transaction.terms	Terms	Invoice	<p>This parent element displays text describing the payment terms that apply to the amount due for payments. It can also describe applicable penalties.</p> <p>If an amount is due for payment (BT-115), then the payment due date (BT-9) or the payment terms (BT-20) is indicated. This element displays a value in the <b>Terms</b> field on a transaction.</p> <p>Example: Net within 30 days</p>
174	cac:AllowanceCharge			Both	<p>This parent element displays a group of business terms with information about applicable allowances of a complete invoice. This element also displays a group of business terms with information about charges and taxes excluding GST for a complete invoice.</p> <p>This element has no value and is displayed in the generated e-document only if there is a Header Discount or Shipping Cost associated with the transaction.</p>
175	cbc:ChargeIndicator		Custom Field	Both	<p>This element displays the charge type used for a transaction. The generated e-document displays the value true when informing about charges and the value false when informing about allowances.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>■ True for shipping and handling.</li> <li>■ False for header discounts.</li> </ul> <p>You can map the value with the required custom field in the SuiteApp.</p>
176	cbc:AllowanceChargeReasonCode		Custom Field	Both	<p>This element displays the document level allowance code or the allowance charge reason code. You can use the UNCL5189 code from the code list for allowance of a subset. For charges, you can use the UNCL7161 code from the code list.</p> <p>The document level allowance reason code and document level allowance reason display the same reason.</p> <p>Example: 95 for Discount SAA for Shipping and Handling</p> <p>This element does not have a value in the template.</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
					You can map the value with the required custom field in the SuiteApp.
177	cbc:AllowanceChargeReason		Custom Field	Both	<p>This element displays document level allowance reason or charge reason in text format. The document level allowance reason code and the document level allowance reason, display the same allowance reason.</p> <p>This element displays the following values:</p> <ul style="list-style-type: none"> <li>■ Shipping and handling for charges</li> <li>■ Discount for allowances</li> </ul> <p>You can map the value with the required custom field in the SuiteApp.</p>
178	cbc:MultiplierFactorNumeric		Custom Field	Both	<p>This element displays the document level allowance or charge percentage.</p> <p>The percentage is used with the document level allowance base amount to calculate the document level or charge amount. the value must be 20.</p> <p>Example: 20</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
179	cbc:Amount	For Discount: transaction.discounttotal  For Shipping and Handling: transaction.altshippingcost+transaction.althandlingcost	For Discount: DISCOUNT ITEM (Summary section of transaction)  For Shipping and Handling:  Transaction's Shipping + handling cost (transaction record)	Both	<p>This element displays the document level allowance or charge amount without GST. This value must be rounded off to a maximum of 2 decimal.</p> <p>value of Discount Item field for allowances.</p> <p>Example: 200</p>
180	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	This element is used with the cbc:Amount element, to display the ISO code of a transaction's currency.
181	cbc:BaseAmount	transaction.subtotal	Subtotal	Both	<p>This element displays the base amount along with the document level allowance or charge percentage. It calculates the document level allowance or charge amount. The value is rounded off to 2 decimal places in the template.</p> <p>A value is provided in the <b>Subtotal</b> field in the Summary section of a transaction</p> <p>Example value: 200</p>
182	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	This element is used with cbc:BaseAmount element, to display the currency's ISO code used by a transaction.
183	cac:TaxCategory			Both	This parent element displays the tax category and its details. This element has no value, but it has child elements with values associated to it in the template.
184	cbc:ID		Custom Field	Both	<p>This element displays a code to identify the GST category.</p> <p>Example:</p> <p>Value: S</p> <p>In ANZ, the following codes are used:</p> <ul style="list-style-type: none"> <li>■ E - Exempt From Tax – Specifies that taxes are not applicable</li> <li>■ S – Standard Rate – Specifies the standard rate.</li> <li>■ Z – Zero Rated Goods – Specifies that the goods are at zero rate</li> <li>■ G- Free Export Item, TAX Not Charged – Specifies that the item is free to export and taxes are not charged.</li> <li>■ O- Services outside scope of tax</li> </ul> <p>Specifies that the goods and services which do not attract GST where invoices are issued by entities who are not registered and not required to be registered for GST in Australia.</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
185	cbc:Percent		Custom Field	Both	<p>This element displays the document level allowance or GST charge rate. The GST rate is displayed as the percentage that applies to the document level allowance or charge.</p> <p>The categories are Shippingtaxcode, Handlingtaxcode, etc.</p> <p>Example: 25</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
					<p>This element does not have a value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
186	cac:TaxScheme			Both	<p>This parent element displays the details of the tax scheme used in a transaction. This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
187	cbc:ID		Fixed Value	Both	<p>This element displays the name of the tax scheme used in a transaction. A fixed GST value is provided in the template for this element.</p>
188	cac:TaxTotal			Both	<p>This parent element displays the total tax details including the amounts for a transaction. When a tax currency code is provided, two instances of tax total must be present and only one instance with tax subtotal must be used. This element does not have a value in the template.</p> <p>This element has no value in the template.</p>
189	cbc:TaxAmount	transaction.taxtotal	Tax	Both	<p>This element displays the total GST amount for an invoice or the total GST amount in the accounting currency as required by the seller.</p> <p>This value is rounded off to a maximum of 2 decimal places in the template.</p> <p>The elements and its values are displayed only if:</p> <ul style="list-style-type: none"> <li>■ A transaction's tax total is not equal to zero.</li> <li>■ The transaction's currency is not equal to the subsidiary's currency of the transaction.</li> </ul>
190	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:TaxAmount element, to display the ISO code of the currency used by the transaction.</p>
191	cac:TaxSubtotal			Both	<p>This parent element displays a group of business terms with information about GST break down based on different categories, rates and exemption reasons.</p> <p>This element has no value, but it has child elements with values associated with it.</p>
192	cbc:TaxableAmount	taxDetails (Refer CDS)	Sum of amount of all items calculated per unique tax category	Both	<p>This element displays all the taxable amount for each item specific to a unique GST category code.</p> <p>This value must be rounded off to 2 decimal places.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
193	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:TaxableAmount element to display the ISO code of the transaction's currency.</p>
194	cbc:TaxAmount	taxableAmount ((Using CDS))	sum of tax amount of all items calculated per category	Both	<p>This element displays the total GST amount for each item specific to a unique GST category code.</p> <p>The value is rounded off to a maximum of 2 decimal places in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
195	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:TaxAmount to display the ISO code of the transaction's currency.</p>
196	cac:TaxCategory			Both	<p>This parent element displays the GST category details. This element has no value, but it has child elements with values associated with it.</p>
197	cbc:ID		Custom Field	Both	<p>This element displays a code to identify the GST category.</p> <p>Example:</p> <p>Value: S</p> <p>In ANZ, the following codes are used:</p> <ul style="list-style-type: none"> <li>■ E - Exempt From Tax – Specifies that taxes are not applicable</li> <li>■ S - Standard Rate – Specifies the standard rate.</li> <li>■ Z - Zero Rated Goods – Specifies that the goods are at zero rate</li> <li>■ G - Free Export Item, TAX Not Charged – Specifies that the item is free to export and taxes are not charged.</li> <li>■ O- Services outside scope of tax</li> </ul> <p>Specifies that the goods and services which do not attract GST where invoices are issued by entities who are not registered and not required to be registered for GST in Australia.</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
198	cbc:Percent		Custom Field	Both	<p>This element displays the GST category rate for each unique tax code used by items for shipping.</p> <p>The GST rate is displayed as percentage for the applicable VAT category.</p> <p>Example: 25</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
199	cbc:TaxExemptionReasonCode		Custom Field	Both	<p>This element displays a GST exemption reason code. A coded statement indicates the reason for exempting the GST amount.</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
200	cbc:TaxExemptionReason		Custom Field	Both	<p>This element displays the GST exemption reason text.</p> <p>Example: Exempt</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
201	cac:TaxScheme			Both	<p>This parent element displays the tax scheme. This element has no value, but it has child elements with values associated with it in the template.</p>
202	cbc:ID		Fixed Value	Both	<p>This element displays the name of the tax scheme used in a transaction. A fixed value GST is assigned to this element.</p>
203	cac:LegalMonetaryTotal			Both	<p>This parent element displays the document totals. This element has no values, but it has child elements that provide monetary totals for an invoice or a credit memo in a template.</p>
204	cbc:LineExtensionAmount	transaction.subtotal	Subtotal	Both	<p>This element displays the sum of all the invoice line net amounts in an invoice. The value is rounded off to a maximum of 2 decimal places in the template.</p> <p>This element displays the <b>Subtotal</b> field value from the transaction's Summary tab.</p>
205	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:LineExtensionAmount element to display the ISO code of a transaction's currency.</p>
206	cbc:TaxExclusiveAmount	transaction.altshippingscost + transaction.althandlingcost + transaction.subtotal - transaction.discounttotal	Subtotal + Allowances - Discount	Both	<p>This element displays the total amount of an invoice without GST. The value of this element must be rounded off to two decimals. In a template, you must add the transaction's amount in the Subtotal field in the Summary tab with the handling and shipping charges to get a value.</p> <p>Example: 3600.0</p> <p>You can add extra allowances and change the template to accommodate this element.</p>
207	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:TaxExclusiveAmount to display the ISO code of the transaction's currency.</p>
208	cbc:TaxInclusiveAmount	transaction.total	Total	Both	<p>This element displays the total amount of a transaction with GST. This value is rounded off to 2 decimal places in the template.</p>
209	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:TaxInclusiveAmount element to display the ISO code of the transaction's currency</p>
210	cbc:AllowanceTotalAmount	transaction.discounttotal	Discount Item	Both	<p>This element displays the sum of allowances at the document level. The value of this element is rounded off to 2 decimal places.</p> <p>Example: 200.0</p>
211	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:AllowanceTotalAmount element to display the ISO code for the transaction's currency.</p>
212	cbc:ChargeTotalAmount	transaction.altshippingscost+ transaction.althandlingcost	Shipping Cost+ Handling Cost	Both	<p>This element displays the shipping cost charge at the document level in the transaction. The value is rounded off to 2 decimal places in the template. It displays total cost of shipping and handling.</p> <p>Example: 10.0</p>
213	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>The element is used with cbc:ChargeTotalAmount to display the ISO code of the transaction's currency.</p>
214	cbc:PrepaidAmount	transaction.amountpaid	Payments	Both	<p>This element displays the sum of the amount paid in advance, for a transaction. The value is rounded off to a maximum 2 decimal places in the template.</p> <p>This value displays the sum of all the payments made for a transaction.</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
					Example: 1000.0
215	@currentcyID	transaction.currency.symbol	Transaction's Currency	Both	This element is used with cbc:PrepaidAmount element to display the ISO code of the transaction's currency.
216	cbc:PayableRoundingAmount		Custom Field	Both	<p>This element displays the rounding amount. The amount is added to the invoice total to round off the amount to be paid. The value is rounded off by maximum two decimals in template.</p> <p>Example: 0.0</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
217	@currentcyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:PayableRoundingAmount element to display the ISO code of the transaction's currency.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
218	cbc:PayableAmount	For Invoice: transaction.amountremainingtot albox  For Credit Memo: transaction.amountremaining	Amount Due	Both	<p>This element displays the amount due for payment and the outstanding amount that is requested to be paid. The value is rounded off to a maximum 2 decimal places in the template. The <b>Amount Due</b> field value in the Summary tab is used for the transaction.</p> <p>Example: 3500.0</p>
219	@currentcyID	transaction.currency.symbol	Transaction's Currency	Both	This element is used with cbc:PayableAmount element to display the ISO code of the transaction's currency.
220	cac:InvoiceLine			Invoice	This parent element displays the line item components. It also has a group of business terms providing information about individual transaction lines. This element has no value, but it has child elements with values in the template.
221	cac:CreditNoteLine			Credit Memo	This parent element displays the line item components on a Credit Memo. It also has a group of business terms providing information about individual invoice lines for the credit memo. This element has no value, but it has child elements with values associated to it in the template. This element is used only for the Credit Memo transaction type.
222	cbc:ID	item.line	Line Sequence Number	Both	<p>This element displays the transaction line ID for each item. It has a unique identifier for an individual line in the transaction. The line sequence number of a line item is used in the template.</p> <p>Example: 12</p>
223	cbc:Note	transaction.memo	Memo	Both	<p>This element is used in the invoice and credit memo line note. It is a text note with unstructured information relevant to the invoice line. The header level <b>Memo</b> field is used as value in the template.</p> <p>Example: New article number 12345</p>
224	cbc:InvoicedQuantity	item.quantity	Quantity	Both	<p>This element displays the value of invoiced quantity of each item in the transaction line. The <b>Quantity</b> field on the line item is used as value in the template.</p> <p>Example: 100</p> <p>The element value with a decimal separator (.) or a whole number is valid. Group separator or more than one decimal separator is invalid.</p>
225	cbc:CreditedQuantity	item.quantity	Quantity	Credit Memo	<p>This element displays the value of invoiced quantity of each item in the credit note line. If the <b>Quantity</b> field for the item has no value, this element will not be printed. This element is used only for the Credit Memo transaction type.</p> <p>Example: 100</p> <p>The element value with a decimal separator (.) or a whole number is valid. Group separator or more than one decimal separator is invalid.</p>
226	@unitCode		Custom Value	Both	<p>This element is used with cbc:InvoicedQuantity element, and displays the invoiced quantity for unit of measure. This unit of measure applies to the invoiced quantity. An alphanumeric code consisting of 2 or 3 letters can be used.</p> <p>Codes for unit of packaging from UNECE Recommendation No. 21 can be used in accordance with the descriptions in the "Intro" section of UN/ECE Recommendation 20, Revision 11 (2015). The 2 character alphanumeric code values in UNECE Recommendation 21 shall be used. To avoid duplication with existing code values in UNECE Recommendation No. 20, each code value from UNECE Recommendation 21 shall be prefixed with an "X", resulting in a 3 alphanumeric code when used as a unit of measure.</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
					<p>Example: C62</p> <p>This element has a fixed value "ZZ" in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
227	cbc:LineExtensionAmount	item.amount	Amount	Both	<p>This element displays the total amount of the invoice line. The total amount includes the line level allowances, charges, and other relevant taxes but excludes VAT. The value is rounded off to a maximum of 2 decimal places. The amount of each element is used as the value in the template.</p> <p>This element is not printed if the <b>Amount</b> field has no value for an item.</p> <p>Example: 2145.00</p>
228	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:LineExtensionAmount element to display the ISO code of the transaction's currency.</p>
229	cbc:AccountingCost		Custom Field	Both	<p>This element displays the accounting reference of the invoice line buyer. A text value specifies the location of where to book relevant data in the buyer's financial accounts.</p> <p>Example: 1287:65464</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
230	cac:InvoicePeriod			Invoice	<p>This is parent element displays a group of business terms providing information about the invoice line period. This element has no value in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
231	cbc:StartDate		Custom Field	Invoice	<p>This element displays the invoice line period start date. The format of the invoice line start date is YYYY-MM-DD.</p> <p>Example: 2017-10-05</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
232	cbc:EndDate		Custom Field	Invoice	<p>This element displays the invoice line period end date. The format of the invoice line end date is YYYY-MM-DD.</p> <p>Example: 2017-10-15</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
233	cac:OrderLineReference			Both	<p>This parent element displays the reference of the order line. This element has no value, but it has child elements with values associated to it in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
234	cbc:LineID		Custom Field	Both	<p>This element displays an object ID of a seller's invoice line.</p> <p>Example: AB12345</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
235	cac:DocumentReference			Both	<p>This parent element displays the details of the line object identifier. This element has no value, but it has child elements with values associated to it in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
236	cbc:ID		Custom Field	Both	<p>This element displays the seller's invoice line object identifier.</p> <p>Example: AB12345</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
237	@schemeID		Custom Value	Both	<p>This element is used with cbc:LineID element to display the ID of the scheme identifier of an invoice line object.</p> <p>Example: ABZ</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
238	cbc:DocumentTypeCode		Fixed value 130 for Invoice	Both	<p>This element displays the document type code.</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
			Fixed Value 130 if Credit Memo is created from Invoice		<p>Code "130" indicates the reference for an invoice object. This code is not used for other documents. In the Invoice template, the code value is always 130. If a credit memo is created from an invoice, the code is always 130 in the credit memo template.</p> <p>Default Value: 130</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
239	cac:AllowanceCharge			Both	<p>This parent element displays a group of business terms with information about allowances or charges of an individual invoice. This element has no value, but it has child elements with values associated with it in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
240	cbc:ChargeIndicator		If line item is of type Discount, display "false" For other type of items, display "true"	Both	<p>This element displays "true" or "false" value for the item type in a transaction. If the item type is discount, the value is "false". For item types other than discount, the value is "true" in the generated e-document.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
241	cbc:AllowanceChargeReasonCode		If line item is of type Discount, the value 95 is displayed For other type of items, display no value	Both	<p>This element displays the line level allowance or charge reason code. If the line item type is Discount, the value of the code is 95; otherwise, the element will not have a value in the template.</p> <p>Example: 95</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
242	cbc:AllowanceChargeReason		If item type is Discount, then display "Discount" For other type of items, the element has no value	Both	<p>This element displays the line level allowance or charge reason in text. If the line item type is Discount, the value of the field is <b>Discount</b> in template; otherwise, the element is not displayed.</p> <p>Example: Discount</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
243	cbc:MultiplierFactorNumeric	item.rate	If item type is Discount, the values of Unit Price or Rate are displayed. For other type of items, the element has no value - Unit Price (Item) or Rate (Item).	Both	<p>This element displays the line level allowance or charge percentage depending on the line level allowance base amount. It calculates the line level allowance or charge amount only for discount item type. For discount item type, the unit price or rate value of the item is displayed. If item type is not discount, the element is not available in the generated e-document.</p> <p>For discount item type, the unit price or rate value of the item is displayed. If unit price or rate is in percentage, the value will be in units.</p> <p>Example: 20% will be 20</p> <p>If unit price or rate is not percentage, the value will be in decimals.</p> <p>Example: 20 will be 20.00</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
244	cbc:Amount	item.amount	If item type is Discount, the Amount value is displayed. For other item types, the element has no value. Amount	Both	<p>This element displays the line level allowance or charge amount without GST. The value is rounded off to maximum two decimals.</p> <p>Example: 200</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
245	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:Amount element to display the ISO code of the transaction's currency.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
246	cbc:BaseAmount		Custom Field	Both	<p>This element displays the line level allowance or charge base amount. The base amount is used according to the line level allowance or charge percentage. This is to calculate the line level allowance or charge amount. The value is rounded off to maximum two decimals.</p> <p>Example: 1000</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
247	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:BaseAmount element to display the ISO code of the transaction's currency.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
248	cac:Item			Both	<p>This parent element displays a group of business terms with information about invoiced goods and services. This element has no value, but it has child elements with values associated with it in the template.</p>
249	cbc:Description	item.description	Description	Both	<p>This element displays the description and features of each item in a transaction. The value of the <b>Description</b> field for items is displayed.</p> <p>Example: Long description of an item</p>
250	cbc:Name	item.name	Item Name	Both	<p>This element displays the item name entered in the <b>Item Name</b> field in a transaction.</p> <p>Example: Item name</p>
251	cac:BuyerItemIdentification			Both	<p>This parent element displays the buyer's item identification and has no value in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
252	cbc:ID		Custom Field	Both	<p>This element displays an identifier assigned by the buyer for an item.</p> <p>Example: 123455</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
253	cac:SellerItemIdentification			Both	<p>This parent element displays the information about seller's identification. This element has no value, but it has child elements with values associated with it in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
254	cbc:ID	item.item	Item Name or Item Number	Both	<p>This element displays an identifier assigned by the seller for an item. The value is displayed only for items without discount and description in the generated e-document.</p> <p>Example: 9873242</p>
255	cac:StandardItemIdentification			Both	<p>This parent element has standard item identification details. This element has no value, but it has child elements with values associated with it in the template.</p>
256	cbc:ID	item1.itemUpcCode (Refer CDS)	UPC Code	Both	<p>This element displays the standard item identifier based on a registered scheme.</p> <p>Example: 10986700</p> <p>This element uses Custom Data Source plug-in to source the value of an item's UPC code in a template.</p>
257	@schemeID		Custom value	Both	<p>This element is used with cbc:ID element, and displays the scheme ID for the item standard identifier. This element has a default value 0160.</p> <p>Example: 0160</p> <p>This element does not have a value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
258	cac:OriginCountry			Both	<p>This parent element is used to display an item's origin country. This element has no value, but it has child elements with values associated with it in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
259	cbc:IdentificationCode	item1.itemCountry	Manufacturer Country	Both	<p>This element displays the ISO code of an item's country of origin.</p> <p>Example: CN</p> <p>This element uses Custom Data Source plug-in implementation to source the value of an item's manufacturer country.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
260	cac:CommodityClassification			Both	<p>This parent element displays the commodity classification code. This is required when members from same country buy from each other, and the traders need statistical information in the invoice.</p> <p>It is that you use the Item Classification Identifier (BT-158) with the code HS, as an identifier list for this purpose.</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
					This element has no value, but has child elements with values associated with it in the template.
261	cbc:ItemClassificationCode	item1.itemUpcCode	UPC Code	Both	<p>This element displays an item classification identifier code to classify items based on its type or nature. Example: 9873242</p> <p>This element uses Custom Data Source plug-in to source the value of an item's UPC code.</p>
262	@listID		Fixed Value	Both	<p>This element is used with cbc:ItemClassificationCode element and displays a scheme ID for item classification identifier. It has a fixed value HS.</p> <p>Example: STI</p> <p>You can map the value with the required custom value in the SuiteApp</p>
263	@listVersionID		Custom Value	Both	<p>This element is used with cbc:ItemClassificationCode element and displays scheme ID for the item classification identifier. This element has no value in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
264	cac:ClassifiedTaxCategory			Both	<p>This parent element displays a group of business terms with information about the GST applicable for invoiced goods and services in an invoice line.</p> <p>This element has no value, but it has child elements with values associated with it in the template.</p>
265	cbc:ID		Custom Field	Both	<p>This element displays the GST category code for an invoiced item.</p> <p>Example: S</p> <p>This element has no value in the template.</p> <p>You can map the value with the required custom field in the SuiteApp.</p>
266	cbc:Percent	item.taxrate1	Tax Rate	Both	<p>This element displays the GST rate percentage applied to the invoiced item. It displays the tax rate value of an item.</p> <p>Example: 25</p>
267	cac:TaxScheme			Both	<p>This element displays information about tax scheme.</p> <p>This parent element has no value, but it has child elements with values associated with it in the template.</p>
268	cbc:ID		Fixed Value	Both	<p>This element displays an ID of the tax scheme used in an invoice. A fixed value GST is added in the template.</p>
269	cac:AdditionalItemProperty			Both	<p>This parent element displays a group of business terms with information about item attributes of invoiced goods and services. This element has no value, but it has various child elements that have values associated with it in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
270	cbc:Name		Custom Field	Both	<p>This element displays the name of the item attribute or the property of the item.</p> <p>Example: Color</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
271	cbc:Value		Custom Field	Both	<p>This element displays a value for the item attribute. The value of the attribute or property of the item.</p> <p>Example: Black</p> <p>You can map the value with the required custom field in the SuiteApp.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
272	cac:Price			Both	<p>This parent element displays the price details. It has child elements with information about the price applied for the invoiced goods and services in an invoice.</p> <p>This element has no value, but it has various child elements with values in the template.</p>
273	cbc:PriceAmount	Item.rate	Unit Price or rate	Both	<p>This element displays the price of an item excluding GST and item price discount. The item net price must be equal to the item gross price, which is less than the item price discount if both the prices are provided. The item price cannot be negative.</p> <p>The unit price or rate of an item is used as a value and must be rounded off to 2 decimal places.</p> <p>Example: 10.8955 is rounded to 11</p>

SL No.	ID	Field ID	Technical Field Name	Applicable for Transaction Type	NetSuite Field Mapping Logic Used for Generic Templates
274	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:PriceAmount to display the ISO code of the transaction's currency.</p>
275	cbc:BaseQuantity	item.quantity	Quantity	Both	<p>This element displays item price base quantity of item units, to which the price applies. Example: 1</p> <p>The element value with a decimal separator (.) or a whole number is valid. Group separator or more than one decimal separator is invalid.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
276	@unitCode		Custom Value	Both	<p>This element displays the unit of measure code applicable to the item price base quantity. Also, the unit of measure must be same as the unit code of the invoiced or credited quantity. The unit of measure value has a 3-digit alphanumeric code. This element has no value in the template.</p> <p>Example: C62</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
277	cac:AllowanceCharge			Both	<p>This element displays allowance details of an invoice. This element has no value, but it has child elements with values in the template.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
278	cbc:ChargeIndicator		Fixed Value	Both	<p>This element indicates the charge. A charge on price level is not valid. This element has a fixed value false.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
279	cbc:Amount	item.amount	Amount	Both	<p>This element is used to calculate the item price discount. The total discount is subtracted from the gross item price to calculate the net item price. This element displays the value if only if the item type is discount.</p> <p>Example value: 100</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
280	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:Amount element to display the ISO code of the transaction's currency.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
281	cbc:BaseAmount	item.amount	Quantity * Unit Price/Rate = Amount	Both	<p>This parent element displays the gross item price. Item amount is used as value for this element. It is the unit price excluding GST before subtracting the item price discount.</p> <p>The item price discount value must not be negative.</p> <p>The formula for calculating gross item price is, quantity multiplied by unit, and price divided by rate per item. Alternatively, the item amount can also be used.</p> <p>Example: 123.5</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>
282	@currencyID	transaction.currency.symbol	Transaction's Currency	Both	<p>This element is used with cbc:BaseAmount element to display the ISO code of the transaction's currency.</p> <p>This element is not supported in the ANZ PEPPOL TEMPLATE.</p>

## Vendor Bill



**Note:** To use the templates, paste the content into the "FIELD MAPPING FOR INBOUND E-DOCUMENT" field of the new E-Document Template creation page.

In the template, while you create, select "ANZ PEPPOL Inbound CDS" from the Inbound Custom Data Source Plug-in Implementation drop down.

If no tax rate is present in for the item in XML reference, then by default, tax code with 0% tax rate will be applied to the item in the bill. This is applicable only if one tax code with tax rate 0% is present in the account.

If there are multiple tax codes of same rates or no tax codes present with specified tax rate, then an error will be thrown. The inbound conversion will be failed.

### Limitations:

- Expenses is not supported in this template; users can change the template and add expenses, if required.
- Items with vendor code will only be considered for inbound conversion. So, users have to make sure that the items in the XML Reference have vendor code value under the Vendors section of the Purchasing Tab in the item records along with the Vendor's Name. The vendor code should be same as the Item name.
- Inbound Conversion using PEPPOL for Australia and New Zealand is not supported in SuiteTax accounts.
- No other language other than English is supported for Inbound Conversion.
- Multi-Subsidiary vendors are not supported for Inbound Conversion.

S.NO	JSON Key in Template	XML Element applied from XML Reference	Information for the Key
1	tranid	D:Invoice/cbc:ID	This key uniquely identifies the Vendor Bill, that is, Reference Number.
2	trandate	D:Invoice/cbc:IssueDate	This key displays the issue date of the bill provided by the vendor, that is, Date. The date is in the format of the system preferences set in the account.  If the element cbc:IssueDate is missing in the XML reference file then it will throw an error "The issue date XPath for the transaction could not be found in the XML reference file." and If the value of the cbc:IssueDate element in XML Reference file is not in the format YYYY-MM-DD then it will throw an error "The issue date of the transaction in the XML reference file is not in correct format.".
3	duedate	D:Invoice/cbc:DueDate	This key displays the payment due date. The date is in the format of the system preferences set in the account. The Bill will be populated only if the XML Reference has relevant value.  If the value of the cbc:DueDate element in XML Reference file is not in the format YYYY-MM-DD, you get the error "The due date of the transaction in the XML reference file is in invalid format."

S.NO	JSON Key in Template	XML Element applied from XML Reference	Information for the Key
4	memo	D:Invoice/cbc:Note	This element displays unstructured note related to the Bill. The Bill will be populated only if the XML Reference has relevant value.
5	currency	D:Invoice/cbc:DocumentCurrencyCode	This key displays the Currency Name in Vendor Bill. The value of this field is derived from the Custom Data Source Plug-in (CDS) used in the template. It retrieves Currency Name from the Currency ISO code present in the XML Reference. Multi Currency feature must be enabled in the account.
6	item	D:Invoice/cac:InvoiceLine	This key iterates over all line items in the XML Reference.
7	vendorcode	D:Invoice/cac:InvoiceLine/cac:Item/cbc:Name	This key displays Vendor Code that is same as the item name. It does not support any other reference apart from item name. Check Limitations section for more details.
8	vendorname	D:Invoice/cac:InvoiceLine/cac:Item/cbc:Name	This key displays Vendor Name that is same as the item name.
9	quantity	D:Invoice/cac:InvoiceLine/cac:Item/cbc:InvoicedQuantity	<p>This key displays Quantity of each line item. Quantity is not added if the quantity of each line item is not present in the XML Reference.</p> <p>The element value with a decimal separator (.) or a whole number is valid. Group separator (Thousands separator) or more than one decimal separator is invalid.</p>
10	rate	D:Invoice/cac:InvoiceLine/cac:Item/cac:Price/cbc:PriceAmount	<p>This key displays net price or unit price of the item. It displays the rate of each line item, exclusive of GST after subtracting item price discount.</p> <p>The element value with a decimal separator (.) or (,), or a whole number is valid.</p> <p>Group separator (Thousands separator) or more than one decimal separator is invalid.</p>
11	taxcode	<p>D:Invoice/cac:InvoiceLine/cac:Item/cac:ClassifiedTaxCategory/cbc:Percent            (We are fetching the tax percent and from that we are finding the corresponding tax code in the account)</p>	<p>This key provides tax codes that is extracted from the tax rate in the XML Reference. [Check the XML Element given in second column]. The tax code which belongs to the specific tax rate for the applicable country.</p> <p>You need to extract the tax codes from this XML Reference: D:Invoice/cac:AccountingSupplierParty/cac:Party/cac:PostalAddress/cac:Country/cbc:IdentificationCode. In OW Accounts, it will retrieve Subsidiary country information. In SI accounts, it will retrieve information about the country from the company information page being populated.</p>

S.NO	JSON Key in Template	XML Element applied from XML Reference	Information for the Key
			<p>1. The following conditions can stop the conversion and throw error:</p> <ul style="list-style-type: none"> <li>■ If you have more than one tax code applicable of <b>Purchase</b> or <b>Both</b> tax type present for a tax rate in the account.</li> <li>■ If no tax code is present for a tax rate.</li> <li>■ If the tax code applicable is inactive.</li> <li>■ If more than one applicable tax code is present for a tax rate.</li> </ul> <p>It throws the below error.</p> <p>"Appropriate tax code could not be found for one of the following reasons:</p> <ul style="list-style-type: none"> <li>-No such tax code exists.</li> <li>-Multiple tax codes with the same tax rate exists.</li> <li>-Tax code is not of type Purchase or Both.</li> <li>-Tax code is inactive."</li> </ul> <p>Rectify the tax codes and make the combination of tax code to tax rate unique for the applicable country.</p> <p>2. If the tax rate is not present in the XML reference file for a line item, the tax code with tax rate 0% is automatically assigned to that line item.</p> <p>In the below case, following conditions can throw an error:</p> <ul style="list-style-type: none"> <li>■ If more than one applicable tax code with tax rate 0% is present in the account.</li> <li>■ If there are no tax codes present for 0%.</li> <li>■ If the tax code applicable is inactive.</li> <li>■ If not of type Both or Purchase.</li> </ul> <p>"Appropriate tax code could not be found for one of the following reasons:</p> <ul style="list-style-type: none"> <li>-No such tax code exists.</li> <li>-Multiple tax codes with the same tax rate exists.</li> <li>-Tax code is not of type Purchase or Both.</li> <li>-Tax code is inactive."</li> </ul> <p>In this case, add a tax code which is active, of type Purchase or Both and of tax rate 0% for the applicable country in the account.</p> <p>3. For Australia and New Zealand, we are not supporting inbound conversion using PEPPOL in SuiteTax Accounts.</p> <p>If you try to use this template along with CDS in SuiteTax enabled accounts, then conversion will fail and you get the error "Inbound conversion is not supported for SuiteTax accounts in Australia and New Zealand."</p> <p>PEPPOL Tax Category is currently not supported for ANZ. If tax category is</p>

S.NO	JSON Key in Template	XML Element applied from XML Reference	Information for the Key
			<p>present in the inbound XML file, it will not be considered for determining the tax code.</p> <p>The element value with a decimal separator (.) or (,), or a whole number is valid. Group separator (Thousands separator) or more than one decimal separator is invalid.</p>
12	amount	D:Invoice/cac:InvoiceLine/cbc:LineExtensionAmount	<p>The key displays invoice line net amount. The amount is "net" without GST, that is, inclusiveness of line level allowances and charges along with other relevant taxes.</p> <p>The element value with a decimal separator (.) or (,), or a whole number is valid. Group separator (Thousands separator) or more than one decimal separator is invalid.</p>
13	description	D:Invoice/cac:InvoiceLine/cac:Item/cbc:Name	This key's description is taken from the item's name. You can change the reference in the XML Reference if needed.
14	inventiorydetailreq		This key is a required field for items. The default value is set to 'False' for each item.

## Understanding Inbound E-Document Templates in JSON Format

Implementing an inbound e-document template enables the system to map which elements in the received XML file will provide data to which fields in the vendor bill record to be created from the XML file.

An inbound e-document template is in JSON format. For more information about JSON objects, go to the w3schools website [JSON Introduction](#).

The bundle also includes a sample JSON template that can be used for parsing XML inbound e-documents for conversion into vendor bills. You can download the sample JSON template from the File Cabinet. The JSON inbound template contains the mapping to basic bill information:

- tranid
- trandate
- currency
- memo
- item
  - vendorname/vendorcode
  - amount
  - rate
  - quantity
  - description
  - tax1amt
- createdfrom



**Note:** The alias for the inbound XML e-document object is 'XML'. Use this when mapping the XML elements to keys. For example, \${XML.ParentElement.ChildElement}.

You can use or customize the sample JSON template that contains the mapping to basic vendor bill information:

```

1  {
2    "tranid": "${XML.Invoice.InvoiceHeader.InvoiceNumber}",
3    "trandate": "${XML.Invoice.InvoiceHeader.InvoiceDate}",
4    "currency": "${XML.Invoice.InvoiceHeader.Currency}",
5    "memo": "${XML.Invoice.InvoiceHeader.Memo}",
6    "createdfrom": "${XML.Invoice.InvoiceHeader.PONumber}",
7    "item": [
8      <#list XML.Invoice.InvoiceDetails.InvoiceItem as item>
9      {
10        "vendorcode": "${item.ItemName}",
11        "quantity": "${item.Quantity}",
12        "rate": "${item.UnitPrice?replace("$", "")}",
13        "amount": "${item.LineItemSubtotal?replace("$", "")}",
14        "description": "${item.Description}",
15        "tax1amt": "${item.TaxAmount?replace("$", "")}"
16      }
17      <if item_has_next>,</if>
18      </list> ],
19    "expense": [
20      <#list XML.Invoice.InvoiceDetails.InvoiceExpense as expense>
21      {
22        "amount": "${expense.Amount?replace("$", "")}",
23        "memo": "${expense.Description}"
24      }
25      <if expense_has_next>,</if>
26      </list> ]
27  }

```

tranid, trandate, currency, memo, item, expense and createdfrom are called key names. Every key name must correspond to a field ID in the vendor bill record to be created from the received XML file. The key name is the reference that points to a field in the vendor bill record. Each key name must extract a value from the received XML file. The value of the key name will be the data that will be entered in the corresponding field of the vendor bill record.

tranid is a required key, used as reference number of the vendor bill. item is another required key name that is an of JSON objects with details of each item in the vendor bill.

createdfrom is a key name used if the vendor bill record to be created is from a Purchase Order. createdfrom will take up the value of the PO# of the source purchase order.

item is another required key name that is an of JSON objects with details of each item in the vendor bill. Under item is vendorcode, which is a required key name if the Multiple Vendor feature is enabled. vendorcode maps to the code assigned to a specific vendor of an item. If the Multiple Vendor feature is not enabled, the vendornname key name must be used. vendornname maps to the vendor name/code field of an item.

expense is also an of JSON objects that takes each expense in the vendor bill. Under expense are the amount and memo key names.



**Note:** Ensure that your item records are updated and must have unique vendornname or vendorcode. Also, you must specify the Default Expense Account in the vendor record if you expect to receive bills for expense lines.

At least the required key names must be present in an inbound e-document template. You must not change or edit required key names.

If you have custom records or fields, you can create your own key names that correspond to the field IDs of those custom fields. But make sure that your custom key names have data to extract from the XML files that you will receive from your vendors or other parties.

After setting up the key names and values of your JSON template, you can now implement it as an inbound e-document template. For more information, refer to step 6 of [Creating E-Document Templates](#).

## Understanding XSD in Inbound E-Document Templates

The XSD file or XML Schema is a text file that defines and validates what XML elements and attributes must be present in the received XML, before it is uploaded as an inbound e-document record. After the received XML file is validated, the system assigns an e-document template to it. The first matched e-document template will be assigned to the newly created inbound e-document.

The following is a sample XSD that you can use as a reference in creating your own XSD file:

```

1 <xsd:element name="edoc" type="edocType"/>
2
3 <xsd:complexType name="edocType">
4   <xsd:sequence>
5     <xsd:element name="tranid" type="xsd:string"/>
6     <xsd:element name="po" type="xsd:integer"/>
7     <xsd:element name="memo" type="xsd:string"/>
8     <xsd:any processContents="skip" minOccurs="0"/>
9   </xsd:sequence>
10  <xsd:attribute name="version" type="xsd:string" use="required" fixed="1.1"/>
11 </xsd:complexType>
12
13 </xsd:schema>
```

This sample XSD validates an XML document for the following:

- The root element is edoc is present, which must have an attribute version, with a value of 1.1.
- The edoc element must have tranid, po, and memo elements in the correct order, where:
  - tranid is a string
  - po is an integer
  - memo is a string

If this sample XSD is used, a received XML file must contain all the elements and attributes defined in the XSD. If it does, the E-Document Template record that the XSD is a part of, will be applied to the received XML file, which is then uploaded as an inbound e-document record. Other elements and attributes can be present in the received XML file in addition to the required ones defined in the XSD. You can define your own required elements and attributes in the XSD file you will create.

## Understanding XSD in Outbound E-Document Templates

The XSD file or XML Schema is a text file. It defines and validates what XML elements and attributes must be present in the generated XML, before it is saved as a generated e-document. A detailed audit trail message is logged if the validation fails.

The following is a sample XSD that you can use as a reference in creating your own XSD file:

```

1 <xsd:element name="edoc" type="edocType"/>
2
3 <xsd:complexType name="edocType">
4   <xsd:sequence>
5     <xsd:element name="tranid" type="xsd:string"/>
6     <xsd:element name="po" type="xsd:integer"/>
```

```

7     <xs:element name="memo" type="xs:string"/>
8     <xs:any processContents="skip" minOccurs="0"/>
9   </xs:sequence>
10  <xs:attribute name="version" type="xs:string" use="required" fixed="1.1"/>
11 </xs:complexType>
12
13 </xs:schema>

```

This sample XSD validates an XML document for the following:

- The root element edoc is present, which must have an attribute version, with a value of 1.1.
- The edoc element must have tranid, po, and memo elements in the correct order, where:
  - tranid is a string
  - po is an integer
  - memo is a string

If this sample XSD is used, the generated XML file must contain all the elements and attributes defined in the XSD. If it does, the e-document is successfully generated. Other elements and attributes can be present in the generated XML file in addition to the required ones defined in the XSD. You can define your own required elements and attributes in the XSD file you will create.

The **Outbound XSD File** field is used for validating the generated e-document, only if you select the XML format from the **Content Type** dropdown list.

## Creating a Digital Signature Plug-in Implementation for E-Documents

A Digital Signature plug-in implementation for e-documents will enable you to generate digitally signed e-documents. You can select a digital signature plug-in from the **Digital signature Plug-in Implementation** field in an e-document template record to digitally sign e-documents. Only the XML or JSON files are generated with a digital signature. The generated PDF files are not digitally signed.

You must first create a custom plug-in implementation for digital signature and then implement it in NetSuite. After this, the plug-in will be available for you to select from the **Digital Signature Plug-in Implementation** field on the e-document template record. To create this plugin implementation, you must first create a JavaScript file for the digital signature plug-in implementation. The JavaScript file must be compatible with SuiteScript 2.0. For more information about creating a Javascript file, refer to the help topic [SuiteScript 2.x Script Creation Process](#).

The plug-in script must return an object with the following function:

```

1 /**
2  * Copyright (c) 2019, Oracle NetSuite its affiliates. All rights reserved.
3  *
4  * @NApiVersion 2.x
5  * @NModuleScope Public
6  * @NScriptType pluginTypeimpl
7 */
8
9 define(["N/crypto/certificate", "N/file"], function(certificate, file) {
10   /**
11    * @param {Object} pluginContext
12    * @param {String} pluginContext.unsignedString
13    * @param {String} pluginContext.subsidiaryId
14    * @param {String} pluginContext.tranType
15    * @param {String} pluginContext.tranId
16    * @param {Number} pluginContext.userId
17    *
18    * @returns {Object} result
19    * @returns {String} result.success

```

```

20  * @returns {String} result.signedString
21  * @returns {String} result.message
22  */
23
24  function signDocument(pluginContext) {
25      var unsignedString = pluginContext.unsignedString;
26
27      /* Extract the other required values from pluginContext:
28          var subsidiaryId = pluginContext.subsidiaryId;
29          var tranType = pluginContext.tranType;
30          var tranId = pluginContext.tranId;
31          var userId = pluginContext.userId;
32      */
33
34      /* Sample params for N/certificate.signXml()
35          var rootTag = "RootTag";
36          var certificateId = "custcertificatesfd";
37          var algorithm = "SHA1";
38      */
39
40      var result = {
41          success: true,
42          signedString: unsignedString,
43          message: "This is a sample implementation of Digital Signature.",
44      };
45
46      /**
47      * Call services to sign the string
48      */
49      try {
50          /*
51              var signedXML = certificate.signXml({
52                  algorithm : algorithm,
53                  certId : certificateId,
54                  rootTag : rootTag,
55                  xmlString : unsignedString
56              });
57
58              result.success = true;
59              result.signedString = signedXML.asString();
60              result.message = "Document signed successfully";
61          */
62      } catch (e) {
63          result.success = false;
64          result.signedString = "";
65          result.message = e.message;
66      }
67
68      return result;
69  }
70
71  return {
72      signDocument: signDocument,
73  };
74 });

```

This script takes the input **pluginContext** which is a JSON object. The parameters of this object are listed in the following table.

Parameter	Type	Description	Remarks
unSignedString	String	This field holds the generated e-document of a transaction in XML or JSON format. The format depends on its selection in the E-Dокумент Template used for generation. The e-document is not digitally signed.	
subsidiaryId	String	This field holds the Subsidiary ID of a transaction.	This parameter can be blank if the subsidiary information is unavailable in the transaction.

tranType	String	This field displays the type of transaction.	tranId along with tranType provides context about the transaction being digitally signed. You can use them together to reference any field value of the transaction using N/search.lookupFields or N/record.load
tranID	String	This field displays the internal ID of the transaction.	
userId	number	This field holds the internal ID of the current logged in user.	This field value can be used wherever there is a requirement to refer to the current logged in user. For example, to update e-document audit trail by shared module API, userId can be used in owner property.

This script must implement digital signature process on an unsigned XML or JSON string. For example Suitescript 2.0 module N/crypto/certificate. This results in a signed XML or JSON string which must be returned in a JSON object with parameters listed in the following table.

Parameter	Type	Description	Required/Optional
success		Valid Values are either true or false. Set the value to true, if the digital signature plug-in implementation is successful and e-document is digitally signed. A string will be generated. Otherwise, set it to false.	Required
signedString	String	This parameter holds the signed string when the digital signature plug—in implementation is successfully completed.	Required
message	String	Message to be passed to Electronic Invoicing. This message gets displayed in E-Document audit trail of the transaction.	Required



**Important:** The digital signature plug-in implementation script must have the @NSScriptType plugintypeimpl.

### To create a plug-in implementation record:

1. Go to Customization > Plug-ins > Plug-in Implementations > New.
2. Create a JavaScript file following the sample script.
3. Select the JavaScript file you created, from the **Script File** field and click **Create Plug-in Implementation** button.
4. Select the Digital Signature for E-Document plug-in type from the Select Plug-in Type record.
5. Type the information in the required fields on the Plug-in Implementation record.
6. Click **Save**.

a digital signature plug-in implementation record is created, you can select it from the E-Document Template record. For more information, see the help topics [Custom Plug-in Creation](#) and [Digital Signing](#).

## Creating an Outbound Validation Plug-in Implementation for E-Documents

An Outbound Validation Plug-in Implementation for e-documents will enable you to validate the e-document during generation process with your custom logic. You can select an Outbound Validation Plug-in Implementation from the **Outbound Validation Plug-in Implementation** field in an E-Document Template.

To use custom validation, you must first create a custom plug-in implementation for outbound validation plug-in for e-documents. After this, the plug-in implementation will be available for you to select in the **Outbound Validation Plug-in Implementation** field on the E-Dокумент Template.

To create this plug-in implementation, you must first create a JavaScript file that must be compatible with SuiteScript 2.0. For more information about creating a Javascript file, see the help topic [SuiteScript 2.x Script Creation Process](#).

The following code is a sample plug-in implementation script for outbound validation:

```

1. /**
2.  * @NApiVersion 2.x
3.  * @NModuleScope Public
4.  * @NScriptType pluginimpl
5. */
6. define([], function() {
7.
8.     /**
9.      *
10.     * @param {Object} pluginContext
11.     * @param {String} pluginContext.content
12.     * @param {String} pluginContext.transactionInfo.transactionId
13.     * @param {String} pluginContext.transactionInfo.transactionType
14.     * @param {Number} pluginContext.userId
15.     * @returns {Object} result
16.     * @returns {string} result.success
17.     * @returns {String} result.message
18. */
19. function validate(pluginContext) {
20.
21.     var result = {
22.         success: false,
23.         message: "Validation failed."
24.     };
25.
26.     try {
27.
28.         /**
29.          * Extract the values from pluginContext
30.          */
31.
32.         // var content = pluginContext.content;
33.         // var userId = pluginContext.userId;
34.         // Connect to validation service and get response.
35.
36.         /**
37.          * Use this information to fetch the transaction data
38.          */
39.
40.         /*
41.         var transactionType = pluginContext.transactionInfo.transactionType;
42.         var transactionId = pluginContext.transactionInfo.transactionId;
43.         var transObj = record.load({
44.             type: transactionType,
45.             id: transactionId
46.         });
47.         */
48.
49.         // If successful
50.         result.success = true;
51.         result.message = "Validation successful!";
52.
53.         // Sample result if not successful
54.         // result.success = false;
55.         // result.message = "Validation failed.";
56.
57.         return result;
58.
59.     } catch (e) {
60.         result.success = false;
61.         result.message = e.message;

```

```

62.         }
63.         return result;
64.     }
65. }
66.
67.     return {
68.         validate: validate
69.     };
70. });

```

This script takes the input **pluginContext** which is a JSON object. The parameters of this object are listed in the following table.

Parameter	Type	Description	Remarks
content	String	This field holds the generated e-document of a transaction in XML or JSON format. The format depends on its selection in the E-Document Template used for generation.	
transactionInfo.transactionType	String	The type of transaction being validated.	transactionId and transactionType parameters provide information about the transaction being validated. You can use them together to reference any field value of the transaction using N/search.lookupFields or N/record.load
transactionInfo.transactionId	String	The internal ID of the transaction being validated.	
userId	number	This field holds the internal ID of the current logged in user.	This field value can be used wherever there is a requirement to refer to the current logged in user. For example, to update e-document audit trail by shared module API, userId can be used in owner property.

This script can implement any custom validation logic on edocString in the validate function. The validate function should return JSON object with parameters listed in the following table.

Parameter	Type	Description	Required/Optional
success		Valid Values are either true or false. Set the value to true, if the validation is successful. Otherwise, set it to false.	Required
message	String	Message to be passed to Electronic Invoicing. This message gets displayed in E-Document audit trail of the transaction.	Required



**Important:** The Outbound Validation Plug-in implementation script must have the @NSScriptType plugintypeimpl.

### To create a plug-in implementation record:

1. Go to Customization > Plug-ins > Plug-in Implementations > New.
2. Create a JavaScript file following the sample script.
3. Select the JavaScript file you created, from the **Script File** field and click **Create Plug-in Implementation** button.
4. Select the Outbound Validation Plug-in type from the Select Plug-in Type record.
5. Type the information in the required fields on the Plug-in Implementation record.

6. Click **Save**.

an Outbound Validation plug-in implementation record is created, you can select it in the E-Document Template record. For more information, see the help topic [Custom Plug-in Creation](#).

## Creating a Custom Plug-in Implementation for E-Document Custom Data Source

A custom plug-in implementation for custom data source will enable you to add custom data sources to an e-document template. With a custom data source plug-in specified in an e-document template, you can add to e-documents more field values from the transactions that the e-documents will be generated from.

You must create a custom plug-in implementation for custom data source first and then implement it in NetSuite so that it will be available for selection on the e-document template record. Create a JavaScript file for the custom data source plug-in implementation. The JavaScript file must be compatible with SuiteScript 2.0.

**Note:** This sample script uses the `define` function, which is required for an entry point script (a script you attach to a script record and deploy). You must use the `require` function if you want to copy the script into the SuiteScript Debugger and test it. For more information, see the help topic [SuiteScript Debugger](#).

The following code is a sample custom plug-in implementation for e-document custom data source.

```

1  * @NApiVersion 2.x
2  * @NScriptType pluginTypeimpl
3  * @NModuleScope Public
4  */
5 define(["N/render"], function(nsrender) {
6  /**
7   * inject - This function will provide the custom data source during the generation process
8   * @param {Object} params
9   * @param {String} params.transactionId
10  * @param {Object} params.transactionRecord
11  * @param {Number} params.userId
12  *
13  * @returns {Object} result
14  * @returns {render.DataSource} result.alias
15  * @returns {string} result.format
16  * @returns {Object | Document | string} result.data
17  */
18
19  function inject(params) {
20    var txnRecord = params.transactionRecord;
21    var txId = params.transactionId;
22    var userId = params.userId;
23    var customObj = {};
24    log.debug("Custom Object", customObj);
25    return {
26      customDataSources: [
27        {
28          format: nsrender.DataSource.OBJECT,
29          alias: "custom",
30          data: customObj
31        }
32      ],
33    };
34  }
35
36  return {
37    inject: inject
38  };
39 });

```

This script takes the input parameters from the JSON object. The parameters of this object are listed in the following table.

Parameter	Type	Description	Remarks
transactionRecord.type	String	The type of transaction being validated.	The id and type parameters provide information about the transaction being validated. You can use them together to reference any field value of the transaction using <a href="#">search.lookupFields(options)</a> or <a href="#">record.load(options)</a>
transactionRecord.id	Number	The internal ID of the transaction being validated.	
transactionId	String	The internal ID of the transaction being validated.	The id provide information about the transaction's internal ID.
userId	Number	This field holds the internal ID of the current logged in user.	This field value can be used wherever there is a requirement to refer to the current logged in user. For example, to update e-document audit trail by shared module API, userId can be used in owner property.



**Important:** The custom data source plug-in implementation script must have the `@NSScriptType pluggintypeimpl`.

After creating the script for plug-in implementation, upload it to Customization > Plug-ins > Plug-in Implementations > New. The type of the custom plug-in implementation must “Custom Data Source for E-Document”. For more information, see the help topics [Custom Plug-in Creation](#), [TemplateRenderer.addCustomDataSource\(options\)](#) and [Using Custom Data Sources for Advanced Printing](#).

The following are guidelines for custom data sources:

- Naming convention - Element names must not begin with digits
- Do not add a large amount of data to the data object in `customDataSource`. This may result in an Out of Memory error.

If the datasource provided by your implementation is in the following format:

```

1 | return {
2 |   customDataSources: [
3 |     {
4 |       format: nsrender.DataSource.OBJECT,
5 |       alias: "custom",
6 |       data: {isOneWorldEnabled: true}
7 |     }
8 |   ],
9 | };

```

You can now include the following custom data in the e-document template by using the datasource format.

- For XML e-document:

```

1 | <xml> ${custom.isOneWorldEnabled} </xml>

```

- For JSON e-document:

```

1 | {
2 |   "key": "${custom.isOneWorldEnabled}"
3 | }

```

## QR String Generation

The QR string generation logic can be implemented in Custom Data Source Plug-in Implementation. After successful implementation of the QR logic, generated string can be populated in **`custbody_qr_string`**. It is hidden and present in transaction record, using Custom Data Source as illustrated in [Sample QR Code Plug-in Implementation\(Custom Data Source\)](#). This populated field can be used in [Advanced PDF/HTML Template](#) to display QR code in generated e-document PDF.

## QR Code Preview

To generate a preview of QR code in the transaction record under **E-Document** subtab, **custbody\_qr\_code** field (rich text) can be updated with img tag containing the source of data URL for the QR code value as illustrated in [Sample QR Code Plug-in Implementation\(Custom Data Source\)](#). The data URL for QR code value can be generated using any library which is capable of generating data URL from a string.

## Sample QR Code Plug-in Implementation(Custom Data Source)

This is an example of QR Code Plug-in implementation (CDS). It shows example usage where SS2.0 can be used to:

- Show QR code preview in Transaction form by generating data URL and using it to update **custbody\_qr\_code** field. This is a rich text field that can be used to display the QR code preview on transaction record.
  - Populate **custbody\_qr\_string**, a hidden field that can be used as a data source in any [Advanced PDF/HTML Template](#) to display QR Code in generated PDF.

```
1  /**
2   * @NApiVersion 2.x
3   * @NScriptType plugintypeimpl
4   * @NModuleScope Public
5   */
6 // Give the relative path of qrcode-generator module responsible for generation of data URL from QR code value
7 define(["N/encode", "N/render", "N/record", "./qrcode-generator"], function (encode, nsrender, record, qrCodeGen) {
8
9   // Implement the logic to return QR string from the input QR data passed
10  function getQRCodeString(inputQRData) {
11    var qrString = '';
12    /*
13     * QR logic implementation goes here, with 'inputQRData' parameter object containing the required properties for generating
14     QR string stored in qrString variable
15     */
16    return qrString;
17  }
18
19 // Function returns the encoded base 64 value of the string passed
20 function getBase64(qrstring) {
21   return encode.convert({
22     string: qrstring,
23     inputEncoding: encode.Encoding.UTF_8,
24     outputEncoding: encode.Encoding.BASE_64_URL_SAFE
25   });
26
27 // Function returns img element which contains the QR Code preview image
28 function getQRCodePreview(qrBase64) {
29   var qrDataUrl = '';
30   /*
31    * Here 3rd party library can be used to convert the 'qrBase64' string to data url generation from QR code value, as 'qrCode
32    Gen' is imported in this example
33    */
34
35    if (qrDataUrl) {
36      return '</img>';
37    }
38  }
39
40  if (qrDataUrl) {
41    return '</img>';
42  }
43
44  if (qrDataUrl) {
45    return '</img>';
46  }
47
48  if (qrDataUrl) {
49    return '</img>';
50  }
51
52  if (qrDataUrl) {
53    return '</img>';
54  }
55
56  if (qrDataUrl) {
57    return '</img>';
58  }
59
60  if (qrDataUrl) {
61    return '</img>';
62  }
63
64  if (qrDataUrl) {
65    return '</img>';
66  }
67
68  if (qrDataUrl) {
69    return '</img>';
70  }
71
72  if (qrDataUrl) {
73    return '</img>';
74  }
75
76  if (qrDataUrl) {
77    return '</img>';
78  }
79
80  if (qrDataUrl) {
81    return '</img>';
82  }
83
84  if (qrDataUrl) {
85    return '</img>';
86  }
87
88  if (qrDataUrl) {
89    return '</img>';
90  }
91
92  if (qrDataUrl) {
93    return '</img>';
94  }
95
96  if (qrDataUrl) {
97    return '</img>';
98  }
99
100 if (qrDataUrl) {
101   return '</img>';
102 }
103
104 if (qrDataUrl) {
105   return '</img>';
106 }
107
108 if (qrDataUrl) {
109   return '</img>';
110 }
111
112 if (qrDataUrl) {
113   return '</img>';
114 }
115
116 if (qrDataUrl) {
117   return '</img>';
118 }
119
120 if (qrDataUrl) {
121   return '</img>';
122 }
123
124 if (qrDataUrl) {
125   return '</img>';
126 }
127
128 if (qrDataUrl) {
129   return '</img>';
130 }
131
132 if (qrDataUrl) {
133   return '</img>';
134 }
135
136 if (qrDataUrl) {
137   return '</img>';
138 }
139
140 if (qrDataUrl) {
141   return '</img>';
142 }
143
144 if (qrDataUrl) {
145   return '</img>';
146 }
147
148 if (qrDataUrl) {
149   return '</img>';
150 }
151
152 if (qrDataUrl) {
153   return '</img>';
154 }
155
156 if (qrDataUrl) {
157   return '</img>';
158 }
159
160 if (qrDataUrl) {
161   return '</img>';
162 }
163
164 if (qrDataUrl) {
165   return '</img>';
166 }
167
168 if (qrDataUrl) {
169   return '</img>';
170 }
171
172 if (qrDataUrl) {
173   return '</img>';
174 }
175
176 if (qrDataUrl) {
177   return '</img>';
178 }
179
180 if (qrDataUrl) {
181   return '</img>';
182 }
183
184 if (qrDataUrl) {
185   return '</img>';
186 }
187
188 if (qrDataUrl) {
189   return '</img>';
189 }
```

```

37     return '';
38 }
39
40 function inject(params) {
41   var tranType = params.transactionRecord.type;
42   var tranId = params.transactionRecord.id;
43
44   // Hidden QR string field in transaction record
45   var QR_CODE_FLD = "custbody_qr_string";
46   // QR preview field in transaction record
47   var QR_CODE_PREVIEW_FLD = "custbody_qr_code";
48   var updateQRFlds = {};
49   var qrCustomObj = {};
50
51   /*
52    * Read the fields of transaction required for QR value generation and pass them to getQRCodeString function as shown below
53    where QR logic can be implemented
54    * Ex:
55    * var data = {
56    *   sellerName: subsidiaryLegal,
57    *   vatNumber: vatRegistrationNo.toString(),
58    *   invoiceTimeStamp: transDate.toISOString(),
59    *   invoiceTotal: totalAmount.toString(),
60    *   invoiceVatTotal: totalTax.toString(),
61    * };
62    */
63
64   //Pass the QR Code input data to return QR string
65   var qrString = getQRCodeString(data);
66
67   //Pass the QR string to get the encoded Base 64
68   var qrBase64 = getBase64(qrString);
69
70   if(qrBase64)
71   {
72     // qrCustomObj will be sent as part of inject function response which can be used in e-document template which is shown in the below
73     example
74     qrCustomObj = {
75       qrCode : qrBase64
76     };
77
78     // Update QR hidden text field to display in generated e document PDF
79     updateQRFlds[QR_CODE_FLD] = qrBase64;
80
81     // Update QR preview field only to display QR code in transaction form under E-document subtab
82     updateQRFlds[QR_CODE_PREVIEW_FLD] = getQRCodePreview(qrBase64);
83
84     // Submitting the QR fields which needs to be updated
85     record.submitFields({
86       type: tranType,
87       id: tranId,
88       values: updateQRFlds
89     });
90
91     return {
92       customDataSources: [
93         {
94           format: nsrender.DataSource.OBJECT,
95           alias: "qrData",
96           data: qrCustomObj
97         }
98       ],
99     };
100
101   return {
102     inject: inject
103   }
104 }

```

You can now include the following custom data of QR code in the e-document template by using the data source format with respect to the alias provided.

- For XML e-document:

```
1 | <xml> ${qrData.qrCode} </xml>
```

- For JSON e-document:

```
1 | {
2 |   "key": "${qrData.qrCode}"
3 | }
```

## Editing E-Document Templates

To edit e-document templates, go to Setup > E-Documents > E-Document Templates. Open the e-document template in edit mode and modify information as needed, then click **Save**.

**i Note:** The **Transaction Type** field in the e-document template becomes unavailable if the e-document template has already been assigned to one or more transaction records. This field remains unavailable unless you remove the e-document template from all transactions.

**i Note:** You cannot edit the native or default templates locked by Localization SuiteApps. In such cases, you can create a copy to modify and use the newly created templates.

## E-Document Certification in the Outbound Process

Most countries that implement electronic invoicing in e-commerce require that businesses or companies submit their e-documents to a certification authority or tax agency. Certification authorities check the validity of e-documents and their compliance to any prescribed format and content. Certified e-documents are returned to the sending company for use in any other business requirements.

The outbound component of the Electronic Invoicing SuiteApp supports the sending of e-documents to authorities, organizations, or a tax agency that provide certification services.

Before e-documents can be sent for certification, the administrator must first assign a certification sending method by checking the **Sending Method for Certification** box on a sending method record. The certification sending method is a different sending method from the sending method applied at the transaction level. The certification sending method is applied to the transaction types and subsidiaries selected on the sending method record. Whether the certification sending method will use email or web service channel, the administrator must specify the certification services provider or authority as the recipient. For more information, see [Creating an E-Document Sending Method Record](#).

**i Note:** Only one certification sending method must be associated with a combination of subsidiaries and transactions you select.

E-documents generated from transaction types and subsidiaries with an assigned certification sending method will display the Certify E-Document button on the record. Clicking the Certify E-Document button sends the e-document to the certification authority.

On the other hand, e-documents generated from transaction types and subsidiaries without an associated certification sending method will display the Send E-Document button.

NetSuite receives successfully certified e-documents, of which certified XML or JSON file you can view from the E-Document subtab. The E-Document Status field on the E-Document subtab indicates the result of e-document certification. If certification fails, details of errors are logged on the E-document Audit Trail.



**Note:** If you have created your own certification sending method, you must store the certified copy of the XML or JSON in the File Cabinet and update the file ID of the certified e-document in the transaction body field certified e-document.

Certified e-documents can be sent to customers, vendors or any third party as recipients using the standard sending method assigned to the transaction. With the appropriate sending method selected, you can click the Send E-Document button on the transaction. For more information, see [Sending the E-Document of a Single Transaction](#).

## E-Document Network Status Overview

The Electronic Invoicing SuiteApp enables you to get e-documents certified from a regulatory body through an API. It is usually generated from a transaction. You can also get network status updates and display the relevant details under the **E-Document** subtab in the transactions.

The following table describes the transaction body fields which contain the network information.

Network Field	Field ID	Description
Network Reference Id	custbody_ei_network_id	It is a unique number generated by network API while certifying or sending e-document.
Network Name	custbody_ei_network_name	It is the name of a network. For instance, PEPPOL, SAT, ARIBA.
Network Status	custbody_ei_network_status	It displays a keyword representing one of the possible network statuses used by the network.
Network Status Updated On	custbody_ei_network_updated_date_time	It displays the time stamp (date and time) of the updated status.

An administrator must implement the getStatus method in the sending method to invoke e-document network status feature. For more info, read [Creating E-Document Sending Methods](#)

To use this feature for custom transaction types (CTT), you must register those custom transaction types again.



- To handle the rejected network status for a transaction, the E-Document Status field (custbody\_psg\_ei\_status) of transaction is set to Certification Data Error. It occurs when the field has Rejected status. If certification is enabled, then set the status to Sending Failed.
- Transfer Order transaction type will not be supported to get e-document network status.

## Providing Access to Get Network Status for Custom Roles

The **Get Network Status** button is hidden by default for custom roles.

### To enable it, the administrator can perform the following steps:

1. Go to Setup > Users/Roles > Manage Roles.
2. Click the **Customize** or **Edit** hyperlink of the role to which you want to give get network status permission.

3. Under the **Authentication** subtab, check the **Allow Manual Updation of Network Status box**.
4. Click **Save**.

## Creating E-Dокумент Sending Methods

An administrator must create e-document sending methods so that they can be available for selection on the transaction record and the e-document package record. Sending methods must be implemented as custom plug-in implementations instead of scripts.

Before users can send e-documents, an administrator must first assign sending methods to the e-document package record. This will enable users to select an e-document sending method on the **E-Document** subtab of the customer's transaction records.

An administrator can select an employee whose name and email address should appear as the sender of e-documents sent by your company or subsidiary. Selecting a designated e-document sender is optional. If there is no designated sender, the system uses the name and email address of the user who sent the e-document as the sender. See [Selecting a Designated E-Document Sender](#).

Another type of sending method, the certification sending method, supports the sending of e-documents to certification authorities, which certifies e-documents for validity and compliance to business requirements. For more information, [E-Document Certification in the Outbound Process](#).

To create sending methods, see the following topics:

- [Setting Up an Email Sending Method for E-Documents](#)
- [Creating Custom Methods for Sending E-Documents](#)
  - [Creating a Script for Sending E-Documents](#)
  - [Creating an E-Document Sending Method Record](#)
  - [Editing an E-Document Sending Method Record](#)

## Setting Up an Email Sending Method for E-Documents

**i Note:** The system can send an e-document by email to a maximum of 10 recipients for each customer or vendor. The system counts each contact added as a recipient. If you add the same contact multiple times, each instance is considered one recipient.

You can send e-documents by email using the **NetSuite Email Custom Plugin** sending method included in the Electronic Invoicing SuiteApp. This sending method is automatically associated with the default e-document package record also provided by the SuiteApp, and cannot be associated with any other e-document package. Also, the **NetSuite Email Custom Plugin** sending method cannot be edited or deleted.

Before you can send e-documents by email to a customer or vendor, the customer or vendor record must be assigned an e-document package that has an email sending channel. You must also define the email recipients for your customer or vendor. For information, see [Creating E-Document Packages](#) and [Defining E-Document Email Recipients](#).

After you send e-documents by email, the system sends you a notification informing you that the e-document sending process is finished. If there are errors found, an error report is included in the notification.

The **E-Document Audit Trail** subtab under the **E-Document** subtab of the transaction shows one of the following statuses to indicate whether the e-document was sent successfully or not:

- **Sent** – This status means the e-document was successfully sent. The **Details** column shows the email addresses of the sender and recipients.
- **Sending Failed** – This status means the e-document was not sent. Information about sending errors are shown in the **Details** column. You must fix the errors before you can successfully resend the e-document.

For more information, see [Outbound E-Document Sending Errors](#).

## Creating Custom Methods for Sending E-Documents

An administrator can create various custom methods for sending e-documents to different customers and vendors. You can use the custom sending methods to get status updates from a network.

To create a custom method for sending e-documents, the administrator must first create an e-document sending method plug-in implementation and then create an e-document sending method record for that implementation.

After custom sending methods are created, they become available for selection on e-document package records and transaction records.

See the following topics:

- [Creating a Custom Plug-in Implementation for Sending E-Documents](#)
- [Creating a Script for Sending E-Documents](#)
- [Creating an E-Document Sending Method Record](#)
- [Editing an E-Document Sending Method Record](#)

## Creating a Custom Plug-in Implementation for Sending E-Documents

A custom plug-in implementation for sending e-documents must be created so that it will be available for selection on the sending method record.

Create a JavaScript file for the custom plug-in implementation. The JavaScript file must be compatible with SuiteScript 2.0.



**Note:** This sample script uses the `define` function, which is required for an entry point script (a script you attach to a script record and deploy). You must use the `require` function if you want to copy the script into the SuiteScript Debugger and test it. For more information, see the help topic [SuiteScript Debugger](#).

The following code is a sample custom plug-in implementation for sending e-documents.

```

1 /**
2 * send - This function is the entry point of our plugin script
3 * @param {Object} pluginContext
4 * @param {String} pluginContext.scriptId
5 * @param {String} pluginContext.sendMethodId
6 * @param {String} pluginContext.eInvoiceContent
7 * @param {Array} pluginContext.attachmentFileIds
8 * @param {String} pluginContext.customPluginImplId
9 * @param {Number} pluginContext.batchOwner
10 * @param {Object} pluginContext.customer
11 * @param {String} pluginContext.customer.id
12 * @param {Array} pluginContext.customer.recipients
13 * @param {Object} pluginContext.transaction
14 * @param {String} pluginContext.transaction.number
15 * @param {String} pluginContext.transaction.id
16 * @param {String} pluginContext.transaction.poNum
17 * @param {String} pluginContext.transaction.tranType

```

```

18 * @param {Number} pluginContext.transaction.subsidiary
19 * @param {Object} pluginContext.sender
20 * @param {String} pluginContext.sender.id
21 * @param {String} pluginContext.sender.name
22 * @param {String} pluginContext.sender.email
23 * @param {Number} pluginContext.userId
24 *
25 * @returns {Object} result
26 * @returns {Boolean} result.success
27 * @returns {String} result.message
28 */
29     function send(pluginContext) {
30
31         var MSG_NO_EMAIL = translator.getString("ei.sending.sendernoemail");
32         var MSG_SENT_DETAILS = translator.getString("ei.sending.sentdetails");
33
34         var senderDetails = pluginContext.sender;
35         var customer = pluginContext.customer;
36         var transaction = pluginContext.transaction;
37         var recipientList = customer.recipients;
38         var result = {};
39         var parameters;
40         if (!senderDetails.email) {
41             parameters = {
42                 EMPLOYEENAME: senderDetails.name
43             };
44             stringFormatter.setString(MSG_NO_EMAIL);
45             stringFormatter.replaceParameters(parameters);
46             result = {
47                 SUCCESS: false,
48                 message: stringFormatter.toString()
49             };
50         } else {
51             var invoiceSendDetails = {
52                 number: transaction.number,
53                 poNumber: transaction.poNum,
54                 transactionType : transaction.type,
55                 eInvoiceContent: pluginContext.eInvoiceContent,
56                 attachmentFileIds: pluginContext.attachmentFileIds
57             };
58             notifier.notifyRecipient(senderDetails.id, recipientList, invoiceSendDetails);
59
60             parameters = {
61                 SENDER: senderDetails.email,
62                 RECIPIENTS: recipientList.join(", ")
63             };
64             stringFormatter.setString(MSG_SENT_DETAILS);
65             stringFormatter.replaceParameters(parameters);
66
67             result = {
68                 SUCCESS: true,
69                 message: stringFormatter.toString()
70             };
71         }
72     }
73
74     return result;
75 }
76
77     return {
78         send: send
79     };
80 });
81 });

```



**Important:** The sending method custom plug-in script must have the `@NSScriptType pluggintypeimpl`.

After creating the script for plug-in implementation, upload it to Customization > Plug-ins > Plug-in Implementations > New. The type of the custom plug-in implementation must be "Sending Plugin". For more information, see the help topic [Custom Plug-in Creation](#).

## Creating a Script for Sending E-Documents



**Important:** Sending methods must be created as custom plug-in implementations instead of scripts. You must recreate existing sending method scripts as new custom plug-in implementations of the type 'Sending Plugin'. For more information, see [Creating a Custom Plug-in Implementation for Sending E-Documents](#). The system will not support sending method scripts in NetSuite 2019.2.

An e-document sending method script must be a JavaScript file that is compatible with SuiteScript 2.0.

The script must return an object with the following function:

**send(scriptContext)**

<b>Description</b>	Executed when sending an e-document.
<b>Returns</b>	A result object.

### Parameters



**Note:** The scriptContext and result parameters are JavaScript objects.

Parameter	Type	Required / Optional	Description
scriptContext.scriptId	string	required	The ID of the document in the file cabinet
scriptContext.sendMethodId	string	required	The ID of the customer's or vendor's selected sending method
scriptContext.eInvoiceContent	string	required	The e-document content as a string  <span style="background-color: #e0f2ff; padding: 5px;"><b>i</b> <b>Note:</b> This content is the generated e-document.</span>
scriptContext.attachmentFileIds	array of strings	optional	The internal ID of the generated e document pdf file in file cabinet
scriptContext.customPluginImpId	string	required	The id (Field ID: scriptid) of the custom plug-in implementation of type sending plug-in set for the transaction.
scriptContext.transaction.id	string	required	The ID of the e-document transaction  <span style="background-color: #e0f2ff; padding: 5px;"><b>i</b> <b>Note:</b> The transaction id specified here is the ID of the document from which the e-document was generated.</span>
scriptContext.transaction.number	string	required	The document number of the e-document transaction
scriptContext.transaction.poNum	string	optional	The PO/check number of the e-document transaction
scriptContext.sender.id	string	required	The ID of the designated sender of the e-document
scriptContext.sender.name	string	required	The name of the designated sender of the e-document

Parameter	Type	Required / Optional	Description
scriptContext.sender.email	string	required	The email address of the designated sender of the e-document
scriptContext.userId	number	required	This internal ID of the current logged in user, it can be used wherever there is a requirement to refer to the current logged in user. For example, to update e-document audit trail by shared module API, userId can be used in owner property.

**i Note:** This sample script uses the define function, which is required for an entry point script (a script you attach to a script record and deploy). You must use the require function if you want to copy the script into the SuiteScript Debugger and test it. For more information, see the help topic [SuiteScript Debugger](#).

The following code is a sample script for sending e-documents.

```

1 /**
2  * @NApiVersion 2.x
3  * @NModuleScope Public
4 */
5 define(["N/record"], function(record, error) {
6     return {
7         /**
8          * send - Sample implementation: This will copy the e-document content to the document's
9          * Memo field
10         *
11         * @param {Object} pluginInContext
12         * @param {String} pluginInContext.scriptId
13         * @param {String} pluginInContext.sendMethodId
14         * @param {String} pluginInContext.eInvoiceContent
15         *
16         * @param {Object} pluginInContext.customer
17         * @param {String} pluginInContext.customer.id
18         * @param {String[]} pluginInContext.customer.recipients
19         *
20         * @param {Object} pluginInContext.transaction
21         * @param {String} pluginInContext.transaction.id
22         * @param {String} pluginInContext.transaction.number
23         * @param {String} pluginInContext.transaction.poNum
24         *
25         * @param {Object} pluginInContext.sender
26         * @param {String} pluginInContext.sender.id
27         * @param {String} pluginInContext.sender.name
28         * @param {String} pluginInContext.sender.email
29         *
30         *
31         * @returns {Object} result
32         * @returns {Boolean} result.success: determines
33         * @returns {String} result.message: a failure message
34     }
35     send: function(pluginInContext) {
36         var result = {
37             success: true,
38             message: "Success"
39         };
40         try {
41             var rec = record.load({
42                 type: record.Type.INVOICE,
43                 id: pluginInContext.transaction.id,
44             });
45             rec.setValue({
46                 fieldId: "memo",
47                 value: [
48                     "Script ID: " + pluginInContext.scriptId,
49                 ]
50             });
51         }
52     }
53 }

```

```

49     "Customer: " + plugInContext.customer.name,
50     "Transaction: " + plugInContext.transaction.number,
51     "Sender: " + plugInContext.sender.name,
52     "Recipients: " + plugInContext.customer.recipients.join("\n"),
53     "Content: " + plugInContext.eInvoiceContent].join("\n\n")
54   });
55   rec.save();
56 } catch (e) {
57   result.success = false;
58   result.message = "Failure";
59 }
60 return result;
61 }
62 );
63 });

```

The sending method gets network status updates and returns document's unique identifier as soon as the document is being sent to the network. The method must also return an additional key networkStatus in the send method implementation response.

```

1 networkStatus:{ 
2   referenceId:"e34fedf343ferterjty",
3   name:"SAT",
4   status: "Certification Processed",
5   updateDateTime: new Date("30/11/2022 3:32:14 PM")
6 }

```

These are the details of parameters added in the response:

Parameter	Type	Description
networkStatus.referenceId	String	It is a unique number generated by network API while certifying or sending e-document.
networkStatus.name	String	It is the name of a network. For instance, PEPPOL, SAT, ARIBA.
networkStatus.status	String	It displays a keyword representing one of the possible network statuses used by the network.
networkStatus.updateDateTime	String	It displays the time stamp (date and time) of the updated status.

The following code is a sample script of a sending method which returns a network status object.

```

1 {
2   success: true,
3   message:"Sent E-Document successfully",
4   networkStatus:{ 
5     referenceId:"pl335ead",
6     name:"PEPPOL",
7     status: "Processed",
8     updateDateTime: new Date("30/11/2022 3:32:14 PM")
9   },
10 }

```

The script returns an object with the function getStatus when the sending method is used to get status updates from an API.

#### getStatus(scriptContext)

<b>Description</b>	Executed to get the network status updates.
<b>Returns</b>	A result object.

## Parameters

Parameter	Type	Required / Optional	Description
scriptContext.transaction.id	string	required	The ID of the e-document transaction  <span style="border: 1px solid #ccc; padding: 5px;"> <b>Note:</b> The transaction id specified here is the ID of the document from which the e-document was generated.</span>
scriptContext.transaction.type	string	required	The type of the e-document transaction.

## getStatus method: Response Parameters

Parameter	Type	Required/Optional	Description
success	boolean	required	In case of successful communication with network status check API, it is set as true. In all failed communication cases, it is set as false.
message	string	required	Message displayed in the <b>E-document Audit Trail</b> .
networkStatus.status	string	required if success parameter is true	Keyword representing one of the possible network statuses used by the network.
networkStatus.updateDateTime	string	required if success parameter is true	Time stamp (date and time) for the status

The following code is a sample script of returned response for getStatus function.

```

1  {
2      success: true,
3      message: "Get the latest network status of e-document is successful",
4      networkStatus:{
5          status: "Processed",
6          updateDateTime: new Date("30/11/2022 3:32:14 PM")
7      },
8  }

```



**Important:** To prevent permission errors, make sure the sending method script has the **@NModuleScope Public** JSDoc .

## Creating an E-Document Sending Method Record

Make sure that you have created a sending method custom plug-in implementation before you create e-document sending method records. For more information, see [Creating a Custom Plug-in Implementation for Sending E-Documents](#).

### To create an e-document sending method record:

1. Go to Setup > E-Documents > E-Document Sending Methods > New.

2. In the **Name** field, enter a name for the e-document sending method.
3. In the **E-Document Package** field, select the e-document package you want to associate this sending method with. For more information, see [Creating E-Document Packages](#).
4. In the **E-Document Sending Method Plugin Implementation** field, select the e-document plug-in implementation for this method.
5. In the **Sending Channel** field, enter the sending channel to use for this method. For example, **email**, **SOAP** or **REST**.



**Note:** If the sending channel is **email** (case-sensitive), the system validates the email recipients upon saving the customer or vendor record and when sending the e-document.

6. In the **Transaction Type** field, select one or more transaction types for which this sending method will be used. To select multiple transaction types, press and hold the **Ctrl** key while selecting the transaction types.

The Transaction Type field only displays the transaction types applicable to or supported by outbound e-document sending, which include:

- Cash Refund
- Cash Sale
- Credit Memo
- Customer Payment
- Estimate
- Invoice
- Item Fulfillment
- Purchase Order
- Return Authorization
- Registered Custom Transaction Types
- Transfer Order
- Vendor Credit or Bill Credit

For more information, see [Transactions and Processes Supported by the Electronic Invoicing SuiteApp](#).



**Note:** The selected transaction types cannot be modified after the sending method has been used in a transaction. You must remove the e-document sending method from the transaction before you can modify this field.

7. In the **Subsidiary** field, select the subsidiaries that this sending method will be associated with. To select multiple subsidiaries, press and hold the **Ctrl** key while selecting the subsidiaries.

If only this sending method is associated with a subsidiary, the supported transactions of that subsidiary will display this sending method on the E-Document Template field on the E-Document subtab. For more information, see [Multi-subsidiary Support in the Outbound Process](#).

8. (Optional) If the sending method is to be used for e-document certification, check the **Sending Method for Certification** box. For more information, see [E-Document Certification in the Outbound Process](#).



**Note:** Only one certification sending method must be assigned to a combination of subsidiaries and transactions you select.

9. Click **Save**.

This sending method can now be selected on e-document package records.

If the **Inactive** box is checked, this record will not be available for selection.

## Editing an E-Document Sending Method Record

To edit an e-document sending method record, go to Setup > E-Documents > E-Document Sending Methods. Open the e-document sending method in edit mode and modify information as needed, then click **Save**.

Sending method records must reference sending method plug-ins, instead of scripts, from the **E-Document Sending Method Plugin Implementation** field. Existing sending method scripts must be recreated as new custom plug-in implementations of the type 'Sending Plugin'. For more information, see [Creating a Custom Plug-in Implementation for Sending E-Documents](#).



**Important:** The system will not support sending method scripts in NetSuite 2019.2, but until that time, you can still edit and use existing sending method scripts.



**Note:** The **Transaction Type** field in the e-document sending method is disabled if the e-document sending method has been assigned to one or more transaction records. To enable the field, you must remove the e-document sending method from all transactions.

## E-Document Email Custom Templates

An administrator can create custom templates to specify default subject and body text for e-document emails. Users can use these templates when sending e-documents by email to any entity selected in outbound e-document sending process.

After the administrator creates the e-document email templates, you can select them from the **E-Document Email Customization** dropdown list. You can find the dropdown list on the Subsidiary record or Company Information page in OneWorld accounts and on Company Information page in Single Instance accounts.

For more information, see [Creating an E-Document Email Custom Template](#) and [Selecting an E-Document Email Custom Template](#).

## Creating an E-Document Email Custom Template

An administrator can create one or more custom templates in the E-Document Email Customization Template page.

### To create an e-document email custom template:

1. Go to Setup > E-Documents > E-Document Email Customization> New.
2. In the **Name** field, enter a name for the e-document email template.
3. In the **Email Subject Text** field, enter text for the subject of the email. You can also enter additional information using the following placeholders as needed:
  - {TRANSTYPE} — Transaction type

- {TRANID} — Transaction ID
- {PONUM} — Transaction purchase order number
- {COMPANYNAME} — Company name

**i Note:** You must enter the placeholders in capital letters along with the braces {}. There should be no spaces inside the braces.

4. In the **Email Body Text** field, enter text for the body of the email. You can also enter additional information using the following placeholders as needed:

- {TRANSTYPE} — Transaction type
- {TRANID} — Transaction ID
- {PONUM} — Transaction purchase order number
- {COMPANYNAME} — Company name

**i Note:** You must enter the placeholders in capital letters along with the braces {}. There should be no spaces inside the braces.

5. Click **Save**.

The template will be available for selection on the Subsidiary record or Company Information page. You can use the template only if you select **Default** option in the **E-Document Sending Method Plugin Implementation** field on the E-Document Sending Method record.

**i Note:** If you do not create a template in the E-document Email Customization record, then a system generated default template is used in the e-document email. This email is sent to customers or vendors during the outbound e-document sending process.

## Selecting an E-Document Email Custom Template

After creating the custom templates for e-document email, an administrator can select the templates on the Subsidiary record or Company Information page. The selected template will be used in sending the e-documents by email to customers or vendors.

### To select an e-document email custom template on the Subsidiary record:

1. Go to Setup > Company > Subsidiaries.
2. Click **Edit** next to the required subsidiary.
3. From the **E-document Email Customization** dropdown list, select the required custom template.
4. Click **Save**.

### To select an e-document email custom template on the Company Information page:

1. Go to Setup > Company > Company Information.
2. From the **E-document Email Customization** dropdown list, select the required custom template.
3. Click **Save**.



**Note:** E-document email custom templates can be selected on both Subsidiary record and Company Information page for the following combinations:

- If you select a different template on both Subsidiary record and Company Information page, then the template you selected on the Subsidiary record is used.
- If you select a template on the Company Information page and not on the Subsidiary record, then the same template is applied on transactions of all the subsidiaries.
- If you select a template on the Subsidiary record for a transaction, then the selected template is used for all transactions that are created for this subsidiary. It overrides the template selected on the Company Information page.



**Note:** If you do not select a template either on Subsidiary record or Company Information page, then a system generated default template is used in the e-document email. This email is sent to customers or vendors during the outbound e-document sending process.

## Selecting a Designated E-Document Sender

An administrator can select an employee whose name and email address should appear as the sender of e-documents sent by your company or subsidiary. This task is optional. If there is no designated sender, the system uses the name and email address of the user who sent the e-document as the sender.

### To select a designated e-document sender:

1. Go to Setup > Company > Company Information.  
If you have a OneWorld account, go to Setup > Company > Classifications > Subsidiaries, and click the edit link to open a subsidiary record.
2. In the **E-Document Sender** field, select an employee.
3. Click **Save**.

## Setting Up Custom Roles to Send E-Documents

The following standard roles can send and re-send e-documents in bulk:

- A/P Clerk
- A/R Clerk
- Accountant
- Administrator
- Bookkeeper
- CEO
- CFO
- Custom Accountant
- Retail Clerk

An administrator can give custom roles access to the bulk sending feature. The following conditions must be met to enable a custom role to send e-documents in bulk:

- The custom role has been added to the script audience for the **Outbound E-Invoicing Form SU** script.
- The custom role has Edit permissions for the transaction types for which your company generates and sends e-documents.
- The custom role has Edit permission for the Add E-Document Sending Batch custom record.

### To grant custom roles access and permissions to send e-documents:

1. Add the custom roles to the script audience by performing the following steps:
  - a. Go to Customization > Scripting > Script Deployments.
  - b. Click the Edit link of the script named **Outbound E-Invoicing Form SU**.
  - c. On the **Audience** subtab, in the **Roles** field, select the custom roles that you want to give access to.
  - d. Click **Save**.
2. Add the required permissions to custom roles by performing the following steps:
  - a. Go to Setup > Users/Roles > Manage Roles.
  - b. Click the Edit link of the custom role that you want to modify.
  - c. Add permission to edit transaction types:
    - i. On the **Permissions** subtab, go to the **Transactions** subtab.
    - ii. In the **Permission** column, on a blank row, select a transaction type.
    - iii. In the **Level** column, select **Edit**.
    - iv. Click **Add** to add the row.
    - v. Add other transaction types that the custom role must have access to.
  - d. Add permission to edit the custom record used for bulk sending:
    - i. On the **Permissions** subtab, go to the **Custom Record** subtab.
    - ii. In the **Record** column, on a blank row, select **Add E-Document Sending Batch**.
    - iii. In the **Level** column, select **Edit**.
    - iv. In the **Restrict** column, select **Editing Only**.
    - v. Click **Add** to add the row.
  - e. Click **Save** to save the changes to the custom role.

## Customizing Roles to Restrict E-Document Generation or Sending

By default, the Generate E-Document and Send E-Document buttons are visible on all outbound transaction types that E-Invoicing SuiteApp supports, regardless of the level of permission the user has on the transaction record.

The administrator can customize standard and custom roles to restrict their permission to manually generate and send outbound e-documents.

### To restrict the permission of standard and custom roles to generate and send outbound e-documents:

1. Go to Setup > Users/Roles > User Management > Manage Roles.
2. Click the Customise or Edit link of the role that you want to restrict permission to generate or send outbound e-documents.
3. Clear the **Allow Manual Generation of E-Document** box or **Allow Manual Sending of E-Document** box.
4. If you want to allow this role to send e-documents for certification, check the **Allow Sending of E-Document for Certification** box; otherwise, clear the box.
5. Click **Save**.

The Generate E-Document button or Send E-Document button will not be available on transaction records viewed by role. If you did not grant the role permission to send e-documents for certification, the Certify E-document button will not be available on transactions viewed by role.

## Deploying the Bulk Generation Script for E-Documents

The Electronic Invoicing SuiteApp enables an administrator to deploy a script to generate multiple e-documents at a specific time, instead of generating e-documents for each transaction manually.

By default, the script's status is set to **Not Scheduled**. The administrator can set a schedule for the script to generate e-documents in bulk.

**Note:** Only an administrator can view, edit, and run the **Generate E-Document Content** script included in the SuiteApp.

The script generates e-documents for transactions that have the following e-document statuses:

- **For Generation** – The script generates e-documents for transaction records that have e-document templates.
- **Generation Failed** – The script regenerates e-documents for transaction records with assigned e-document templates that previously encountered errors during generation. Errors must be fixed before e-documents can be regenerated. For more information, see [Outbound E-Document Generation Errors](#).

After the system generates e-documents, it updates the value in the **E-Document Status** field on the **E-Document** subtab of each transaction record. The field shows one of the following values:

- **Ready for Sending** – This status means the e-document was generated successfully and can be sent to the customer or vendor.
- **Generation Failed** – This status means the e-document was not generated. Information about generation errors are shown in the **Details** column of the **E-Document Audit Trail** subtab. The user must first fix the errors before the e-document can be regenerated for the transaction record.

**Note:** If e-document generation fails, NetSuite does not send an email notification to the administrator. Instead, it sends an email notification containing the error details to the user who created the transaction record. The user must fix the errors before an e-document can be regenerated for the transaction record.

For more information about generation errors, see [Outbound E-Document Generation Errors](#).

See also [Generating and Regenerating E-Documents in Bulk](#).

### To deploy the bulk generation script for e-documents:

1. Go to Customization > Scripting > Script Deployments.
2. Click the Edit link of the **Generate E-Dокумент Content MR** script.
3. Clear the **Deployed** box if you do not want to deploy the script yet. A script will not run in NetSuite until the **Deployed** box is selected.
4. Select a status in the **Status** field:
  - Testing
  - Not Scheduled
  - Scheduled

For more information, see the help topic [Setting Script Deployment Status](#).

5. Choose an event type on the **Schedule** subtab:
  - **Single event** – The script generates e-documents only one time.
  - **Daily event** – Enter the interval between days if this event is every day or every few days, or select every weekday if this event is every day except Saturdays and Sundays.  
Enter 1 as the interval if this event is every day, for example, or enter 2 if the event is every other day.
  - **Weekly event** – Enter the interval between weeks, and select the day of the week this event repeats on.
  - **Monthly Event** – If you want to generate e-documents on the same day of every month or every few months, enter the date the generation of e-documents repeats, and select the interval between months.  
If you want to generate e-documents on the same day of the week every month or every few months, select the week, the day of the week, and enter the interval between months.
  - **Yearly Event** – If you want to generate e-documents one time every year, select the month and day to generate e-documents, or select the week, day, and month.
  - **Start Date** – You must enter the date you want to schedule the generation of e-documents for.
  - **Start Time** – Enter the time you want the generation of e-documents to start.
  - **Repeat** – Select how often you want the script to generate e-documents. On the day the script is scheduled to run, e-documents are generated at the specified **Start Time** and repeat every *n* hours until midnight.
  - **End By** – Set the date the script stops generating e-documents.  
If you are scheduling a single event, e-documents will be generated on the date entered in the **Start Date** field.  
If you are scheduling a repeat event, e-documents will be generated according to the schedule you set starting on the date entered in the **Start Date** field and ending on the date entered in the **End By** field.
  - **No End Date** – Check this box if you want to repeat the schedule indefinitely.
6. Click **Save and Execute** to generate e-documents immediately. Click **Save** to generate e-documents according to the schedule you chose.

## Deploying the Script for Scheduled Sending of E-Documents

Like the script for generating e-documents in bulk, the Electronic Invoicing SuiteApp also has a script for automatic scheduled sending of outbound e-documents. The script sends only outbound e-documents whose status is Ready for Sending.



**Note:** By default, only the administrator can view, edit, and run the Automatic Send E-Document MR script included in the SuiteApp, because the administrator has the permission to view, edit and run Script Deployment records.

By default, the script's status is set to Not Scheduled. The administrator can set a schedule for the script to send outbound e-documents.

### To deploy the script for scheduled sending of e-documents:

1. Go to Customization > Scripting > Script Deployments
2. Click the Edit link of the **Automatic Send E-Document MR** script.
3. Clear the **Deployed** box if you do not want to deploy the script yet. A script will not run in NetSuite until the Deployed box is selected.
4. Select a status in the **Status** field:
  - Testing
  - Not Scheduled
  - Scheduled
 For more information, see the help topic [Setting Script Deployment Status](#).
5. Choose an event type on the **Schedule** subtab:
  - **Single Event** – The script sends outbound e-documents only one time.
  - **Daily Event** – Enter the interval between days if this event should occur every day or every few days, or select every weekday if this event should occur every day except Saturdays and Sundays. For example, enter 1 as the interval if this event should occur every day, or enter 2 if the event should occur every other day.
  - **Weekly Event** – Enter the interval between weeks, and select the day of the week this event should be repeated.
  - **Monthly Event** – If you want to send e-documents on the same day of every month or every few months, enter the date when you want e-document sending to repeat and then select the interval between months. If you want to send e-documents on the same day of the week every month or every few months, select the week, the day of the week, and enter the interval between months.
  - **Yearly Event** – If you want to send e-documents one time every year, select the month and day, or select the week, day, and month.
  - **Start Date** – You must enter the date when you want to schedule the sending of e-documents .
  - **Start Time** – Enter the time when you want the sending of e-documents to start.
  - **Repeat** – Select how often you want the script to send e-documents. On the day the script is scheduled to run, e-documents will be sent at the specified Start Time and the process repeats every n hours until midnight.
  - **End By** – Set the date when the script should stop sending e-documents.
 

If you are scheduling a single event, e-documents will be sent on the date entered in the Start Date field.

If you are scheduling a repeat event, e-documents will be sent according to the schedule you set, starting from the date entered in the Start Date field and ending on the date entered in the End By field.

    - **No End Date** – Check this box if you want to repeat the schedule indefinitely.
  - 6. Click **Save and Execute** to send e-documents immediately. Click **Save** to send e-documents according to the schedule you chose.

The script automatically searches for outbound e-documents whose status are Ready for Sending, and then sends them. After the outbound e-documents are sent, their status is changed to either Sent or Sending Failed. Details of any error during sending are recorded in the E-Document Audit Trail. Error notification is sent to the Recipient of E-Document Notifications. If Recipient of E-Document Notifications has not been defined yet, the notifications are sent to active administrators. The notification has a CSV file attached containing details of the error.

## Scheduled Script for Getting Network Status of E-Documents

The Electronic Invoicing SuiteApp provides a script to automatically get network status for bulk transactions. The script has an additional parameter **Transactions Filtering Plugin ID** (id: custscript\_ei\_filtertransid). This parameter populates the implementation id of the Transactions Filtering Plugin ID field, which contains the logic for filtering transactions. By default, the script selects transactions if the following conditions are met:

1. The **Network Reference Id** must be populated.
2. The transactions must be created the current or previous day.

By default, the script's status is set to **Scheduled** and it runs every 2 hours. The administrator can set a different schedule as per their requirements.

## Implementing Transaction Filtering Plug-in

The Electronic Invoicing SuiteApp has added a custom plug-in type called the Transaction Filtering Plugin (id: customscript\_transactionfilterplugin). This plug-in contains the following method which the administrator must implement:

**getFilteredTransactions**: This method filters the transactions for which the **Automatic Get Network Status MR** script updates the statuses. The method returns an array of objects. The objects have two keys that contain the details of transactions.

The details of objects in the returned array are given in the following table:

Parameter	Type	Description
id	String	It is the unique number to identify the transaction.
recordType	String	It specifies the type of transaction.

In the default implementation, the **getFilteredTransactions** method returns the list of transactions where:

- The **Network Reference Id** is populated.
- The transactions are created the current or previous day.

```

1 /**
2  * @NApiVersion 2.x
3  * @NModuleScope Public
4  * @NScriptType plugintypeimpl
5  */
6 define(["N/search", "N/format"], function (search, format) {
7 /**

```

```

8  * getFilteredTransactions - This function is the entry point of our plugin script
9  *
10 * @returns {Array} transactions
11 * @returns {Object} transaction
12 * @returns {String} transaction.recordType
13 * @returns {String} transaction.id
14 */
15 function getFilteredTransactions() {
16     var transactions = [];
17     var yesterdayDate = new Date();
18     yesterdayDate.setDate(yesterdayDate.getDate() - 1);
19     var transSearch = search.create({
20         type: search.Type.TRANSACTION,
21
22         filters: [
23             [
24                 "trandate",
25                 search.Operator.NOTBEFORE,
26                 format.format({
27                     value: yesterdayDate,
28                     type: format.Type.DATE,
29                 }),
30             ],
31             "and",
32             ["custbody_ei_network_id", search.Operator.ISNOTEMPTY, ""],
33             "and",
34             ["mainline", "is", "T"],
35         ],
36     });
37     var PAGE_SIZE = 1000;
38
39     var searchResult = transSearch.runPaged({ pageSize: PAGE_SIZE });
40     searchResult.pageRanges.forEach(function (pageRange) {
41         var currPage = searchResult.fetch({ index: pageRange.index });
42
43         currPage.data.forEach(function (result) {
44             transactions.push({
45                 recordType: result.recordType,
46                 id: result.id,
47             });
48         });
49     });
50     return transactions;
51 }
52
53 return {
54     getFilteredTransactions: getFilteredTransactions,
55 };
56 });

```

The administrator must add a plug-in implementation for the Transaction Filtering Plugin type. You must provide the id for implementation in the **Automatic Get Network Status MR** script's parameter **Transaction Filtering Plugin ID**.

## Adding Transaction Filtering Plugin ID

The administrator must add a plug-in implementation for the Transaction Filtering Plugin type.

### To add Transaction Filtering Plugin ID for bulk network status:

1. Go to Customization > Scripting > Script Deployments.
2. Click **Edit** on the **Automatic Get Network Status MR** script.
3. Go to **Parameters** subtab.
4. In the **Transactions Filtering Plugin ID** field, enter the relevant ID.
5. Click **Save**.

## Updating E-Document Certification Statuses

E-document certification statuses are not automatically updated by the Electronic Invoicing SuiteApp. E-document certification statuses can be updated through the certification sending method implementation. If your company or organization is implementing its own e-document certification sending, you can use the format of the following sample script for capturing sending method return, which can be used to update the e-document certification statuses and log corresponding messages in the E-document Audit Trail.



**Note:** This sample script uses the `define` function, which is required for an entry point script (a script you attach to a script record and deploy). You must use the `require` function if you want to copy the script into the SuiteScript Debugger and test it. For more information, see the help topic [SuiteScript Debugger](#).

The following code is a sample script to implement e-document certification status updates.

```

1  /**
2   * Copyright (c) 2017, Oracle and/or its affiliates.
3   *
4   * @NApiVersion 2.x
5   * @NScriptType plugintypeimpl
6   * @NModuleScope public
7   */
8  define([], function() {
9      /**
10      * send - This function is the entry point of our plugin script
11      * @param {Object} pluginContext
12      * @param {String} pluginContext.scriptId
13      * @param {String} pluginContext.sendMethodId
14      * @param {String} pluginContext.eInvoiceContent
15      * @param {Object} pluginContext.customer
16      * @param {String} pluginContext.customer.id
17      * @param {Array} pluginContext.customer.recipients
18      * @param {Object} pluginContext.transaction
19      * @param {String} pluginContext.transaction.number
20      * @param {String} pluginContext.transaction.id
21      * @param {String} pluginContext.transaction.poNum
22      * @param {Object} pluginContext.sender
23      * @param {String} pluginContext.sender.id
24      * @param {String} pluginContext.sender.name
25      * @param {String} pluginContext.sender.email
26      * @param {Array} pluginContext.attachmentFileIds
27      *
28      * @returns {Object} result
29      * @returns {Boolean} result.success
30      * @returns {String} result.message
31      */
32      return {
33          send: function(pluginContext) {
34              var result = {
35                  success: true,
36                  message: '',
37                  eiStatus: {
38                      "transactionId": pluginContext.transaction.id,
39                      "transactionType": pluginContext.transaction.tranType,
40                      "entity": customer.id,
41                      "eDocStatus": "3",
42                      "eventType": "3",
43                      "details": "The e-Doc successfully certified and is ready for sending.",
44                      "owner": pluginContext.sender.id,
45                      "isUpdateFields": "true",
46                      "extraFieldsForUpdate": {},
47                      "bundleId": "",
48                      "bundleName": ""
49                  }
50              }
51          }
52      }
53  })

```

```

49     }
50   );
51   return result;
52 }
53 });
54 });

```

If e-document certification is successful, the script will change the e-document status of the transaction to Ready for Sending and add the following details in the E-Document Audit Trail:

The e-document was successfully certified and is ready for sending.

The following table lists the parameters that the object eiStatus in the script takes up.

Parameter	Type	Description	Values
transactionId	String	ID of transaction record	Required
transactionType	String	Transaction type such as INVOICE, VENDOR_BILL, ITEM_FULFILLMENT, and others	Required
entity	String	ID of entity record; for example, Customer, Vendor, and others.	Required
eDocStatus	String	E-document Status  Use numbers for status types. The following numbers can be used: <ul style="list-style-type: none"><li>■ READY_FOR_SENDING : 3</li><li>■ CERTIFICATION_IN_PROGRESS : 20</li><li>■ CERTIFICATION_DATA_ERROR : 21</li><li>■ CERTIFICATION_FAILED : 22</li></ul>	Required
eventType	String	E-document Audit Trail Event  Use numbers for event types. The following numbers can be used: <ul style="list-style-type: none"><li>■ READY_FOR_SENDING : 3</li><li>■ CERTIFICATION_IN_PROGRESS : 20</li><li>■ CERTIFICATION_DATA_ERROR : 21</li><li>■ CERTIFICATION_FAILED : 22</li></ul>	Required
details	String	Details	Required
owner	String	ID of user or Employee	Required
isUpdateFields	String	Flag; if TRUE, the EI Status is updated, fields are provided in extraFieldsForUpdate field and an audit trail is created. Otherwise, an audit trail is created.	Required
extraFieldsForUpdate	JSON	A JSON object containing the key-value pair for the transaction body fields, where key is the field ID and value is the field value. It can be used to update any transaction body field required by a Country SuiteApp.	Required (can be an empty object)
bundleId	String	BundleId of the requester	Required
bundleName	String	BundleName of the requester	Required

The parameter details will be included in the E-Document Audit Trail.

## Processing E-Documents Automatically for Individual Transactions

A user with required permissions can process e-documents automatically for individual transactions. All the e-document processes like generate, certify and send can be automated by clicking a single **Process E-Document** button. Before processing the e-documents automatically for a transaction, the transaction's e-document template, sending method and certification must be set up.

 **Tip:** Processing e-documents automatically rather than manually can be useful in reducing the time and effort, especially when it is used along with auto-selection of template and sending method. For more information, see **Template and Sending Method Auto-selection** in [Multi-subsidiary Support in the Outbound Process](#)

The automatic processing of e-document feature is disabled by default. To enable this feature, an administrator must select an **E-Document Automation Type** from the dropdown available in the Automatic E-Invoicing tab on the E-Document Preferences page.

### To process e-documents automatically for individual transactions:

1. Go to Setup > E-Documents > E-Document Preferences.
2. In the E-Document Preferences page, select one of the following options from the **E-Document Automation Type** dropdown present in the Automatic E-Invoicing tab for each subsidiary.
  - Generate, Certify, Send – The **Process E-Document** button becomes available on the transactions. It enables you to perform e-document's generate, certify and send processes by clicking the **Process E-Document** button.

 **Note:** E-document certification process is not performed if a valid certification sending method is not defined.

- Generate, Certify – The **Process E-Document** button becomes available on the transactions. It enables you to perform e-document's generate and certify processes by clicking the **Process E-Document** button.
- Certify, Send – The **Process E-Document** button becomes available on the transactions. It enables you to perform e-document's certify and send processes by clicking the **Process E-Document** button.
- Disable – This option is selected by default. The **Process E-Document** button is not available on the transactions. The e-document's generate, certify and send processes must be performed manually using the **Generate E-Document**, **Certify E-Document** and **Send E-Document** buttons. For more information, see [Overview of Outbound E-Document Process](#).

Only a user with administrator role can configure the e-document processes for automation. After this you can use the **Process E-Document** button on individual transactions to process the e-documents automatically for a subsidiary.

## Electronic Invoicing Inbound Email Capture

Inbound electronic invoicing supports the capturing of vendor invoices in XML format sent through email. This means your vendors can send you invoices in XML through email, which you can directly view as inbound e-document records.

When the system receives email with an XML invoice attached, it first determines the vendor based on either the vendors' email domain or sender email address. If the vendor is determined, the system

automatically captures the email with the XML invoice and then stores the XML content as an inbound e-document record. Inbound e-document records from email capture are automatically added to the Inbound E-Documents List with the e-document status indicated.

If the vendor is not determined from the email e-document, or other errors are encountered, the system sends a notification to the Recipient of E-Document Notifications, informing that user of the problems encountered with the received email e-document. If the Recipient of E-Document Notifications is not set up, the notifications are sent to all active administrators.

## Setting Up Inbound Email Capture

Set up the Inbound Email Capture feature in your account by performing the following tasks:

- Enable the Inbound Email Capture PIug-in.
- Set up Vendor E-Document Email Sender.

## Enabling Inbound Email Capture Plug-in

The Electronic Invoicing SuiteApp comes with an inbound email capture plug-in that you must first enable.

### To enable the inbound email capture plug-in:

1. Go to Customizations > Plug-ins > Manage Plug-ins.
2. On the Manage Plug-In Implementations page, check the **Inbound Email Capture PI** box.
3. Click **Save**.

Take note of the email address indicated, with the domain emails.netsuite.com. This is the email address where your vendors should send their XML invoices to. This email address is different for every account. It is that you set up an alias for this email address in your email facility, so that your vendors can easily record or remember the email address. For more information about setting up an email alias, see the help topic [Create an Email Alias and Set Up Forwarding](#)

To view details of events or errors associated with the inbound email capture plug-in, go to Customization > Plug-ins > Plug-in Implementations. Click the View link of the Inbound Email Capture PI and then on the Plug-In Implementation page, click the Execution Log subtab. For more information about the email capture plug-in, see the help topics [Email Capture Plug-in Overview](#) and [Administering an Email Capture Plug-in Implementation](#)

Aside from the plug-in execution log, the system also sends inbound e-document email capture notification messages to your designated recipient, informing that user of the details of any event or error associated with the inbound email capture plug-in.

## Setting Up Vendor E-Document Email Sender

After enabling the e-document email capture plug-in, you must identify or assign the Vendor E-Document Sender in vendor records.

### To set up the vendor e-document email sender

1. Edit a vendor record and click the **E-Document** subtab.
2. You can enter the email domain associated with the vendor in the **Sender Email Domain** field. If you enter an email domain in this field, the system will use the email domain to determine the vendor of the captured email e-documents.



**Note:** The vendor email domain you enter must be unique for each vendor. No vendors must have the same email domain.

3. Enter the email address of the vendor's designated user in the **Vendor E-Document Email Sender** sublist. Add as many vendor sender email addresses as necessary. The system will use the email addresses you entered to determine the vendor of captured email e-documents.



**Note:** Only designated vendor e-document email senders can create inbound e-documents.

4. If you want the system to use the email addresses you entered in **Vendor E-Document Email Sender** instead of the domain, for determining the vendor of captured e-document email, you can check the **Use Sender Email List** box. If you check the box, you must enter email addresses in the **Vendor E-Document Email Sender** field, which becomes required and replaces the domain as the basis for determining the vendor.
5. Click **Save**.

If you do not enter an email domain or check the Use Email Sender List box, the system will not be able to determine the vendor of the email e-document and an inbound e-document will not be created. In this case, the system will send a notification to the recipient you designated, informing that user that e-document email was received but no vendor was associated with the e-document.

To apply these settings to your existing vendor records, you can use CSV Import.

#### To update vendor records to apply or update vendor e-document email sender or recipient:

1. Go to Setup > Import/Export > Import Tasks > Import CSV Records.
2. In the **Import Type** field, select Custom Records.
3. In the **Record Type** field, select Vendor E-Document Email Sender or Recipient.
4. Click **Select**, and select the CSV file to upload.
5. Click **Next**.
6. On Import Options, select the appropriate Data Handling option. Click **Next**.
7. On Field Mapping, if you are updating the vendor e-document email sender, make sure that email and vendor are mapped to appropriate values. If you are updating the vendor e-document email recipient, make sure that the contact and vendor are mapped to appropriate values. Click **Next**.
8. In the **Import Map Name**, enter a unique name. Click **Save & Run**.

For more information about performing CSV Import, see the help topics [CSV Imports Overview](#) and [Importing CSV Files with the Import Assistant](#).

## Using SOAP Web Services for Inbound Processing

Inbound e-document processing supports SOAP web services, which offer advantages in interoperability, flexibility and security in sending information over the internet. By enabling the web service features in NetSuite, you can receive XML files sent through web service and then capture information in XML for subsequent creation of e-documents. You can set up vendor or employee records to authorize them as web service senders. As authorized web service senders, these entities can send single or multiple XML files in one web service request.

### Setting Up Inbound E-Document SOAP Web Services

1. Go to Setup > Company > Setup Tasks > Enable Features.

- a. On the **SuiteCloud** subtab, under SuiteTalk (Web Services), check **SOAP Web Services**.
- b. Under Manage Authentication, check **Token-Based Authentication**.
- c. On the **Web Presence** subtab, under Access, check **Vendor Access**.
- d. Click **Save**.

For more information, see the help topics [SuiteCloud Features](#) and [Commerce Features](#).

2. Create an integration record to generate authentication keys. Go to Setup > Integration > Integration Management > Manage Integrations > New. For more information, see the help topic [Creating an Integration Record](#).
  - a. Enter or select values in the **Name** and **Description** fields. Retain the default value in the **State** field, which is **Enabled**.
  - b. On the **Authentication** subtab, check **Token-based Authentication**. The **User Credentials** box is checked by default, retain this.
  - c. Click **Save**.

The Consumer Key and Consumer Secret are generated and displayed on the integration record. Copy the Consumer Key and Consumer Secret to send to the vendors or parties that you want to authorize as web service senders.
3. If you upgraded the Electronic Invoicing SuiteApp from a previous version, make sure that the custom role, Inbound E-Document Web Service Role, is included in the target audience of the E-Document Inbound UE script deployment. But if the SuiteApp is a new installation, you do not have to perform these steps.
  - a. Go to Customization > Scripting > Scripts.
  - b. Click the Deployments link of the E-Document Inbound UE script and edit the record.
  - c. On the **Audience** subtab, under Roles, select **Inbound E-Document Web Service Role** to add it to the list of roles.
  - d. Click **Save**.
4. Assign the Inbound e-Document Web Services Role to the vendor or employee whom you want to authorize to be the sender of XML files through web service.
  - a. Edit the vendor record that you authorized as web service sender.
  - b. On the vendor record **E-Document** subtab, enter appropriate values in the **Web Service ID** and **Web Service Sender** fields.

For the Web Service ID, you can assign any unique identifier of the vendor, like the tax ID. For the Web Service Sender, you can choose from the list a vendor or an employee, whom a vendor designated to send e-documents of its behalf.

  - c. Click **Save**.

For more information, see the help topics [Assigning Roles to an Employee](#) and step 6 under the Access section in [Creating a Vendor Record](#).
5. Create access tokens for each vendor or employee that you authorized as web service sender.
  - a. Go to Setup > Users/Roles > User Management > Access Tokens > New.
  - b. In the **Application Name** field, select the integration record that you created in step 2, which will be associated with the access token you will create.
  - c. In the **User** field, select the vendor or employee whom you assigned the Inbound e-Document Web Service Role.
  - d. In the **Role** field, select **Inbound e-Document Web Service Role**. The Token Name is automatically assigned a value.
  - e. Click **Save**.

The Token ID and Token Secret are generated and displayed on the Access Token page. Copy the values for the Token ID and Token Secret before going to another page.



**Important:** For security reasons, the values for the Token ID and Token Secret are only displayed on the initial setup page. They cannot be retrieved from the system. If you did not take note or copied the Token ID and Token Secret values, you will need to create a new token.

For more information, see the help topic [Getting Started with Token-based Authentication](#).

6. Send or convey the following web service connection and authentication details to authorized web service senders:

- Consumer Key
- Consumer Secret
- Token ID
- Token Secret
- Account ID

Go to Setup > Company Information. In the **Account ID** field, obtain the value that you need to send.

- Inbound E-Document Web Service RESTlet External URL

Go to Customization > Scripting > Scripts. View the Inbound E-Document Web Service RL, RESTlet. On the **Deployments** subtab, click **Inbound E-Document Web Service RL**. The External URL is displayed.

- The vendor Web Service ID.

After setting up SOAP web services for inbound processing, you are ready to receive XML e-documents from SOAP web services. For more information, see [Receiving E-Document XML Files from Web Service](#).

## Web Service Sender Setup Tasks

The authorized web service sender can either be a vendor or an employee. Web service senders must also perform setup tasks before they can send e-document files through web service.

First, a web service sender must obtain the web service connection and authentication details from you. Then, they must meet the following requirements in constructing the web service request.

Requirements	Web Service Details
OAuth 1.0 Authentication	Use the authentication details from the NetSuite user: <ul style="list-style-type: none"> <li>■ Consumer Key</li> <li>■ Consumer Secret</li> <li>■ Token ID</li> <li>■ Token Secret</li> <li>■ Realm (Account ID)</li> </ul>
OAuth Data	<ul style="list-style-type: none"> <li>■ Consumer Key = generated from Netsuite</li> <li>■ Signature Method = HMAC-SHA1 or HMAC-SHA256</li> </ul>

Requirements	Web Service Details
	<ul style="list-style-type: none"> <li>■ OAuth Nonce = unique string that is generated</li> <li>■ Timestamp = timestamp in seconds</li> <li>■ OAuth Version = 1.0</li> <li>■ OAuth Token = generated from Netsuite</li> <li>■ OAuth Signature = Key is secret key, Base String. For more information, see the help topics <a href="#">Using TBA for RESTlet Authentication (OAuth)</a> and <a href="#">Step One Obtain An Unauthorized Request Token</a>.</li> </ul>
Content-type header	Application/json
Request Method	POST
Web Service Request Body	<p>JSON Object</p> <p><b>Sample code for sending a single XML file in one request:</b></p> <pre data-bbox="523 813 1010 939">1  { 2    "identifier": "vendor1", 3    "fileName": "vendor1.xml", 4    "content": "&lt;XML&gt;content&lt;/XML&gt;" 5 }</pre> <p>If multiple XML files must be sent in one web service request, the format must be an of JSON objects.</p> <p><b>Sample code for sending multiple XML files in one request:</b></p> <pre data-bbox="523 1151 1010 1467">1  [ 2    { 3      "identifier": "vendor1", 4      "fileName": "vendor1-1.xml", 5      "content": "&lt;XML&gt;content&lt;/XML&gt;" 6    }, 7    { 8      "identifier": "vendor1", 9      "fileName": "vendor1-2.xml", 10     "content": "&lt;XML&gt;content&lt;/XML&gt;" 11   }, 12   { 13     "identifier": "vendor2", 14     "fileName": "vendor2.xml", 15     "content": "&lt;XML&gt;content&lt;/XML&gt;" 16   } ]</pre> <div data-bbox="523 1488 1010 1636" style="border: 1px solid #f0e68c; padding: 10px; background-color: #fff;">  <b>Important:</b> There is a limit of 10MB per string used as RESTlet input or output. For more information, see the help topic <a href="#">RESTlet Governance</a>. </div>

## Setting Up Custom Roles that can Convert Inbound E-Documents

An administrator can create custom roles or customize standard roles, to grant them the permission to perform conversion of inbound e-documents into transaction records.

## To grant a role the permission to convert inbound e-documents into transaction records:

1. Set up the permission. Go to Setup > Users/Roles > User Management > Manage Roles. Select the role you want to grant permission to and click its Customize link. On the Role page, do the following:
  - On the **Custom Record** subtab, do the following:
    - In the Inbound E-Documents row, click the entry in the Level column. Select **Edit**.
    - In the E-Document Audit Trail row, click the entry in the Level column. Select **Edit**.
  - On the **Transactions** subtab, do the following:
    - In the Bills row, click the entry in the Level column. Select **Full**.
    - In the Purchase Order row, click the entry in the Level column. Select **View**.
  - On the **Lists** subtab, do the following:
    - In the Vendors row, click the entry in the Level column. Select **View**.
    - In the Perform Search row, click the entry in the Level column. Select **View**.
    - In the Documents and Files row, click the entry in the Level column. Select **View**.
  - Click **Save**.
2. Make the Convert button visible to the roles with permission to perform e-document conversion. To do so, the inbound e-document script deployment must be enabled. Go to Customization > Scripting > Script Deployments. On the Script Deployments page, do the following:
  - Click the Edit link of the script customdeploy\_edoc\_ue\_inbound, **E-Document Inbound UE**.
  - On the **Audience** subtab, select the roles that the script will execute for. To select multiple roles, press and hold down the Ctrl key while selecting the roles.
  - Click **Save**.

For more information about associating a role with script deployment, see the help topic [Executing Scripts Using a Specific Role](#).
3. Assign the role with the permission to perform inbound conversion, to employees. See the help topic [Assigning Roles to an Employee](#).

For more information, see the help topics [Customizing or Creating NetSuite Roles](#) and [Setting Permissions for a Custom Record Type](#).

## Inbound Validation Plug-ins

The Electronic Invoicing SuiteApp provides support for plug-ins that can perform inbound e-document validation according to the requirements of tax agencies or standards regulatory bodies.

You can create or customize a plug-in that communicates with external sites or systems of a third party, such as a tax agency, which validates e-documents and then returns the results to the plug-in. Or you can create or customize a plug-in that itself performs the required e-document validation.

An inbound e-document validation plug-in must be set up first and integrated with a vendor e-document package. With the validation plug-in set up, every time a user converts inbound e-documents to vendor bills, the plug-in automatically triggers the validation process that is transparent to the user. Depending on your implementation of the plug-in, it can either send the XML e-documents to a validating third party or validate them locally.

The result of the validation process can be viewed from the status of the inbound e-document. If the inbound e-document passed validation, then parsing and conversion will proceed. However, if the

inbound e-document is invalid, conversion will not proceed and the status of the inbound e-document will be set to Conversion Failed. The audit trail of the inbound e-document will indicate the details of a failed local conversion process or information from a validating third party.

## Creating a Custom Plug-in for Inbound E-Document Validation

A custom plug-in implementation for validating inbound e-documents must be created so that it will be available in the vendor e-document package.

A validation custom plug-in implementation script is included in Electronic Invoicing SuiteApp. Use the sample script as a reference or a template for creating your own custom validation plug-in.

The validation plug-in must be a JavaScript file that is compatible with SuiteScript 2.0.

To open the sample validation plug-in script, go to Documents > Files > File cabinet > SuiteBundles > Bundle 436209 > src > comp > pl > pl\_inbound\_validation\_sample.js. Open the script in a programming editor to view the script.



**Note:** This sample script uses the define function, which is required for an entry point script (a script you attach to a script record and deploy). You must use the require function if you want to copy the script into the SuiteScript Debugger and test it. For more information, see the help topic [SuiteScript Debugger](#) SuiteScript Debugger.

The following code is a sample validation plug-in script.

```

1  /**
2   * Copyright (c) 2017, Oracle and/or its affiliates.
3   *
4   * @NModuleScope public
5   * @NApiVersion 2.x
6   * @NScriptType pluginTypeimpl
7   */
8
9  define([], function() {
10    /**
11     * validate - This function is the entry point of our plugin script
12     * @param {Object} pluginContext
13     * @param {Object} pluginContext.eDocument
14     * @param {String} pluginContext.eDocument.id
15     * @param {String} pluginContext.eDocument.scriptId
16     * @param {String} pluginContext.eDocument.content
17     * @param {Object} pluginContext.eDocument.source
18     * @param {String} pluginContext.eDocument.source.id
19     * @param {String} pluginContext.eDocument.source.text
20     * @param {Object} pluginContext.eDocument.template
21     * @param {String} pluginContext.eDocument.template.id
22     * @param {String} pluginContext.eDocument.template.text
23     * @param {Object} pluginContext.eDocument.status
24     * @param {Integer} pluginContext.eDocument.status.id
25     * @param {String} pluginContext.eDocument.status.text
26     * @param {Object} pluginContext.eDocument.package
27     * @param {String} pluginContext.eDocument.package.id
28     * @param {String} pluginContext.eDocument.package.text
29     * @param {Object} pluginContext.eDocument.transactionType
30     * @param {String} pluginContext.eDocument.transactionType.id
31     * @param {String} pluginContext.eDocument.transactionType.text
32     * @param {Object} pluginContext.eDocument.vendor
33     * @param {String} pluginContext.eDocument.vendor.id
34     * @param {String} pluginContext.eDocument.vendor.text
35     * @returns {Object} result
36     * @returns {Boolean} result.success
37     * @returns {String} result.message
38     */
39     function validate(pluginContext) {

```

```

40     var eDoc = pluginContext.eDocument;
41     var result = {
42       success: false,
43       message: ""
44     };
45
46
47     // Connect to validation service
48
49
50
51     // If successful
52     result.success = true;
53     result.message = "Validation successful!";
54
55
56     // Sample result if not successful
57     // result.success = false;
58     // result.message = "Service returned a failed response";
59
60
61     return result;
62   }
63
64
65   return {
66     validate: validate
67   };
68
69 });

```

 **Important:** The validation custom plug-in script must have the @NSScriptType pluggintypeimpl.

You can create as many validation plug-ins implementation as needed in your e-document processing, but you can only assign one validation plugin implementation per e-document package.

After creating the custom validation plug-in script, upload it to Customization > Plug-ins > Plug-in Implementations > New . The type of the custom plug-in implementation must be "Validation Plugin". For more information, [Custom Plug-in Creation](#).

## Setting Up the Inbound E-Document Validation Plug-in

Before inbound e-document processing can use the validation plug-in you created, you must set up the plug-in first and then integrate it with an e-document package.

### To set up the inbound e-document validation custom plug-in and integrate it with a vendor e-document package:

1. Go to Setup > E-Documents > Inbound E-Document Validation Plugin > New.
2. On the Inbound E-Document Validation Plugin page, enter the name of the plug-in in the **Name** field. In the **Script** field, select the plug-in you created from the dropdown list.
3. Click **Save**.
4. Go to Setup > E-Documents > E-Document Package.
5. On the E-Document Package List page, click the Edit link of the vendor e-document package that you want to integrate the inbound e-document validation plug-in with.
6. On the E-Document Package page, in the **Inbound E-Document Validation Plugin Implementation** field, select the custom plug-in you created.
7. Click **Save**.

The validation process or logic that you programmed into the plug-in will now be integrated with the inbound e-document conversion process.

## Deploying Automatic Bulk Conversion Script for Inbound E-Documents

The administrator can create schedules to run periodic and automatic bulk conversion of all inbound e-documents of which status is For Conversion. The Electronic Invoicing SuiteApp includes a Script Deployment for running automatic bulk conversion of inbound e-documents. By default, the status of the script is set to **Not Scheduled**. The administrator can set a schedule for the script to convert inbound e-documents in bulk.

### **Deploy the bulk conversion script for inbound e-documents:**

1. Go to Customization > Scripting > Script Deployments.
2. Expand the Filters and select **Map/Reduce** in the Type field. Click the Edit link of the script **Convert Inbound E-Document MR**.
3. Clear the **Deployed** box if you do not want to deploy the script yet. A script will not run in NetSuite until the Deployed box is selected.
4. Select a status in the **Status** field:
  - Testing
  - Not Scheduled
  - Scheduled
 For more information, see the help topic [Setting Script Deployment Status](#).
5. Choose an event type on the Schedule subtab:
  - **Single Event** – The script converts inbound e-documents only one time.
  - **Daily Event** – Enter the interval between days if this event should occur every day or every few days, or select every weekday if this event should occur every day except Saturdays and Sundays. For example, enter 1 as the interval if this event should occur every day, or enter 2 if the event should occur every other day.
  - **Weekly Event** – Enter the interval between weeks, and select the day of the week this event should be repeated.
  - **Monthly Event** – If you want to convert inbound e-documents on the same day of every month or every few months, enter the date when you want inbound e-document conversion to repeat and then select the interval between months. If you want to convert inbound e-documents on the same day of the week every month or every few months, select the week, the day of the week, and enter the interval between months.
  - **Yearly Event** – If you want to convert inbound e-documents one time every year, select the month and day, or select the week, day, and month.
  - **Start Date** – You must enter the date when you want to schedule inbound e-document conversion.
  - **Start Time** – Enter the time when you want inbound e-document conversion to start.
  - **Repeat** – Select how often you want the script to convert inbound e-documents. On the day the script is scheduled to run, inbound e-documents will be converted at the specified Start Time and the process repeats every n hours until midnight.
  - **End By** – Set the date when the script should stop converting inbound e-documents.

If you are scheduling a single event, inbound e-documents will be converted on the date entered in the Start Date field.

If you are scheduling a repeat event, inbound e-documents will be converted according to the schedule you set, starting from the date entered in the Start Date field and ending on the date entered in the End By field.

- **No End Date** – Check this box if you want to repeat the schedule indefinitely.
- 6. Click **Save and Execute** to convert inbound e-documents immediately. Click **Save** to convert inbound e-documents according to the schedule that you set.

**Note:** Before deploying the bulk conversion script, make sure that the date format in inbound e-documents is the same as the date format set in Setup > Company > Preferences > General Preferences.

After running the automatic bulk conversion, all inbound e-documents with status For Conversion, will be converted into transaction records. Successfully converted inbound e-documents will have their status set to Converted; otherwise, the status will be Conversion Failed.

If any of the inbound e-documents in the batch failed conversion, a notification will be sent to the email address of the user defined in the Recipient of E-Document Notifications. If all inbound e-documents in the batch were successfully converted, no notification email will be sent. The email notification will have an attached CSV file listing the inbound e-documents that underwent conversion. The list has columns indicating the Internal ID, Vendor, and details of the conversion process and errors. Likewise, you can view the error details of an inbound e-document that failed conversion by clicking the View link of that record. Then, on the E-Document Audit Trail subtab, in the Details column, view the error scope and details.

## Electronic Invoicing User Guide

**Note:** To fully use the outbound and inbound e-document processing of the Electronic Invoicing SuiteApp, make sure your account administrator has already created and applied the e-document templates, sending method, email recipients, email capture, inbound validation plug-in, and other settings. For information about setting up Electronic Invoicing, see [Electronic Invoicing Administrator Guide](#).

The following topics are intended for users of the Electronic Invoicing SuiteApp.

### General User Task

- [Displaying the E-Documents Portlet on the Home Page](#)

### Outbound E-Document Processing User Tasks

- [Overview of Outbound E-Document Process](#)
- [Outbound E-Document Statuses](#)
- [Assigning E-Document Packages to Customer or Vendor Records](#)
- [Defining E-Document Email Recipients](#)
- [Selecting E-Document Packages, Templates and Sending Methods on Transactions](#)
- [Enabling PDF File Reference Generation](#)
- [Generating and Regenerating E-Documents](#)
  - [Generating E-Documents for Single Transactions](#)
  - [Regenerating E-Documents for Single Transactions](#)
  - [Generating and Regenerating E-Documents in Bulk](#)

- Sending and Resending E-Documents
  - Sending the E-Document of a Single Transaction
  - Resending the E-Document of a Single Transaction
  - Resending E-Documents in Bulk

### **Inbound E-Document Processing User Tasks**

- Overview of Inbound E-Document Processing
- Inbound E-Document Statuses
- Receiving Inbound E-Documents by Email Capture
- Receiving E-Document XML Files from Web Service
- Uploading Received XML Files as Inbound E-Documents
- Converting Inbound E-Documents into Transaction Records
  - Converting an E-Document into Vendor Bill Linked to Purchase Order
  - Prerequisites and Conditions for Conversion
- Common Scenarios in Vendor Bill Conversion
- Converting Individual Inbound E-Documents into Vendor Bills
- Converting Failed Inbound E-Documents
- Compatibility of Approval Workflows with Vendor Bill Conversion
- Canceling Inbound E-Documents

See also the following topics:

- Electronic Invoicing Overview
  - Understanding E-Documents and E-Document Packages
  - Electronic Invoicing Permissions and Access Levels
  - Electronic Invoicing Limitations and Best Practices
- Electronic Invoicing Errors
  - Electronic Invoicing Error Codes
  - Outbound E-Document Generation Errors
  - Outbound E-Document Sending Errors

## Displaying the E-Documents Portlet on the Home Page

The Electronic Documents Dashboard SuiteApp Portlet provides links to outbound and inbound e-document processes and features. For more information, see [Electronic Documents Dashboard SuiteApp Portlet](#)

If you have access permission to the e-documents portlet, you can display it on your NetSuite Home page like any dashboard portlet.

### **To display the E-Documents Portlet on the Home page:**

1. Go to the NetSuite Home page.
2. Click **Personalize**.
3. On the Personalize Dashboard menu, select **SuiteApps**.
4. Click the Electronic Documents portlet, or drag the icon to the dashboard.

For more information about dashboard portlets, see the help topic [Dashboards](#).

## Overview of Outbound E-Document Process

From a user's perspective, the following steps describe the end-to-end process of generating XML or JSON e-documents from NetSuite transaction records and then sending the e-documents:

1. Assign an e-document package to a customer or vendor by performing the following steps:
  - a. Create or edit a customer or vendor record. If you are using an e-document sending method with an email channel, be sure to create or add the email recipients of the e-document. Then, on the **E-Document Email Recipient** subtab under the **E-Document** subtab, select contacts. You can select only contacts associated with the customer or vendor. Contacts must have valid email addresses. See [Defining E-Document Email Recipients](#).
    - If the customer or vendor is a company, create one or more contact records.
    - If the customer or vendor is an individual, add a valid email address on the record.
  - b. On the **E-Document** subtab, in the **E-Document Packages** field, select the e-document package to use for the customer's or vendor's transaction records.

**Note:** The outbound template and sending method included in the e-document package you will select must have subsidiaries belonging to this customer or vendor. For more information, see [Multi-subsidiary Support in the Outbound Process](#).

2. Create or edit a transaction record. Select an e-document template and an e-document sending method. For more information, see [Transactions and Processes Supported by the Electronic Invoicing SuiteApp](#) and [Selecting E-Document Packages, Templates and Sending Methods on Transactions](#).
  - If the e-document to be generated must be sent to a certification authority for certification, make sure that a certification sending method is assigned to the transaction and subsidiary. For more information, see [E-Document Certification in the Outbound Process](#).
3. Generate an e-document for the transaction record. You can generate e-documents individually or in bulk. For more information, see [Generating and Regenerating E-Documents](#).
4. Send the e-document. You can send e-documents individually or in bulk. For more information, see [Sending and Resending E-Documents](#).

**Note:** If a certification sending method is assigned to the transaction and subsidiary, the Certify E-Document button is displayed, which sends the e-document to the specified certification authority. Certified e-documents can be sent again to customers, vendors or a tax agency.

5. If an administrator enables and configures the automatic e-document processing feature, you can process the e-document automatically by clicking the **Process E-Document** button. For more information, see [Processing Multiple E-Document Processes Using Automatic E-Invoicing](#)

To understand errors that can be encountered during the generation or sending of e-documents, read the following topics:

- [Outbound E-Document Generation Errors](#)
- [Outbound E-Document Sending Errors](#)

To understand the outbound electronic invoicing process flow and e-document status definitions, see the following topics:

- [Outbound Electronic Invoicing](#)
- [E-Document Audit Trail and Statuses](#)

## Outbound E-Document Statuses

E-document statuses are indicated in the **E-Document Status** field on the **E-Document** subtab. Outbound e-document statuses describe whether a transaction or e-document is in the process of e-document generation, sending or certification. The following table lists the e-document generation statuses.

E-Document Generation Status	Description
For Generation	The e-document can be generated. An outbound e-document template was associated with the transaction and the e-document can be generated. Audit trail log indicates Tagged for Generation.
Generating	The e-document is being generated.
Generation Failed	The e-document was not generated. An error caused generation to fail. See details of e-document generation errors on the E-Document Audit Trail subtab. You must first fix the errors before you can regenerate an e-document. For more information, see <a href="#">Outbound E-Document Generation Errors</a> .
Untagged for E-Invoice Generation	The e-document template was removed from the transaction record. The e-document of the transaction record will not be generated.

The following table lists the e-document sending statuses.

E-Document Sending Status	Description
Ready for Sending	The e-document was generated successfully and the e-document can be sent to specified recipients. Whether the e-document was generated manually or through bulk generation is indicated.
Sending	The e-document is being sent to specified recipients.
Sending Failed	The e-document was not sent. Information about sending errors are shown in the Details column. You must fix the error and regenerate the e-document before you can resend it.
Sent	The e-document was successfully sent. If the default sending method is email, the Details column shows the email addresses of the sender and recipients.
<b>Note:</b> When an e-document is sent, the designated e-document sender's name and email address is displayed as the sender. If you did not designate a sender, the system uses the name and email address of the person who initiated the sending process as the sender of the e-document. For more information, see <a href="#">Selecting a Designated E-Document Sender</a> .	

The following table lists the e-document certification statuses.

 <b>Note:</b> E-document certification statuses are not automatically updated by the system, you must implement the updates through the certification sending method. See <a href="#">Updating E-Document Certification Statuses</a>
---

E-Document Certification Status	Description
Ready for Certification	The e-document has a certification sending method assigned to its transaction and subsidiary, and it can be sent to the certification authority.

E-Document Certification Status	Description
Certification in Progress	The e-document has been sent to the certification authority and is being certified. The certified e-document has not been received.
Certification Failed	Certification failed due to an error, of which details are on the E-Document Audit Trail subtab. You must fix the error and regenerate the e-document before you can send it again for certification.
Certification Data Error	Details of certification errors are on the E-Document Audit Trail subtab.
Ready for Sending	The e-document was successfully certified and is ready for sending.

## Assigning E-Document Packages to Customer or Vendor Records

The e-document templates and sending methods assigned to the e-document package are available for selection on the **E-Document** subtab of every customer or vendor transaction.

### To assign an e-document package to a customer or vendor:

1. Create or edit a customer or vendor record.
  2. Click the **E-Document** subtab of the customer or vendor record.
  3. In the **E-Document Package** field, select the appropriate e-document package for the customer or vendor.
- Note:** The outbound template and sending method included in the e-document package you will select, must have subsidiaries belonging to this customer or vendor. For more information, see [Multi-subsidiary Support in the Outbound Process](#).
4. Click **Save**.
- Note:** Assigning e-document packages with email sending methods can only be done by editing the customer or vendor record.

## Defining E-Document Email Recipients

If you want to use an e-document sending method that has an email sending channel, you must first define the email recipients for your customer or vendor. If the customer or vendor is a company, there should be at least one email recipient defined on the **E-Document** subtab on the customer or vendor record. If the customer or vendor is an individual, make sure it has an email address.

**Note:** The system can send an e-document by email to a maximum of 10 recipients for each customer or vendor. The system counts each contact added as a recipient. If you add the same contact multiple times, each instance is considered an individual recipient.

### To define e-document email recipients:

1. Create or edit a customer or vendor record.
2. On the **E-Document Email Recipient** subtab under the **E-Document** subtab, select a contact or select **New** to create a new contact.



**Note:** When you select an email recipient, only contacts with valid email addresses associated with the customer or vendor are available for selection in the dropdown list.

3. Click **Add** after selecting each contact.
4. Click **Save**.



**Note:** For transfer order, basic and journal style custom transaction types, you can define the email recipients directly in the **E-Document Email Recipient** field from the **E-Document** subtab. The email recipients are also applicable for item fulfillment transactions.

## Selecting E-Document Packages, Templates and Sending Methods on Transactions

Before you can generate an e-document, you must specify on the transaction record the e-document template and sending method to use for generating and sending the e-document.

### To Select E-Document Package, Template and Sending Method on Transactions:

1. Create or edit any supported transaction record that you want to generate and send an e-document.
2. Go to the **E-Document** subtab.
3. Select an e-document package associated with the e-document template and sending method on the **E-Document Package** field.

For item fulfillment records, you must first define an Item Fulfillment template and sending method along with Transfer Order template and sending method in the same package. This is to enable the selection of Item Fulfillment template and sending method directly from the Item Fulfillment record.



**Note:** The **E-Document Package** field is specific only for transfer order, its following item fulfillment, basic and journal style custom transaction types.

4. Select an e-document template that you want the system to use for generating the e-document on the **E-document Template** field.

If you do not want the transaction to be tagged for e-document generation, leave this field blank.



**Note:** The e-document templates available for selection are based on the e-document package assigned to the customer. For more information, see [Creating E-Document Packages](#)

5. Select the sending method to use for sending the e-document on the **E-Document Sending Method** field.
6. Click **Save**.

## Enabling PDF File Reference Generation

The Electronic Invoicing SuiteApp supports the creation, sending, and receiving of PDF file references of transaction records undergoing the e-document process. The PDF version of transaction records will enable you to view and check if the content of e-document is correct.

In outbound e-document processing, if you enable the setting Generate PDF, the PDF file is created when the XML or JSON file of a transaction record is generated. After the PDF file is created, you can view it in the **Generated PDF** field, on **E-Document** subtab of transaction records. When sending the e-document using **E-document Sending Method Plugin Implementation** field, the PDF file is attached along with the XML or JSON e-document.



**Note:** PDF file reference generation is not supported for vendor bill records. The Generate PDF box and Generated PDF link are not available on vendor bill records with generated outbound e-documents.

## Generate PDF File References

You can generate PDF file references of transaction records to be converted into e-documents.

The PDF file generated will follow the default printing layout of the transaction. You can set a default preference on entity records.

### Enabling Generate PDF as a Default in Entity Records

1. Edit a customer or vendor record and go to the **E-Document** subtab.
2. Check the **Generate PDF** box. Consequently, all transaction types supported by electronic invoicing from this customer or vendor will have the Generate PDF box automatically checked. When e-documents are generated from the transactions, corresponding PDF file references will be automatically created. When sending the e-document using **E-document Sending Method Plugin Implementation** field, the PDF file is attached along with the XML or JSON e-document.
3. Click **Save**.

When the e-document is generated, the PDF file reference can be viewed and downloaded under **Generated PDF** on the **E-Document** subtab of the transaction. Also do note that the QR code will be visible only on successful QR generation, else it will be hidden.

If you do not want to create PDF file references of individual transaction records, you can still override PDF generation enabled on the customer or vendor record. To do so, edit the transaction record that you do not want to generate a PDF file reference of, and then clear the **Generate PDF** box. If Generate PDF is cleared on a transaction record where it was initially enabled and a PDF file was generated, during the next e-document generation, no PDF file will be generated and any previously generated PDF file will be deleted.

## Generating and Regenerating E-Documents

The e-documents of transaction records in NetSuite can be generated using the Electronic Invoicing SuiteApp. The generated e-documents can then be sent to your customers or tax authorities through the outbound process.

You can generate the XML or JSON e-documents of the following transaction records:

- Bill (for self-billing)
- Cash Sale
- Cash Refund
- Credit Memo
- Customer Payment
- Estimate

- Invoice
- Item Fulfillment
- Purchase Order
- Return Authorization
- Transfer Order
- Vendor Credit or Bill Credit

To generate an e-document, you must first select an e-document template on the transaction. The system cannot generate an e-document if there is no e-document template associated with the transaction.

The e-document templates available for selection in each transaction record are templates you have assigned to the customer's e-document package. For more information, see [Creating E-Document Packages](#).

The e-document templates available for selection in each transaction record are templates that have been assigned to that transaction type. For more information, see [Creating E-Document Templates](#).

You can generate e-documents individually or in bulk, regardless of transaction type.

The system supports e-document generation for supported transactions created using the user interface, CSV import, SOAP web services, and SuiteScript.

## Generating E-Documents from the Electronic Documents Portlet

You can quickly access outbound e-documents with status For Generation from the Electronic Documents portlet on the Home page. For more information about the Electronic Documents portlet, see [Electronic Documents Dashboard SuiteApp Portlet](#).

### To generate e-documents from the Electronic Documents portlet:

1. Go to the Electronic Documents portlet on the Home page.
2. Click the number under **Outbound E-Documents for Generation**.  
The Outbound E-Documents For Generation results page is displayed.
3. Click the **View** link of the transaction record that you want to generate an e-document from.
4. On the selected transaction record, click the **Generate** button.

## Generating E-Documents for Single Transactions

You can also start the process of generating e-documents from transaction records.

### To generate an e-document:

1. Create or edit a transaction record.
2. Enter items and other required information about the transaction record.
3. Click the **E-Document** subtab.
4. In the **E-Document Template** field, select the e-document template that you want the system to use for generating the e-document. If you do not want this transaction to be tagged for e-document generation, leave this field blank.



**Note:** The e-document templates available for selection are based on the e-document package assigned to your customer or vendor. For more information, see [Creating E-Document Packages](#).



**Important:** Use caution when you want to edit the transaction record. Each time you edit and save a transaction, the **E-Document Status** field is updated to **For Generation**. You will need to generate and send the e-document again. Note that this does not apply to e-documents with the **Sent** status.

5. Click **Save**. The system updates the transaction record, and a **Generate E-Document** button is displayed at the top. On the **E-Document** subtab, the value of the **E-Document Status** field is updated to **For Generation**.



**Note:** The **Generate E-Document** button appears only when viewing a transaction record. It is not displayed in edit mode.

6. Click **Generate E-Document**.

The outbound e-document is generated. A PDF version of the source transaction is also generated if Generate PDF is enabled on the transaction record.

After generating the e-document, the system updates the value in the **E-Document Status** field on the **E-Document** subtab. The field shows one of the following statuses:

- **Ready for Sending** – This status means the e-document was generated successfully and can be sent to the email recipients defined in the customer record or your e-document sending method.
- For more information, see [Creating E-Document Sending Methods](#).
- **Generation Failed** – This status means the e-document was not generated. Information about generation errors are shown in the **Details** column of the **E-Document Audit Trail** subtab. You must first fix the errors before you can regenerate an e-document.

For more information, see [E-Document Audit Trail and Statuses](#).

## Regenerating E-Documents for Single Transactions

You can regenerate an e-document for individual supported transaction records at any time. The transaction records must have an e-document template assigned to them, and the status in the **E-Document Status** field must not be blank or **Sent**. However, if an error occurs during generation and the error is not fixed, the system will not be able to generate an e-document for the transaction.

If the value in the **E-Document Status** field on the **E-Document** subtab shows **Generation Failed**, you can regenerate the e-document by doing the following:

### To regenerate an e-document:

1. Fix the errors identified in the audit trail.
2. View the transaction and click **Generate E-Document**.



**Important:** Use caution when you want to edit the transaction record. Each time you edit and save an document, the **E-Document Status** field is updated to **For Generation**. You will need to generate and send the e-document again. This can result in generating and sending multiple e-documents for one transaction record. Note that this does not apply to e-documents with the **Sent** status.

For more information about generation errors, see [Outbound E-Document Generation Errors](#).

## Generating and Regenerating E-Documents in Bulk

If your account administrator has set up a schedule to run the bulk generation script periodically, the script automatically generates e-documents for those transactions that have any of the following statuses:

- For Generation
- Generation Failed

Upon successful bulk generation of outbound e-documents, PDF versions of the transactions are also generated if those transactions have the Generate PDF setting enabled.

For failed generation, errors must be fixed first before e-documents can be regenerated.

When the system encounters an error during bulk generation of e-documents, it sends an email notification to the Recipient of E-Document Notifications. If the Recipient of E-Document Notifications is not assigned yet, the notification will be sent to the active administrator of the account. For more information about assigning the Recipient of E-Document Notifications, see [Prerequisites for Using Electronic Invoicing](#), step 4.

The email message contains an attachment in CSV format that lists the transaction records that encountered errors during bulk generation.

The user who created the transaction record must fix the errors for each transaction record.

After the errors are fixed, the e-documents will be generated during the next scheduled bulk generation of e-documents. Alternatively, the user can regenerate an e-document for the individual transaction record.

For more information about generation errors, see [Outbound E-Document Generation Errors](#).

If you want to generate or regenerate e-documents for multiple transactions outside of the scheduled script run, contact your account administrator for assistance. See [Deploying the Bulk Generation Script for E-Documents](#).

## Sending and Resending E-Documents

A successfully generated e-document displays the **Send E-Document** button on the transaction record and its e-document status indicating Ready for Sending.

A generated e-document with a certification sending method assigned to its transaction and subsidiary, displays the **Certify E-Document** button on the transaction.



**Note:** Generating a new e-document overwrites the previous e-document.

## Sending the E-Document of a Single Transaction

Perform the following steps to send the e-document of a single transaction.

### To send an e-document:

1. Open the transaction record whose e-document status is Ready for Sending.  
If the e-document is for certification and a certification sending method is assigned to the transaction and subsidiary, the status is Ready for Certification.
2. Make sure that the **E-Document Sending Method** field specifies the sending method appropriate for this transaction record.

If the e-document is for certification, make sure that a certification sending method is assigned to the transaction and subsidiary.

3. Click the **Send E-Document** button.

If the e-document is for certification and a certification sending method is assigned to the transaction and subsidiary, click the **Certify E-Document** button.

A banner appears on the transaction record indicating that sending of the e-document is in progress. The e-document is sent through the selected sending method and the value of the E-Document Status field on the E-Document subtab is updated. For more information, see [Outbound E-Document Statuses](#).

Successfully certified e-documents are returned by the certification authority. You can view or download the XML or JSON file of the certified e-document by clicking the corresponding links under **Certified E-Document** on the **E-Document** subtab of the transaction.

## Resending the E-Document of a Single Transaction

Perform the following steps to resend the e-document of a single transaction.

### To resend an e-document:

1. If failure in the previous e-document sending was caused by an error, amend the transaction as necessary and save it.
2. Open the amended transaction and click **Generate E-Document**.  
After the system generates the e-document, an audit trail is created and the status in the **E-Document Status** field changes to **Ready for Sending**.
3. Click **Send E-Document** to send the e-document.  
After the system sends the e-document, the status displayed in the **E-Document Status** field changes to **Sent**, and an audit trail is created.

## Resending E-Documents in Bulk

 **Note:** Resending e-documents in bulk requires certain permissions and access levels for roles. For more information, see [Setting Up Custom Roles to Send E-Documents](#).

The Electronic Invoicing SuiteApp provides a search filter to enable you to resend in bulk the e-documents that failed initial sending. The system searches for all e-documents with **Sending Failed** status.

Errors must be fixed before e-documents can be successfully resent. For more information, see [Outbound E-Document Sending Errors](#).

To access e-documents that failed generation or sending, go to the Electronic Documents portlet on the Home page, and then click the number under **E-Documents with Errors**. On the E-Documents with Errors results page, view or edit the transaction records that you want to fix.

After fixing the errors, you can proceed with either regenerating or resending the e-documents.

### To resend e-documents in bulk:

1. Go to the Electronic Documents portlet on the Home page, and then click the link **Send Failed E-Documents**.
2. In the Select entity type filter, choose either **Customer** or **Vendor**.

3. To view a list of transactions for a particular customer or vendor, select that customer or vendor from the **Customer** or **Vendor** field. If no customer or vendor is selected, the search result will display all transaction belonging to the subsidiary, regardless of customer or vendor.
4. In the **Subsidiary** field, select a subsidiary to view the transactions for that subsidiary only.
5. In the **Transaction Date From** field, set the date to define the beginning of the date range. The search result displays all created transaction records starting from the date you set in this field.
6. In the **Transaction Date To** field, set the date to define the end of the date range. The search result displays all created transaction records starting from the date you set in this field.
7. If you chose the Customer entity type, select one or more transaction types in the **Transaction Type** field, for each e-document you want to resend. To select multiple transaction types, press and hold the Ctrl key while selecting each transaction type. If no transaction type is selected, the search result will show all e-documents with Sending Failed status, regardless of transaction type.

If you chose the Vendor entity type, Purchase Order is automatically selected in the Transaction Type field, because only purchase order is applicable to vendor entities.

Only transaction types that are supported by or applicable to outbound e-document sending are shown in the transaction list, which include:

- Bill (for self-billing)
- Cash Sale
- Cash Refund
- Credit Memo
- Customer Payment
- Estimate
- Invoice
- Item Fulfillment
- Purchase Order
- Return Authorization
- Registered Custom Transaction Types
- Transfer Order
- Vendor Credit or Bill Credit

8. Click **Search**.

The search result displays all e-documents with Sending Failed status, according to your search criteria.

9. Click **Send** to resend all e-documents displayed in the search results.

After the system sends the e-documents, it updates the value in the **E-Dокумент Status** field on the **E-Dокумент** subtab on each transaction record. The field shows one of the following statuses:

- **Sent** – This means the e-document was successfully sent. The **Details** column shows the email addresses of the sender and recipients.
- **Sending Failed** – This means the e-document was not sent. Information about sending errors are shown in the **Details** column. You must fix the errors before you can successfully resend the e-document.

For more information, see [Outbound E-Document Sending Errors](#).

The system sends an email notification to the person who initiated the sending process when the e-documents have been sent.

The email message contains an attachment in CSV format that lists the transaction records and the status of each transaction. If errors were encountered, the details are included in the attachment.

## Processing Multiple E-Dокумент Processes Using Automatic E-Invoicing

Only users with administrator role can enable and configure the e-document processes for automation. After the configuration, only users with required permissions can get access to the **Process E-Document** button on individual transactions. After this the users can perform the pre-configured e-document processes automatically for a subsidiary by clicking the **Process E-Document** button .

## E-Dокумент Network Status

The Electronic Invoicing SuiteApp enables you to get the e-documents certified from a regulatory body and get network status through an API. The latest network status can be obtained by clicking the **Get Network Status** button on the transaction.

The **Get Network Status** button is available on the transaction as soon as the sending method returns the network details for the transaction. It can happen either in the certify or send stage of the transaction, or both. If the transaction is in certify stage, the certification sending method returns the status of the transaction, else if the transaction is in send stage, the sending method assigned to the transaction returns the status of the transaction.

The following table describes the transaction body fields which contain the network information.

Network Field	Field ID	Description
Network Reference Id	custbody_ei_network_id	It is a unique number generated by network API while certifying or sending e-document.
Network Name	custbody_ei_network_name	It is the name of a network. For instance, PEPPOL, SAT, ARIBA.
Network Status	custbody_ei_network_status	It displays a keyword representing one of the possible network statuses used by the network.
Network Status Updated On	custbody_ei_network_updated_date_time	It displays the time stamp (date and time) of the updated status.

## Overview of Inbound E-Dокумент Processing

The following describe the end-to-end process of creating inbound e-documents and then converting them into NetSuite transaction records using the Electronic Invoicing SuiteApp.

### To upload and convert inbound e-documents:

1. The administrator creates an inbound e-document template, which will map the data elements of the inbound XML e-document to corresponding NetSuite transaction fields when converted. See [Creating E-Dокумент Templates](#).
2. The administrator associates the inbound e-document template with an e-document package, which is assigned to a customer or vendor. See [Assigning E-Document Packages to Customer or Vendor Records](#).
3. The administrator designates an employee or group who will receive an email notification after scheduled automatic bulk conversion of inbound e-document records to transaction records.
4. If the inbound e-document is received through means other than email capture, users can upload it manually. See [Uploading Received XML Files as Inbound E-Documents](#).

5. Users can convert inbound e-document records with status For Conversion, into transaction records. You can manually convert e-documents or run automatic scheduled bulk conversion. See [Converting Inbound E-Documents into Transaction Records](#).

The status of newly created vendor bills from e-document conversion is dependent on the default status of Accounting Preference or whether Vendor Approval Routing is enabled.

## Inbound E-Document Statuses

E-document statuses are indicated in the **E-Document Status** field on the **E-Document** subtab. Inbound e-document statuses describe the state of a received e-document in the e-document conversion process. The following table lists the inbound e-document statuses.

E-Document Conversion Status	Description
Incomplete	The inbound e-document is missing some details or content. Audit trail log indicates Tagged as Incomplete.
For Conversion	The inbound e-document is ready for conversion into a vendor bill. A vendor, XML file reference, and e-document template have been defined in the inbound e-document. Audit trail log indicates Tagged for Conversion.
Converting	The e-document is being converted to a transaction record.
Converted	The inbound e-document was successfully converted into a vendor bill.
Conversion Failed	An error occurred and the inbound e-document was not converted. Information about conversion errors can be found in the Details column of the E-Document Audit Trail subtab.
Canceled	The inbound e-document was canceled. If cancellation of the inbound e-document encountered an error and did not proceed, the audit trail log will indicate Cancellation Failed.

## Receiving Inbound E-Documents by Email Capture

After your administrator has set up the Inbound Email Capture plug-in and Vendor E-Document Sender, you are ready to receive inbound e-documents from your vendors. The system automatically receives inbound e-documents sent through email and then queues them as Inbound E-Documents for Conversion if an e-document template is assigned.

The number of e-documents received from email capture is displayed on the Electronic Documents portlet, under Inbound E-Documents for Conversion. You can click the number displayed to go to the Inbound E-Documents for Conversion: Results page, which lists the inbound e-documents for conversion. For more information, see [Electronic Documents Dashboard SuiteApp Portlet](#).

If you have automatic bulk conversion set up, the bulk conversion script will automatically convert the e-documents in the list into transaction records at a defined schedule. For more information, see [Deploying Automatic Bulk Conversion Script for Inbound E-Documents](#). Or, you can convert individual e-documents in the list by clicking the View link of an e-document, and then click the Convert button on the inbound e-document record page.

Inbound e-document email capture receives XML files along with their PDF file reference attachment, if any were generated. Received PDF files are stored in the File Cabinet and can be viewed by clicking a link in the PDF File Reference field on the newly created inbound e-document record. When the inbound e-document is converted into vendor bill, the PDF file reference can also be viewed from the E-Document subtab.



**Note:** Inbound email capture can only receive one XML e-document and one PDF file per email.

## Editing Incomplete Inbound E-Documents

Inbound e-documents that do not clearly define the vendor who sent them, will also be missing the e-document template, which is defined by the vendor. These inbound e-documents will be tagged as incomplete. They are queued as Incomplete Inbound E-Documents, which is displayed on the e-document dashboard portlet. You can click the number displayed under Incomplete Inbound E-Documents to go to a results page, where you can click the Edit link of an inbound e-document to complete it.

On the Inbound E-Document record page, you can complete the inbound e-document by selecting the right vendor in the Vendor field and then the system automatically selects the e-document template that is applicable to the inbound e-document based on the XML content and the selected vendor. But, you can still manually select or override the default e-document template. After completing the inbound e-document, the E-Document Status is changed to For Conversion.

## Receiving E-Document XML Files from Web Service

XML files sent through web service are received by NetSuite and are displayed in the Electronic Documents dashboard portlet. But before receiving e-documents, you must first set up and select the appropriate inbound e-document templates to be used. For more information about setting up inbound templates, see the help topics [Understanding Inbound E-Document Templates in JSON Format](#) and [Understanding XSD in Inbound E-Document Templates](#).

XML e-documents received by NetSuite through web service are validated and checked if the information they contain is complete. The system automatically identifies the sender using the Token ID and Token Secret provided in the request. The Web Service ID is used to identify the vendor whom the e-document should be associated with. The integrity of the XML file is also checked. The results of these validations determine if an inbound e-document record will be created from the received XML file. If created, the inbound e-document record will indicate Web Service in the Source field. The inbound e-document is added to the queue corresponding its processing status displayed on the Electronic Documents dashboard portlet.

NetSuite responds to web service requests by confirming the success or failure of e-document creation. You can view web service requests and responses by going to Customization > Scripting > Scripts, then click the View link of the Inbound E-Document Web Service RL script. The logs are on the Execution Log subtab.

## Web Service Errors

The following tables list the web service errors that can be encountered by the recipient and sender of web service requests. The recipient is the user in your company who can receive the XML files from web service. The sender is the vendor you authorized to issue web service requests to NetSuite to send XML files.

### Errors Encountered by Web Service Recipients

Error Message	Description and Solution
The Web Service ID is already being used by another vendor. Enter a different Web Service ID.	This message is displayed when user tries to save a vendor record that has a web service identifier that is already being used by another vendor.  A different web service identifier must be entered.

Error Message	Description and Solution
The inbound e-document is incomplete, as the correct template cannot be determined. Either select a template in the inbound e-document record, or set up the XSD in the e-document template record to enable template autoselection.	This message is sent through email to the Recipient of E-Document Notifications, informing the recipient that an inbound e-document record was created but is incomplete.  Either manually set the template in the inbound e-document record, or set up the template's XSD to enable template auto selection.
The inbound e-document is incomplete, as the correct vendor cannot be determined. Either select a vendor in the inbound e-document record, or set the Web Service ID in the associated vendor record.	This message is sent through email to the Recipient of E-Document Notifications, informing the recipient that an inbound e-document record was created but is incomplete.  Either manually set the vendor in the inbound e-document record, or correctly set up the vendor's web service ID field.

### Errors Encountered by Web Service Sender

Error Message	Description and Solution
The following keys are missing: {KEYS}, which you must provide in the web service request.	The web service request failed because any of the following keys is missing: <ul style="list-style-type: none"><li>■ identifier</li><li>■ filename</li><li>■ content</li></ul> Ensure that the missing keys are provided in the web service request.
The body of the web service request must be a JSON object or an array of JSON objects using Content-Type: 'application/json'.	The web service request failed.  The content of the web service request must be JSON objects.
The selected XML File Reference is not a valid XML file. Ensure that the file you select has the .xml extension.	The web service request failed.  Provide a valid XML file for the XML File Reference.
No vendor is associated with the Web Service ID: {IDENTIFIER}. Ensure that the correct Web Service ID is used.	The web service request failed.  Ensure that the Web Service ID is associated with a vendor or the party sending XML files.
The selected XML File Reference is not a well-formed XML document.	The web service request failed.  Check the XML File Reference and ensure that content adheres to XML syntax, with properly defined content, structure and tags.

## Uploading Received XML Files as Inbound E-Documents

Aside from email capture, you can receive vendor bills in XML format through other means like storage media. You can manually upload these XML file references to NetSuite as inbound e-document records, which are subsequently converted into vendor bills records.

### To upload an XML file reference as an inbound e-document record:

1. On the E-Documents Portlet on the Home page, click the **Upload Inbound E-Document** link. The Inbound E-Documents page displays the following fields:
    - Transaction Type indicates the record (Bill) that will be created consequently of the conversion.
    - Source indicates Manual Upload by default.
    - Reference Number and PO Number receive values from the XML file reference that you will upload. These fields will have values after conversion to a transaction record.
    - E-Document Status indicates the status of the newly created inbound e-document.
  2. In the **Vendor** dropdown list, select the vendor who sent the XML file.
  3. In the **XML File Reference** field, click + and select the XML file that you will convert into a transaction record.
- The XML File Reference dropdown list displays files stored in the File Cabinet. Ensure that you select an XML document that is well-formed and valid, with the .xml file extension; otherwise an error will be generated.
4. In the **E-Document Template** field, select an inbound e-document template.



**Note:** If an XSD file was selected in a template record that is appropriate for this inbound e-document record, that template is automatically selected in the E-Document Template field.

5. If you want to attach the PDF version of the XML document, click + on the **PDF File Reference** field, then select the PDF file you want to associate with this record.
6. Click **Save**.

A new inbound e-document record is created and displayed with its E-Document Status set to For Conversion. The **E-Document Audit Trail** subtab displays information about the inbound e-document record including the date created, entity, event type, owner, and details. If the uploaded XML file has an attached PDF file reference, the attachment can be viewed and downloaded in PDF File Reference field on the created inbound e-document record.

## Converting Inbound E-Documents into Transaction Records

### Converting an E-Document into Vendor Bill Linked to Purchase Order

The process of converting purchase order to vendor bill is viewed from the perspective of the NetSuite user being a customer, who purchased items or services from a vendor. In this scenario, the NetSuite user initially sends a purchase order to a vendor, by outbound electronic invoicing. The purchase order is received by the vendor and processed on their side. The vendor will enter the details of the purchase order into their system and eventually generates an invoice record, which is then converted into an XML e-document and returned to the NetSuite user. The XML e-document from the vendor is received by the NetSuite user through the inbound electronic invoicing feature. The NetSuite user uploads the received XML file to NetSuite as an inbound e-document record. It is this inbound e-document record that will undergo conversion to a vendor bill that is linked to the original purchase order.

### Purchase Order Items and Expenses in Inbound E-Documents

As an item or an expense is required in a purchase order, either of them must also be included in the e-document for conversion; otherwise, conversion to vendor bill will not proceed.

In addition, items or expenses specified in the inbound e-document must include relevant details:

- For items, either the vendor code or vendor code/name must be included, depending on whether the Multiple Vendors feature is enabled or not.
- For expense, the amount must be included, and it is that the Default Expense Account in the vendor record has a value. If there are specified amounts with no corresponding accounts, the Default Expense Account is automatically made the default account in the new vendor bill created from conversion.

For more details about the prerequisites for converting inbound e-documents with purchase orders items or expenses, see the following topic.

## Prerequisites and Conditions for Conversion

### Permission to Convert E-Documents

Your user role must have the permission to convert e-document records into NetSuite transactions. For more information about granting a role the permission to perform inbound e-document conversion, see [Setting Up Custom Roles that can Convert Inbound E-Documents](#).

### Status of Purchase Order

The inbound XML e-document must not be linked to a purchase order whose status is fully billed, unapproved, rejected, cancelled or closed. Otherwise, the conversion will fail. Conversion will also fail if the vendor in the inbound XML e-document is different from the vendor in the purchase order.

### Inbound E-Document has Reference Number with Mapping in the Template

The Reference Number must be included in the inbound XML e-document and the e-document template must contain the mapping for the reference number. Without the reference number in the inbound XML e-document or the correct reference number mapping in the template, conversion of the e-document will fail.

### Inbound E-Document has Purchase Order Number and the Template has Mapping to the createdfrom Field

The Purchase Order Number must be included in the inbound XML e-document and the e-document template must contain the mapping for the createdfrom field. Without the Purchase Order Number in the inbound XML e-document or the correct reference number mapping in the template, conversion will result in a stand-alone vendor bill. See [Converting an Inbound E-Document Without a Purchase Order Number](#).

### Purchase Order must have either an Item or Expense

The reference purchase order of the inbound e-document for conversion, must have either an item or expense specified. Without item or expense, the inbound e-document will not be converted. Moreover, purchase order items or expenses must have required details included in the inbound e-document for conversion. The required details of items or expenses are discussed in the following sections.

### Required Account and Amount for an Expense

If the XML e-document has an expense line, the required Account and Amount fields of the bill must have values. The Account field of the expense line references the value of Default Expense Account on the Financial subtab of the vendor record. Therefore, the vendor must have a Default Expense Account setup or the conversion will fail.

### Required Vendor Code for Item Records

If the purchase order has item records, the vendor code of these item records must be included in the inbound XML e-document. If the Multiple Vendors feature is enabled during conversion, you must enter the Vendor Code on the Vendors subtab of the item record. The vendor code will be used to map the items to their corresponding vendor. If the vendor code in the XML e-document does not match any vendors, conversion will fail. The vendor code field is highlighted in the following screenshot.

The screenshot shows the NetSuite Inventory Item screen for an item named 'Black Backpack'. The 'Vendors' subtab is selected. The 'VENDOR NAME/CODE' field for the item record is highlighted with a red box. Other fields visible include 'DISPLAY NAME/CODE' (Black Pack 1.0 Enhanced), 'SUBITEM OF', 'INTERNAL ID' (9), 'ITEM NAME/NUMBER' (Black Backpack), 'SUBSIDIARY' (Parent Company), 'DEPARTMENT', 'LOCATION' (Manila), 'CLASS', 'PURCHASE DESCRIPTION', 'STOCK DESCRIPTION', 'MATCH BILL TO RECEIPT', 'REORDER MULTIPLE', 'TRANSFER PRICE', 'MANUFACTURER', 'MPN', 'MANUFACTURER COUNTRY', and various vendor bill matching tolerance fields.

### Required Vendor Code/Name for Item Records

If the Multiple Vendors feature is disabled, the vendor code/name on the main tab of item records must be included in the inbound XML e-document. Otherwise, the conversion will fail. If two or more items have the same vendor code/name, implying duplicates, the conversion will fail. The vendor code/name field is highlighted in the following screenshot.

The screenshot shows the NetSuite Inventory Item screen for an item named 'Black Backpack'. The main tab is selected. The 'VENDOR NAME/CODE' field for the item record is highlighted with a red box. Other fields visible include 'DISPLAY NAME/CODE' (Black Pack 1.0 Enhanced), 'SUBITEM OF', 'INTERNAL ID' (9), 'ITEM NAME/NUMBER' (Black Backpack), and 'VENDOR NAME/CODE' (BACKBLK01).

## Common Scenarios in Vendor Bill Conversion

The following tables summarize the conversion of inbound e-documents with reference purchase orders that have items and expenses. Consider the scenarios that result in successful conversion and avoid those where conversion will fail.

### Conversion Scenarios with Purchase Order Items

Case	Details of Reference Purchase Order	Details of Inbound XML E-Document based from the Purchase Order	Details of Inbound E-Document Template	Expected Conversion Result
Partial billing of some line items	The purchase order contains multiple line items.	The inbound XML e-document contains less line items than the total number of line items in the reference purchase order.	The template is valid with correct mapping.	Conversion will succeed. The created vendor bill includes only the line items contained in the XML e-document. The vendor bill is linked to the reference purchase order.
Duplicate line items	The purchase order contains duplicate line items.	The inbound XML e-document contains only one instance of each line item in the purchase order.	The template is valid with correct mapping.	Conversion will succeed. The created vendor bill contains no duplicate line items. The vendor bill is linked to the reference purchase order.
E-document template contains no mapping	The purchase order contains line items with values for amount, quantity, tax code, and rate fields.	The inbound XML e-document contains line items with field values.	The e-document template is valid but has no mapping for line item field values.	Conversion is successful. Field values for line items will take the default values entered in the purchase order.
The XML e-document contains additional line items that are not in the purchase order.	The purchase order is not yet billed and contains multiple line items.	The inbound XML e-document contains all or some of the purchase order line items, plus additional line items that are not in the purchase order.	The e-document template is valid with correct mapping.	Conversion will succeed. . All purchase order line items that are included the XML e-document, are entered in the created vendor bill. The additional line items included in the XML e-document but not in the purchase order, are also entered in the vendor bill. If all line items from the purchase order are included in the vendor bill. The status of the reference purchase order becomes Fully Billed.
Converting a Partially Received purchase order with additional line items that are not in the reference purchase order	The purchase order is Partially Received and contains multiple line items, one of which has been received.	The inbound XML e-document contains some (but not all) purchase order line items, including the ones that have been received, plus additional line items that are not in the purchase order.	The e-document template is valid with correct mapping.	Conversion will succeed. All purchase order line items in the XML e-document that are not yet received, are included in the created vendor bill. Line items already received but still included in the XML e-document, are taken as additional line items in the bill. Additional line items that are not in the purchase

Case	Details of Reference Purchase Order	Details of Inbound XML E-Document based from the Purchase Order	Details of Inbound E-Document Template	Expected Conversion Result
				order are included in the vendor bill. The status of the reference purchase order remains Partially Received.
The inbound XML e-document contains line items with no field values or values are set to 0.	The purchase order contains line items with values for amount, quantity, tax code, or rate.	The inbound XML e-document contains line items but with no values for the amount or quantity fields, or values are set to 0.	The e-document template is valid with correct mapping.	Conversion will fail. Details of the error are indicated in the E-Document Audit Trail.
The inbound XML e-document contains no line items.	The purchase order contains multiple line items.	The inbound XML e-document contains no line items.	The e-document template is valid with correct mapping.	Conversion will fail. Details of the error are indicated in the E-Document Audit Trail. The transaction must contain at least one line item.
The XML e-document contains only line items that are not in the reference purchase order.	The purchase order contains multiple line items.	All line items in the XML e-document are not in the reference purchase order.	The e-document template is valid with correct mapping.	Conversion will fail. Details of the error are indicated in the E-Document Audit Trail. The XML e-document must contain at least one line item from the reference purchase order.

## Conversion Scenarios with Purchase Order Expenses

Case	Details of Reference Purchase Order	Details of Inbound XML E-Document based from the Purchase Order	Details of Inbound E-Document Template	Expected Conversion Result
The XML inbound e-document is for partial billing. Not all purchase order expenses are billed.	The purchase order contains expenses with the required amount and corresponding account.	The inbound XML e-document does not include all amounts for each expense line.	The e-document template is valid with correct mapping.	Conversion will succeed. The created vendor bill has the amount of the expenses mapped to their corresponding accounts.
An extra expense amount is included in the inbound e-document.	The purchase order contains expenses with the required amount and corresponding account.	The inbound XML e-document contains an expense amount with no corresponding account.	The e-document template is valid with correct mapping.	Conversion will succeed. The expense amount with no corresponding account will automatically be assigned to the Default Expense Account specified in the vendor record.
The purchase order has expense amount and account, but the inbound XML e-document does not	The purchase order contains expenses with the required amount and corresponding account.	The inbound XML e-document does not have an expense included.	The e-document template is valid with correct mapping.	Conversion will only proceed if the inbound XML e-document includes at least one line item from the reference purchase order. If neither line item nor expense is

Case	Details of Reference Purchase Order	Details of Inbound XML E-Document based from the Purchase Order	Details of Inbound E-Document Template	Expected Conversion Result
have an expense included.				included in the inbound e-document, conversion will not proceed.
The purchase order has item lines and no specified expense amount and account, but the inbound e-document has expense amounts	The purchase order has no specified expense amount and account, but has at least one line item.	The inbound XML e-document has expense amounts included, without corresponding accounts.	The e-document template is valid with correct mapping.	Conversion will only proceed if there is at least one line item included, and a Default Expense Account is specified in the vendor record. The expense amounts in the inbound e-document will be assigned to the Default Expense Account.

## Converting Individual Inbound E-Documents into Vendor Bills

If the conditions and requirements for conversion have been met, you are ready to convert an inbound e-document into a vendor bill. See [Prerequisites and Conditions for Conversion](#)

### To convert an inbound e-document into a vendor bill

1. Go to the E-Documents Portlet on the Home page, and then click the number under **Inbound E-Documents for Conversion**.
2. On the Inbound E-Document for Conversion results page, click the View link of the inbound e-document that you want to convert into a vendor bill.
3. On the Inbound E-Documents page, click **Convert**.



**Note:** Before proceeding with conversion, make sure that the date format in the XML e-documents is the same as the date format set in Set Preferences.

The progress of conversion is displayed as a banner on the page. If conversion is successful, the created vendor bill is displayed. On the **E-Document** subtab of the vendor bill, the **Inbound E-Document** field indicates the record link of the converted inbound e-document. The **E-Document Audit Trail** subtab of the inbound e-document record will indicate the details of the successful conversion and the status is set to Converted. Details of any error or failure in conversion will also be listed in the e-document audit trail. If the converted inbound e-document has an attached PDF file reference, the attachment can be viewed and downloaded on the **E-Document** subtab of the created vendor bill.

## Converting an Inbound E-Document Without a Purchase Order Number

An inbound e-document without a purchase order number can be converted into a stand-alone vendor bill. Even if the inbound e-document is neither derived from, associated with, nor linked to an existing purchase order record, it can still be converted into a vendor bill. But the inbound e-document that will undergo conversion, must meet the following requirements:

- It must be a valid and well-formed XML document.

- It must include a reference number.
- It must include at least one item purchased or an expense.
- If it includes expense line, vendor record must have a Default Expense Account.

Convert this type of (stand-alone) inbound e-document by clicking **Convert** on the record. Or, use a script to run automatic bulk conversion of inbound e-document records with status For Conversion. For more information, see [Deploying Automatic Bulk Conversion Script for Inbound E-Documents](#).

## Converting Failed Inbound E-Documents

Inbound e-documents that initially failed conversion can be manually converted again, individually or in bulk, into vendor bills.

First, you must review the inbound e-documents that failed conversion and then resolve the errors in those e-documents. Without resolving the errors, the conversion process will fail again.

After resolving the errors, you need to search for the inbound e-documents that you fixed and finally proceed with the bulk conversion.

### To search and convert inbound e-documents that failed initial conversion:

1. Go to the Electronic Documents portlet on the Home page, and then click the number under **Convert Failed Inbound E-Documents**.

The Convert Inbound E-Documents page is displayed with search filters where you can define criteria for refining the search for failed inbound e-documents.

2. Enter a date in the **Date Created From** and **Date Created To** fields.



**Note:** The Date Created From must be an earlier date than the Date Created To.

The combined dates that you entered define a date range during which all inbound e-documents created within that period will be displayed as the result.

3. (Optional) Select a vendor in the **Vendor** field

The Vendor field further refines your search criteria by searching for inbound e-documents from a particular vendor.



**Important:** The search will not proceed if any inbound e-document within the specified date range or from the selected vendor, is undergoing the conversion process at the same time. In this case, a message is displayed instructing you to change your search criteria or try again later when the ongoing conversion is finished.

4. Click **Search**.

A results page is displayed listing the failed inbound e-documents that meet your search criteria. The results page displays the first 25 inbound e-documents, if there are more inbound e-documents found, they are displayed on succeeding pages. On the results page, you can click the Internal ID link of an inbound e-document is a link that to open the inbound e-document record.

5. Click **Convert**.

A message is displayed on the banner, confirming that the e-documents are being converted.

After the conversion is completed, an email notification is sent to the user who performed the conversion and to the email address in the Email Recipient Notification setup, if any is specified. If errors are encountered during conversion, the email will have an attached CSV file listing the inbound e-documents that underwent conversion, including columns for the Internal ID, Vendor, and details of the conversion process and errors.

For more information about choosing the recipient of notifications, see step 4 of [Prerequisites for Using Electronic Invoicing](#).

**Note:** You can also convert inbound e-documents that failed conversion, individually or one at a time. For more information, see [Converting Individual Inbound E-Documents into Vendor Bills](#).

## Compatibility of Approval Workflows with Vendor Bill Conversion

If Vendor Bill Approval Workflow is enabled, it is automatically applied to vendor bills created from inbound e-document conversion, without further configuration.

With approval workflow enabled, a vendor bill created from inbound e-document conversion is also assigned an approval status. If the new vendor bill is still pending approval when it was created, the approval process is continued. The vendor bill is then properly routed to the next approver defined in the workflow. For more information, see the help topic [Vendor Bill Approval Workflow](#).

If 3 Way Match Vendor Bill Approval Workflow is enabled, you can determine the discrepancy between the created vendor bill and the reference purchase order. Clicking Bill Exception on the new vendor bill will display any discrepancy resulting from the validation performed by the approval workflow against set exception criteria. A vendor bill and its corresponding purchase order may have discrepancies in terms, quantity tolerance, quantity difference, or amount. For more information, see the help topic [3 Way Match Vendor Bill Approval Workflow](#).

**Note:** The standard workflow included in the Vendor Approval SuiteApp does not support checking for tax amount discrepancy. If you want to enable checking for tax amount discrepancy, see the help topic [Customizing for the Vendor Bill Approval Workflow](#).

## Cancelling Inbound E-Documents

You can cancel an inbound e-document if you do not want to convert it into a NetSuite transaction.

To cancel an inbound e-document, view or open the inbound e-document you want to cancel and then click **Cancel**. A banner is displayed on the inbound e-document confirming cancellation of the record. The E-Document Audit Trail on the E-Document subtab also indicates the details of cancellation. Canceled e-documents cannot be converted to NetSuite records anymore.

You cannot cancel an inbound e-document if it is already converted or canceled. A canceled or converted e-document does not display the Cancel button. Cancellation will also fail if the inbound e-document is concurrently being converted or canceled by another user.

If cancellation of the record failed, a banner is displayed on the inbound e-document indicating the failure. The E-Document Audit Trail on the E-Document subtab also indicates the cause of the failure to cancel the inbound e-document.

## Electronic Invoicing Errors

To understand outbound and inbound e-document processing errors, see the following topics:

- [Electronic Invoicing Error Codes](#)
- [Outbound E-Document Generation Errors](#)
- [Outbound E-Document Sending Errors](#)
- [Inbound E-Document Conversion Errors](#)

## Electronic Invoicing Error Codes

Error code	Message	Description	Solution
EI_TEMPLATE_CSV_ERROR	The XML template contains errors. XML format must be well-formed.	The XML input for Template for Outbound E-Invoices field is invalid.	You must fix the XML input for the <b>Template for Outbound E-Invoices</b> field.
EI_TEMPLATE_VALIDATOR_CSV_ERROR	The <b>REGEX</b> field contains an incorrect regular expression. Proper syntax must be used.	The Regex input for XML validators is invalid.	You must fix the regex input for <b>Regex Validation Expression</b> field in the XML Validators sublist of the template.
EI_INACTIVE_CUSTOMER	Transactions with inactive customers are not supported by e-document.	The customer who owns the transaction is inactive.	You must remove the e-document template from the transaction or activate the inactive customer.
EI_SENDING_NO_RECIPIENTS	The e-document cannot be sent because the customer has no email address. Before you can send this e-document by email, an email address must be provided on the customer record.	The customer (individual) does not have any email address specified in the Customer record.	You must enter a valid email address in the Customer record.
EI_SENDING_NO_RECIPIENTS	There are no e-document recipients for this customer. To send electronic documents by email to this customer, at least one contact must be added to the list of e-document recipients.	The customer (company) does not have any e-document recipients defined in the Customer record.	You must enter e-document recipients in the Customer record.
EI_SENDING_RECIPIENT_NO_EMAIL	One or more recipients of the e-document, associated with this transaction, does not have an email address. Verify that the recipients for this customer have valid email addresses.	The customer (company) has an e-document recipient that does not have an email address.	You must enter a valid email address in the e-document recipient in the Customer record.
EI_SENDING_INVALID_METHOD	Select a valid sending method for {TYPE} #{TRANSACTIONNUMBER}.	The transaction does not have an e-document sending method defined.	You must enter an e-document sending method in the Transaction record.
EI_SEND_INVALID_RESULT	The plug-in failed to return a valid result.	The plug-in used for sending, did not return a result.	You must fix the plug-in implementation so that it will return a result object.
EI_SENDING_PLUG_IN_ERROR	<ERROR MESSAGE>	The plug-in used for sending, encountered an error.	You must fix the plug-in implementation based on the error message.
PROBLEM_LOADING_PLUG_IN	E-document Sending encountered an error while loading a custom plug-in. Error Code: <ERROR CODE> Message: <MESSAGE>	The plug-in file used for sending, did not load properly.	You must fix the plug-in implementation based on the error message.
EI_CANNOT_CREATE_DEFAULT_DOCUMENT_PACKAGE	The {DEFAULT_DOCUMENT_PACKAGE} record already exists. You cannot create an e-document package record with the same name. Rename your e-document package record and try again.	Creating an e-document package with the name "Default E-Document Package" is not allowed. (this error is shown in CSV import).	You must create a new e-document package with another name.

Error code	Message	Description	Solution
EI_CANNOT_EDIT_DEFAULT_DOCUMENT_PACKAGE	Editing the {DEFAULT_DOCUMENT_PACKAGE} record Name or Description is not allowed.	Editing the e-document package, "Default E-Document Package", is not allowed. (this error is shown in CSV import).	Avoid editing the default e-document package record.
EI_CANNOT_DELETE_DEFAULT_DOCUMENT_PACKAGE	Deleting the {DEFAULT_DOCUMENT_PACKAGE} record is not allowed.	Deleting the e-document package, "Default E-Document Package", is not allowed. (this error is shown in CSV import)	Avoid deleting the default e-document package record.
EI_VENDOR_CODE_FIELD_NOT_FOUND	The vendorcode field is missing in the e-document template. Modify the e-document template or select another template that includes vendorcode field mapping.	If the Multiple Vendor feature is enabled in the account, the vendorcode must be indicated in the Mapping Format of the E-Document Template that is being used for Inbound E-Document conversion.	You must add the mapping for vendorcode field in the Field Mapping for Inbound E-Documents in the template.
EI_NO_VENDOR_CODE_VALUE	At least one of the items has no vendor code. Cancel this e-document and submit another e-document with the correct value for the XML element mapped to the vendor code field.	The vendorcode of one of the items has no value. This error occurs if the Multiple Vendor feature is enabled in the account.	You must modify the XML by defining a value for the XML that is mapped to the vendorcode field.
EI_VENDOR_NAME_FIELD_NOT_FOUND	The vendorname field is missing in the e-document template. Modify the e-document template or select another template that includes vendorname field mapping.	If the Multiple Vendor feature is disabled in the account, the vendorname field must be indicated in the Mapping Format of the E-Document Template that is being used for Inbound E-Document conversion.	You must add the mapping for the vendorname field in the Field Mapping for Inbound E-Documents in the template.
EI_NO_VENDOR_NAME_VALUE	At least one of the items has no vendor name/code. Cancel this e-document and submit another e-document with the correct value for the XML element mapped to the vendor name/code field.	The vendorname of one of the items has no value. This error occurs if the Multiple Vendor feature is disabled in the account.	You must modify the XML by defining a value to the XML that is mapped to the vendorname field.
EI_CREATED_FROM_TRANSACTION_NOT_FOUND	Record {{TRANSTYPE}#{TRANSID}} was not found in the system. Cancel this e-document and submit another e-document with the correct value for the XML element mapped to the createdfrom field.	The transaction that the vendor bill will be created from is not found in the system.	You must define the correct value for the XML that is mapped to the createdfrom field.
EI_CREATED_FROM_TRANSACTION_ENTITY_NOT_MATCH	Record {{TRANSTYPE}#{TRANSID}} is assigned to a different entity. Select the correct entity and convert this e-document.	The referenced transaction has a different entity compared to the inbound e-document.	You must select the correct entity and convert the e-document again.
EI_TRANSACTION_REF_NUM_ALREADY_EXISTS	A vendor bill with the same reference number already exists. Cancel this e-document and submit another e-document with the correct reference number value for the XML element mapped to the tranid field.	This error occurs during conversion of inbound e-document to vendor bill. The SuiteApp will not proceed with conversion if the reference number on the inbound e-document already exists in one of the vendor bills.	You must define another value for the XML that is mapped to the tranid field.

Error code	Message	Description	Solution
EI_TRANSACTION_REF_NUM_NOT_FOUND	The required reference number is missing in the inbound e-document. Cancel this e-document and submit another e-document that includes an XML element for the reference number, mapped to the tranid field.	Reference number must be present in the XML. It has to be mapped to the E-Document template tranid field.	You must add an XML that maps to the tranid field.
EI_TRANSACTION_ITEM_ERROR	The following vendor codes: {ITEMLIST}, are associated with multiple item records. Modify the item records and ensure that vendor codes are unique for each item per vendor.	There are items in the XML (in the inbound e-Document) that have multiple matches in the item record.	You must edit the vendor codes of the duplicate item records to ensure the uniqueness of the items.
EI_TRANSACTION_ITEM_ERROR	The following vendor name/codes: {ITEMLIST}, are associated with multiple item records. Modify the item records and ensure that vendor name/codes are unique for each item per vendor.	There are items in the XML (in the inbound e-document) that have multiple matches in the item record.	You must edit the vendor name/codes of the duplicate item records to ensure the uniqueness of the items.
EI_TRANSACTION_ITEM_ERROR	The following vendor codes: {ITEMLIST}, are not associated with any item records.	There are items in the XML (in the inbound e-document) that have no match in the system.	You must create an item record for those items with no matches.
EI_TRANSACTION_ITEM_ERROR	The following vendor name/codes: {ITEMLIST}, are not associated with any item records.	There are items in the XML (in the inbound e-document) that have no match in the system.	You must create an item record for those items with no matches.
EI_TRANSACTION_ITEM_ERROR	There are no items in the inbound e-document that is included in the referenced transaction. Check the status of the referenced transaction if it can be transformed. If it can be transformed, cancel this e-document and submit another e-document with the correct value for the XML element mapped to the createdfrom field.	The user cannot convert the transaction or the referenced transaction does not have any matching items to the items in the inbound e-document.	A different transaction must be referenced or a different inbound e-document with the correct items must be used.
EI_SENDING_IN_PROGRESS	The system cannot perform a search using the filters you selected because e-document sending is already in progress for transactions within the date range {{TRANDATE_FROM}} - {{TRANDATE_TO}} for subsidiary {{SUBSIDIARY}}. Please change your search criteria or try again later.	The records you are searching for might already be undergoing the sending process.	You can change the parameters for sending, or wait for the current sending process to finish before sending again.
EI_CONVERSION_IN_PROGRESS	The system cannot perform a search using the filters you selected because inbound e-document conversion is already in progress within the date range {{DATECREATED_FROM}} - {{DATECREATED_TO}}. Please change your search criteria or try again later.	The records you are searching for might already be undergoing the conversion process.	You can change the parameters for conversion, or wait for the current conversion process to finish before converting again.
EI_CERTIFICATION_SENDING_METHOD_ALREADY_EXISTS	You cannot set this sending method as the Certification Sending Method because '{CERTIFICATION_SENDING_METHOD_NAME}' is already selected as the Certification Sending Method for the Subsidiaries [{SUBSIDIARIES}] and transactions [{TRANSACTIONS}]. To save this record, you must unassign '{CERTIFICATION_SENDING_METHOD_	A certification sending method is assigned to a combination of a transaction and subsidiary that already has a certification sending method. Only one certification sending method can be assigned to	You must unassign the existing certification sending method or make it inactive, to be able to assign a new certification sending method and save it.

Error code	Message	Description	Solution
	NAME}' as a Certification Sending Method, or make it inactive.	a particular combination of transaction and subsidiary.	
EI_TEMPLATE_ERROR	<p>Any of the following error messages can be displayed:</p> <ul style="list-style-type: none"> <li>■ You selected an outbound transaction type, but the template content in the <b>Template for Outbound E-Documents</b> field is missing. Enter the XML or JSON content in the field and try again.</li> <li>■ There are missing field values. For an outbound transaction, specify a valid XML or JSON content in the <b>Template for Outbound E-Documents</b> field. For an inbound transaction, specify the JSON content in the <b>Field Mapping for Inbound E-Documents</b> field.</li> <li>■ The selected XSD file is not a valid XSD file. Ensure that the file you select has the .xsd extension.</li> </ul>	<p>Any of these errors will be displayed if validations related to template are not met. For example, if any of the required fields <b>Template for Outbound E-Documents</b> or <b>Field Mapping for Inbound E-Documents</b> are missing, or when an invalid XSD file is selected.</p>	You must provide the missing fields or select a valid XSD file.

## Electronic Invoicing Common Errors

Message	Error on User Interface	Description	Solution
An error occurred during conversion. Check the E-Document Audit Trail on the E-Document subtab for details.	Banner on the record	A red banner is displayed on the inbound e-document record indicating failed conversion.	Check the audit trail for more information.
Unable to convert this inbound e-document because the selected vendor is inactive. The E-Document Status field has not been updated and an audit trail has not been created. Clear the Inactive box on the vendor record, then try converting the e-document again.	Banner on the record	A yellow banner is displayed on the inbound e-document record indicating that an inactive vendor is selected.	Activate the inactive vendor before converting the inbound e-document.
Unable to convert this inbound e-document because the selected customer is inactive. The E-Document Status field has not been updated and an audit trail has not been created. Clear the Inactive box on the customer record, then try converting the e-document again.	Banner on the record	A yellow banner is displayed on the inbound e-document record indicating that an inactive customer is selected.	Activate the inactive customer before converting the inbound e-document.
Parsing failure. Check the Field Mapping for Inbound E-documents.	Audit Trail log	The Audit Trail of the inbound e-document logs this error after a failed conversion due to failure in XML parsing.	Check the inbound template for errors.
Conversion failure.	Audit Trail log	The Audit Trail of the inbound e-document logs	Check the inbound e-document for errors.

Message	Error on User Interface	Description	Solution
		the error after the failed conversion process.	
Conversion failed because the status of the inbound e-document record is '{STATUS}'	Audit Trail log	The Audit Trail of the inbound e-document logs the error after a failed conversion due to a wrong status detected during conversion process.	A concurrent conversion process might be running, or the record is already cancelled when conversion was started. You can wait for the other conversion process to finish, or verify if the record has been canceled.
An error occurred during cancellation. Check the E-Document Audit Trail on the E-Document subtab for details.	Banner on the record	A banner is displayed indicating an error in cancellation.	You must check the inbound e-document for errors; most likely, the error is due to an invalid status.
The selected XML File Reference is not a valid XML file. Ensure that the file you select has the .xml extension.	Popup message	The inbound e-document record has an invalid e-document file.	You must make sure that the file attached to the inbound e-document is an XML file.
The selected XML File Reference is not a well-formed XML document.	Popup message	The inbound e-document record has a malformed XML file.	You must check the attached XML file if it is well-formed.
The Transaction Date From must not be later than the Transaction Date To. Change the dates so that the Transaction Date From is earlier than the Transaction Date To.	Popup message	The selected Transaction Date To is earlier than the Transaction Date From.	You must change the selected dates to make the Transaction Date From earlier than the Transaction Date To.
The Date Created From must not be later than the Date Created To. Change the dates so that the Date Created From is earlier than the Date Created To.	Popup message	The selected Date Created To is earlier than the Date Created From.	You must change the selected dates to make the Date Created From earlier than the Date Created To.
The selected XSD file is not a valid XSD file. Ensure that the file you select has the .xsd extension.	Popup message	This message is displayed while creating/editing a template record when the selected file is not an XSD file.	You must select an XSD file
There is no e-document email sender for this vendor. To receive e-documents through email from this vendor, you must enter at least one email address in the Vendor E-Document Email Sender list.	Popup message	The vendor does not have any e-document senders defined in the vendor record. This error appears when saving the record with Use Sender List box checked.	You must add at least one sender email address, or clear the box and input a sender domain.
The sender email address already exists.	Popup message	The sender email address entered in the sublist is already existing for the same vendor.	You must enter a different email address or remove the current entry.
The sender email domain is already being used by a different vendor.	Popup message	The sender domain is already being used by another vendor.	You must enter a different domain, or use the sender list to enter specific email addresses.

Message	Error on User Interface	Description	Solution
This account does not have an active license to use the Electronic Invoicing SuiteApp in multiple countries. To convert e-documents in bulk, please contact your account administrator to configure the E-Document Country for Free Use on the Company Information page.	Popup message	This message is displayed when trying to convert failed inbound e-documents in bulk using the Suitelet.	The administrator must set up the E-Document Country for Free Use field.
This account does not have an active license to use the Electronic Invoicing SuiteApp in multiple countries. To convert this e-document to a transaction, please contact your account administrator to specify a country in the E-Document Country for Free Use field on the Company Information page.	Popup message	This message is displayed when trying to convert an inbound e-document individually from the record page.	The administrator must set up the E-Document Country for Free Use field.
This account does not have an active license to use the Electronic Invoicing SuiteApp in multiple countries. To convert this e-document to a transaction, please contact your NetSuite account manager to purchase a license.	Popup message	This message is displayed when trying to convert an inbound e-document individually from the record page.	Ask the account manager to purchase a license.
This account does not have an active license to use the Electronic Invoicing SuiteApp in multiple countries. To convert this e-document to a transaction, please set up the default billing address of the selected vendor.	Popup message	This message is displayed when trying to convert an inbound e-document individually from the record page.	You must add a default billing address to the vendor of this e-document.

## Outbound E-Document Generation Errors

The following errors can occur during the generation of e-documents:

- [Malformed Template](#)
- [Malformed XPath](#)
- [Malformed Regex](#)
- [Unsupported XPath](#)
- [Data Failed Validation Checking](#)
- [Inactive Customer Record](#)

### Malformed Template

Generation of e-documents will fail if the selected template contains FreeMarker syntax errors.

To fix this error, select a different template or update the content of the e-document template, then try generating the e-document again.

Make sure that e-document templates are well-formed.

Click the link to the e-document template in the **E-Document Template** field to view and edit your e-document template.

For more information, see [XPath and Regex Examples for E-Document Templates](#).

## Malformed XPath

Generation of e-documents will fail if there is no xpath to validate your e-document template, or the xpath format is invalid.

To fix this error, select a different template or verify that an xpath exists and the format is correct in your e-document template, then try generating the e-document again.

Make sure that e-document templates use the proper syntax for the xpath.

Click the link to the e-document template in the **E-Document Template** field to view and edit your e-document template.

For more information, see [XPath and Regex Examples for E-Document Templates](#).

## Malformed Regex

Generation of e-documents will fail if the regex validation expression that validates your e-document template is invalid.

To fix this error, select a different template or check the regex validation expression's format in your e-document template, then try generating the e-document again.

Make sure that e-document templates use the proper syntax for the regex validation expression.

Click the link to the e-document template in the **E-Document Template** field to view and edit your e-document template.

For more information, see [XPath and Regex Examples for E-Document Templates](#).

## Unsupported XPath

Generation of e-documents will fail if the e-document template does not contain the information required by the xpath used to validate it.

To fix this error, select a different template, update the e-document template, or delete the xpath, then try generating the e-document again.

Click the link to the e-document template in the **E-Document Template** field to view and edit your e-document template.

For more information, see [XPath and Regex Examples for E-Document Templates](#).

## Data Failed Validation Checking

Generation of e-documents will fail if the transaction does not contain the information needed by the e-document template you selected. For example, if your e-document template requires a shipping address with a 5-digit postal code, but the shipping address on your transaction record has a 4-digit postal code, an error will occur.

To fix this error, select a different template, update the transaction record with the required information, or update the e-document template validations, then try generating the e-document again.

## Inactive Customer Record

Generation of e-documents will fail if the customer associated with the transaction record is inactive.

To fix this error, set the **E-Document Template** field on the transaction to blank. If the **E-Document Template** field is blank, the system will not generate an e-document for the transaction, and no error message will be shown. If you intend to generate an e-document for the transaction, you must make the customer record active, then try generating the e-document again.

To activate the customer record, go to Lists > Relationships > Customers and find the customer record. Click **Edit**, then go to the **System Information** subtab and clear the **Inactive** box.

To regenerate e-documents, see the following topics:

- [Regenerating E-Documents for Single Transactions](#)
- [Generating and Regenerating E-Documents in Bulk](#)

## Outbound E-Document Sending Errors

Errors can occur when sending e-documents. When the system encounters an error, NetSuite sends an email notification containing the error details to the user who initiated the sending process.

An audit trail containing the error details is also created on the **E-Document Audit Trail** subtab on the transaction record.

Fixing errors may require an Administrator role. Be sure to inform your account administrator about the error immediately. If the error is not fixed, the e-document cannot be sent.

The following errors can occur when sending e-documents:

- [Invalid Sending Method](#)
- [Invalid E-Document Sender](#)
- [Invalid Email Recipient](#)
- [Plug-in Script Error](#)

## Invalid Sending Method

Sending of e-documents will fail if no sending method has been selected on the transaction record.

To fix this error, edit the transaction record and select a sending method, then try sending the e-document again.

## Invalid E-Document Sender

Sending of e-documents will fail if the e-document sender has no email address.

To fix this error, change the e-document sender or make sure the e-document sender has a valid email address, then try sending the e-document again.

To view the **E-Document Sender** field, go to Setup > Company > Company Information.

If you have a OneWorld account, go to Setup > Company > Classifications > Subsidiaries. Then, click the subsidiary to view the assigned employee in the **E-Document Sender** field.

To add or change the email address, go to Lists > Employees and edit the employee record.

## Invalid Email Recipient

Sending of e-documents will fail if the selected sending method has no email recipients or if one or more email recipients have no email addresses.

To fix this error, select a different sending method or make sure the selected sending method has at least one email recipient with an email address. If the customer is a company, there should be at least one email recipient defined on the **E-Document Email Recipient** subtab on the customer record. If the customer is an individual, enter the email address.

To view and add email recipients for the selected sending method, go to the **E-Document** subtab of the customer record. On the **E-Document Email Recipient** field, verify that your email recipients have email addresses. Click **New E-Document Email Recipient** to add an email recipient. Only contacts associated with the customer record are available for selection in the dropdown list.

## Plug-in Script Error

Sending of e-documents will fail if the selected custom sending method is invalid.

To fix this error, make sure that the script of your custom sending method is valid.

For more information, see [Creating Custom Methods for Sending E-Documents](#).

## Inbound E-Document Conversion Errors

Errors can occur when converting e-documents. When an error is encountered in automatic scheduled bulk conversion, the system sends an email notification with error details to the Recipient of E-Document Notifications. If the Recipient of E-Document Notifications is not assigned, the system sends the notification to all active administrators.

An audit trail containing the error details is also created on the **E-Document Audit Trail** subtab on the inbound e-document record.

Fixing errors may require an Administrator role. Be sure to inform your account administrator about the error immediately. If the error is not fixed, the e-document cannot be converted.

The following errors can occur when converting inbound e-documents:

- Purchase Order is not Ready
- Incorrect PO Number
- Item vendorcode is not Unique
- Duplicate Reference Number
- Incorrect Mapping to Transaction Fields
- Mapping to Required Vendor Bill Fields is Missing
- Missing Default Expense Account

## Purchase Order is not Ready

Inbound e-document conversion will fail, if the e-document to be converted was generated from a purchase order that is not ready for billing, closed or fully billed.

To fix this error, ensure that the PO status is ready for billing.

## Incorrect PO Number

Inbound e-document conversion will fail, if the e-document to be converted was generated from a purchase order with incorrect PO Number.

To fix this error, contact the vendor or party whom the e-document came from, inform them to make corrections to the PO Number, and have them send the amended XML file.

## Item vendorcode is not Unique

Inbound e-document conversion will fail, if the e-document to be converted contains items whose vendorcode are not unique.

To fix this error, update the item records and ensure that each item has a unique vendorcode.

## Duplicate Reference Number

Inbound e-document conversion will fail, if the Reference Number of an e-document to be converted has a duplicate. Duplicate Vendor Bill Detection is enabled for the Electronic Invoicing SuiteApp, so a duplicate Reference Number and vendor will cause the error.

To fix this error, review the existing bill, and then determine if it is duplicate. If it is a duplicate, cancel the e-document. If it is not duplicate, ask the vendor to resend an updated XML document, and then cancel the e-document.

## Incorrect Mapping to Transaction Fields

Inbound e-document conversion will fail, if the JSON template has incorrect mapping to transaction fields.

To fix this error, review the inbound e-document template and make sure that JSON objects map to the correct vendor bill transaction fields.

## Mapping to Required Vendor Bill Fields is Missing

Inbound e-document conversion will fail, if the JSON template is missing the mapping to required vendor bill fields.

To fix this error, review the inbound e-document template and define the JSON objects that will map to required vendor bill transaction fields.

## Missing Default Expense Account

Inbound e-document conversion will fail, if the XML contains an expense line but the vendor record does not have a default expense account specified. To fix this error, edit the vendor record and on the **Financial** tab, specify a Default Expense Account.