

# Oracle

## Financials Cloud

# Implementing Receivables Credit to Cash

---

**Release 12**

This guide also applies to on-premises implementations

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 installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. 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 Xeon 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, Opteron, the AMD logo, and the AMD Opteron 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.

The business names used in this documentation are fictitious, and are not intended to identify any real companies currently or previously in existence.

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

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.

# Contents

## Preface

i

<b>1</b>	<b>Define Common Accounts Receivable Configuration</b>	<b>1</b>
	Simple Configuration to Operate Receivables: Explained	1
	Predefined Receivables Data in Subledger Accounting: Explained	2
	Notes Mapping for Receivables: Explained	12
	Setting Up a Salesperson: Procedure	12
	FAQs for Define Common Accounts Receivable Configuration	14
	Legal Entity Document Sequencing	14
	Define Receivables Activities	18
	Define AutoCash Rule Sets	21
	Define Approval Limits	29
	FAQs for Distribution Sets	30
	Define Statements	30
	FAQs for Standard Messages	34
<b>2</b>	<b>Manage Receivables System Options</b>	<b>35</b>
	Updating System Option Records: Critical Choices	35
	Using Header Level Rounding: Example	36
	Tax Invoice Printing Options	37
	Tuning Segments: Explained	38
	Log File Message Levels: Explained	39
	FAQs for Manage Receivables System Options	40
	Define E-Mail Delivery	41
	Define B2B XML	45

<b>3</b>	<b>Define Customer Billing Configuration</b>	<b>51</b>
	Define AutoInvoice	51
	Define Payment Terms	58
	Define AutoAccounting	64
	Define Transaction Types	70
	Define Transaction Sources	74
	Define Memo Lines	79
	Define Balance Forward Billing Cycles	80
	FAQs for Salesperson Account References	83
	FAQs for Remit-to Addresses	83
<b>4</b>	<b>Define Customer Payments</b>	<b>85</b>
	Managing Shared Services in Receivables: Procedures	85
	Define Application Rule Sets	86
	Define Receipt Classes and Methods	92
	FAQs for Receipt Sources	96
	Define Lockbox	97
	Define Transmission Formats for Lockbox	125
	Define AutoMatch Rule Sets	134
	Define Application Exception Rule Sets	139
	Define Customer Paying Relationship Assignments	141
<b>5</b>	<b>Configure Payment System Connectivity</b>	<b>145</b>
	Validations: Critical Choices	145
	Setting Up Formats: Explained	146
	Transmission Protocol: Explained	149
	Transmission Configuration: Explained	149
	Configuring Pretty Good Privacy (PGP) Encryption and Digital Signature for Outbound and Inbound Messages: Explained	150
	Testing the Transmission Configuration: Explained	153
	Configuring a Communication Channel to a Payment System: Explained	154
	Setting Up a Payment System: Explained	155
	Payment System Account: Explained	157
	Auditing Payments Business Objects	158
	Importing a Security Credential File: Procedures	163
	FAQs for Configure Payment System Connectivity	165



<b>6</b>	<b>Define Funds Capture</b>	<b>167</b>
	Funds Capture Process Profile: Explained	167
	Settlement Grouping Rules: Example	168
	Routing Rules: Explained	169
	Using Oracle BI Publisher Enterprise to Modify Templates for Use with Formats: Explained	170
	FAQs for Define Funds Capture	172
<b>7</b>	<b>Define Payments Security</b>	<b>175</b>
	System Security Options: Critical Choices	175
	Enabling Encryption of Sensitive Payment Information: Procedure	178
<b>8</b>	<b>Define Customer</b>	<b>181</b>
	Define Customer Account	181
	Manage Receivables Customer Profile Classes	184
	Manage Customers	188
<b>9</b>	<b>Define Bills Receivable</b>	<b>197</b>
	Updating System Options for Bills Receivable: Explained	197
	Setting Up a Bills Receivable Creation Receipt Method: Procedure	197
	Preparing Transactions for Bills Receivable Batches: Critical Choices	198
	Setting Up a Bills Receivable Remittance Receipt Method: Explained	201
	Setting Up a Remittance Bank Account for Bills Receivable: Explained	202
	FAQs for Define Bills Receivable	203
<b>10</b>	<b>Define Revenue Management Configuration for Receivables</b>	<b>205</b>
	Evaluating Revenue Policy: Points to Consider	205
	Event-Based Revenue Management: How It Works	206
	Revenue Contingencies: Explained	207
	Payment-Based Contingencies: Explained	208
	Removal Events: Explained	209
	Using Revenue Contingency Assignment Rules: Examples	210
	FAQs for Define Revenue Management Configuration for Receivables	211

<b>11</b>	<b>Define Cash Management and Banking Configuration</b>	<b>213</b>
	Bank, Branch, and Account Components: How They Work Together	213
	Creating Accounts: Points to Consider	214
	Parse Rule Sets: Overview	215
	Transaction Type Mapping: Overview	216
	Tolerance Rules: Overview	217
	Reconciliation Matching Rules: Explained	218
	Reconciliation Rules Sets: Overview	221
	Bank Statement Transaction Codes: Overview	222
	Bank Statement Transaction Creation Rules: Overview	222
	Create Banks, Branches, and Accounts in Spreadsheet	222
	Setting Up Cash Positioning and Forecasting	224
	Bank Account Validation	227
<b>12</b>	<b>Define Advanced Collections Configuration</b>	<b>285</b>
	Manage Collections Preferences	285
	Manage Aging Methods	289
	Manage Collectors	292
	Manage Dunning Configuration	301
	Define Scoring	303
	Define Strategy	312
<b>13</b>	<b>Define Revenue Management</b>	<b>319</b>
	Source Document Types: Explained	319
	System Options: Explained	319
	Contract Identification Rules: Explained	320
	Performance Obligation Identification Rules: Explained	321
	Creating Pricing Dimension Structures and Value Sets: Explained	322
	Creating a Pricing Dimension Structure and Value Sets: Worked Example	323
	Pricing Dimensions: Explained	325
	Understanding Pricing Dimension Bands: Explained	327
	Item Groups: Explained	329
	Standalone Selling Price Profiles: Explained	330
	Standalone Sales Pool Exclusion Rules: Explained	330
	FAQs for Define Revenue Management	331


# Preface

This preface introduces information sources that can help you use the application.

## Oracle Applications Help

Use the help icon  to access Oracle Applications Help in the application. If you don't see any help icons on your page, click the Show Help icon  in the global header. Not all pages have help icons. You can also access Oracle Applications Help at <https://fusionhelp.oracle.com>.

## Using Applications Help

 **Watch:** This video tutorial shows you how to find help and use help features.

## Additional Resources

- **Community:** Use [Oracle Applications Customer Connect](#) to get information from experts at Oracle, the partner community, and other users.
- **Guides and Videos:** Go to the [Oracle Help Center](#) to find guides and videos.
- **Training:** Take courses on Oracle Cloud from [Oracle University](#).

## Documentation Accessibility

For information about Oracle's commitment to accessibility, see the [Oracle Accessibility Program](#).

## Comments and Suggestions

Please give us feedback about Oracle Applications Help and guides! You can send e-mail to: [oracle\\_fusion\\_applications\\_help\\_ww\\_grp@oracle.com](mailto:oracle_fusion_applications_help_ww_grp@oracle.com).





# 1 Define Common Accounts Receivable Configuration

## Simple Configuration to Operate Receivables: Explained

You can create an operational Oracle Fusion Receivables environment with seven configurations. The remaining configurations are either optional or have predefined values.

If applicable, your Receivables configuration must include a plan to migrate your customer information from your legacy system.

### Receivables Configuration Tasks

There are seven configuration tasks necessary to create an operational Receivables environment. Before you perform these tasks, you must ensure that you have completed all of the required implementation tasks for Oracle Fusion Financials.

Perform these seven tasks in the order indicated:

1. Set Receivables System Options

Set Receivables system options to customize your Receivables environment. During Receivables setup, you specify your accounts, customer and invoice parameters, and how the AutoInvoice and Automatic Receipts programs operate.

2. Define Receivables Activities

Define receivables activities to create default accounting for all activities other than transactions and receipts, including, for example, miscellaneous cash, discounts, late charges, adjustments, and write-offs.

3. Define AutoAccounting Rules

Defining AutoAccounting is a required configuration task for processing customer billing.

Define AutoAccounting to specify how you want Receivables to determine the default general ledger accounts for transactions. Receivables creates default accounts for revenue, receivable, freight, tax, unearned revenue, unbilled receivables, late charges, and AutoInvoice clearing (suspense) accounts using your AutoAccounting setup.

4. Define Receipt Classes and Methods

Defining receipt classes and receipt methods is a required configuration task for processing customer payments.

Receipt classes determine the required processing steps for receipts to which you assign receipt methods with this class. These steps include confirmation, remittance, and clearance. Receipt methods specify accounting for receipt entries and applications, determine customer remittance bank account information, and configure automatic receipt processing and fund transfer error handling.

5. Define Remit-to Addresses

Define remit-to addresses to let your customers know where to send payment for open receivables. Receivables uses the addresses to provide default remit-to information when you enter transactions.

You must provide a remit-to address to complete a transaction.

If you use AutoInvoice, but you haven't defined a remit-to address for a particular customer site, AutoInvoice rejects all transactions for which it couldn't determine a remit-to address.

6. Define Approval Limits
- Define approval limits to determine whether a Receivables user can approve adjustments or credit memo requests. You define approval limits by document type, amount, and currency.
7. Define Statement Cycles
- Define statement cycles to control when you create customer statements. You assign statement cycles to customer profiles.


Related Topics

- [How can I use remit-to addresses?](#)
- [AutoAccounting Account Types and Segment Values](#)
- [Updating System Option Records: Critical Choices](#)

# Predefined Receivables Data in Subledger Accounting: Explained

Oracle Fusion Receivables provides predefined data for Oracle Fusion Subledger Accounting that you can use to integrate the two applications.

When you run Create Receivables Accounting, the program accepts the default accounting information from AutoAccounting without change and uses the predefined data to create accounting in the subledger. Subledger Accounting transfers the final accounting to Oracle Fusion General Ledger.

 **Note:**

You can optionally define your own subledger accounting rules to overwrite the default accounts from the accounting events.

Receivables predefines one application in Subledger Accounting named **Receivables**. Most of the data that Receivables predefines for Subledger Accounting is associated with the **Receivables** application.

This table shows the attribute values that Receivables predefines for the **Receivables** application:

Field	Value
Application Name	Receivables
Drilldown Procedure	AR_DRILLDOWN_PUB_PKG.DRILLDOWN
Use Security	Yes
Policy Function	XLA_SECURITY_POLICY_PKG.MO_POLICY
Journal Source	Receivables

Field	Value
Third Party Control Account Type	Customer
Subject to Validation	No
Calculate Reporting Currency Amounts	Yes

This table lists the setup information that Receivables predefines for the event entities:

Application	Entity Name	Description	Gapless Event Processing
Receivables	Adjustments	Adjustments	No
Receivables	Receipts	Receipts	No
Receivables	Transactions	Transactions	No

Receivables predefines process categories for the **Receivables** application. These process categories are:

- Adjustments
- Miscellaneous Receipts
- Standard Receipts
- Third Party Merge
- Transactions

Additional considerations for Receivables predefined data for subledger accounting include:

- Event Classes and Event Class Options
- Sources, Source Assignments, and Accounting Attribute Assignments
- Journal Line Types
- Account Derivation Rules
- Journal Lines Definitions
- Application Accounting Definition

## Event Classes and Event Class Options

Receivables predefines event classes for each event entity that belongs to the **Receivables** application.

This table lists the event classes that Receivables predefines for the **Receivables** application:

Entity	Event Class Name
Adjustments	Adjustment
Receipts	Miscellaneous Receipt

Entity	Event Class Name
Receipts	Receipt
Transactions	Chargeback
Transactions	Credit Memo
Transactions	Debit Memo
Transactions	Invoice

Accounting event class options define attributes of an event class. Receivables defines the accounting event class options for each predefined event class.

This table lists the accounting event class options that Receivables predefines for the **Receivables** application:

Event Class	Process Category	Default Journal Category	Transaction View	Balance Types
Adjustment	Adjustments	Adjustment	AR_ADJ_INF_V	Actual
Miscellaneous Receipt	Miscellaneous Receipts	Misc Receipts	AR_CR_INF_V	Actual
Receipt	Standard Receipts	Receipts	AR_CR_INF_V	Actual
Chargeback	Transactions	Chargebacks	AR_TRX_INF_V	Actual
Credit Memo	Transactions	Credit Memos	AR_TRX_INF_V	Actual
Debit Memo	Transactions	Debit Memos	AR_TRX_INF_V	Actual
Invoice	Transactions	Sales Invoices	AR_TRX_INF_V	Actual

## Sources, Source Assignments, and Accounting Attribute Assignments

Receivables predefines sources, source assignments, and accounting attribute assignments for Subledger Accounting.

You can't make changes to predefined sources, source assignments, or accounting attribute assignments. However, you can define your own custom sources.

If you choose to define your own journal line types or application accounting definitions, you can override the default accounting attribute assignments.

## Journal Line Types

Receivables predefines journal line types for each predefined event class. Receivables specifies conditions for the use of each journal line type.

This table lists the journal line types that Receivables predefines for the **Receivables** application:

Event Class	Name	Balance Type	Side
Adjustment	Adjustment	Actual	Credit
Adjustment	Adjustment Charge	Actual	Credit
Adjustment	Adjustment Default Receivable	Actual	Credit
Adjustment	Adjustment Deferred Tax	Actual	Credit
Adjustment	Adjustment Charge Nonrecoverable Tax	Actual	Credit
Adjustment	Adjustment Nonrecoverable Tax	Actual	Credit
Adjustment	Adjustment Tax	Actual	Credit
Chargeback	Chargeback Default Receivable	Actual	Debit
Chargeback	Chargeback Revenue	Actual	Credit
Credit Memo	Credit Memo Default Deferred Tax Application	Actual	Credit
Credit Memo	Credit Memo Default Application	Actual	Credit
Credit Memo	Credit Memo Default Tax Application	Actual	Credit
Credit Memo	Credit Memo Refund Application	Actual	Credit
Credit Memo	Credit Memo Charges	Actual	Credit
Credit Memo	Credit Memo Default Receivable	Actual	Debit
Credit Memo	Credit Memo Default Revenue	Actual	Credit
Credit Memo	Credit Memo Default Tax	Actual	Credit
Credit Memo	Credit Memo Rounding	Actual	Credit
Credit Memo	Credit Memo Unbilled Receivable	Actual	Credit
Credit Memo	Credit Memo Deferred Revenue	Actual	Credit

Event Class	Name	Balance Type	Side
Debit Memo	Debit Memo Charges	Actual	Credit
Debit Memo	Debit Memo Default Receivable	Actual	Debit
Debit Memo	Debit Memo Freight	Actual	Credit
Debit Memo	Debit Memo Revenue	Actual	Credit
Debit Memo	Debit Memo Rounding	Actual	Credit
Debit Memo	Debit Memo Tax	Actual	Credit
Debit Memo	Debit Memo Unbilled Receivable	Actual	Debit
Debit Memo	Debit Memo Deferred Revenue	Actual	Credit
Invoice	Invoice Charges	Actual	Credit
Invoice	Invoice Default Receivable	Actual	Debit
Invoice	Invoice Freight	Actual	Credit
Invoice	Invoice Revenue	Actual	Credit
Invoice	Invoice Rounding	Actual	Credit
Invoice	Invoice Tax	Actual	Credit
Invoice	Invoice Unbilled Receivable	Actual	Credit
Invoice	Invoice Deferred Revenue	Actual	Credit
Miscellaneous Receipt	Miscellaneous Receipt Bank Charges	Actual	Credit
Miscellaneous Receipt	Miscellaneous Receipt Cleared Cash	Actual	Credit
Miscellaneous Receipt	Miscellaneous Receipt Confirmed Cash	Actual	Credit
Miscellaneous Receipt	Miscellaneous Receipt Short Term Debt	Actual	Credit
Miscellaneous Receipt	Miscellaneous Receipt Factored Cash	Actual	Credit

Event Class	Name	Balance Type	Side
Miscellaneous Receipt	Miscellaneous Receipt Miscellaneous Cash	Actual	Credit
Miscellaneous Receipt	Miscellaneous Receipt Remitted Cash	Actual	Credit
Miscellaneous Receipt	Miscellaneous Receipt Tax	Actual	Credit
Receipt	Receipt On Account Application	Actual	Credit
Receipt	Receipt Application to Freight	Actual	Credit
Receipt	Receipt Application to Revenue	Actual	Credit
Receipt	Receipt Application to Rounding	Actual	Credit
Receipt	Receipt Application to Suspense Revenue	Actual	Credit
Receipt	Receipt Application to Tax	Actual	Credit
Receipt	Receipt Application to Unbilled Revenue	Actual	Credit
Receipt	Receipt Application to Earned Revenue	Actual	Credit
Receipt	Receipt Bank Charges	Actual	Credit
Receipt	Receipt Cleared Cash	Actual	Credit
Receipt	Receipt Credit Card Chargeback Application	Actual	Credit
Receipt	Receipt Chargeback Application	Actual	Credit
Receipt	Receipt Confirmed Cash	Actual	Credit
Receipt	Receipt Currency Rounding	Actual	Credit
Receipt	Receipt Short Term Debt	Actual	Credit
Receipt	Receipt Default Application	Actual	Credit
Receipt	Receipt Deferred Tax	Actual	Credit

Event Class	Name	Balance Type	Side
Receipt	Receipt Earned Discount	Actual	Credit
Receipt	Receipt Earned Discount on Freight	Actual	Debit
Receipt	Receipt Earned Discount Nonrecoverable Tax	Actual	Credit
Receipt	Receipt Earned Discount on Revenue	Actual	Debit
Receipt	Receipt Earned Discount on Tax	Actual	Debit
Receipt	Receipt Exchange Gain	Actual	Credit
Receipt	Receipt Exchange Gain Loss	Actual	Gain/Loss
Receipt	Receipt Exchange Loss	Actual	Credit
Receipt	Receipt Factored Cash	Actual	Credit
Receipt	Receipt Payment Netting Application	Actual	Credit
Receipt	Receipt Prepayment Application	Actual	Credit
Receipt	Receipt Refund Application	Actual	Credit
Receipt	Receipt Remitted Cash	Actual	Credit
Receipt	Receipt Tax	Actual	Credit
Receipt	Receipt Unapplied Cash	Actual	Debit
Receipt	Unapplied Cash	Actual	Debit
Receipt	Receipt Unapplied for Gain Loss lines	Actual	Debit
Receipt	Receipt Unearned Discount	Actual	Credit
Receipt	Receipt Unearned Discount on Freight	Actual	Debit



Event Class	Name	Balance Type	Side
Receipt	Receipt Unearned Discount on Nonrecoverable Tax	Actual	Credit
Receipt	Receipt Unearned Discount on Revenue	Actual	Debit
Receipt	Receipt Unearned Discount on Tax	Actual	Debit
Receipt	Receipt Unidentified Cash	Actual	Credit
Receipt	Receipt Write-Off Application	Actual	Credit

## Account Derivation Rules

Receivables predefines account derivation rules. When Subledger Accounting uses the predefined account derivation rules that Receivables provides, it accepts the default accounting that Receivables generates using AutoAccounting without change.

You can optionally define your own account derivation rules for an Accounting Flexfield or for a segment. In this case, Subledger Accounting overrides the default accounts that Receivables generates, or individual segment values in the default accounts, when it creates the draft or final subledger accounting.

The name account derivation rules that Receivables predefines for the **Receivables** application are as follows:

- Credit Memo Distribution GL Account
- Collection Bank Charges Account
- Collection Bank Account Cash Account
- Collection Bank Factoring Charges Account
- Distribution GL Account
- Remit Bank Unapplied Account
- System Gain GL Account
- System Loss GL Account
- Transaction Distribution GL Account
- Transaction Distribution GL Account with reference

## Journal Lines Definitions

Receivables predefines journal lines definitions that group the predefined journal line types and account derivation rules within each of the predefined event types. Receivables assigns each predefined journal lines definition to all event types within an event class.

This table lists the journal lines definitions that Receivables predefines for the **Receivables** application:

Event Class	Journal Lines Definition Name	Journal Line Types
Adjustment	Adjustments - Default Accrual	Adjustment, Adjustment Charge, Adjustment Charge Nonrecoverable Tax, Adjustment Default Receivable, Adjustment Deferred Tax, Adjustment Nonrecoverable Tax, Adjustment Tax
Chargeback	Chargebacks - Default Accrual	Chargeback Default Receivable, Chargeback Revenue
Credit Memo	Credit Memos - Default Accrual	Credit Memo Charges, Credit Memo Default Application, Credit Memo Default Deferred Tax Application, Credit Memo Default Receivable, Credit Memo Default Revenue, Credit Memo Default Tax, Credit Memo Default Tax Application, Credit Memo Deferred Revenue, Credit Memo Refund Application, Credit Memo Rounding, Credit Memo Unbilled Receivable
Debit Memo	Debit Memos - Default Accrual	Debit Memo Charges, Debit Memo Default Receivable, Debit Memo Deferred Revenue, Debit Memo Freight, Debit Memo Revenue, Debit Memo Rounding, Debit Memo Tax, Debit Memo Unbilled Receivable
Invoice	Invoices - Default Accrual	Invoice Charges, Invoice Default Receivable, Invoice Deferred Revenue, Invoice Freight, Invoice Revenue, Invoice Rounding, Invoice Tax, Invoice Unbilled Receivable
Miscellaneous Receipt	Miscellaneous Receipts	Miscellaneous Receipt Bank Charges, Miscellaneous Receipt Cleared Cash, Miscellaneous Receipt Confirmed Cash, Miscellaneous Receipt Factored Cash, Miscellaneous Receipt Miscellaneous Cash, Miscellaneous Receipt Remitted Cash, Miscellaneous Receipt Short Term Debt, Miscellaneous Receipt Tax
Receipt	Receipts - Default Accrual	Receipt Bank Charges, Receipt Chargeback Application, Receipt Cleared Cash, Receipt Confirmed Cash, Receipt Credit Card Chargeback Application, Receipt Currency Rounding, Receipt Default Application, Receipt Deferred Tax, Receipt Earned Discount, Receipt Earned Discount Nonrecoverable Tax, Receipt Exchange Gain Loss, Receipt Factored Cash, Receipt On Account Application, Receipt Payment Netting Application, Receipt Prepayment Application, Receipt Refund Application, Receipt Remitted Cash, Receipt Short Term Debt, Receipt Tax, Receipt Unapplied Cash, Receipt Unearned Discount, Receipt Unearned Discount on Nonrecoverable Tax, Receipt Unidentified Cash, Receipt Write-Off Application, Unapplied Cash


Event Class	Journal Lines Definition Name	Journal Line Types
Receipt	Receipt - Basis Journal Lines Definition	Receipt Application to Earned Revenue, Receipt Application to Freight, Receipt Application to Revenue, Receipt Application to Rounding, Receipt Application to Suspense Revenue, Receipt Application to Tax, Receipt Application to Unbilled Revenue, Receipt Bank Charges, Receipt Cleared Cash, Receipt Confirmed Cash, Receipt Currency Rounding, Receipt Earned Discount, Receipt Earned Discount on Freight, Receipt Earned Discount on Revenue, Receipt Earned Discount on Tax, Receipt Exchange Gain Loss, Receipt Factored Cash, Receipt On Account Application, Receipt Payment Netting Application, Receipt Prepayment Application, Receipt Refund Application, Receipt Remitted Cash, Receipt Short Term Debt, Receipt Unapplied Cash, Receipt Unapplied for Gain Loss lines, Receipt Unearned Discount, Receipt Unearned Discount on Freight, Receipt Unearned Discount on Revenue, Receipt Unearned Discount on Tax, Receipt Unidentified Cash, Receipt Write-Off Application, Unapplied Cash

## Application Accounting Definition

Receivables predefines the **Receivables Default Accrual** application accounting definition.

This table lists the assignments for the **Receivables Default Accrual** application accounting definition that Receivables predefines for the **Receivables** application:

Event Class Assignments	Event Type Assignments	Create Accounting	Journal Line Definition Assignments
Adjustment	All	Yes	Adjustments - Default Accrual
Chargeback	All	Yes	Chargebacks - Default Accrual
Credit Memo	All	Yes	Credit Memos - Default Accrual
Debit Memo	All	Yes	Debit Memos - Default Accrual
Invoice	All	Yes	Invoices - Default Accrual
Miscellaneous Receipt	All	Yes	Miscellaneous Receipts
Receipt	All	Yes	Receipts - Default Accrual

 **Note:** Subledger Accounting provides a predefined subledger accounting method that groups the predefined application accounting definitions for subledger applications. You can optionally create your own subledger accounting method. Receivables assigns the predefined **Receivables Default Accrual application** accounting definition to the predefined Standard Accrual subledger accounting method. You can assign this subledger accounting method to your ledgers.

#### Related Topics

- [Accounting Attribute Assignments: Points to Consider](#)
- [Accounting Event Model: Explained](#)

## Notes Mapping for Receivables: Explained

The Notes common component portlet is available on all Oracle Fusion Receivables transaction pages. You can create, view, update, and delete notes on transactions throughout the entire transaction cycle: incomplete, complete, posted.

There are two types of Notes: Internal and Private. Internal notes are available to all users. Private notes are available to authors only.

There are three configuration tasks to perform to use the Notes portlet on Receivables transactions.

### Notes Mapping Configuration Tasks

There are three configuration tasks for Notes mapping to Receivables.

Perform these tasks in the order indicated:

#### 1. Manage Receivables Note Type

Define the lookups you need for the Note Type.

#### 2. Manage Receivables Note Type Mapping

Use the CRM component to map the Note Type to the Receivables Transaction object (AR\_TRANSACTIONS).

This table displays the predefined note type mapping for Receivables:

OBJECT_CODE	MAPPING_TYPE_CODE	MAPPED_LOOKUP_CODE	Meaning	Description
AR_TRANSACTION	ZMM_NOTE_TYPE	MAINTAIN (default)	Maintain	Maintain Receivables transactions
AR_TRANSACTION	ZMM_NOTE_TYPE	AR_APPROVAL	Receivables Credit Memo Request Approval	Receivables credit memo request approval note type

#### 3. Manage Receivables Note Descriptive Flexfields

You can optionally define descriptive flexfields for Notes. You can use Notes descriptive flexfields on Receivables transactions to capture additional information for your business requirements.

## Setting Up a Salesperson: Procedure

Set up salespersons for your transactions and sales credit assignments.

Oracle Fusion Receivables provides the default salesperson No Sales Credit. If the Receivables system option **Require salesperson** is enabled and no salesperson is defined, then No Sales Credit is populated by default on all transactions. If AutoAccounting depends on salesperson, then you must assign revenue, freight, and receivable general ledger reference accounts to No Sales Credit. These accounts are required when creating a debit memo reversal or when entering transactions with No Sales Credit.

To set up a salesperson, complete these tasks:

- Create a User as a Salesperson
- Assign Resources to the User
- Define Receivables Salesperson Reference Accounts

### Create a User as a Salesperson

You can create a new user or update an existing user as a salesperson.

To create a new user as a salesperson:

1. Navigate to the Create User page.
2. Enter personal details for the user.
3. In the Employment Information section, select **Employee** in the **Person Type** field.
4. In the Resource Information section, select **Salesperson** in the **Resource Role** field.
5. Complete the rest of the page according to your requirements, and save.

### Assign Resources to the User

Assign a salesperson role and reference set to the user you just created.

To assign resources to the salesperson user:

1. Navigate to the Manage Resources page.
2. Search for and select the user.
3. In the Resource page that appears, navigate to the Roles tab.
4. In the **Role Name** field, assign the user a role of **Salesperson** or **Sales Manager**.
5. If necessary, update the **From Date** and **To Date** fields.
6. Navigate to the Salesperson tab.
7. In the **Set** field, assign a reference set to the salesperson.
8. Complete the remaining fields according to your requirements, and save.

### Define Receivables Salesperson Reference Accounts

Assign the necessary general ledger reference accounts to the user. If AutoAccounting depends on salesperson, Receivables uses the general ledger reference accounts that you enter for the salesperson in combination with the AutoAccounting rules that you define to determine the default revenue, freight, and receivable accounts for your transactions.

To assign general ledger reference accounts to the salesperson user:

1. Navigate to the Manage Salesperson Account References page.
2. Search for and select the salesperson user.
3. Enter or update the business unit.
4. Enter the revenue, freight, and receivable accounts to assign to this user, and save.

#### *Related Topics*

- [Sales Credits and AutoAccounting: Explained](#)

## FAQs for Define Common Accounts Receivable Configuration

### Why can't I create transactions or generate accounting?

You must open an accounting period before you can perform basic Receivables activities. This is also true of new installations: manually open an accounting period once your installation is complete.

## Legal Entity Document Sequencing

### Legal Entity Document Sequencing in Receivables: Points to Consider

You can set up your primary ledger to allow document sequencing at the legal entity level instead of at the ledger level.

This means that if you have more than one legal entity assigned to the same ledger, you can assign separate document sequences to Receivables transactions, adjustments, and receipts belonging to each legal entity.

Legal entity level document sequencing helps you conform to local and governmental authority requirements, while still being able to organize multiple legal entities under the same primary ledger.

There are these points to consider if you want to use document sequencing at the legal entity level for Receivables:

- Document Sequencing in Receivables
- Receivables Document Categories
- Gapless Document Sequencing
- AutoInvoice Processing

### Document Sequencing in Receivables

To use document sequencing at the legal entity level for Receivables, in the Sequencing section of the Specify Options page of your primary ledger, perform one or both of these tasks:

1. Set the **Sequencing By** option to Legal Entity.

2. Optionally enable the **Enforce Document Sequencing** option for Receivables.

If you perform step 1 and step 2, then document sequencing is always used in all legal entities assigned to the ledger. You must assign a document sequence to every document category generated by Receivables events.

If you perform step 1 but not step 2, then you can optionally assign document sequences to the Receivables document categories that meet your business requirements. For example, you can use sequential numbering for automatic receipt processing only and not for transactions.

## Receivables Document Categories

If the primary ledger is enabled for document sequencing at the legal entity level (**Sequencing By** option set to Legal Entity), then Receivables creates a document category for each of these Receivables events in all legal entities assigned to the ledger:

- Invoice transaction type
- Credit memo transaction type
- Debit memo transaction type
- Bills receivable transaction type
- Standard receipt
- Adjustment

To assign a document sequence to a document category:

1. Navigate to the Manage Document Sequence Categories page.
2. Search for the document categories that you want.

You can use the **Category Code** field to limit your search by entering, for example, Invoice or Receipt.


3. Review the search results to find the document category that you want
4. If necessary, update the category name according to your requirements.
5. Save your changes.
6. Navigate to the Manage Receivables Document Sequences page.
7. Search for the document sequence name that you want, or create a new document sequence.
8. In the Search Results section, update or complete the document sequence setup according to your requirements.
9. In the Assignments section, select the document category to assign to this document sequence.
10. Save your changes.

For each Receivables event, the document sequence number is generated when the following related action takes place:

- Transactions: At the time of either saving or completing the transaction, depending on the setting of the Receivables system option **Document Number Generation Level** for the applicable business units.
- Adjustments: When the adjustment is submitted.
- Receipts: When the receipt is submitted.
- Bills Receivable: When the bill receivable is completed.

These rules apply to Receivables document sequencing:


- Document sequence date is the accounting date (not, for example, the transaction date or receipt date).

 **Note:** If a bill receivable requires drawee acceptance, then Receivables uses the transaction date instead of the accounting date to assign the document number. This is because a bills receivable document number is generated when the bill is completed, not accepted.

- You can't change the legal entity on any transaction that has a document sequence number.
- By default, you can't delete any transaction or receipt that has a document sequence number. If the Receivables system option **Allow payment deletion** is enabled, then you can delete receipts and bills receivable transactions only for the applicable business units.

## Gapless Document Sequencing

Document sequencing uses the accounting date as the document sequence date. To ensure gapless document sequencing for your transactions, you can enforce the sequencing of document numbers in chronological order.

 **Note:** Gapless document sequencing applies to invoice, credit memo and debit memo transactions only, either entered manually or imported using AutoInvoice. It doesn't apply to receipts, adjustments or bills receivable.

To ensure gapless document sequencing at the legal entity level for Receivables transactions, in the Sequencing section of the Specify Options page of your primary ledger perform these three tasks:

1. Set the **Sequencing By** option to Legal Entity.
2. Enable the **Enforce Document Sequencing** option for Receivables.
3. Enable the **Enforce Chronological Order on Document Date** option.

You must perform all three steps for gapless document sequencing. Enabling the **Enforce Chronological Order on Document Date** option enforces the correlation between the document sequence accounting date and the transaction accounting date. This prevents Receivables from creating a transaction with an accounting date that is earlier than the accounting date of the last sequenced document within the same document sequence.

For example, you create an invoice with an accounting date of 01-Jan-2014. This invoice is assigned the document number 100. The next invoice you create is assigned the document number 101 provided the accounting date of the invoice is 01-Jan-2014 or later. If the accounting date is earlier than 01-Jan-2014, then Receivables doesn't create the transaction.

## AutoInvoice Processing

If you are using document sequencing at the legal entity level and the primary ledger option **Enforce Chronological Order on Document Date** is enabled, then you must set the Receivables system option **Accounting Dates Out of Order**. Setting this option determines how AutoInvoice processes transactions when the accounting date is out of order within the document sequence.

In the Receivables System Options page: Billing and Revenue tab: AutoInvoice section, set the **Accounting Dates Out of Order** field to **Reject** or **Adjust**:

- **Reject:** If the transaction accounting date is out of order within the document sequence, reject the transaction and transfer it to the error table.
- **Adjust:** If the transaction accounting date is out of order within the document sequence, adjust the accounting date to conform to the document sequence accounting date.


### Related Topics

- [What legal entity is assigned to a transaction?](#)
- [Deriving the Accounting Date during AutoInvoice Import: How It Works](#)
- [Document Sequences: Explained](#)
- [Document Sequence Categories: Explained](#)



# AutoInvoice and Legal Entity Document Sequencing: How Transactions are Processed

If the primary ledger is enabled for document sequencing, AutoInvoice validates and assigns document numbers to transactions according to the requirements of the ledger settings.

 **Note:** If an imported transaction line already has a document number, then AutoInvoice accepts this document number without further validation.

## Settings That Affect AutoInvoice Transaction Processing


If the primary ledger option **Enforce Chronological Order on Document Date** is enabled, then you must set the Receivables system option **Accounting Dates Out of Order**. Setting this option determines how AutoInvoice processes transactions when the accounting date is out of order within the document sequence.

Before you run the Import AutoInvoice program, you must also create and assign document sequences to the document categories of the transaction types that you plan to assign to imported transactions.

## How Transactions Are Processed

If the primary ledger is set for document sequencing at the legal entity level, AutoInvoice processes transactions in this way:

1. Validate and group transaction lines into transactions.
2. Derive the legal entity for each transaction and assign the legal entity to the transaction.
3. If the transaction doesn't have an accounting date, then derive the accounting date.
4. If the primary ledger option **Enforce Chronological Order on Document Date** is not enabled:
  - a. Retrieve the document sequence for the document category of each transaction type assigned to transactions.
  - b. Assign the document number to the transaction according to the combination of legal entity, transaction type, document sequence, and accounting date.
5. If the primary ledger option **Enforce Chronological Order on Document Date** is enabled:
  - a. Order transactions by legal entity and then by transaction type.
  - b. Retrieve the document sequence for the document category of each transaction type assigned to transactions.
  - c. Order transactions by document sequence and then by accounting date in ascending chronological order.
  - d. Compare the document sequence accounting date to the accounting date of the transaction.
  - e. If the transaction accounting date is equal to or later than the document sequence accounting date, assign the document number to the transaction according to the combination of legal entity, transaction type, document sequence, and accounting date.
  - f. If the transaction accounting date is earlier than the document sequence accounting date, retrieve the value of the Receivables system option **Accounting Dates Out of Order**.
  - g. If the **Accounting Dates Out of Order** option is set to:
    - **Reject:** Reject the transaction and transfer the transaction to the error table.
    - **Adjust:** Update the transaction accounting date with the document sequence accounting date.

 **Note:** If the document sequence accounting date is in a closed period, then adjust this date to the first open accounting period.

- h.** Assign the document number to the transaction according to the combination of legal entity, transaction type, document sequence, and accounting date.

#### *Related Topics*

- [AutoInvoice Import: How Data Is Processed](#)
- [AutoInvoice Validations: Points to Consider](#)

## FAQs for Legal Entity Document Sequencing

### [Can I share document sequences across ledgers or business units?](#)

If document sequencing is enabled for a ledger, it is not recommended to share the same document sequence across ledgers or business units.

Document sequencing uses the accounting date as the document sequence date. Because ledgers can have different accounting periods open, each ledger could derive a different accounting date for the same document sequence.

Because a document category is created for each transaction type you create, it is recommended to create and maintain a separate set of transaction types in each business unit.

## Define Receivables Activities

### Receivables Activity Types

Receivables activity types provide default accounting information for all activities in Oracle Fusion Receivables other than transactions and receipts.

#### [Using Receivables Activity Types](#)

##### **Adjustment**

You use activities of this type when creating adjustments. You must create at least one activity of this type.

There are also three related activities that are reserved for internal use only:

- Chargeback Adjustment
- Adjustment Reversal
- Chargeback Reversal

You must define general ledger accounts for the Chargeback Adjustment activity before creating chargebacks.

When you reverse a receipt, if an adjustment or chargeback exists, Receivables automatically generates off-setting adjustments using the Adjustment Reversal and Chargeback Reversal activities.

### **Bank Error**

You use activities of this type when entering miscellaneous receipts. You can use this type of activity to help reconcile bank statements using Oracle Fusion Cash Management.

### **Bills Receivable Funds Recovery**

You use activities of this type when you need to unapply a receipt from a bill receivable. Because you can't reverse the receipt in Oracle Fusion Cash Management, you use this activity to create a negative miscellaneous receipt and apply it to Bills Receivable Funds Recovery.

### **Credit Card Chargeback**

You use activities of this type when recording credit card chargebacks. You must define a general ledger clearing account for the Credit Card Chargeback activity that Receivables provides before recording credit card chargebacks.

Receivables credits the clearing account when you apply a credit card chargeback, and then debits the account after generating the negative miscellaneous receipt. If you later determine the chargeback is invalid, then Receivables debits the clearing account when you unapply the credit card chargeback, and then credits the account after reversing the negative miscellaneous receipt. Only one Credit Card Chargeback activity within a business unit can be active at a time.

### **Credit Card Refund**

You use activities of this type when processing refunds to customer credit card accounts. This activity includes information about the general ledger clearing account to use to clear credit card refunds. You must create at least one activity of this type to process credit card refunds.

### **Earned Discount**

You use activities of this type to adjust a transaction if payment is received within the discount period, as determined by the payment terms on the transaction.

### **Late Charges**

You use activities of this type to define a late charge policy. You must define a Late Charges activity if you record late charges as adjustments against overdue transactions. If you assess penalties in addition to late charges, then define a separate Late Charges activity for penalties.

### **Miscellaneous Cash**

You use activities of this type when entering miscellaneous receipts. The Miscellaneous Cash activity uses a distribution set to automatically distribute miscellaneous cash across various accounts. You must create at least one activity of this type.

If the Tax Rate Code Source for this activity is Activity, then you must define asset and liability tax rate codes to account for tax on miscellaneous receipts and miscellaneous payments.

### **Payment Netting**

You use activities of this type when applying a receipt against other open receipts. You must define a general ledger clearing account to use when offsetting one receipt against another receipt. Only one Payment Netting activity within a business unit can be active at a time.

### Prepayments

You use activities of this type when creating prepayment receipts. You must define a general ledger account for prepayment receipts that use the Prepayments activity. Only one Prepayments activity within a business unit can be active at a time.

### Receipt Write-off

You use activities of this type when writing off receipts. You must define the general ledger account to credit when you write off an unapplied amount or an underpayment on a receipt.

### Refund

You use activities of this type to process automated non-credit card refunds. You must define the general ledger clearing account to use to clear refunds. You must create at least one activity of this type. Only one Refund activity within a business unit can be active at a time.

### Short Term Debt

You use activities of this type to record advances made to creditors by the bank when bills receivable are factored with recourse. You select a short-term debt receivables activity when you create or update remittance banks to use with bills receivable remittance receipt methods.

### Unearned Discount

You use activities of this type to adjust a transaction if payment is received after the discount period, as determined by the payment terms on the transaction.

## GL Account Source

When you define a Receivables activity, you use the **GL Account Source** to indicate how Oracle Fusion Receivables derives the accounts for the expense or revenue generated by the activity.

### GL Account Source Options

#### Activity GL Account

Allocate the expense or revenue to the general ledger account that you specify for the Receivables activity. If the activity type is Bank Error, Late Charges, Prepayments, or Receipt Write-off, you can only select this option.

#### Distribution Set

Allocate the expense or revenue to the distribution set that you specify. This value is only used with Miscellaneous Cash activities.


#### Revenue on Invoice

Allocate the expense or revenue net of any tax to the revenue accounts specified on the invoice. If **Tax Rate Code Source** is set to None, allocate the gross amount to these accounts. You can only choose this option if the activity type is Adjustment, Earned Discount, or Unearned Discount.

If the revenue on the invoice is unearned, then AutoAccounting derives the anticipated revenue accounting distribution accounts and amounts. Receivables then uses this information to allocate the adjustment or discount amount to these derived revenue accounts.

### Tax Rate Code on Invoice

Allocate the net portion using the expense/revenue accounts specified by the tax rate code on the invoice. If **Tax Rate Code Source** is set to None, allocate the gross amount to these accounts. You can only choose this option if the activity type is Adjustment, Earned Discount, or Unearned Discount.

 **Note:** In the event of an adjustment to an invoice with zero amount revenue distributions, the adjustment activity **GL Account Source** must not be set to Revenue on Invoice or Tax Rate Code on Invoice.

#### Related Topics

- [Adjusting Transactions: How It Works](#)

## Tax Rate Code Source

When you define a receivables activity, you use the **Tax Rate Code Source** to indicate how Oracle Fusion Receivables derives the tax rate code for an activity.

### Tax Rate Code Source Options

#### None


Allocates the entire tax amount according to the **GL Account Source** you specified. You use this option if you don't want to account for tax separately.

#### Activity

Allocate the tax amount to the asset or liability tax accounts specified by the activity.

#### Invoice

Distribute the tax amount to the tax accounts specified by the tax rate code on the invoice. You can't choose this option if the activity type is Miscellaneous Cash or Late Charges.

 **Note:** In the event of a tax adjustment to an invoice with zero amount tax distributions, the adjustment activity **Tax Rate Code Source** must not be set to Invoice.

If the **Tax Rate Code Source** is Activity or Invoice, you must indicate whether tax for this activity is recoverable or non-recoverable.

#### Related Topics

- [Accounting for Tax on Receivables Transactions: Explained](#)

## Define AutoCash Rule Sets

### Using AutoCash Rules: Examples

You create an AutoCash rule set from a combination of the five AutoCash rules. You enter the rules in the order in which you want Oracle Fusion Receivables to use them to apply a receipt to an open debit item.

The AutoCash rules are:

- Match Payment with Invoice
- Clear the Account
- Clear Past Due Invoices
- Clear Past Due Invoices Grouped by Payment Terms
- Apply to the Oldest Invoice First

When you apply a receipt, Receivables uses the first rule in the AutoCash rule set. If the first rule in the set doesn't find a match, Receivables uses the next rule in the sequence, and so on until it can apply the receipt.

These examples illustrate how each rule applies receipts to transactions and updates customer balances.

#### Match Payment with Invoice

The Match Payment with Invoice rule applies a receipt to a single invoice, debit memo, or chargeback only if the receipt amount exactly matches the amount of the debit item. If more than one debit item has an open amount that matches the receipt amount, Receivables applies the receipt to the item with the earliest due date. If more than one debit item has the same amount and due date, Receivables applies the receipt to the item with the lowest payment schedule ID number (internal identifier).

Receivables uses the values specified for the AutoCash rule set open balance calculation and the number of discount grace days assigned to the customer profile to determine the remaining amount due on the debit item. The rule ignores the value of the **Apply partial receipts** option.

For example, consider the following scenario:

Item/Option	Value
Discounts	Earned Only
Late Charges	No
Receipt	\$1800
Receipt Date	14-JAN-03
Discount Grace Days	5

The invoice details are:

Invoice Number	Invoice Amount	Discount	Payment Terms	Invoice Date	Due Date
600	\$2000	\$20	10% 10/Net 30	01-JAN-03	30-JAN-03

The payment terms assigned to this invoice include a 10% discount if the invoice is paid within 10 days, and the open balance calculation on the AutoCash rule set allows for earned discounts. Even though the invoice is paid after the 10 day period, Receivables adds the 5 discount grace days, making this invoice eligible for a 10% discount.

The remaining amount due on the invoice on January 14 is \$1800. Since the remaining amount due matches the receipt amount, the receipt is applied. If there had been no discount grace days, Receivables couldn't apply the receipt because the remaining amount of the invoice would be \$2000.

## Clear the Account

The Clear the Account rule applies a receipt only if the receipt amount exactly matches the customer open balance. Receivables includes all open debit and credit items when calculating the customer open balance. Open credit items include credit memos, on-account credits, and on-account and unapplied cash. The rule ignores the value of the **Apply partial receipts** option.

The Clear the Account rule uses the following equation to calculate the open balance for each debit item:

$$\text{Open Balance} = \text{Original Balance} + \text{Late Charges} - \text{Discount}$$

Receivables then adds the balance for each debit item to determine the total account balance. The rule uses this equation for each invoice, chargeback, debit memo, credit memo, and application of an unapplied or on-account receipt to a debit item.

Receivables uses the values specified for the AutoCash rule set open balance calculation and the number of discount grace days assigned to the customer profile to determine the customer open balance.

For example, consider the following scenario:

Item/Option	Value
Late Charges	Yes
Items in Dispute	Yes
Receipt	\$590

This table shows the customer activity:

Past Due Debits/Credits	Invoice Amount	Late Charges	In Dispute
Invoice 45	\$500	\$40	Yes
Invoice 46	\$300	\$0	N/A
Credit Memo 100	\$50	N/A	N/A

Past Due Debits/Credits	Invoice Amount	Late Charges	In Dispute
Unapplied Cash	\$200	N/A	N/A

Since the **Late charges** and **Items in dispute** options are enabled, the open balance for this customer is \$590. Because the receipt amount matches the customer open balance, the receipt can be applied.

If the receipt amount didn't exactly match the customer account balance, Receivables would use the next rule in the set to attempt to apply the receipt.

## Clear Past Due Invoices

The Clear Past Due Invoices rule applies a receipt only if the receipt amount exactly matches the customer past due account balance. Receivables includes all open past due debit and credit items when calculating the past due account balance. The rule ignores the value of the **Apply partial receipts** option.

The Clear Past Due Invoices rule only applies the receipt to items that are currently past due. A debit item is considered past due if the invoice due date is earlier than or equal to the date of the receipt that is currently being applied. Receivables uses the receipt date for unapplied and on-account cash, and the credit memo date for credit memos and on-account credits, to determine whether to include these amounts as part of the customer past due account balance.

For example, if you apply a receipt with a receipt date of 10-JAN-03, all unapplied and on-account cash, and all credit memos and on-account credits, that have a transaction date (receipt date or credit memo date) equal to or earlier than 10-JAN-03 are included when calculating the customer past due account balance.

Receivables uses the values specified for the AutoCash rule set open balance calculation and the number of discount grace days assigned to the customer profile to determine the customer past due account balance. The settings of the **Late charges** and **Items in dispute** options may prevent a past due debit item from being closed, even if the receipt amount matches the customer past due account balance.

For example, consider the following scenario:

Item/Option	Value
Late Charges	No
Items in Dispute	No
Receipt	\$420

This table shows the customer activity:

Past Due Debits/Credits	Invoice Amount	Late Charges	In Dispute
Invoice 209	\$300	\$0	N/A
Invoice 89	\$250	\$0	Yes
Invoice 7	\$120	\$30	N/A



Since the **Late charges** and **Items in dispute** options are not enabled, Receivables doesn't include Invoice 89 (\$250) or late charges for Invoice 7 (\$30) in the calculation of the customer past due account balance. Therefore, the past due account balance for this customer is \$420. Because the receipt amount matches the customer past due account balance, the receipt is applied. However, Invoice 7 and Invoice 89 are still open, past due debit items.

## Clear Past Due Invoices Grouped by Payment Terms

The Clear Past Due Invoices Grouped by Payment Terms rule applies a receipt only if the receipt amount exactly matches the sum of the customer credit memos and past due invoices. This rule is similar to the Clear Past Due Invoices rule, but it first groups past due invoices by payment terms and uses the oldest transaction due date within the group as the group due date.

A debit item is considered past due if the invoice due date is earlier than the date of the receipt that is currently being applied. For credit memos, Receivables uses the credit memo date to determine whether to include these amounts in the customer account balance.

For example, if you apply a receipt with a receipt date of 10-JAN-03, credit memos that have a transaction date equal to or earlier than 10-JAN-03 are included. Credit memos don't have payment terms, so they are included in each group.

Receivables uses the values specified for the AutoCash rule set open balance calculation and the number of discount grace days assigned to the customer profile to determine the sum of the customer credit memos and past due invoices. The settings of the **Late charges** and **Items in dispute** options may prevent a past due debit item from being closed, even if the receipt amount matches the sum of the customer credit memos and past due invoices.

For example, consider a \$900 receipt applied on 25-JUN. This table shows the related customer activity:

Transaction Number	Payment Terms	Due Date	Invoice Amount
1	A	25-MAY	\$500
2	A	25-JUN	\$200
3	A	25-JUN	\$200
4	B	20-JUN	\$900
5	C	25-MAY	\$905

Receivables groups these transactions as follows:

- Group 1: Transactions 1,2,3 Amount: \$900 Group Due Date: 25-MAY
- Group 2: Transaction 4 Amount: \$900 Group Due Date: 20-JUN
- Group 3: Transaction 5 Amount: \$905 Group Due Date: 25-MAY

Since both Groups 1 and 2 match the receipt amount, Receivables selects the group with the oldest due date (Group 1) and applies the receipt to the transactions in this group.

## Apply to the Oldest Invoice First

The Apply to the Oldest Invoice First rule applies receipts to customer debit and credit items, starting with the item with the oldest due date. Receivables uses the values specified for the AutoCash rule set open balance calculation to determine the oldest outstanding item for the customer.

For example, consider the following scenario:

Item/Option	Value
Apply Partial Receipts	Yes
Late Charges	No
Receipt	\$200

This table shows the customer activity:

Invoice Number	Invoice Amount	Late Charges	Due Date
801	\$0	\$35	01-DEC-02
707	\$450	\$0	01-JAN-03

If you compare only the due dates for the two invoices, invoice 801 is the oldest invoice. However, Receivables also checks the open balance calculation and automatic matching rule options for the AutoCash rule set. Since the **Late charges** option is not enabled, Receivables ignores invoice 801 (because the remaining amount only consists of late charges) and applies the \$200 receipt to invoice 707.

If the **Apply partial receipts** option were not enabled, Receivables couldn't apply this receipt and would look at the next rule in the sequence.

### Related Topics

- [Applying Receipts and On-Account Credit Memos: Points to Consider](#)

## Using an AutoCash Rule Set: Worked Example

This example demonstrates how to create and use an AutoCash rule set. You create an AutoCash rule set to manage the payments received from Global Freight Carriers. You have an earned discount arrangement with this company but with no payment or discount grace days, and you don't add late charges for payments received beyond the due date.

## Creating the AutoCash Rule Set

Create the AutoCash Rule set using these values:

Field	Value
<b>Open Balance Calculation: Discounts</b>	Earned Only
<b>Open Balance Calculation: Late Charges</b>	No
<b>Open Balance Calculation: Items in Dispute</b>	No
<b>Automatic Matching Rules: Apply Partial Receipts</b>	Yes
<b>Automatic Matching Rules: Remaining Remittance Amount</b>	On Account
<b>AutoCash Rule</b>	1. Match Payment with Invoice
<b>AutoCash Rule</b>	2. Clear The Account
<b>AutoCash Rule</b>	3. Apply To The Oldest Invoice First

## Processing Payment Using the AutoCash Rule Set


Global Freight Carriers has the following outstanding invoices, none of which are in dispute:

Number	Amount Remaining	Due Date	Discount Date	Discount Amount
123	\$200	11-DEC-02	01-DEC-02	\$20
124	\$300	08-DEC-02	30-NOV-02	\$30
125	\$150	13-DEC-02	28-NOV-02	\$15

A payment was entered for Global Freight Carriers for \$600 with a deposit date of 10-DEC-02.

Using the AutoCash rule set that you created, Oracle Fusion Receivables processes the payment in this way:

1. AutoCash rule 1, Match Payment with Invoice, fails because none of the customer open items have a remaining amount due that is equal to the amount of the receipt (\$600).
2. Receivables looks at AutoCash rule 2.
3. AutoCash rule 2, Clear the Account, fails because the customer calculated account balance (\$650) is not the same as the amount of the receipt.
4. Receivables looks at AutoCash rule 3.

5. Receivables uses AutoCash rule 3, Apply to the Oldest Invoice First.
- a. Receivables first applies the receipt to the oldest invoice, Invoice 124 for \$300, and performs these calculations:
    - Since the discount date of 30-NOV-02 has passed and the **Discount** field is set to Earned Only, the \$30 discount is no longer available. The amount due remaining for this invoice is now equal to either \$0 or the amount of any late charges previously assessed for this item.
    - Because the **Late Charges** option is set to No, late charges are not included in the customer open balance calculation. The remaining receipt amount is now \$300.00.
  - b. Receivables now applies \$200 to the next oldest invoice, Invoice 123, and performs these calculations:
    - As with Invoice 124, the discount date for Invoice 123 has passed and the \$20 discount is no longer available. The amount due remaining for this invoice is now equal to either \$0 or the amount of any late charges previously assessed for this item.
    - Because the **Late Charges** option is set to No, late charges are not included in the customer open balance calculation. The remaining receipt amount is now \$100.
  - c. Receivables applies the remaining \$100 to Invoice 125 (\$150) as a partial receipt because the **Apply partial receipts** option is set to Yes.
-  **Note:** If the **Apply partial receipts** option were set to No, Receivables couldn't apply the remaining amount to Invoice 125. Instead, it would be placed on account, because the **Remaining Remittance Amount** option is set to On Account.
- As with the other invoices, the discount date for Invoice 125 has passed and the \$15 discount is no longer available.
  - If there are no late charges for this invoice, the amount due remaining is reduced from \$150 to \$50, and remains open.

## FAQs for Define AutoCash Rule Sets

### How is an AutoCash rule set selected and used?

During payment processing, Oracle Fusion Receivables uses the Match Receipts By rules to attempt to match receipts to open transactions, and either apply receipts automatically or present recommendations for receipt application.

If transactions cannot be matched or transaction information is not available, Receivables uses the AutoCash rule set defined for the customer profile either at the customer site or customer level to apply the receipt. If the customer does not have an AutoCash rule set assigned to a profile, Receivables uses the AutoCash rule set assigned to system options and the number of discount grace days defined in the customer site or customer profile to apply the receipt.

If none of the rules in the AutoCash rule set apply, Receivables places the remaining amount either unapplied or on-account, depending on the setting of the **Remaining Remittance Amount** option on the AutoCash rule set.

### How can I use partial receipts?

Use the AutoCash rule set **Apply partial receipts** option with the Apply to the Oldest Invoice First rule. If you enable this option, you can apply a receipt to a transaction that is less than the amount required to close the debit item.

If the AutoCash rule set does not use partial receipts but does include late charges in the open balance calculation, then Oracle Fusion Receivables can interpret a receipt application against a transaction amount plus late charges as a partial receipt.

For example, you intend to close a \$100 transaction by applying a \$100 receipt, but the transaction has since accumulated a \$10 late charge. If the **Apply partial receipts** option is not enabled, Receivables cannot apply the \$100 receipt to what is now a \$110 open debit item.

## Define Approval Limits

### Approval Limits Document Types

You can define approval limits for your users for specific transaction activities and amount ranges per currency. The document types identify the transaction activities that a user can approve.

#### Document Types

##### **Adjustment**

Define Adjustment approval limits by currency and amount. Oracle Fusion Receivables uses approval limits that have a document type of Adjustment when you create or approve an adjustment.

When you enter an adjustment that is outside your approval limit range, Receivables assigns the adjustment a status of Pending until someone with the appropriate approval limits either approves or rejects it.

##### **Receipt Write-off**

Define Receipt Write-off approval limits by currency and amount. Receivables uses approval limits with this document type whenever you attempt to write off either an unapplied receipt amount or an underpayment on a receipt.

You can't write off a receipt amount that is outside your approval limit range. In addition, the approval limits for write-offs are separate from, but can't exceed, the Receivables system options write-off amounts.

##### **Credit Memo Refund**

Define Credit Memo Refund approval limits by currency and amount. Receivables uses approval limits with this document type whenever you attempt to refund an on-account credit memo.

#### *Related Topics*

- [Adjusting Transactions: How It Works](#)
- [Applying Receipts and On-Account Credit Memos: Points to Consider](#)
- [Write-offs and Receipts: Explained](#)

## FAQs for Define Approval Limits

### How can I manage the users that have approval limits?

You can only assign approval limits to valid users that are defined in your organization. The combination of user, document type, and currency identify a specific approval limit record. You can, for example, define multiple approval limit ranges for the same user and document type in each currency defined in your system.

Be sure to update approval limits when personnel changes occur.

## FAQs for Distribution Sets

### What are distribution sets?

Use distribution sets to account for miscellaneous, or non-invoice related, receipts. Distribution sets are groups of general ledger code combinations that you define to determine the credit accounts for positive miscellaneous receipt amounts and the debit accounts for negative receipt amounts.

## Define Statements

### Setting Up for Statements: Procedure

Print statements to provide your customers with a complete record of their invoice, debit memo, chargeback, receipt, on-account credit, credit memo, and adjustment activity for a specified period.

To set up for statements, complete these tasks:

- Set Receivables System Options for Statements
- Define Remit-to Addresses
- Define Statement Cycles
- Define Customer Site Profiles

### Set Receivables System Options for Statements

Set the necessary Receivables system options for statement processing.

To set Receivables system options for statements:

1. Navigate to the Create System Options or Edit System Options page.
2. Select the applicable business unit.
3. Click the Billing and Revenue tab, if it is not already displayed.
4. Enable the **Print remit-to address** option to print your remit-to addresses on your customer statements.

5. Enable the **Print home country** option to print your home country on transactions and statements that refer to addresses in that country.
6. In the **Default Country** field, select a country to use as your home country. This is the default country value for your remit-to addresses. The home country is also used in tax calculations and for taxpayer ID and tax registration number validation.
7. Complete or update the rest of Receivables system options according to your requirements, and save.

## Define Remit-to Addresses

Define remit-to addresses to provide default remit-to information on statements and transactions.

To define a remit-to address:

1. Navigate to the Create Remit-to Address page.
2. In the **Remit-to Address Set** field, select a reference data set.
3. The **Country** field displays the default country defined in Receivables system options. If necessary, select another country.
4. Enter the address details and save.
5. In the Receipt from Criteria section, click the **Create** icon to open the Create Receipt from Criteria window. Use this window to assign the remit-to address you just created to customer bill-to sites in specified locations.
6. In the **Country** field, select a country to assign customer bill-to sites this remit-to address.
7. If necessary, use the **State** field and **Postal Code** fields to further limit the remit-to address assignment, and save.

## Define Statement Cycles

Define statement cycles to determine when to send statements to your customers.

To define a statement cycle:

1. Navigate to the Manage Statement Cycles page.
2. In the Search Results section, click the **Add** icon.
3. In the **Name** field, enter a name for this statement cycle.
4. In the **Interval** field, select the interval that determines how often to generate statements: weekly, monthly, quarterly.
5. In the Cycle Dates section, click the **Add** icon.
6. In the **Business Unit** field, select the business unit that will use this statement cycle.
7. In the **Statement Date** field, enter the first date on which to print statements for the statement cycle.
8. Repeat steps 5 to 7 until you have the appropriate number of rows for the statement cycle interval for each applicable business unit.

For example, enter four rows for a quarterly interval or twelve rows for a monthly interval to cover the period of one year.

9. Enable the **Skip** option on a row if you want to skip an interval in the statement cycle.

For example, after creating a statement cycle with a monthly interval and twelve monthly statement dates, you decide to send statements bi-monthly instead of monthly. You can enable the **Skip** option on every other row to reduce the number of statements to six per year.

10. Complete the remaining fields according to your requirements, and save.

## Define Customer Site Profiles

After you define remit-to addresses and statement cycles, you must enable the appropriate profile settings on each customer bill-to site to which you plan to send statements. If you are not using one statement site for the customer, you can, depending on your requirements, assign a different statement cycle to the bill-to sites belonging to the same customer account. In this way each bill-to site can have its own statement for its site transactions.

To define customer site profiles for statements:

1. Navigate to the Edit Site page of the applicable customer site.
2. Navigate to the Statement and Dunning section.
3. Enable the **Send statement** option.
4. In the **Statement Cycle** field, select the statement cycle to use for this site.
5. Click the Late Charges tab.
6. Navigate to the Currency Settings section.
7. Click the **Add** icon.
8. In the **Currency** field, select the currency used by this site.
9. In the **Minimum Statement Amount** field, enter a minimum amount to generate statement. Receivables generates statements for the site whenever the minimum outstanding balance in the given currency is greater than this amount.
10. Complete the remaining fields according to your requirements, and save.

#### Related Topics

- [Profile Classes: Explained](#)
- [How can I use remit-to addresses?](#)

## Statement Cycles: Example

Use statement cycles to determine when to send statements to your customers. You assign statement cycles to customer and site level profiles.

If you print statements for a specific customer, then:

- If you defined a statement site for the customer, Oracle Fusion Receivables uses the statement cycle defined in the customer account profile as the default statement cycle to use for printing.
- If you didn't define a statement site, Receivables uses the statement cycle defined in the customer site profile for each applicable bill-to site included in the print run.

If you don't select a customer, then Receivables prints statements for all customers that have a statement cycle that matches the statement cycle you enter for the print run.

When you create a statement cycle, you define the interval to use for the cycle (weekly, monthly, quarterly) and the dates on which to print statements for the cycle. You can also indicate if Receivables should skip certain statement dates.

Receivables includes all activity from the last time you printed a statement for this customer to the current statement date, even if the customer statement cycle is set up to skip printing on one or more statement dates. Receivables also includes open debit items from prior periods in the statement.

## Scenario

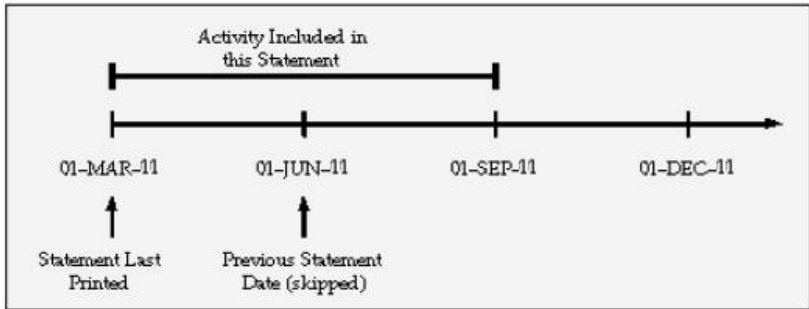
Consider the following criteria:

- System Date: 03-SEP-11
- Statement Date: 01-SEP-11
- Previous Statement Date: 01-JUN-11 (skipped)
- Statement Cycle: Quarterly

The activity included in this statement spans the date the statement was last printed on 01-MAR-11 to the current statement date of 01-SEP-11. The previous statement dated 01-JUN-11 had been skipped, so the activity for that period now shows on the current statement.



This diagram illustrates the activity included in this statement:



This table illustrates the logic used to include particular invoices on the statement:

Invoice Creation Date	Included in Statement?
Invoice Date: 28-FEB-11	No, unless it is either still open or was closed between 01-MAR-11 and 31-AUG-11.
Invoice Date: 30-AUG-11	Yes, because the invoice date is between the date the statement was last printed and the statement date.
Invoice Date: 02-SEP-11	No, because the invoice date is later than the statement date.

Related Topics

- [Aging Method: Explained](#)

## FAQs for Define Statements

### How can I create a statement site?

You can designate one of the sites belonging to a customer as a statement site. If you create a statement site, Oracle Fusion Receivables generates a single, consolidated statement for all the customer bill-to sites, rather than a statement for each site. You can only define one active statement site per customer.

To create a statement site:

- Assign the site the Statements business purpose.
- Set the **Statement, Dunning, and Late Charges Site Profiles Used** profile option to Yes.

### How do on-account and unapplied receipts appear on statements?

All receipts, including on-account and unapplied receipts, appear on the statement of the corresponding bill-to site. If there are any on-account or unapplied receipts without a bill-to site assignment, and if you don't have a statement site defined for the customer, then these receipts are not included on any of the bill-to site statements for the customer.

If you create a statement site for the customer, then Oracle Fusion Receivables summarizes on-account and unapplied receipts as credits and prints them on a separate page of the consolidated statement, before a summarized listing of subtotals for each of the customer bill-to sites.

## FAQs for Standard Messages

### Can I write messages in different languages?

Yes, you can write a message in any language that suits the needs of your enterprise. For any language, the text of a standard message can't exceed 255 characters. Oracle Fusion Receivables doesn't perform any other validation on messages.

### How can I add a message to a document?

You can print standard messages on customer statements, and on late charges documents presented as debit memos or interest invoices.

For statements, active standard messages appear as list of value choices in the Create Customer Statements program. The message you select appears at the bottom of the customer statement.

For late charge documents, active standard messages appear on the choice list of the **Message Text** field in the Late Charges tabbed section of the applicable customer or site profile. The message you select appears in the Notes section of the late charges document for this customer.

## 2 Manage Receivables System Options

### Updating System Option Records: Critical Choices

Certain Receivables system option settings have critical implications for the way Oracle Fusion Receivables functions for a given business unit. You may need to do some advance planning before deciding how to set certain Receivables system options.

Considerations for Receivables system option settings include:

- Salespersons
- Header Level Rounding
- Allow Change to Printed Transactions
- Discounts

#### Salespersons

If you intend to use revenue accounting, you must enable the **Require salesperson** Receivables system option. Revenue accounting requires that you assign sales credits to all transactions that can be adjusted for either revenue or sales credits.

If you enable the **Require salesperson** Receivables system option, use the **Sales Credit Percent Limit** field to limit the percentage of revenue plus non-revenue sales credit that a salesperson can have on any transaction line.

If you don't enter a value in the **Sales Credit Percent Limit** field, then no sales credit limit validation is performed during revenue accounting.


#### Header Level Rounding

Depending on the legal requirements of your home country, you may need to round amounts at the transaction header level for the receivable account, and then account for and post the difference in a separate account between this rounded amount and the sum of the rounded line amounts for the respective revenue accounts. To do this, enable the **Use header level rounding** option and define a **Header Rounding Account**.

The rounding difference between the header level and line level rounding is assigned to the **Header Rounding Account**.


If you enable the **Use header level rounding** option, then Receivables displays a rounding distribution line for all transactions, regardless of currency. If the transaction is in the ledger currency, then the amount of this line is zero.

If you don't enable the **Use header level rounding** option, Receivables rounds amounts at the line level and posts any rounding difference to the receivable account.

 **Caution:** Once you enable Header Level Rounding and save the Receivables system options record, you can't disable the feature.

## Allow Change to Printed Transactions

To allow updates to transactions that have been printed, enable the **Allow change to printed transactions** option. This option also determines whether you can update a customer address when printed, posted, or applied transactions are assigned to that address.

 **Note:** You can't update a transaction if it has activity against it, regardless of how you set this option. Examples of activity include payments, credit memos, adjustments, accounting, and assigning the transaction to a balance forward bill.

## Discounts

To allow Receivables to accept unearned discounts, enable the **Allow unearned discounts** option. Unearned discounts are discounts a customer takes after the discount period passes. The Receivables system options record is the only place that determines whether you can accept unearned discounts for the given business unit.

To allow discounts to be taken for partial payments against open debit items, enable the **Discount on partial payment** option. A partial payment is a payment that is less than the remaining amount due. If this option is enabled, you can still decide to disallow discounts on partial payments at the transaction level when defining payment terms.

If you never allow discounts on partial payments, then do not enable the **Discount on partial payment** option.

### Related Topics

- [Revenue Recognition Settings: Explained](#)
- [Calculating Discounts: Explained](#)

## Using Header Level Rounding: Example

This example illustrates how header level rounding processes currency conversions and accounts for rounding differences.

## Scenario

ABC Company uses euros as the ledger currency, and it receives an invoice with three line items in Norwegian krone. For this example, the conversion rate between the krone and the euro is 6.55957.

## Transaction Details

The **Use header level rounding** Receivables system option is enabled for the applicable business unit and a **Header Rounding Account** is defined.

This table shows the calculations performed to convert each line amount on the invoice:

Item/Description	Amount in Krone	Conversion Rate	Amount in Euros	Comment
Paper	15.00	6.55957	2.29	rounded up
Pens	12.00	6.55957	1.83	rounded up
Envelopes	25.00	6.55957	3.81	rounded down
Subtotal	52.00	N/A	7.93	sum of items
Rounding Difference	N/A	N/A	- 0.01	N/A
Total Amount	52.00	6.55957	7.92	rounded down

## Analysis

Because the **Use header level rounding** Receivables system option is enabled, Oracle Fusion Receivables must calculate the rounding difference between the currency conversion of the total invoice amount at the header level assigned to the receivable account and the sum of the currency conversions at the line level assigned to each revenue account. This difference is placed in the designated header rounding account.

## Conversion Results

Receivables first converts each line item separately from krone to euros, and then adds them together, for a total of 7.93 EUR. Receivables then separately adds the line amounts in the invoice currency (krone) and then converts to the ledger currency, for a total of 7.92 EUR.

The rounding difference of .01 is assigned to the header rounding account as defined in Receivables system options.

## Tax Invoice Printing Options

The **Tax Invoice Printing Options** Receivables system option identifies the method Oracle Fusion Receivables uses to print tax amounts on transactions. The value you enter here becomes the default value for customer profile classes.

### Tax Invoice Printing Options

#### European Tax Format

Does not itemize tax information for each line, but does print tax rates as the last column of invoice lines. Prints freight items last. At the end of the invoice, the Tax Summary by Tax Name section includes a summary of taxable amounts and tax charged for each tax rate code.

#### Itemize and Sum

Itemizes tax information for each invoice line. At the end of the invoice, the Tax Summary by Tax Name section includes a summary of the tax charged for each tax rate code. At the end of the invoice, Receivables prints the invoice subtotal, tax, shipping, and invoice total.

### **Itemize Taxes**

Itemizes tax information for each invoice line.

### **Itemize With Recap**

Itemizes tax information for each invoice line. At the end of the invoice, the Tax Summary by Tax Name section includes a summary of the tax charged for each tax rate code.

### **Recap**

Does not itemize tax information for each line. At the end of the invoice, the Tax Summary by Tax Name section includes a summary of the tax charged for each tax rate code.

### **Sum Taxes**

Does not itemize tax information for each line. At the end of the invoice, the Tax Summary by Tax Name section includes a summary of the tax charged for each tax rate code. At the end of the invoice, Receivables prints the invoice subtotal, tax, shipping, and invoice total.

### **Summarize By Tax Name**

Does not itemize tax information for each line. At the end of the invoice, the Tax Summary by Tax Name section includes a summary of the tax charged for each printed tax name and rate.

### **Total Tax Only**

Displays only the total tax amount at the bottom of the document.

#### *Related Topics*

- [Tax Handling on Receivables Transactions: Examples](#)
- [How do I print transactions?](#)

## Tuning Segments: Explained

Use the AutoInvoice: Tuning Segments section of the Billing and Revenue tab of the Create and Edit Receivables System Options pages to designate Accounting and System Items flexfield segments as tuning segments. Tuning segments help increase performance of AutoInvoice. The tuning segment is the segment most frequently accessed by AutoInvoice.

### Accounting Flexfield Tuning Segment

If you want to increase the performance of AutoInvoice and indices already exist for the GL\_CODE\_COMBINATIONS table, use the value that you specified for your index as the Accounting Flexfield tuning segment. If you defined a concatenated index, use the first column of your concatenated index.

If no indices exist for the `GL_CODE_COMBINATIONS` table, enter the segment with the most distinct values for your Accounting Flexfield tuning segment.

## System Items Flexfield Tuning Segment

If you want to increase the performance of AutoInvoice and indices already exist for the `MTL_SYSTEM_ITEMS` table, use the value that you specified for your index as your System Items Flexfield tuning segment. If you defined a concatenated index, use the first column of your concatenated index.

If no indices exist for the `MTL_SYSTEM_ITEMS` table, enter the segment with the most distinct values for your System Items Flexfield tuning segment.

## Log File Message Levels: Explained

In the **Log File Message Level** field of the AutoInvoice section of the Billing and Revenue tab of the Create or Edit System Options page, enter a value from 0 to 5 to indicate the amount of detail that you want to display in the AutoInvoice log file.

For day-to-day business needs and to improve performance, set the level to 0. If you consistently experience errors while running AutoInvoice, you can set the output to a higher level to review more detailed information in the log about the errors.

## Meaning of the Log File Message Levels

Message Level 0 provides the following entries in the log file:

- Product Version
- Program Name
- AutoInvoice Start Time
- AutoInvoice Concurrent Request Arguments
- Error and Warning Messages
- AutoInvoice End Time
- AutoInvoice Logical Steps

Message Level 1 provides all of the entries for Message Level 0 plus:

- Time-Stamped function labels

Message Level 2 provides all of the entries for Message Levels 0 and 1 plus:

- Sizes of allocated arrays
- Dynamic SQL statements
- Number of rows updated, inserted, and deleted

Message Level 3 provides all of the entries for Message Levels 0, 1, and 2 plus:

- Method IV SQL array values

Message Level 4 provides all of the entries for Message Levels 0, 1, 2, and 3 plus:

- Values of all variables that are used to call FND or Tax routines

Message Level 5 provides all of the entries for Message Levels 0, 1, 2, 3, and 4 plus:

- Details of all bad lines and rejected lines. This provides all messages needed for C debugging of AutoInvoice.

## FAQs for Manage Receivables System Options

### What's the days per posting cycle?

The **Days per Posting Cycle** setting lets you process the transactions you are posting in smaller groups to ensure that you don't run out of rollback space during posting.

For example, if your accounting period is 30 days and you set this value to 30, the posting program uses only one cycle. If your accounting period is 30 days and you set this value to 17, the posting program uses two cycles. Best practice is to set this field to a value that is less than the number of days in your accounting period.

### What happens if I allow transaction deletion?

Enable the **Allow transaction deletion** option if you want to let users delete transactions from Oracle Fusion Receivables after the transactions have been saved. If you don't enable this option, all Receivables users are prevented from deleting transactions.

If an installation is legally required to number transactions sequentially with no missing transaction numbers, then best practice is to not enable this option.

If you enable the **Allow transaction deletion** option, you can still control which users can delete transactions using function security.

### How can I determine the memory allocation?

Enter in the **Maximum Memory in Bytes** field the value that represents the amount of memory to allocate to AutoInvoice for validation. The default is 65535 bytes. For best results, enter a value that is the maximum number of records that you import, rounded to an even number, multiplied by 1024. For example, if you use AutoInvoice to import no more than 100 records at a time, enter a value of 102400.

During AutoInvoice processing, if you receive a message that indicates the application failed to allocate memory, then enter a lower number. If you receive a message that the memory is not large enough, then enter a higher number.

### When do I use days to AutoApply a receipt?

Enter in the **Days to AutoApply a Receipt** field the age in days of receipts that AutoApply considers for application against customer transactions. Use this field if your customers often pay for transactions before they are created.

AutoApply looks for and attempts to apply open receipts to transactions created within the number of days that you specify. For example, if you enter 10, then AutoApply considers all receipts created within the past 10 days for application to transactions. Receipts created longer than 10 days ago are not considered.



If you don't enter a value in this field, AutoApply only attempts to match receipts created on the system date.

## What are the exception rule activities?

The active application exception rule set determines the action to perform on overpayment and underpayment amounts after receipt application. You can define the default receivables activity to use to process these payments when the action is either a billing adjustment or a write-off.

The **Exception Rule Adjustment Activity** field provides the default receivables activity to use for adjustments on overpayments or underpayments. The **Exception Rule Write-off Activity** field provides the default receivables activity to use for write-offs of overpayments or underpayments.

## When are receipts required for a bill-to site?

Enable the **Require billing location for receipts** option to require that a bill-to site be associated with a receipt. If enabled, Oracle Fusion Receivables doesn't create receipts that don't have a bill-to site.

Use this option for customers without statement sites. If you don't enable this option, and you have receipts for customers without statement sites and without a bill-to site associated with the receipt, the unapplied amount of the receipt won't appear on any of the statements for this customer.

## What's the difference between the realized gains and losses accounts and the cross currency rounding account?

The realized gains and realized losses accounts are used to account for the conversion rate gain or loss in the ledger currency resulting from a cross-currency receipt application. For example, if the conversion rate for a foreign currency invoice is 1.7 and the conversion rate of the payment for this invoice is 2.0, Oracle Fusion Receivables posts the difference as a gain to the realized gains account.

The cross-currency rounding account is used to record rounding error amounts created during a cross-currency receipt application. You must define a rounding error account if you create cross-currency receipts.

## Define E-Mail Delivery

### Transaction Delivery E-Mail Subject Line: Examples

Use the E-Mail Subject fields in the Transaction Delivery using E-Mail section of the Billing and Revenue tabbed region of Receivables system options to create an e-mail subject line for the transactions you deliver to customers using e-mail.

Receivables inserts a period after the business unit and the transaction number, and inserts a space between each of the field values.

The following examples illustrate how the subject line appears depending on the settings you use.

## Example 1

The resulting e-mail subject line for the settings in this table is: "Your invoice is now ready for review. Vision Operations. INV1234."

Field	Value
E-Mail Subject	Your invoice is now ready for review.
Include Business Unit in E-Mail Subject	Last
Include Transaction Number in E-Mail Subject	Last

## Example 2

The resulting e-mail subject line for the settings in this table is: "Vision Operations. Your invoice is now ready for review. INV1234."

Field	Value
E-Mail Subject	Your invoice is now ready for review.
Include Business Unit in E-Mail Subject	First
Include Transaction Number in E-Mail Subject	Last

## Example 3

The resulting e-mail subject line for the settings in this table is: "Your invoice is now ready for review. INV1234."

Field	Value
E-Mail Subject	Your invoice is now ready for review.
Include Business Unit in E-Mail Subject	Do Not Include
Include Transaction Number in E-Mail Subject	Last

## Transaction Delivery Using E-Mail: How It Works

Print and send Receivables transactions to designated customers using e-mail.

When you run the Print Receivables Transactions program, the program sends transactions as either a PDF or Zipped PDF file to the designated e-mail addresses of the customer accounts and sites that are set up for e-mail delivery.

 **Note:** Delivery using e-mail is for individual transactions only, and not statements or balance forward bills.


### Settings That Affect Delivery Using E-Mail

These settings affect print delivery using e-mail:

- **Print Option:** Set the **Print Option** to Print or Do Not Print at the customer site or customer account level, depending upon whether you want to print transactions for specific customer sites or all sites belonging to a customer account.

If there is no **Print Option** value enabled at the site or account level, Receivables uses the required **Print Option** setting on the transaction type.

If necessary, you can exclude or include individual transactions in a print run by changing the **Print Option** setting on transactions.

 **Note:** If the transaction type doesn't have the **Open Receivable** option enabled (that is, it is a Void transaction), then the e-mail delivery process ignores the **Print Option** settings at the customer site and account level and uses the setting assigned to the transaction type. You can still override this setting on the transaction.

- **Preferred Delivery Method:** Set the **Preferred Delivery Method** field to E-Mail at the customer site or customer account level, depending upon whether you want to deliver printed transactions using e-mail for specific customer sites or all sites belonging to a customer account.
- **Customer contacts:** Assign at least one customer contact at the account or site level both an e-mail address and the Bill-to purpose. These rules apply to customer contacts:
  - The contact must be active when the Print Receivables Transactions program is run.
  - If a customer account or site doesn't have any contact with both an e-mail address and a Bill-to purpose, then transactions are not delivered for this customer account or site.
  - If a customer account or site does have a contact with both an e-mail address and a Bill-to purpose, but the same contact e-mail address is also assigned the Collections or Dunning purpose, then this contact is excluded from e-mail delivery.
- **E-Mail Receivables system options:** Use the Transaction Delivery using E-Mail section of the Billing and Revenue tabbed region of the Receivables System Option pages to set up the details of transaction delivery using e-mail for the applicable business units. Enter these values in the corresponding fields:
  - **From Name:** Name of your enterprise.
  - **From E-Mail:** E-mail address of your enterprise.
  - **Reply-to E-Mail:** E-mail address of your enterprise that your customers can send an e-mail to.
  - **E-Mail Subject:** Text of the e-mail subject line.

- **Include Business Unit in E-Mail Subject:** Option to include the name of your business unit in the subject line.
- **Include Transaction Number in E-Mail Subject:** Option to include the transaction number in the subject line.
- **E-Mail Body:** Text of the e-mail message. Include appropriate formatting.

❗ **Important:** If you are printing and delivering transactions using e-mail, then you must enter an e-mail address in either the **From E-Mail** field or the **Reply-to E-Mail** field. All other fields are optional.

## How Transactions are Delivered Using E-Mail

Once your setup is complete, use the Print Receivables Transactions program to print customer transactions. Use the **Output File Type** parameter to specify the output to use for the print run. For print delivery using e-mail, you must select either PDF or Zipped PDFs. If you select Zipped PDFs, the file includes an index file of the print run, to identify the first and last page of each printed transaction.

When you create a transaction, Receivables looks in your setup for the **Print Option** setting to assign to the transaction in this order:

- Customer site profile
- Customer account profile
- Transaction type

During print processing, the e-mail delivery process verifies for each transaction:

- **Print Option** is set to Print.

You can update the **Print Option** setting on individual transactions before printing.

- **Preferred Delivery Method** field is set to E-Mail for the customer account or site.
- At least one active customer contact has both an e-mail address and Bill-to purpose. The e-mail delivery process looks for an e-mail address to send transactions to in this order:
  - E-mail address of the bill-to customer contact on the transaction.
  - E-mail addresses of the contacts of the customer site that are assigned a Bill-to purpose, but not assigned the Collections or Dunning purpose.
  - E-mail addresses of the contacts of the customer account that are assigned a Bill-to purpose, but not assigned the Collections or Dunning purpose.
- Receivables system options contains an active e-mail address in either the **From E-Mail** field or the **Reply-to E-Mail** field for the applicable business unit.

The name of the output PDF file for transactions delivered using e-mail uses the format:

**<INV\_NUMBER>\_<INSTALLMENT\_NUMBER>.PDF**

## FAQs for Define E-Mail Delivery

### Why can't I print transactions using e-mail delivery?

Your e-mail delivery setup is incomplete.

This is most likely due to one of two settings:

- In Receivables system options, you must enter an active e-mail address in either the **From E-Mail** field or the **Reply-to E-Mail** field of the Transaction Delivery using E-Mail section of the Billing and Revenue tabbed region for the applicable business unit.
- At least one customer contact must have both a Bill-to purpose assignment and an active e-mail address.

If the transaction doesn't have a bill-to customer contact, or if that contact doesn't have an e-mail address, then the e-mail delivery process looks for all active customer contacts at the account or site level that have both a Bill-to purpose and an e-mail address.

If any exist, then the transaction is delivered to all of these contacts, with the exception of contact e-mail addresses also assigned the Collections or Dunning purpose.

If none exist, then the transaction is not printed.

## Define B2B XML

### Setting Up Oracle B2B to Send Receivables Transactions in XML: Procedures

Oracle B2B Server is an Oracle SOA Suite component that manages interactions between deploying companies and trading partners. Oracle Fusion Receivables supports an outbound Oracle B2B flow using Oracle B2B Server to send transactions to customer trading partners in XML format.

The setup of the Oracle B2B flow for Receivables makes use of these existing elements:

- XML Schema document guideline
- OAG-7.2.1-PROCESS\_INVOICE\_002-OAG\_DEF document definition
- Host trading partner MyCompany

To set up Oracle B2B to send Receivables transactions in XML:

- Configure the host and remote trading partners
- Configure agreements between the host and remote trading partners

### Configuring Trading Partners

Configure your enterprise as the host trading partner, and all of your customers that receive XML documents as remote trading partners.

To configure trading partners:

1. Log in to the Oracle B2B Server.
2. Navigate to the Administration page.
3. Click the Document tab.
4. Load the OAG-7.2.1-PROCESS\_INVOICE\_002-OAG\_DEF document definition file.
5. Click the Types tab.
6. Add a new Internal Identifier with the name B2B Trading Partner Code.  
This name matches the field name on the customer account profile.
7. Navigate to the Partners page.
8. In the Partner regional area, select the default host partner MyCompany.
9. If necessary, update the default host partner name to reflect your enterprise.
10. Click the Documents tab.
11. Verify that the OAG-7.2.1-PROCESS\_INVOICE\_002-OAG\_DEF document definition is assigned to the host trading partner.
12. Ensure that the Sender option is enabled.
13. In the Partner regional area, click the Add icon.
14. Enter the name of a remote trading partner.
15. Select the **Internal Identifier Type** B2B Trading Partner Code that you previously created and enter the Value for the identifier.  
This is the value that you will enter in the **B2B Trading Partner Code** field on the customer account profile of this remote trading partner.
16. Click the Documents tab.
17. Click the **Add** icon to associate the OAG-7.2.1-PROCESS\_INVOICE\_002-OAG\_DEF document definition with the remote trading partner.
18. Enable the Receiver option.
19. Click the Channel tab.
20. Define a channel for the remote trading partner.  
The channel determines how the XML transaction is delivered to the remote trading partner from B2B: directly; via the Oracle Supplier Network (OSN), or via a third party.  
If you are communicating using Oracle Supplier Network (OSN), select the Generic Identifier and enter the IP Address of the OSN machine.  
For more information about configuring a channel, refer to the Oracle Fusion Middleware User Guide.
21. Repeat steps 13 to 20 for each remote trading partner.

## Configuring Agreements

A trading partner agreement defines the terms that enable two trading partners, the sender and the receiver, to exchange business documents. The agreement identifies the trading partners, trading partner identifiers, document definitions and channels.

An agreement consists of two trading partners, the host trading partner and one remote trading partner, and represents one type of business transaction between these partners. For example, if the host trading partner MyCompany and the remote trading partner ABC Solutions regularly exchange both purchase orders and invoices, then two separate agreements are needed for each document definition.

To configure agreements between the host and remote trading partners:

1. Log in to the Oracle B2B Server.
2. Navigate to the Partners page.

3. In the Agreement regional area, click the Add icon to open a new agreement for the host trading partner MyCompany.
4. Enter the agreement ID and Name.
5. Enter the agreement parameters.
6. Select the Document Definition OAG-7.2.1-PROCESS\_INVOICE\_002-OAG\_DEF for this agreement.
7. Select the remote trading partner to include in this agreement.
8. Select the channel for the remote trading partner.
9. Add identifiers for the remote trading partner.
10. Click Save to save the agreement.
11. Click Validate to validate the agreement.
12. Click Deploy to deploy the agreement.

Deployment is the process of activating an agreement from the design-time repository to the run-time repository.

13. Repeat steps 3 to 12 for each agreement between the host trading partner and this remote trading partner.

## Setting Up B2B XML Invoicing for Receivables: Explained

Send Oracle Fusion Receivables transactions electronically in XML format to designated customers using the Oracle B2B outbound service. Oracle B2B is a SOA Suite component that manages the interactions between your enterprise and your customers.

A customer designated for receiving transactions in XML format is called a trading partner. You set up your customer trading partners in Oracle B2B to send XML transactions using one of these transmission methods:

- Directly to the customer system.
- Using Oracle Supplier Network.
- Using a third-party service.

In Receivables, you must set up the related customer accounts for B2B XML using the Invoicing section of the account profile. The settings that you use apply to all sites belonging to the customer account.

To set up a customer account for B2B XML invoicing:

1. In the Invoicing section of the Account Profile tab, select XML in the **Preferred Delivery Method** field.
2. In the **B2B Trading Partner Code** field, enter the connecting identifier of the trading partner.

If applicable, you can assign the same trading partner code to multiple customer accounts.

**! Important:** This is a mandatory value for XML delivery. If you don't enter the connecting identifier, then you can't send transactions in XML format to this customer account.

3. Using the **Enable for XML Invoicing** check boxes, select the transactions that you want to send to this trading partner in XML format. Receivables selects all eligible transactions whenever you run the XML Invoicing Program for this customer account.

After you set up your customer accounts for B2B XML invoicing, you can customize the XML invoice template to enhance the information included in the invoice output by mapping attributes in the system to attributes in the User Area section of the XML invoice template.

Once your setup is complete, use the XML Invoicing Program to send customer transactions in XML format. If any transactions fail during the transmission process, use the Manage Transactions page to review failed transactions and pursue corrective actions. You can then resend these transactions using the XML Invoicing Program.

## Processing XML Transactions for Receivables: Explained

Use the XML Invoicing Program to send Oracle Fusion Receivables transactions to your customer trading partners.

These restrictions apply to the transactions you send:

- Transactions belong to customer accounts set up for XML invoicing.
- Transactions are complete.
- Transactions have a status of Print Pending.
- Transactions (invoice, credit memo, debit memo, chargeback) are enabled for XML invoicing in the related customer account profiles.
- Transactions have the printing option on the transaction type set to Print.

### Processing New Transactions

Select New Transactions Only in the **Transactions Included** field to send new transactions in XML format to your customers. You can process all eligible transactions for all business units and customers, or you can use the available parameters to limit the transactions processed according to your needs.

### Processing Failed Transactions

Transactions can fail to transmit correctly at one of three stages in the transmission process.

These three stages are:

- Receivables process
- SOA process
- B2B process

This table describes the potential errors that can occur at each stage of the B2B XML transmission process:

XML

Process	Error Condition	Error Notification	Corrective Action
Receivables	Customer Account is missing the B2B trading partner code.	Transaction marked as Failed with corresponding error reason.	Add the B2B trading partner code to the appropriate account.
SOA	Receivables system or server is down.	Transaction is not marked because it was not processed. System error notification only on the EM console.	System administrator restarts and reinitiates SOA for the transaction.
SOA	Transformation errors.	Transaction marked as Failed with corresponding error reason.	System administrator fixes the transformation errors.
SOA	B2B server is down.	Transaction marked as Failed with corresponding error reason.	System administrator restarts the B2B server.
B2B	B2B setup is either incomplete or incorrect.	Transaction marked as Failed with corresponding error reason.	B2B administrator fixes the B2B setup errors.



Process	Error Condition	Error Notification	Corrective Action
---------	-----------------	--------------------	-------------------

## FAQs for Define B2B XML

### Why can't I configure B2B XML on the profile class?

You configure and maintain B2B XML invoicing parameters in the related customer account profile. Customer sites share the settings of the related customer account.



## 3 Define Customer Billing Configuration

### Define AutoInvoice

#### Setting Up Data for AutoInvoice: Points to Consider

To ensure that the AutoInvoice process works properly, you need to prepare your Receivables environment for any new data that you want to import. If your original system uses any setup data which is not yet defined in Receivables, you must define this data before using AutoInvoice.

There are these points to consider when setting up data for AutoInvoice:

- Data Checklist
- AutoInvoice Setup
- Transaction Flexfield

#### Data Checklist

Ensure that you have set up and updated the appropriate records in Receivables and related applications.

Add or update this setup data:

- Add or import customers, if your original system contains data for customers that are not yet defined in Receivables.
- Add units of measure, if your original system uses units of measure not yet defined.
- Add or update in Oracle Fusion General Ledger this data:
  - Currencies, if your original system uses currencies not yet defined.
  - Accounting flexfield segment values, if your original system uses values not yet defined.
- Add or update in Oracle Fusion Tax this tax data:
  - Tax rates assigned to tax rate codes that are not yet defined.
  - Tax rates associated with products shipped to specific locations.
  - Full or partial customer and item tax exemptions.
- Add or update these Receivables lookup codes:
  - Free on Board (FOB) lookup codes, if your original system uses FOB point codes not yet defined.
  - Freight carrier lookup codes.
- Add or update this Receivables data:
  - AutoAccounting (This is a required setup to use AutoInvoice)
  - Payment terms
  - Transaction types
  - Transaction sources

- Salespersons
- Revenue scheduling rules

## AutoInvoice Setup

Review and update Receivables data specific to AutoInvoice.

Review and update this data:

- AutoInvoice Grouping Rules: Define additional grouping rules or update the default grouping rule provided by Receivables. AutoInvoice uses grouping rules to determine how to create transactions.

AutoInvoice uses the following hierarchy when determining the grouping rule to use:

- Transaction source
  - Customer site
  - Customer profile
  - System options
- AutoInvoice Line Ordering Rules: Define line ordering rules for AutoInvoice to determine how to order transaction lines. AutoInvoice randomly orders lines on your transactions if you don't define line ordering rules.
  - AutoInvoice Transaction Source Automatic Receipt Handling: If you want AutoInvoice to automatically evaluate imported credits for receipt handling, enable the **Receipt Handling for Credits** option on the AutoInvoice transaction source.
  - Receivables System Options: Set Receivables system options for AutoInvoice in the Billing and Revenue tab:
    - Customers section: **Grouping Rule** field: Assign an AutoInvoice grouping rule to use as part of the default hierarchy for selecting a grouping rule during transaction processing.
    - AutoInvoice section: **Purge interface tables** option: Enable this option to purge data automatically after running AutoInvoice.
    - AutoInvoice section: **Maximum Memory in Bytes** field: Enter a value that represents the amount of memory to allocate to AutoInvoice for validation.
    - AutoInvoice section: **Log File Message Level** field: Enter a level from 0 to 5 to indicate the amount of detail that you want to display in the AutoInvoice log file.
    - AutoInvoice section: **Accounting Dates Out of Order** field: Select **Reject** or **Adjust** to determine how AutoInvoice processes transactions when the accounting date is out of order within the document sequence.

 **Note:** You only use this setting when the primary ledger is enabled for document sequencing.

- Profile Options: Set these profile options for AutoInvoice:
  - **ID Flexfield Code:** Specify the ID of the flexfield code used by AutoInvoice.
  - **Maximum Lines per AutoInvoice Worker:** Specify the maximum number of lines per AutoInvoice worker.
  - **Source Code:** Specify the source code used by AutoInvoice.
  - **Use Parallel Hint:** Enable parallel hints in AutoInvoice.
  - **AutoInvoice Gather Statistics Allowed:** If you set this profile option to Yes, then when you submit AutoInvoice, the program first analyzes the interface tables (RA\_INTERFACE\_LINES\_ALL, RA\_INTERFACE\_DISTRIBUTIONS\_ALL, and RA\_INTERFACE\_SALESCREDITS\_ALL) and gathers statistics to determine how best to execute the transaction import.

If the number of records to be imported and the number of worker processes are approximately the same as the previous submission of AutoInvoice, then you can set this profile option to No and skip this analysis.

## Transaction Flexfield

Receivables uses the transaction flexfield to uniquely identify each transaction and transaction line you import using AutoInvoice. Transaction flexfields are also used to refer to and link transaction lines.

You must define both a line-level and a header-level transaction flexfield. All segments in the line-level transaction flexfield that refer to header information must also exist in the header-level transaction flexfield. For example, if you define a line-level transaction flexfield with four segments, and only the last two segments refer to line-level information, define the header-level transaction flexfield using the first two segments.

If you don't create Reference and Link-to transaction flexfields, then Receivables will use the line-level transaction flexfield structure to link and reference different lines. You don't have to define separate Reference and Link-to transaction flexfields in this case.

However, if you are planning to create a customized form to enter interface data to display the Reference and Link-to transaction flexfields, then you must define these transaction flexfields. These flexfields must have the same flexfield structures as the line-level transaction flexfield.

### *Related Topics*

- [Legal Entity Document Sequencing in Receivables: Points to Consider](#)

## Using AutoInvoice Grouping Rules: Example

This example illustrates how to use grouping rules to group transaction lines into transactions during AutoInvoice import.

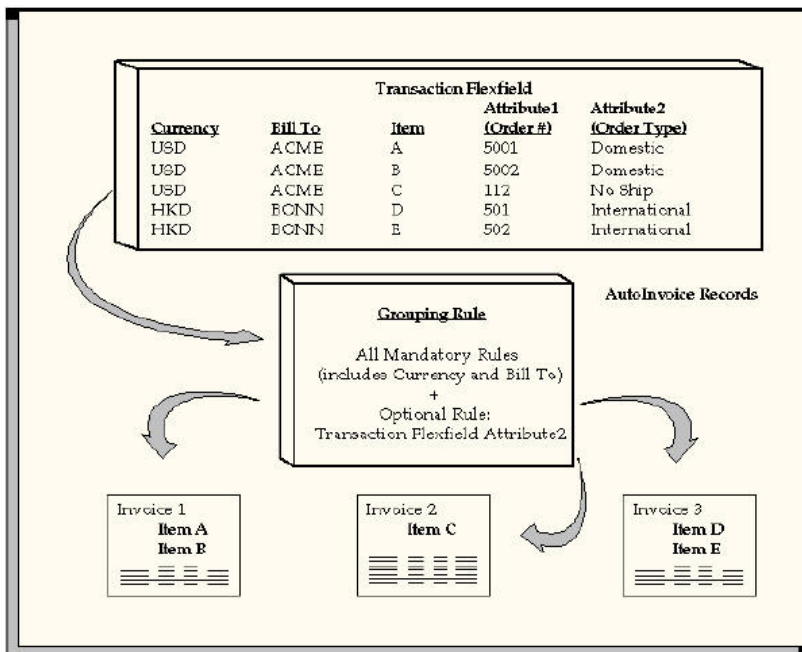
### Scenario

Define an AutoInvoice grouping rule that specifies that to appear on the same invoice, items must match on all mandatory attributes, such as currency (CURRENCY\_CODE) and customer bill-to address (ORIG\_SYSTEM\_BILL\_ADDRESS\_ID), and must also match on the optional attribute of sales order type (SALES\_ORDER\_SOURCE).

### Transaction Details

During AutoInvoice import, assume that all mandatory attributes match other than currency and customer bill-to address.

This diagram illustrates how three imported invoices are created according to the AutoInvoice grouping rule defined in this example:



## Analysis

Items A and B share the same currency and sales order type, so they appear on the same invoice (Invoice 1). Item C has the same currency as A and B, but it has a different sales order type, so it appears on its own invoice (Invoice 2). Items D and E share the same currency and sales order type, so they appear on the same invoice (Invoice 3).

## Result

Because of the optional attribute of sales order type, AutoInvoice created three invoices. If the grouping rule had designated only mandatory attributes, then AutoInvoice would have created only two invoices.

## Mandatory and Optional Grouping Rule Attributes: Explained

AutoInvoice grouping rules contain transaction attributes that must be identical for all items on the same transaction. For example, transaction number (TRX\_NUMBER) is a mandatory attribute of all grouping rules. If you have two records in the interface tables with different transaction numbers, AutoInvoice creates separate transactions for each record.

The AutoInvoice grouping rule provides both mandatory and optional transaction attributes for imported transactions. You can't delete a mandatory attribute from any grouping rule, but you can add optional attributes to the mandatory attributes to create a new grouping rule.

## Mandatory Transaction Attributes

The AutoInvoice grouping rule provides the following mandatory transaction attributes from the RA\_INTERFACE\_LINES\_ALL table. All of these transaction attributes apply to all transactions created using AutoInvoice grouping rules:

- BILL\_PLAN\_NAME
- BILL\_PLAN\_PERIOD
- COMMENTS
- CONS\_BILLING\_NUMBER
- CONVERSION\_DATE
- CONVERSION\_RATE
- CONVERSION\_TYPE
- CREDIT\_METHOD\_FOR\_ACCT\_RULE
- CREDIT\_METHOD\_FOR\_INSTALLMENTS
- CURRENCY\_CODE
- CUSTOMER\_BANK\_ACCOUNT\_ID
- CUST\_TRX\_TYPE\_ID
- DOCUMENT\_NUMBER
- DOCUMENT\_NUMBER\_SEQUENCE\_ID
- GL\_DATE
- HEADER\_ATTRIBUTE1-15
- HEADER\_ATTRIBUTE\_CATEGORY
- HEADER\_GDF\_ATTRIBUTE1-30
- INITIAL\_CUSTOMER\_TRX\_ID
- INTERNAL\_NOTES
- INVOICING\_RULE\_ID
- ORIG\_SYSTEM\_BILL\_ADDRESS\_ID
- ORIG\_SYSTEM\_BILL\_CONTACT\_ID
- ORIG\_SYSTEM\_BILL\_CUSTOMER\_ID
- ORIG\_SYSTEM\_SHIP\_CONTACT\_ID
- ORIG\_SYSTEM\_SHIP\_CUSTOMER\_ID
- ORIG\_SYSTEM\_SOLD\_CUSTOMER\_ID
- ORIG\_SYSTEM\_BATCH\_NAME
- PAYMENT\_SERVER\_ORDER\_ID
- PAYMENT\_SET\_ID
- PREVIOUS\_CUSTOMER\_TRX\_ID
- PRIMARY\_SALESREP\_ID
- PRINTING\_OPTION
- PURCHASE\_ORDER
- PURCHASE\_ORDER\_DATE
- PURCHASE\_ORDER\_REVISION
- REASON\_CODE

- RECEIPT\_METHOD\_ID
- RELATED\_CUSTOMER\_TRX\_ID
- SET\_OF\_BOOKS\_ID
- TERM\_ID
- TERRITORY\_ID
- TRX\_DATE
- TRX\_NUMBER

## Optional Transaction Attributes

The AutoInvoice grouping rule provides the following optional transaction attributes from the RA\_INTERFACE\_LINES\_ALL table. You can assign one or more of these attributes to transaction classes within a grouping rule:

- ACCOUNTING\_RULE\_DURATION
- ACCOUNTING\_RULE\_ID
- ATTRIBUTE1-15
- ATTRIBUTE\_CATEGORY
- INTERFACE\_LINE\_ATTRIBUTE1-15
- INTERFACE\_LINE\_CONTEXT
- INVENTORY\_ITEM\_ID
- REFERENCE\_LINE\_ID
- RULE\_START\_DATE
- SALES\_ORDER
- SALES\_ORDER\_DATE
- SALES\_ORDER\_LINE
- SALES\_ORDER\_REVISION
- SALES\_ORDER\_SOURCE
- TAX\_CODE
- TAX\_RATE

## Line Ordering Rule Transaction Attributes: Explained

AutoInvoice uses line ordering rules to determine how to order and number each line of a transaction, after AutoInvoice has grouped transactions into invoices, debit memos, and credit memos. You can specify an AutoInvoice line ordering rule for each AutoInvoice grouping rule that you create.

### Transaction Attributes

AutoInvoice provides the following transaction attributes from the RA\_INTERFACE\_LINES\_ALL table for use with line ordering rules:

- ACCOUNTING\_RULE\_DURATION
- ACCOUNTING\_RULE\_ID
- ACCOUNTING\_RULE\_NAME
- AMOUNT



- ATTRIBUTE\_CATEGORY
- ATTRIBUTE1-15
- FOB\_POINT
- INTERFACE\_LINE\_ATTRIBUTE1-15
- INTERFACE\_LINE\_CONTEXT
- ORIG\_SYSTEM\_SHIP\_ADDRESS\_ID
- QUANTITY
- QUANTITY\_ORDERED
- REASON\_CODE
- REASON\_CODE\_MEANING
- REFERENCE\_LINE\_ATTRIBUTE1-15
- REFERENCE\_LINE\_CONTEXT
- REFERENCE\_LINE\_ID
- SALES\_ORDER
- SALES\_ORDER\_DATE
- SALES\_ORDER\_LINE
- SALES\_ORDER\_SOURCE
- SHIP\_DATE\_ACTUAL
- SHIP\_VIA
- TAX\_CODE
- UNIT\_SELLING\_PRICE
- UNIT\_STANDARD\_PRICE
- UOM\_CODE
- UOM\_NAME
- WAYBILL\_NUMBER

## FAQs for Define AutoInvoice

### Why did AutoInvoice reject transactions?

During AutoInvoice processing, if you have transaction lines that fail validation, the import process looks at the value of the **Invalid Line** field in the transaction source to determine what to do about the transaction.

If the value is Reject Invoice, then AutoInvoice rejects all of the transaction lines that make up one invoice according to the grouping rule, if any one of the transaction lines are invalid. For example, if a grouping rule specifies that three transaction lines should be created as one invoice and one of the transaction lines has an error, AutoInvoice rejects all three transaction lines and does not create an invoice.

However, if the value is Create Invoice, AutoInvoice rejects the one invalid transaction line and creates an invoice from the two remaining valid transaction lines.

### Why did AutoInvoice create transactions with duplicate transaction numbers?

During AutoInvoice processing, the import process validates that transaction and document numbers are unique after grouping has completed. In certain cases, AutoInvoice will create multiple invoices in the same group with the same

transaction or document number. Once grouping is completed, AutoInvoice checks for duplicate transaction and document numbers and reports any lines that fail validation.

For example, two lines are imported with the same transaction number, but they have different currencies. These lines are split into two separate invoices during grouping due to the different currencies. Once grouping has completed, both of the invoices will fail validation due to identical transaction numbers.

## What happens if AutoInvoice processes a transaction class that is not defined for a grouping rule?

If AutoInvoice uses grouping rules and it is processing a transaction class that is not defined for a grouping rule, then AutoInvoice only uses the mandatory transaction attributes to group transactions.

## When does a grouping rule need a line ordering rule?

Assign an AutoInvoice line ordering rule to an AutoInvoice grouping rule when you want to organize the transaction lines belonging to a transaction created by the grouping rule in a specific order. Use the **Order By Type** to specify whether to order the values belonging to a transaction attribute from least to greatest (Ascending) or greatest to least (Descending).

For example, if you are importing transactions from Oracle Fusion Distributed Order Orchestration, you can define a line ordering rule with the attribute SALES\_ORDER\_LINE to list the items on the invoice in the same order as they appear on the sales order.

Or, you can define a line ordering rule with the attribute AMOUNT and an **Order By Type** of Descending to ensure that the highest invoice line amounts are listed first on the transactions created by the grouping rule.

# Define Payment Terms

## Payment Terms: Explained

Use payment terms to identify due dates and discount dates on your customer transactions.

After you create payment terms, you can optionally assign them both to customer account and site profiles and to transaction types. The payment terms you assign are then assigned by default to transactions you create manually using the related customer account or site, or transaction type.

Considerations for payment terms include:

- Payment Terms and Customer Profiles
- Payment Terms and Discounts
- Split Payment Terms with Installments
- Prepayment Payment Terms

## Payment Terms and Customer Profiles

The setup of payment terms on customer account and site profiles influences the use and availability of payment terms on transactions you create manually.

When you create a transaction manually, Receivables looks for payment terms to assign to the transaction in this order:

1. Payment terms assigned to the site profile of the bill-to customer.
2. Payment terms assigned to the account profile of the bill-to customer.
3. Payment terms assigned to the transaction type.


You must enable the **Override terms** option on the customer account or site profile in order to change the payment terms assigned to the transaction. Enabling the **Override terms** option provides more flexibility in assigning payment terms to manual transactions.

If you do not enable the **Override terms** option on customer account and site profiles, then:

- You cannot change the payment terms assigned by default.
- If you select a bill-to customer, and no payment terms were assigned either to the account or site profiles of this customer or to the transaction type, then no payment terms are available for use on the transaction.

You can alternatively either select another bill-to customer or select payment terms before selecting a customer.

- If you select payment terms and then select a bill-to customer, and no payment terms were assigned to the account or site profiles of this customer, then you cannot change the payment terms originally assigned.

 **Tip:** If you intend to leave the **Override terms** option disabled on customer account and site profiles, then make sure that you assign payment terms to the transaction types that you intend to use for manual transactions.

## Payment Terms and Discounts

Define standard payment terms for your customers to specify the due date and discount date for their open items. Payment terms can include a discount percent for early payment, and you can assign multiple discounts to each line of your payment terms.

For example, the payment terms named 2% 10, Net 30 indicates that a customer is allowed a two percent discount if payment is received within 10 days. After 10 days, the entire balance is due within 30 days of the transaction date with no applicable discount.

Enable the **Allow discount on partial payments** option to let your customers take discounts for partial payments on items associated with payment terms. A partial payment is a payment that is less than the remaining amount due. If you do this, you must also ensure that the **Discount on partial payment** Receivables system option is enabled.

Use the **Discount Basis** field to determine what amount to use to calculate discounts for the payment terms. If the payment terms use installments, you can assign discount percentages to each installment.

Use the **Discount Basis Date** field to select the date to use to calculate discounts. The choices are:

- Receipt Date: Date the receipt is created.
- Receipt Application Date: Date the receipt is applied to the transaction.
- Deposit Date: Date the receipt is deposited into the remittance bank.

The discount is applied if the transaction is paid within the payment terms discount date. The formula is: **Transaction Date + Discount Due By Period >= Discount Basis Date (Receipt Date/Receipt Application Date/Deposit Date)**.

## Split Payment Terms with Installments

Create split payment terms for invoice installments that have different due dates. The payment terms determine the amount of each installment.

Use the **Installment Option** field to determine how to allocate the freight and tax charged to transactions. You can either distribute tax and freight charges across all installments, or allocate all freight and tax charges to the first installment.

Define the payment schedule for the split payment terms. The payment schedule determines when each installment is due, how much in each installment is due, and how much discount to offer in each installment.

## Prepayment Payment Terms

You can optionally define prepayment payment terms by enabling the **Prepayment** option. You assign prepayment payment terms to transactions to indicate which transactions require prepayment for goods and services.

Prepayment payment terms do not require the capture of funds in advance of invoicing or the delivery of prepaid goods or services. You must establish specific business practices at your enterprise if you want to capture these funds in advance.

## Installment Options and Amounts Due: Explained

Split payment terms derive different amounts due in each installment of the payment schedule, depending on the setting of the **Installment Option** field.

If the base amount is different from the relative amount, and you set the **Installment Option** field to **Allocate tax and freight**, Oracle Fusion Receivables prorates the base amount across the relative amounts of the payment schedule based upon the ratio you define. Receivables uses the following equation to determine the original amount due for each installment:

$$\text{Amount Due} = \text{Relative Amount} / \text{Base Amount} * \text{Invoice Amount}$$

If you set the **Installment Option** field to **Include tax and freight in first installment**, the base amount and the relative amounts that you specify for the payment schedule only indicate how the original line amounts of the invoices to which you assign these payment terms are distributed across different installments.

In this case, the original freight and tax amounts are included in the first installment, in addition to the line amount allocated by the ratio of the base amount and the relative amount that you specify for the first payment. Receivables uses the following equation to determine the original amount due for the first installment:

$$\text{Amount Due} = (\text{Relative Amount} / \text{Base Amount} * \text{Base Line Amount}) + \text{Base Freight Amount} + \text{Base Tax Amount}$$

## Payment Terms Discount Basis

The payment terms **Discount Basis** field determines on what basis Oracle Fusion Receivables calculates the discount amount.

### Discount Basis

#### Invoice Amount

Calculates the discount amount based on the sum of the tax, freight, and line amounts of transactions.

#### Lines Only

Calculates the discount amount based on only the line amounts of transactions.

#### Lines, Freight Items and Tax

Calculates the discount amount based on the amount of line items and their freight and tax amounts, but excludes freight and charges at the transaction header level.

#### Lines and Tax, not Freight Items and Tax

Calculates the discount amount based on the line items and their tax amounts, but excludes freight items and their tax lines.

## Calculating Discounts: Explained

Oracle Fusion Receivables uses different formulas to calculate discounts, depending on your setup, the payment terms on the transaction, and the type of payment received.

Receivables provides formulas for these discount events:

- Maximum Discount
- Earned Discounts and Partial Payments Allowed
- Unearned Discounts with Partial Payment Discounts Allowed
- Earned Discounts with Partial Payment Discounts Not Allowed
- Unearned Discounts and Partial Payments Not Allowed
- Discount on Lines Only

### Maximum Discount

Receivables uses the following formula to determine the maximum discount amount:

$$\text{Maximum Discount} = (\text{Amount Due Original}) * (\text{Highest Discount Percent} - \text{Discount Taken})$$

### Earned Discounts and Partial Payments Allowed

If the receipt amount is greater than the remaining amount due, less the discount, Receivables uses the following formula to determine the earned discount:

$$\text{Earned Discount} = \text{Amount Due Remaining} * \text{Discount Percent}$$

If the receipt amount is either the equal to or less than the remaining amount due, less the discount, Receivables uses the following formula to determine the earned discount:

$$\text{Earned Discount} = (\text{Receipt Amount} * \text{Discount Percent}) / (1 - \text{Discount Percent})$$

### Unearned Discounts with Partial Payment Discounts Allowed

Receivables uses the following formula to determine unearned discounts if partial payments are allowed:

$$\text{Unearned Discount} = \text{Maximum Discount} - \text{Earned Discount}$$

### Earned Discounts with Partial Payment Discounts Not Allowed

If the **Allow discount on partial payments** option on the payment terms is not enabled, Receivables only takes earned discounts if the receipt amount closes the installment.

Receivables uses the following formula to determine earned discounts, if partial payment discounts are not allowed and the receipt amount closes the installment:

$$\text{Earned Discount} = \text{Amount Due Original} * \text{Discount Percent}$$

### Unearned Discounts and Partial Payments Not Allowed

If the **Allow discount on partial payments** option on the payment terms is not enabled, Receivables only takes unearned discounts if the receipt amount closes the installment.

Receivables uses the following formula to determine unearned discounts, if partial payments are not allowed and the receipt amount closes the installment:

$$\text{Unearned Discount} = (\text{Amount Due Original}) * (\text{Maximum Discount Percent} - \text{Earned Discount})$$

## Discount on Lines Only

If the **Discount Basis** field on the payment terms is set to Lines Only, Receivables does not take discounts on receipt amounts applied to tax, freight, or late charges. Receivables uses the following formula to determine the discount amount:

$$\text{Line Percent} = \text{Discount Percent} * (\text{Sum of Lines} + \text{Sum of Line Adjustments} - \text{Sum of Line Credits}) / (\text{Amount Due Original} + \text{Sum of Adjustments} - \text{Sum of Credits})$$

Once you determine the discount line percent, use this as the discount percent in the formulas above.

## Deriving Discount Amounts: Example

This example illustrates how Receivables derives discount information based on the date of the receipt.

When you enter receipts manually, Oracle Fusion Receivables determines whether discounts are allowed based on the payment terms, discount grace days, Receivables system options, transaction date, and the payment terms discount basis date (receipt date, receipt application date, or receipt deposit date). If discounts are allowed, Receivables determines the amount of both earned and unearned discounts, as determined by the details of your setup.

### Scenario

Assume that you are using the following information:

- Unearned Discounts = Yes
- Payment Terms: 10/10, 5/15, Net 30
- Discount Basis Date: Receipt Date
- Discount Grace Days = 0
- Calculate Discount on Lines Only = No
- Allow Discount on Partial Payments = Yes

This table shows the discount details:

Percent	Date	On Lines Only	On Partial Payments
5	17-DEC-10	NO	YES
10	12-DEC-10	NO	YES

Assume these invoice details:

- Invoice #101
- Invoice Date: 02-DEC-10
- Due Date: 01-JAN-11
- Amount: \$1100

The following table displays the default discount amounts based on different receipt application dates. You can also see the amount of earned and unearned discounts that your customers can take.

Receipt Date	Receipt Amount	Default Discount Amount	Message Line	Earned Discount Allowed	Unearned Discount Allowed
From 02-DEC-10 to 12-DEC-10	\$990	\$110	Discount Earned = 110  Total = 110	\$110	None
After 17-DEC-10	\$990	0  To take the unearned discount, you must update the amount.	Discount Earned = 0  Total = 110	None	\$110
From 02-DEC-10 to 12-DEC-10	\$1000  \$10 of the receipt is left as Unapplied after the default discount.	\$110	Discount Earned = 110  Total = 110	\$110	None
From 13-DEC-10 to 17-DEC-10	\$1000  After the default discount of \$52.63, the receipt is fully applied. However, there is still a remaining balance of \$47.37 on the invoice.	\$52.63  To take the unearned discount, you must update the amount.	Discount Earned = 52.63  Total = 110	\$52.63	\$57.37
After 17-DEC-10	\$1000  Since there is no default discount, the receipt is fully applied. There is a remaining balance of \$100 on the invoice.	0  To take the unearned discount, you must update the amount.	Discount Earned = 0  Total = 110	None	\$110

## FAQs for Define Payment Terms

### What's the difference between balance forward payment terms and other payment terms?

Balance forward billing payment terms pass the balance forward billing cycle to the Create Balance Forward Bill program. The billing cycle determines when customer balance forward bills are generated.

Because balance forward bills can't be split across installments, all settings related to installments on balance forward billing payment terms are disabled. You can't change existing payment terms back and forth for use as both non-balance forward billing and balance forward billing payment terms.

## Define AutoAccounting

### AutoAccounting Account Types and Segment Values

Define AutoAccounting to specify how to determine the default general ledger accounts for transactions that you enter manually or import using AutoInvoice. You must define AutoAccounting before you can enter transactions in Oracle Fusion Receivables. When you enter or update transactions, you can override the default general ledger accounts that AutoAccounting creates.

#### Account Types

Define an AutoAccounting record for each type of account. You can then assign either a table name or constant value to each segment of the account.

##### **AutoInvoice Clearing**

The clearing account for imported transactions. Receivables uses the clearing account to hold any difference between the specified revenue amount and the selling price times the quantity for imported invoice lines. Receivables only uses the clearing account if you have enabled this option on the transaction source used for imported transactions.

##### **Bills Receivable**

The bills receivable account for your transactions. Receivables uses this account when you apply transactions to bills receivable.

##### **Factored Bills Receivable**

The factored bills receivable account for your bills receivable transactions.

##### **Freight**

The freight account for transactions.

##### **Receivable**

The receivable account for transactions.

##### **Remitted Bills Receivable**

The remitted bills receivable account for your bills receivable transactions.

##### **Revenue**

The revenue and late charges account for transactions.

##### **Tax**



The tax account for transactions.

### **Unbilled Receivable**

The unbilled receivable account for transactions. Receivables uses this account when the transaction uses the In Arrears invoicing rule. If the revenue scheduling rule on the transaction recognizes revenue before the invoicing rule bills it, Receivables uses this account.

### **Unearned Revenue**

The unearned revenue account for transactions. Receivables uses this account when a transaction uses the In Advance invoicing rule. If the revenue scheduling rule on the transaction recognizes revenue after the invoicing rule bills it, Receivables uses this account.

### **Unpaid Bills Receivable**

The unpaid bills receivable account for your bills receivable transactions.

## **Table Names**

Enter either the table name or constant value that you want Receivables to use to retrieve information for each accounting flexfield segment of a given account.

Enter a constant value instead of a table name if you want AutoAccounting to always use the same value for a given segment. You must ensure that you enter information that is valid for this segment. For example, if you defined your Company segment as a two-character segment with valid values ranging from 00 to 10, you must enter a two-character value within this range.

### **Bill-to Site**

Use the bill-to site of the transaction to determine this segment of revenue, freight, receivable, AutoInvoice clearing, tax, unbilled receivable, and unearned revenue accounts.

### **Drawee Site**

Use the drawee site table to determine this segment of your bills receivable, factored bills receivable, remitted bills receivable, and unpaid bills receivable account.

### **Remittance Banks**

Use the remittance banks table to determine this segment of your factored bills receivable and remitted bills receivable account.

### **Salesperson**

Use the salesperson table to determine this segment of revenue, freight, receivable, AutoInvoice clearing, tax, unbilled receivable, and unearned revenue accounts.

If you select this option for AutoInvoice clearing, tax, or unearned revenue accounts, Receivables uses the revenue account associated with the salesperson on the transaction. If you select this option for the unbilled receivable account, Receivables uses the receivable account associated with the salesperson on the transaction.

If the transaction has a line type of Line with an inventory item of Freight, AutoAccounting uses the revenue scheduling rules for the freight account rather than the revenue account.

## Standard Lines

Use the memo line or inventory item on the transaction to determine this segment of revenue, AutoInvoice clearing, freight, tax, unbilled receivable, and unearned revenue accounts.

If you select this option for AutoInvoice clearing, freight, tax, unbilled receivable or unearned revenue accounts, Receivables uses the revenue account associated to the memo line item or inventory item.

If the transaction has a line type of Line with an inventory item of Freight, AutoAccounting uses the revenue scheduling rules for the freight account rather than the revenue account.

## Tax

Use the tax account assigned to the tax rate codes on the transaction.

## Transaction Types

Use the transaction types table to determine this segment of revenue, freight, receivable, AutoInvoice clearing, tax, unbilled receivable, and unearned revenue accounts.

If the transaction has a line type of Line with an inventory item of Freight, AutoAccounting uses the revenue scheduling rules for the freight account rather than the revenue account.

# AutoAccounting Structure: Points to Consider

To implement AutoAccounting, you first define your AutoAccounting structure and then define information for each salesperson, transaction type, product, and tax rate code in order for AutoAccounting to properly create your default accounts.

You must define your AutoAccounting structure before you can enter invoices and credit memos, and you can only define one structure for each account type. During transaction creation, if AutoAccounting can't determine all of the accounting flexfield segments, it derives what it can and displays an incomplete accounting flexfield. You must provide any missing accounting flexfield information before you can complete a transaction.

There are these points to consider for each account type when creating your AutoAccounting structure:

- AutoInvoice Clearing Account
- Freight Account
- Receivable Account
- Revenue Account
- Tax Account
- Unbilled Receivable Account
- Unearned Revenue Account
- Available Information for Each Account

This table indicates the information that you can use to create each type of account. (Rec) and (Rev) indicate whether the account information is taken from the corresponding Receivables or Revenue accounting flexfield.

Information Source / AutoAccounting Type	Constant	Customer Bill-to Site	Salesperson	Transaction Type	Standard Item	Tax Rate Code
AutoInvoice Clearing	Yes	Yes	Yes (Rev)	Yes	Yes (Rev)	No
Freight	Yes	Yes	Yes	Yes	Yes (Rev)	No
Receivable	Yes	Yes	Yes	Yes	No	No
Revenue	Yes	Yes	Yes	Yes	Yes	No
Tax	Yes	Yes	Yes (Rev)	Yes	Yes (Rev)	Yes
Unbilled Receivable	Yes	Yes	Yes (Rec)	Yes	Yes (Rev)	No
Unearned Revenue	Yes	Yes	Yes (Rev)	Yes	Yes (Rev)	No

Notes on the table:

- If AutoAccounting for the AutoInvoice Clearing, Tax, Unbilled Receivable, or Unearned Revenue account is based on Standard Item, Receivables uses the segment from the standard item Revenue accounting flexfield.
- If AutoAccounting for the AutoInvoice Clearing, Tax, or Unearned Revenue account is based on Salesperson, Receivables uses the segment from the salesperson Revenue accounting flexfield.
- If AutoAccounting for Unbilled Receivable is based on salesperson, Receivables uses the segment from the salesperson Receivable accounting flexfield.
- If the AutoInvoice Clearing, Revenue, Tax, Unbilled Receivable, or Unearned Revenue account is based on salesperson, and there are multiple salespersons on the transaction, then Receivables creates separate distributions for each salesperson.

## AutoInvoice Clearing Account

During AutoInvoice processing, Receivables uses the AutoInvoice clearing account to store any differences between the specified revenue amount and the (price \* quantity) for imported invoice lines.

Receivables only uses the AutoInvoice clearing account if you enabled the **Create clearing** option on the transaction source assigned to imported transactions. However, you must define a clearing account whether or not you enable this option.

You can use constant value, customer bill-to site, salesperson, transaction type, and standard item for your AutoInvoice clearing account. If you select salesperson or standard item, Receivables uses the specified Revenue accounting flexfield.

## Freight Account

The freight account controls the account in general ledger to which you post freight amounts. You can use constant value, customer bill-to site, salesperson, transaction type, and standard item to specify your freight account.

If you choose standard item, Receivables uses the specified Revenue accounting flexfield. In addition, you can't import transactions with header-level freight through AutoInvoice.

If the transaction has a line type of LINE with an inventory item of freight, AutoAccounting uses the revenue scheduling rules for the freight account rather than the revenue account.

## Receivable Account

The receivable account controls the account in your general ledger to which you post receivable amounts. You can use transaction type, customer bill-to site, salesperson, and constant value to specify your receivable account.

## Revenue Account

The revenue account controls the account in your general ledger to which you post your revenue amounts. You can use transaction type, customer bill-to site, standard item, salesperson, and constant value to specify your revenue account.

## Tax Account

The tax account controls the account in your general ledger to which you post tax amounts. You can use tax rate codes, customer bill-to site, salesperson, transaction type, standard item, and constant value to specify your tax account.

If you select salesperson or standard item, Receivables uses the specified Revenue accounting flexfield.

## Unbilled Receivable Account

Receivables uses the unbilled receivable account for transactions that have invoicing and revenue scheduling rules. Whenever the revenue scheduling rule recognizes revenue on the transaction before the invoicing rule bills for the transaction, Receivables posts this amount to the unbilled receivable account.

You can select constant value, customer bill-to site, salesperson, transaction type, and standard item for your unbilled receivable account.

If you select standard item, Receivables uses the specified Revenue accounting flexfield. If you select salesperson, Receivables uses the salesperson Receivable accounting flexfield.

## Unearned Revenue Account

Receivables uses the unearned revenue account for transactions that have invoicing and revenue scheduling rules. Whenever the revenue scheduling rule recognizes revenue on the transaction after the invoicing rule bills for the transaction, Receivables posts this amount to the unearned revenue account.

You can select constant value, customer bill-to site, salesperson, transaction type, and standard item for your unearned revenue account.

If you select salesperson or standard item, Receivables uses the specified Revenue accounting flexfield.

# Using AutoAccounting to Derive Accounting Flexfield Segments: Example

This example illustrates how to derive default accounting flexfield segments in AutoAccounting.

Oracle Fusion Receivables uses AutoAccounting to derive the default accounting on transactions, and uses the predefined setup in Oracle Fusion Subledger Accounting so that the Create Receivables Accounting program accepts the default accounts that AutoAccounting derives without change. However, if you modify the Subledger Accounting setup to define custom accounting, then you can select a constant value for all accounting flexfield segments.

These two scenarios illustrate how Receivables uses the AutoAccounting structure you define to determine your accounting flexfield defaults.

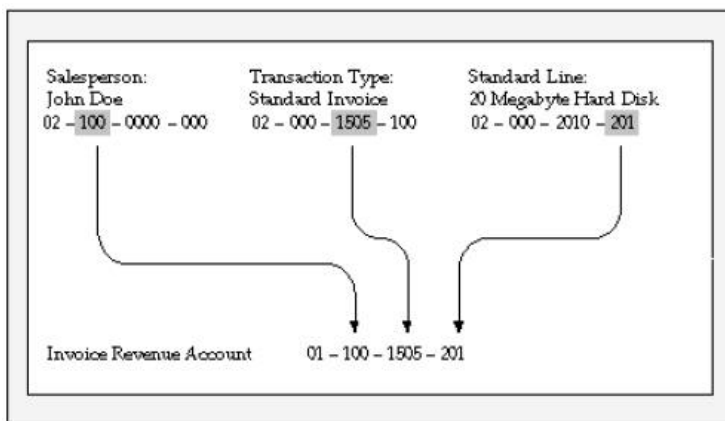
## Flexfield with Four Definitions

You want to define a four segment revenue flexfield: 00-000-0000-000 (Company-Cost Center-Account-Product). You can define AutoAccounting to derive defaults for each segment:

- Company, the first segment, is the constant 01.
- Cost Center, the second segment, derives from the salesperson (John Doe).
- Account, the third segment, derives from the transaction type (Standard Invoice).
- Product, the fourth segment, derives from the standard line (20 Megabyte Hard Disk).

Salesperson John Doe enters a one line Standard Type invoice for a 20 Megabyte Hard Drive.

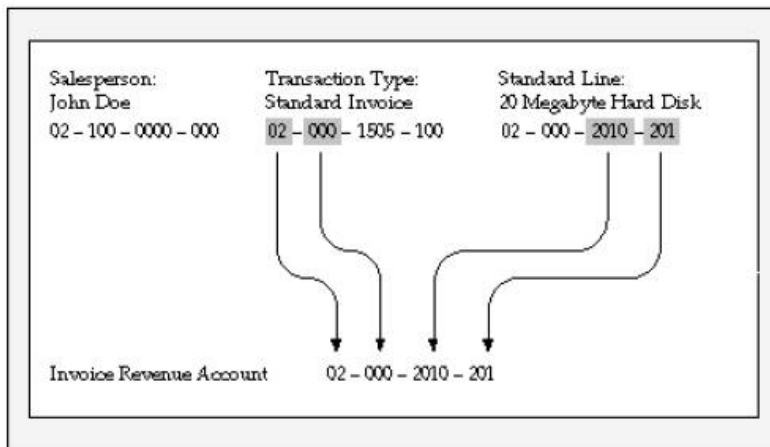
This graphic illustrates how AutoAccounting derives the revenue flexfield based on a separate definition for each segment:



## Flexfield with Two Definitions

You redefine the structure so that AutoAccounting only uses information from the transaction type (Standard Invoice) for segments 1 and 2, and standard line (consulting services) for segments 3 and 4.

This graphic illustrates how AutoAccounting derives the revenue flexfield based on the same definitions for segments 1 and 2 and segments 3 and 4:



## FAQs for Define AutoAccounting

### When does AutoAccounting create subledger accounting?

AutoAccounting derives the default accounting for each transaction accounting event in Oracle Fusion Receivables according to your setup. AutoAccounting assigns valid accounting flexfields to invoices and credit memos, and automatically generates valid accounting flexfields for all related accounts: freight, receivable, revenue, AutoInvoice clearing, tax, unbilled receivable, and unearned revenue.

When you submit the Create Receivables Accounting program, this creates the subledger accounting entries. Oracle Fusion Subledger Accounting transfers the final accounting to Oracle Fusion General Ledger.

You can optionally define your own accounting rules in Subledger Accounting to create accounting that meets your business requirements. If you customize the Subledger Accounting setup to create your own accounting, then Subledger Accounting overwrites the default accounts, or individual segments of accounts, that AutoAccounting originally derived during transaction entry.

## Define Transaction Types

### Recording Posted and Non-Posted Activities using Transaction Types: Critical Choices

Use the **Open Receivable** and **Post to GL** options on the transaction type to manage posted and non-posted activities on transactions.

If the **Open Receivable** option is enabled, Oracle Fusion Receivables updates your customer balances each time you create a complete debit memo, credit memo, chargeback, or on-account credit with this transaction type. Receivables also includes these transactions in the standard aging and collection processes.

If the **Post to GL** option is enabled, Receivables posts transactions with this transaction type to general ledger. If this option is not enabled, then no accounting is generated for transactions with this transaction type.

Considerations for defining transaction types include:

- Creating a Void Transaction Type
- Updating Customer Accounts and Aging
- Updating Accounting Only

## Creating a Void Transaction Type

You can void a debit memo, credit memo, on-account credit, or invoice by defining a Void transaction type. When you define a Void transaction type, set the **Open Receivable** and **Post to GL** options to No. Then, as long as there is no activity against the transaction, and it has not been posted to general ledger, you can make the transaction invalid by changing the transaction type to Void.

This activity is not included on the Review Customer Account Details page since the activity does not modify the customer balance.

## Updating Customer Accounts and Aging

If you set the **Open Receivable** option to Yes and **Post to GL** option to No, Receivables updates customer accounts with the transaction activity of transactions assigned this transaction type. There is no effect on accounting.

Use transaction types with these settings during your initial implementation, where the transaction amount is included in the general ledger beginning balance for the receivable account, but activity still needs to be aged and payment collected against it. All related activities against the transaction, such as credit memos, payments, and adjustments, are accounted as affecting the customer balance. You can review these activities on the Review Customer Account Details page.

## Updating Accounting Only

If you set the **Open Receivable** option to No and **Post to GL** option to Yes, Receivables updates accounting without any impact on the customer balance.

Use transaction types with these settings when you want to adjust accounting activity, such as when you rebill a customer in order to reclassify the general ledger account. A credit memo and invoice with the **Open Receivable** option set to No are created where the credit memo reverses the general ledger account of the original invoice, and the invoice creates accounting with the new general ledger account. This activity is transparent to the customer because the original invoice is used for the cash application when payment is received.

This activity is not included on the Review Customer Account Details page since the activity does not modify the customer balance.

### *Related Topics*

- [When do I credit and rebill a transaction?](#)

## Natural Application and Overapplication: Explained

The transaction type that you assign to a transaction defines the type of application that is permitted against the transaction balance. This definition is managed by the **Natural Application Only** and **Allow Overapplication** options on the transaction type.

Natural application only lets you apply a payment or credit to a transaction that brings the transaction balance close to or equal to zero. For example, if an invoice has a balance due of \$400, you can make applications against this transaction up to \$400 only. With natural application, you can only bring the balance to zero.

Overapplication lets you apply more than the balance due on a transaction. For example, if you apply a \$500 receipt to a \$400 invoice, this overapplies the invoice by \$100 and reverses the balance sign from positive to negative.

### Using Natural Application and Overapplication

Whether or not a transaction allows overapplication determines the actions that you can take on that transaction.

If a transaction that allows natural application only is paid in full, then in order to credit the transaction you must first unapply the receipt from the transaction before creating the credit.

If you want to use AutoInvoice to import credit memos against paid invoices and evaluate these credits for automatic receipt handling, then the transaction type of the paid invoices must allow natural application only. However, if the **Receipt Handling for Credits** option is not enabled on the transaction source of the transaction, AutoInvoice leaves the related credit memo in the interface tables until you unapply the receipt from the invoice.

#### *Related Topics*

- [Automated Receipt Handling for Credits: How It Works](#)

## Reference Accounts and Transaction Types: Points to Consider

Define the accounting for transaction types of class Invoice, Chargeback, Credit Memo, and Debit Memo. Oracle Fusion Receivables uses this information along with your AutoAccounting definition to determine the accounts to use for transactions with the applicable transaction type.

There are points to consider for each reference account on a transaction type:

- Revenue
- Freight
- Receivable
- AutoInvoice Clearing
- Tax
- Unbilled Receivable
- Unearned Revenue



## Revenue

Enter a revenue account, unless the transaction type doesn't allow freight.

If the **Invoice Accounting Used for Credit Memos** profile option is set to No, then a revenue account is not required for Credit Memo transaction types.

## Freight

Enter a freight account, unless the transaction type doesn't allow freight.

## Receivable

Enter a receivable account for all transaction types.

If the **Post To GL** option on the transaction type is enabled, Receivables creates a receivable transaction record using this account in order to transfer accounting to general ledger and create a journal entry.

For Chargeback transaction types, enter the Receivable Chargeback account. The offset to the Receivable account on the original debit transaction is generated by the chargeback adjustment.

If the **Invoice Accounting Used for Credit Memos** profile option is set to No, then a receivable account is not required for Credit Memo transaction types.

## AutoInvoice Clearing

If this is an Invoice or Debit Memo transaction type, enter an AutoInvoice clearing account. Receivables uses this account to hold any difference between the revenue amount specified for the revenue account and the selling price times the quantity for imported invoice lines.

Receivables only uses the AutoInvoice clearing account for imported transactions that have a transaction source with the **Create clearing** option enabled. If the **Create clearing** option is not enabled, then AutoInvoice requires that the revenue amount on the invoice be equal to the selling price times the quantity, or the invoice is rejected.

## Tax

If this is an Invoice, Credit Memo, or Debit Memo transaction type, enter a tax account.

## Unbilled Receivable

If this is an Invoice or Credit Memo transaction type, enter an unbilled receivable account. This account is for transactions that use the In Arrears invoicing rule.

## Unearned Revenue

If this is an Invoice or Credit Memo transaction type, enter an unearned revenue account. This account is for transactions that use the In Advance invoicing rule.

# FAQs for Define Transaction Types

## How can I arrange the creation of transaction types?

Define your transaction types in the following order:

- Credit memo transaction types
- Invoice transaction types

- Debit memo transaction types
- Chargeback transaction types

If applicable, define the transaction types that you want to add to your transaction sources before defining transaction sources.

If you are using late charges, define a transaction type with a class of Debit Memo for debit memos, and a transaction type with a class of Invoice for interest invoices. Specify the receivable and revenue accounts for these transaction types. Oracle Fusion Receivables uses these accounts instead of AutoAccounting when generating late charges.

## How can I use transaction types to review and update customer balances?

Use the **Open Receivable** option on the transaction type to implement an approval cycle for any temporary or preliminary transactions.

For example, if you have particularly sensitive debit memos, credit memos, on-account credits, chargebacks, or invoices that you want to review after creation, you can define a transaction type called Preliminary with **Open Receivable** set to No and assign it to the applicable transactions. This transaction type does not update your customer balances.

Once you review and approve these transactions, you can define a transaction type called Final with **Open Receivable** set to Yes and assign it to the same transactions. This will now update your customer balances on these transactions.

# Define Transaction Sources

## Managing Transaction Numbering: Points to Consider

Use the various options on the transaction source assigned to a transaction to manage your transaction numbering requirements.

There are these points to consider when defining transaction numbering for transactions assigned to specific transaction sources:

- Defining Document Sequences
- Using Automatic Transaction Numbering
- Copying Document Numbers to Transaction Numbers
- Allowing Duplicate Transaction Numbers
- Using the Credit Memo Transaction Source

## Defining Document Sequences

If necessary, define document sequences to assign unique numbers to each transaction, in addition to the transaction number automatically assigned by Oracle Fusion Receivables.

Ensure that the necessary setups for document sequences are completed, according to your requirements.

## Using Automatic Transaction Numbering

To automatically number new transactions you create using a transaction source, enable the **Automatic transaction numbering** option and enter a number in the **Last Number** field.

For example, to start numbering transactions with 1000, enter a last number of 999. Receivables automatically updates the **Last Number** fields on transaction sources, so you can review the transaction source later to see the last transaction number that was generated.

 **Note:** The last transaction number on the transaction source is an approximation only, due to caching.

You can use automatic transaction numbering with both Imported and Manual transaction sources.

## Copying Document Number to Transaction Number

If you are using document sequences and you want to use the same value for both the document number and the transaction number for transactions assigned to a transaction source, enable the **Copy document number to transaction number** option.

If you are using Gapless document sequences, you should enable this option if you require gapless transaction numbering. This ensures that transaction numbers are generated sequentially and that there are no missing numbers.

## Allowing Duplicate Transaction Numbers

Enable the **Allow duplicate transaction numbers** option to allow duplicate transaction numbers within a transaction source.

You cannot use this option with automatic transaction numbering.

## Using the Credit Memo Transaction Source

Assign a credit memo transaction source to an invoice transaction source, if you want to number credit memos differently from the invoices that they credit.

### Related Topics

- [Document Sequences: Explained](#)

## Sales Credits on Imported Transactions: Explained

During AutoInvoice processing, whether you must provide sales credit information on imported transaction lines depends on the settings of the **Allow sales credits** option on the transaction source and the **Require salesperson** Receivables system option.


These are the requirements for passing sales credit information on imported transaction lines:

- If the **Require salesperson** system option and the **Allow sales credits** option on the transaction source are both enabled, you must provide sales credit information.
- If the **Require salesperson** system option is not enabled and the **Allow sales credits** option on the transaction source is enabled, you can provide sales credit information, but it is not required.
- If the **Require salesperson** system option is enabled and the **Allow sales credits** option on the transaction source is not enabled, you must provide sales credit information.
- If neither the **Require salesperson** system option nor the **Allow sales credits** option on the transaction source are enabled, you cannot provide sales credit information. AutoInvoice ignores any values that you pass.

## Validating Imported Transactions: How It Works

Use the AutoInvoice Options and Import Information sections of an Imported transaction source to define how AutoInvoice validates imported transaction lines.


You don't have to pass values for all of the fields that are referenced in the transaction source. If you don't want AutoInvoice to pass certain data, then where available you can set the related option to **None**.

 **Note:** Even if you set a transaction source data option to **None** in order to avoid importing this information into the interface tables, AutoInvoice can still validate and reject transaction lines with invalid data.

### Settings That Affect the Validation of Imported Transactions

These settings affect the validation of imported transactions:

- **Invalid Line** field: Indicate how AutoInvoice handles imported transactions with invalid lines by selecting either Reject Invoice or Create Invoice.
  - If you select Reject Invoice, AutoInvoice doesn't import this transaction or any of its lines into the interface tables.
  - If you select Create Invoice, AutoInvoice creates a transaction with valid lines only. For example, you import an invoice with three invoice lines and one of the lines is invalid. AutoInvoice creates the invoice with the two valid lines only and rejects the invalid line. You can use the Edit Transaction page to add the rejected line.
- **Accounting Date in a Closed Period** field: Indicate how AutoInvoice handles imported transactions that have lines in the interface lines table that are in a closed accounting period.
  - Select Adjust to have AutoInvoice automatically adjust the accounting dates to the first accounting date of the next open or future enterable period.
  - Select Reject to reject these transaction lines.
- In the Import Information sections, where applicable select Number, Value, Segment, or ID for each option to indicate how AutoInvoice validates information:
  - Select Number to import a record into the interface tables using its assigned number.
  - Select Value to import a record into the interface tables using its actual name.

 **Note:** Use Value if you intend to use the transaction source to import data from a non-Oracle system.

- Select Segment to use the flexfield segment.
- Select ID to use the internal identifier of the record.
- Select Amount or Percent to indicate how AutoInvoice validates Sales Credits and Revenue Account Allocations on transaction lines.

### How Imported Transactions Are Validated

AutoInvoice validates imported transactions based on the settings of the assigned Imported transaction source. Transactions that fail validation appear in the Import AutoInvoice Validation report.

AutoInvoice ensures that certain column values agree with each other. These values can be within an interface table or multiple interface tables. For example, if the transaction source indicates that a revenue scheduling rule can't be used, AutoInvoice ignores any values passed for invoicing rule, revenue scheduling rule, and revenue scheduling rule duration.

AutoInvoice performs these validations on transaction lines with revenue scheduling rules:

- Requires that these transactions also include an invoicing rule, if you import transactions that use revenue scheduling rules.
- Rejects lines, if the revenue scheduling rule has overlapping periods.
- Rejects lines, if the designated accounting periods don't exist for the duration of the revenue scheduling rule.

## Setting Up for CPQ Cloud and Receivables Processing: Procedure

Set up Oracle CPQ Cloud and Oracle Fusion Receivables to enable the integrated quote-to-cash cloud service. This integrated service lets you manage the entire process of quote, pricing, order entry, credit checking and invoicing.

To set up the CPQ Cloud/Receivables integration service, perform one or more of these tasks:

- Set up CPQ Cloud and CPQ Cloud data.
- Set up Receivables transaction sources.
- Set Up for multiple CPQ Cloud Instances.

### Set Up CPQ Cloud

Perform the necessary setups in CPQ Cloud using the CPQ Administration pages.

Create a data table definition for each of the integration services used by the CPQ Cloud/Receivables quote-to-cash cloud service:

- Credit Check Service: Service that verifies customer creditworthiness.
- Receivables Invoice Service: Service that transfers order lines from CPQ Cloud to the AutoInvoice interface tables in Receivables.

Each data table definition includes these attributes:

- Endpoint URL: The URL to use to contact the service.
- Username: The valid user name recognized by the service.
- Password: The accompanying password for the valid user.

### Set Up the CPQ Cloud Transaction Source

Receivables provides an Imported transaction source under the Common reference set to use to configure the CPQ Cloud/Receivables integrated service.

Use this transaction source to:

- Manage the transfer of transaction information from CPQ Cloud to the AutoInvoice interface tables.
- Manage the Call Back service to send Receivables transaction information to CPQ Cloud after the transactions are created.

To set up the transaction source for CPQ Cloud:

1. Navigate to the Edit Transaction Source page.

2. Open the CPQ Cloud transaction source.
3. If applicable, enter the legal entity.
4. If applicable, update the **From Date** and **To Date** fields.
5. In the Oracle CPQ Cloud Integration section, check the **Enabled** option.
6. Click the **Define** button to open the Define Endpoint Policy window.
7. In the **URL** field, enter the URL to use to contact the Call Back service.
8. In the **Security Policy** field, enter `oracle/wss_username_token_over_ssl_client_policy`.
9. In the **User Name** field, enter the user name to use for the Call Back service.
10. In the **Password** field, enter the accompanying password for the valid user.
11. Click the **Save and Close** button to exit the Define Endpoint Policy window.
12. Save and close the Edit Transaction Source page.

## Set Up for Multiple CPQ Cloud Instances

If you plan to have more than one CPQ Cloud instance sending data to Receivables, you must represent each CPQ instance as a business unit and define a transaction source for each instance.

To set up for multiple CPQ Cloud instances:

1. Define one reference set for each CPQ Cloud instance.
2. Associate this reference set to the business unit for the reference object transaction source.
3. Create a transaction source for each reference set:
  - a. Open the Create Transaction Source page.
  - b. In the **Transaction Source Set** field, select the applicable reference set for the business unit.
  - c. If applicable, enter the legal entity.
  - d. In the **Name** and **Description** fields, enter a name and description for this transaction source.
  - e. In the **Type** field, select Imported.
  - f. If applicable, update the **From Date** and **To Date** fields.
  - g. Complete the remaining fields in the transaction source with the same settings as the predefined transaction source.
  - h. In the Oracle CPQ Cloud Integration section, check the **Enabled** option and complete the endpoint policy.
  - i. Save and close the Create Transaction Source page.

## FAQs for Define Transaction Sources

### What do I create before creating transaction sources?

You may want to create certain records before creating your transaction sources.

You can optionally create these objects for all transaction sources:

- Transaction types: Define the transaction types that you want to appear by default on transactions assigned to your transaction sources.
- Invoice transaction flexfield: Define the reference information that you want to capture in the invoice transaction flexfield and display on imported transactions, such as a purchase order number.
- Credit memo transaction source: Define a transaction source for credit memos before you define a transaction source for invoices. Use this transaction source to number the credit memos created against invoices differently from the invoices they are crediting.

You can optionally create these objects for Imported transaction sources:

- AutolInvoice grouping rule: Define the grouping rule to appear by default on imported transaction lines.
- AutolInvoice clearing account: Define an AutolInvoice clearing account, if you intend to enable the **Create clearing** option. AutolInvoice puts any difference between the revenue amount and the selling price times the quantity for a transaction into this account.

## How can I manage credit memos with transaction sources?

Special conditions may apply to the creation of transaction sources for credit memos.

Review these considerations for transaction sources assigned to credit memos:

- Define Manual transaction sources for credit memos created by the credit memo request approval process.
- Enable the **Copy transaction information flexfield to credit memo** option on Manual transaction sources used for credit memos, to copy the invoice transaction flexfield reference information to the credit memo that is crediting the invoice.
- Define and assign transaction sources for credit memos to transaction sources for invoices, if you want to number the credit memos created against invoices differently from the invoices they are crediting.

## What happens if I don't enter an AutolInvoice grouping rule?

Assign the AutolInvoice grouping rule to Imported transaction sources that AutolInvoice uses to group imported transaction lines.

If you don't assign a grouping rule to an Imported transaction source, AutolInvoice uses the following hierarchy to determine which rule to use:

1. Grouping rule assigned to the transaction source of the transaction line.
2. Grouping rule assigned to the bill-to customer site profile of the transaction line.
3. Grouping rule assigned to the bill-to customer profile of the transaction line.
4. Grouping rule assigned to Receivables system options.

## What happens if I don't create a clearing account?

If you don't use an AutolInvoice clearing account and enable the **Create clearing** option on the transaction source, AutolInvoice requires that the revenue amount be equal to the selling price times the quantity for all of the transactions it processes. AutolInvoice rejects any transaction line that does not meet this requirement.

# Define Memo Lines

## Revenue Accounts and Memo Lines: Explained

You can optionally associate a revenue account with a memo line.

If AutoAccounting depends on memo line, Oracle Fusion Receivables uses the revenue account segment values defined for the memo line, in combination with the rest of your AutoAccounting structure, to determine the default revenue, freight, AutolInvoice clearing, tax, unbilled receivable, unearned revenue, and receivable accounts for invoices that include the memo line.

When you create a debit memo or on-account credit memo with memo lines, Receivables uses the revenue account from the original receivable item as the credit account. However, when you create debit memo reversals or chargebacks, Receivables uses instead the revenue flexfield from the original receivable item as the credit account.

### *Related Topics*

- [Debit Memo Reversals: Points to Consider](#)

## FAQs for Define Memo Lines

### [When do I use memo lines?](#)

Use memo lines on your transactions when the item is not an inventory item. For example, you can define a memo line called Consulting Services to identify charges for consulting activities. You can assign memo lines to debit memos, on-account credits, debit memo reversals, chargebacks, and invoices.

### [How can I use tax memo lines?](#)

You can use tax memo lines on transactions only if your tax definition lets you enter manual tax lines on transactions. After you enter a tax memo line on a transaction, you can specify the amount of tax to assign to the transaction line.

## Define Balance Forward Billing Cycles

### Account and Site Balance Forward Billing: Explained

You can enable and maintain balance forward billing details at the customer account level and customer site level. The rules governing the interaction between the account and account sites determine what sites are included in balance forward bills.

### [Rules for Account and Site Balance Forward Billing](#)

These rules apply to balance forward billing at the account level and site level:

- Account level customer profile values become the default values for the related customer sites when you create site profiles.
- If balance forward billing is enabled at the account level, you still need to enable balance forward billing on the related site profiles if you want to include these sites for balance forward billing.
- Once you create a site profile and define balance forward billing details, the site profile no longer references the account profile. This lets you define different balance forward billing details for a particular site.
- Balance forward billing definitions at the site level take precedence over balance forward billing definitions at the account level.

### *Related Topics*

- [Balance Forward Billing: Explained](#)

## Setting Up Balance Forward Billing: Worked Example

This example demonstrates how to set up balance forward billing for the customer Business World. The example includes the setup of a balance forward billing cycle, balance forward billing payment terms, and the Business World customer profile.



This table summarizes key decisions for this scenario.

Decisions to Consider	In This Example
What billing cycle is used?	Monthly
What day of the month is used on the payment terms?	15th
How are balance forward bills consolidated?	Account level
Can the customer exclude transactions from the balance forward bill?	Yes

## Summary of the Tasks

To generate balance forward bills for a customer:

1. Define a balance forward billing cycle.
2. Define balance forward billing payment terms.
3. Set up the customer profile class for balance forward billing.
4. Assign the profile class to the Business World customer account.

### Define a Balance Forward Billing Cycle

Define the balance forward billing cycle to use for this customer:

1. Open the Balance Forward Billing Cycle window.
2. Complete the fields, as shown in this table:

Field	Value
<b>Name</b>	Name that identifies this balance forward billing cycle for this customer.
<b>Frequency</b>	Monthly
<b>Repeat Every</b>	1
<b>Day of Month</b>	Last Day of Month

### Define Balance Forward Billing Payment Terms

Define the balance forward billing payment terms to use for this customer:

1. Open the Create Payment Terms page.
2. Complete the fields, as shown in this table:

Field	Value
<b>Name</b>	Name that identifies these payment terms for this customer.


Field	Value
<b>Billing Cycle</b>	The billing cycle you defined in the previous step.
<b>Due By Day of Month</b>	15

## Setting Up the Customer Profile Class for Balance Forward Billing

Set up the customer profile for Business World for balance forward billing:

1. Open the Create Receivables Customer Profile Class page.
2. Complete the general required fields for the profile class.
3. Complete the fields on the Profile Class tab, as shown in this table:

Field	Value
<b>Balance Forward Billing Enable</b>	Enable this option.
<b>Bill Level</b>	Account
<b>Bill Type</b>	Detail
<b>Payment Terms</b>	The balance forward billing payment terms you defined in the previous step.
<b>Override terms</b>	Enable this option.

 **Note:** The **Override terms** option makes available to transactions both non-balance forward billing payment terms and the one balance forward billing payment terms you defined in the previous step. This lets you exclude individual transactions from balance forward billing by assigning the transaction non-balance forward payment terms.

## Assign the Profile Class to the Business World Customer Account

Assign the profile class to the Business World account:

1. Open the Edit Account page for Business World.
2. Click the Profile tab.
3. Open the Insert Account Profile page and update the Business World customer account with the profile you defined in the previous step.
4. Make sure that balance forward billing is enabled at the site profile level for all sites belonging to the customer account that you want to include in balance forward billing.

### Related Topics

- [How can I change a customer's billing cycle?](#)
- [Transactions for Balance Forward Billing: How They Are Selected](#)

## FAQs for Define Balance Forward Billing Cycles

### What's a balance forward billing cycle?

The balance forward billing cycle determines the dates in the year to generate balance forward bills, and the transactions that are included in balance forward bills. You assign balance forward billing cycles to payment terms.

When you run the Create Balance Forward Bill program, you must select a billing cycle. You can also select specific customers, customer sites, and payment terms. A run of the Create Balance Forward Bill program includes all transactions not included on a previous balance forward bill that are assigned the payment terms with the selected billing cycle, or a subset of these payment terms as determined by the other program parameters.

## FAQs for Salesperson Account References

### What are salesperson reference accounts?

Assign general ledger accounts to your salespersons. When AutoAccounting depends on salesperson, Oracle Fusion Receivables uses the account references that you define here to derive the accounts to use on transactions that are assigned a particular salesperson.

## FAQs for Remit-to Addresses

### How can I use remit-to addresses?

The remit-to address lets your customers know where to send payment for their open debit items. After you create a remit-to address, you can assign it to the bill-to addresses of the customers and customer sites that you designate by country and, if applicable, by region and postal code range.

If the **Print remit-to address** system option is enabled, Oracle Fusion Receivables prints the remit-to address on the related dunning letters and statements.

### How does AutoInvoice validate remit-to addresses?

During the import process, AutoInvoice rejects all invoices for which it cannot determine a remit-to address. In order for AutoInvoice to import an invoice, you must either define a remit-to address for the geographical location of each applicable bill-to site or define a remit-to address to use as default for one or more locations.

### How can I define a default remit-to address?

Create or select a remit-to address, then open the Receipts from Criteria dialog box. Select the country that you want to assign to this remit-to address.

If you only select a country, then all customer bill-to sites in this country are assigned this remit-to address.

If you want to assign this remit-to address to specific locations within the country, you can optionally select a state or region within the country, and a range of postal codes.

## Why did the country appear?

When you create a remit-to address, a country appears by default if one was defined in Receivables system options. You can change the default to the applicable country of the remit-to address.

## Why do I verify the address?

If you have Oracle Fusion Trading Community Data Quality installed, you can expose a **Verify Address** button on the Create and Edit Remit-to Address pages for applicable countries.

After you enter a remit-to address, use the **Verify Address** button to confirm that the address is in the Oracle Fusion Trading Community Model registry. If it is not, Oracle Fusion Receivables either presents alternative addresses or lets you optionally add the address you entered to the registry.

## 4 Define Customer Payments

### Managing Shared Services in Receivables: Procedures

Use service provider relationships to create a shared service model in Oracle Fusion Receivables to centralize the processing of customer payments.

In the shared service model, you define one servicing business unit for customer payments. You then create a service provider relationship between this servicing business unit and one or more client billing business units.

 **Note:** All business units must belong to the same ledger.

Once you define this relationship, the servicing business unit can receive and process payments for all of its client billing business units. The shared service model for centralized customer payment processing supports all receipt creation methods in Receivables:

- Manual receipts.
- Receipts uploaded using a spreadsheet.
- Lockbox receipts.
- Automatic receipts.

Receivables provides additional functionality for cross-business unit receipt processing, to facilitate the processing of customer payments:

- Allow a billing business unit outside the service provider relationship to have payments for its transactions processed by any other business unit in the same ledger.
- Allow an individual receipt to be applied to transactions of any business unit in the same ledger.

Perform these procedures to manage shared services in Receivables:

- Define Customer Payments Service Providers
- Define Receivables System Options for Cross-Business Unit Receipt Processing
- Enable a Receipt for Cross-Business Unit Processing

### Define Customer Payments Service Providers

Perform the steps below to set up a servicing business unit and client billing business units.

1. Create a servicing business unit and client business units.
2. Assign the appropriate business functions to each business unit:
  - a. Open the Assign Business Functions page for the servicing business unit.
  - b. Assign business functions to the servicing business unit. At a minimum, you must assign the servicing business unit the **Customer Payments** business function.
  - c. Save your work.
  - d. Open the Assign Business Functions page for the first client business unit.
  - e. Assign business functions to the client business unit. At a minimum, you must assign the client business unit the **Billing and Revenue Management** business function.

- f. Save your work.
        - g. Repeat steps 4 to 6 for each client business unit.
3. Assign the servicing business unit to each client business unit:
  - a. From the Manage Service Provider Relationships task, open the Manage Service Providers page for the first client business unit.
  - b. In the Customer Payments Service Providers section, click the **Add** icon.
  - c. In the Search and Select window, select the servicing business unit.
  - d. Save your work.
  - e. Repeat steps 1 to 4 for each client business unit.

## Define Receivables System Options for Cross-Business Unit Receipt Processing

Perform the steps below to allow a billing business unit outside the service provider relationship to have payments for its transactions processed by any other business unit in the same ledger.

1. From the Manage Receivables System Options task, open the Edit System Options page for the business unit that you want.
2. Navigate to the Billing and Revenue tabbed region.
3. Enable the **Allow any business unit to process receipts** option.
4. Save your work.

This business unit can now have payments for its transactions processed by any other business unit in the same ledger.

## Enable a Receipt for Cross-Business Unit Processing

Perform the steps below to apply an individual receipt to one or more transactions belonging to any business unit in the same ledger.

1. Use the Create Receipt page to create a new receipt or the Edit Receipt page to open an unapplied receipt.
2. In the Application tabbed section, click the **Add Open Receivables** button.
3. In the Add Open Receivables window, enable the **Include transactions from all business units** option.
4. Search for and select the transactions that you want to apply to the receipt. You can select transactions from any business unit in the same ledger.
5. Click the **Add** button and then the **Done** button.
6. If necessary, update the figure in the **Applied Amount** field for each applicable transaction.
7. Click the **Save** button to apply the receipt to the selected transactions.

## Define Application Rule Sets

### Application Rules: Explained

When you apply a receipt or credit memo to a transaction, the application rule set determines how Oracle Fusion Receivables reduces the open balance of the line, tax, freight, and late charges amounts on a transaction.

Receivables uses the application rule set assigned to the transaction type to process payment applications. If no application rule set is assigned to the transaction type, then Receivables uses the Receivables system options application rule set.

You can arrange the order of the line types and application rules in an application rule set according to your needs. Each line type must appear in an application rule set, and appear only once. The Overapplication rule is always last in the sequence.

## Line First - Tax After Rule

The Line First - Tax After rule first applies the payment to open line amounts, and then applies the remaining amount to the associated tax.

If the payment is greater than the sum of the open line and tax amounts, Receivables attempts to close each remaining open item by applying the remaining amount in the following order, stopping when the payment has been fully applied:

1. Freight
2. Late charges

After the payment is fully applied, if there is still a remaining receipt amount this amount is managed by the Overapplication rule.

## Line and Tax Prorate

The Line and Tax Prorate rule applies a proportionate amount of the payment to open line and tax amounts for each line.

If the payment is greater than the sum of the open line and tax amounts, Receivables attempts to close each remaining open item by applying the remaining amount in the following order, stopping when the payment has been fully applied:

1. Freight
2. Late charges

After the payment is fully applied, if there is still a remaining receipt amount this amount is managed by the Overapplication rule.

## Prorate All

The Prorate All rule applies a proportionate amount of the payment to each open amount associated with a debit item (that is, any line, tax, freight, and late charge amounts for this item).

Receivables uses the following formula to determine the applied amount:


$$\text{Applied Amount} = \frac{\text{open application line type amount}}{\text{sum of application line types in rule details}} * \text{Receipt Amount}$$

After the payment is fully applied, if there is still a remaining receipt amount this amount is managed by the Overapplication rule.

## Overapplication Rule

The Overapplication rule is always the last rule in an application rule set. This rule manages any remaining receipt amount after the open balance of all transaction lines has been reduced to zero.

If the transaction type for the debit item allows overapplication, Receivables applies the remaining amount to transaction lines, making the balance due negative. If the transaction type for the debit item does not allow overapplication, you can either place the remaining amount on-account or leave it unapplied.

 **Note:** Lockbox uses the AutoCash rule set rather than the application rule set to determine how to apply a remaining amount.

### Related Topics

- [Natural Application and Overapplication: Explained](#)
- [Using AutoCash Rules: Examples](#)

## Using Application Rules: Examples

These examples show how Oracle Fusion Receivables uses each application rule in an application rule set to apply a payment to a transaction.

Invoice 123 contains these details:

Field	Value
Line	\$1000
Tax	\$140
Freight	\$200
Total	\$1340

Your customer remits a partial payment of \$1040 for this invoice. This table shows how Receivables applies the payment using each of the three application rules:

Application Rule	Total Amount Applied	Line Amount Applied	Tax Amount Applied	Freight Amount Applied
Line First - Tax After	1040	1000	40	0
Line and Tax Prorate	1040	912.28	127.72	0
Prorate All	1040	776.12	108.66	155.22

This table shows the calculations used by each application rule:

Application Rule	Calculations
Line First - Tax After	<ol style="list-style-type: none"> <li>1. Apply payment to open line amount.</li> <li>2. Apply any remaining amount to tax.</li> </ol>
Line and Tax Prorate	<ol style="list-style-type: none"> <li>1. <math>(1040/1140) * 1000 = 912.28</math> (Receipt Amount / Total Line and Tax) * Line Amount = Line Amount Applied</li> <li>2. <math>(1040/1140) * 140 = 127.72</math> (Receipt Amount / Total Line and Tax) * Open Tax Amount = Tax Amount Applied</li> </ol>
Prorate All	<ol style="list-style-type: none"> <li>1. <math>(1040/1340) * 1000 = 776.12</math> (Receipt Amount / Invoice Total) * Open Line Amount = Line Amount Applied</li> </ol>



Application Rule	Calculations
	<ol style="list-style-type: none"> <li>(1040/1340) x 140 = 108.66 (Receipt Amount / Invoice Total) * Open Tax Amount = Tax Amount Applied</li> <li>(1040/1340) x 200 = 155.22 (Receipt Amount / Invoice Total) x Open Freight Amount = Freight Amount Applied</li> </ol>

## Line First - Tax After

The Line First - Tax After rule first applies the payment to the line amount, reducing the balance due to zero. Receivables then applies the remaining amount (\$40) to the tax charges, reducing the open tax amount to \$100. Since the payment is not enough to close these items, the freight balance is not affected.

This table compares each line type before and after you apply an amount using the Line First - Tax After rule:

Transaction Amount	Remaining Amount	Line Items	Line Items Remaining	Tax	Tax Remaining	Freight	Freight Remaining
\$1340	\$300	\$1000	\$0	\$140	\$100	\$200	\$200

## Line and Tax Prorate

The Line and Tax Prorate rule applies a proportionate amount to the open line and tax charges. Since the amount applied is not enough to close these items, the freight balance is not affected.

This table compares each line type before and after you apply an amount using the Line and Tax Prorate rule:

Transaction Amount	Remaining Amount	Line Items	Line Items Remaining	Tax	Tax Remaining	Freight	Freight Remaining
\$1340	\$300	\$1000	\$87.72	\$140	\$12.28	\$200	\$200

This table shows the calculations used to arrive at the proportionate amounts:

Item	Calculations
Line Items	$1000 - 912.28 = 87.72$ Amount Line Items - Line Amount Applied = Open Line Amount
Tax	$140 - 127.72 = 12.28$ Tax Original - Tax Amount Applied = Open Tax Amount

## Prorate All

The Prorate All rule applies a proportionate amount of the receipt to the line, tax, and freight for this transaction.

This table compares each line type before and after you apply an amount using the Prorate All rule:

Transaction Amount	Remaining Amount	Line Items	Line Items Remaining	Tax	Tax Remaining	Freight	Freight Remaining
\$1340	\$300	\$1000	\$223.88	\$140	\$31.34	\$200	\$44.78

This table shows the calculations used to arrive at the proportionate amounts:

Item	Calculations
Line Items	$1000 - 776.12 = 223.88$ Amount Line Items - Line Amount Applied = Open Line Amount
Tax	$140 - 108.66 = 31.34$ Tax Original - Tax Amount Applied = Open Tax Amount
Freight	$200 - 155.22 = 44.78$ Freight Original - Freight Amount Applied = Open Freight Amount

## Using Application Rules with Transactions with Mixed Sign Balances: Example

This example shows how Oracle Fusion Receivables uses the Prorate All application rule to apply a payment to a transaction that has mixed sign balances. In a transaction with mixed sign balances, the charges that make up the transaction have a combination of positive and negative signs.

When you apply a payment to a transaction that has mixed sign balances, Receivables applies the payment only to those amounts that have the same sign as the payment. For example, if the payment is for a positive amount (that is, it is not a credit memo), Receivables only reduces the charges that have a positive balance. Any negative balances are not affected.

As with transactions having a same sign balance, Receivables applies any remaining amounts according to the Overapplication rule assigned to the application rule set.

### Prorate All with Mixed Sign Balances

Invoice 101 contains these details:

Field	Value
Line	<\$100>
Tax	\$100
Freight	\$30
Late Charges	\$10

Field	Value
-------	-------

Assume that you are using the Prorate All application rule. Your customer remits a receipt of \$100. Receivables prorates the positive receipt amount among the positive tax, freight, and late charges amounts. The line amount of <\$100> is not affected.

This table shows the new balance of Invoice 101 after the receipt application:

Field	Value
Line	<\$100>
Tax	\$28.56
Freight	\$8.58
Late Charges	\$2.86

This table compares each line type for this invoice before and after you apply the payment using the Prorate All rule:

Line Items	Line Items Remaining	Tax	Tax Remaining	Freight	Freight Remaining	Late Charges	Late Charges Remaining
<\$100>	<\$100>	\$100	\$28.56	\$30	\$8.58	\$10.00	\$2.86

This table shows the amount applied to each line type:

Total Amount Applied	Line Amount Applied	Tax Amount Applied	Freight Amount Applied	Late Charges Amount Applied
100	0	71.44	21.42	7.14

This table shows the calculations used by the Prorate All application rule:

Item	Calculations
Tax	$100 - (21.42 + 7.14) = 71.44$
Freight	$(30 * 100) / 140 = 21.42$
Late Charges	$3(10.00 * 100) / 140 = 7.14$

## FAQs for Define Application Rule Sets

### What's the tax treatment?

The **Tax Treatment** option on the application rule set determines how to reduce the tax amount in relation to the line amount when a payment is applied.

The tax treatment options are:

- Prorate: Proportionately reduce the net amount of the line and associated tax amounts.
- Before: First reduce the open tax amount, then apply any remaining amount to the line.
- After: Reduce the open line amount, then apply any remaining amount to the associated tax.

### How can I use the rounding correction?

You assign a rounding correction to one of the line types (Line, Freight, or Charges) belonging to an application rule set. When an amount is prorated across more than one line type, the application rule set uses the line type you indicate to account for the rounding adjustment. The line amount of the designated line type is adjusted accordingly to account for any rounding corrections within the application rule set.

## Define Receipt Classes and Methods

### Remittance Methods and Clearance Methods

Define a remittance method and clearance method for each receipt class. These settings determine the remittance and clearing behavior for receipts with a given receipt class.

#### Remittance Methods

Use the remittance method to determine the accounts that Oracle Fusion Receivables uses for receipts that you create using the receipt method assigned to this receipt class.

##### Standard

Use the remittance account for automatic receipts assigned to a receipt method with this receipt class.

##### Factoring

Use the factoring account for automatic receipts assigned to a receipt method with this receipt class.

##### Standard and Factoring

Receivables selects receipts assigned to this receipt class for remittance regardless of the batch remittance method. In this case, you can specify either of these remittance methods when creating your remittance batches.

##### No Remittance

For Manual receipts only. Remittance is not required for manual receipts assigned to this receipt class.

## Clearance Methods

Use the clearance method to require receipts created using a receipt method assigned to this receipt class to be reconciled before posting them to the general ledger cash account.

### Directly

This method is for receipts that you don't expect to be remitted to the bank and subsequently cleared.

It is assumed that these receipts are cleared at the time of receipt entry and require no further processing.

### By Automatic Clearing

Use this method to clear receipts using the Clear Receipts Automatically program.

 **Note:** You can also clear receipts using this method in Oracle Fusion Cash Management.

### By Matching

Use this method to clear receipts manually in Cash Management.

#### Related Topics

- [Managing Remittances: Explained](#)
- [Clearing Receipts: Explained](#)

## Automatic Receipt Processing: Points to Consider


Define the attributes of an automatic receipt method to determine how automatic receipts are processed against selected transactions.

There are these points to consider when defining an automatic receipt method:

- Receipts Inherit Transaction Numbers
- Number of Receipts Rule
- Receipt Maturity Date Rule
- Automatic Print Template
- Lead Days
- Customer Payment Method

### Receipts Inherit Transaction Numbers

If you are using One per Invoice as the **Number of Receipts Rule**, you can enable the **Receipts inherit transaction numbers** option to ensure that the automatic receipt number is always the same as the transaction number to which it is applied. Enabling this option helps track automatic receipts.


 **Note:** Don't enable this option if you want to assign document numbers to automatic receipts.

## Number of Receipts Rule

The **Number of Receipts Rule** determines the way in which the automatic receipt process creates and applies receipts against transactions.

Select one of these rules:

- One per Customer: Create one receipt for each customer.
- One per Customer Due Date: Create one receipt for each customer and due date. This option creates several payments for a customer if the invoices of the customer have several due dates.
- One per Invoice: Create one receipt for each invoice.
- One per Site: Create one receipt for each customer site.
- One per Site Due Date: Create one receipt for each customer site and due date.

 **Note:** The **Number of Receipts Rule** assumes an additional grouping by payment instrument. For example, if you use the One per Customer rule, and two invoices belonging to the same customer are to be paid with different credit cards, the automatic receipt process create two receipts, one for each credit card number.

## Receipt Maturity Date Rule

Use the **Receipt Maturity Date Rule** to pay invoices that have different due dates with a single receipt.

Select Earliest to use the earliest due date of all of the invoices that the receipt covers as the receipt maturity date. Select Latest to use the latest due date of all of the invoices that the receipt covers as the receipt maturity date.

When you remit a receipt, Oracle Fusion Receivables uses the maturity date to determine when to transfer funds from the customer bank account to your remittance bank account.

## Automatic Print Template

Enter the automatic print template to use for transmissions using this receipt method.

Receivables provides one standard receipt print template to format the output of payment selection and creation programs when you create the receipt document. To use a different receipt print template, you must copy and modify this standard receipt print template.

## Lead Days

The number of lead days is the number of days before the invoice due date that an invoice can be selected for application by the automatic receipt process using this receipt method.

This option is useful, for example, when customer approval is required. You can set the value to the number of days normally required to receive customer approval.

## Customer Payment Method

Select the funds capture payment method that the customer will use to remit payment for automatic receipts using this receipt method.

Oracle Fusion Payments provides predefined funds capture payment methods, but you can define your own.

### Related Topics

- [Processing Automatic Receipts: How It Works](#)

## Fund Transfer Error Handling: Explained

Define fund transfer error handling on automatic receipt methods to manage error correction during automatic receipt processing.

You define fund transfer error handling to automatically correct errors encountered either during credit card authorization or payment capture, or during bank account transfer.

You set up error handling in the Fund Transfer Error Handling section of the Automatic Processing tab of the Create or Edit Receipt Class and Methods page, for the applicable automatic receipt methods.

### Mapping Third-Party Error Codes

To define fund transfer error handling for a given automatic receipt method, you map the error codes from a third-party credit card provider or financial institution to the corrective actions in Oracle Fusion Receivables for each category of transaction.

When an error occurs during payment processing of automatic receipts with the specified receipt method, Receivables applies the corrective action that corresponds to the given third-party error code.

Map each error code to a corresponding action for each category of transaction. This table indicates the corrective actions available for each category:

Category	Available Actions
Invoice, Debit Memo, Credit Memo	Clear Payment Information, Retry
Receipt	Retry, Reverse Receipt
Refund	Retry, Reverse Receipt

### Mapping Error Codes: Example

You map a credit card provider error code of GW-0062 to the **Invoice** category and the **Retry** action.

When credit card authorization fails and the credit card provider returns the error code of GW-0062 for multiple transactions, then Receivables automatically deletes this error code on these failed transactions. These transactions then become available for inclusion in the next automatic receipt batch.

## Remittance Bank Accounts: Explained

Define remittance bank account information for each receipt method assigned to a receipt class. Remittance bank account information includes the general ledger accounts to use when you enter or apply receipts.

### Remittance Bank Accounts and Receipt Currencies

If you remit receipts in one currency only, you can enter more than one remittance bank account for a receipt method, but you must mark one account as the primary bank account for the receipt method.

If you remit receipts in more than one currency for a receipt method, then you must enter at least one remittance bank account per currency and mark one account per currency as primary.

During receipt entry and processing, Oracle Fusion Receivables uses the primary bank account as the default remittance bank account for the receipt. You can accept this value or enter any other bank account defined for the receipt method that is in the same currency as the receipt.

## Factored Receipts

If the receipt class of the receipt method allows factoring, you can specify the number of **Risk Elimination Days** for factored receipts for a given bank account.

When you factor receipts, Receivables creates a short term debt to account for risk in case of customer default. When you clear or risk eliminate these receipts, the debt is cleared after the receipt maturity date plus the number of risk elimination days that you enter.

### *Related Topics*

- [Creating Accounts: Points to Consider](#)

## FAQs for Define Receipt Classes and Methods

### What creation method should a receipt class have?

Use an Automatic creation method to use the automatic receipt process to create a batch of receipts from selected transactions. For these receipts, Oracle Fusion Payments is responsible for the funds capture process. Use a Manual creation method for receipts either entered manually or imported using lockbox.

### What's the relationship between the receipt class and receipt methods?

The receipt methods assigned to a receipt class account for receipt entries and applications. They also determine the customer remittance bank information.

The receipt class determines the processing steps that are required for receipts. These steps include automatic or manual creation, the remittance method, the bank clearance method, and whether receipts require confirmation by the customer.

### What legal entity is assigned to a receipt?

Legal entities are linked to remittance and internal bank accounts. You set up remittance banks in Oracle Fusion Cash Management. The receipt method determines which remittance bank account, and therefore which legal entity, is assigned to a receipt.

All receipts inherit the legal entity from the bank account, and all refunds inherit the legal entity of the original receipt.

You can perform receipt applications across legal entities, if the receipt and the transactions to which it is applied are in the same business unit. Receipt applications and receipt clearing across legal entities is recorded in the subledger as intercompany accounting.

## FAQs for Receipt Sources



## How does automatic batch numbering work?

The automatic receipt and remittance processes assign a number to an automatic receipt or remittance batch by adding +1 to the previous batch number. When you define the Automatic Receipts receipt source, enter a number in the **Batch Number Starts After** field as the starting point from which to increment the next automatic receipt batch number.

For example, to number automatic receipt batches starting with 1000, enter the number 999. The automatic receipt process adds +1 to each subsequent number as new batches are created.

## Where are receipt sources used?

You only use receipt sources with receipt and remittance batches. This includes automatic receipt batches, lockbox receipts, and receipts created via spreadsheet.

The Automatic Receipts receipt source is used automatically by the automatic receipt and remittance processes. You don't need to enter this receipt source.

## Define Lockbox


### Processing Lockbox Files: How It Works

Use lockbox both to create receipts in Oracle Fusion Receivables from data supplied by your remittance bank and to apply the receipts to customer transactions.

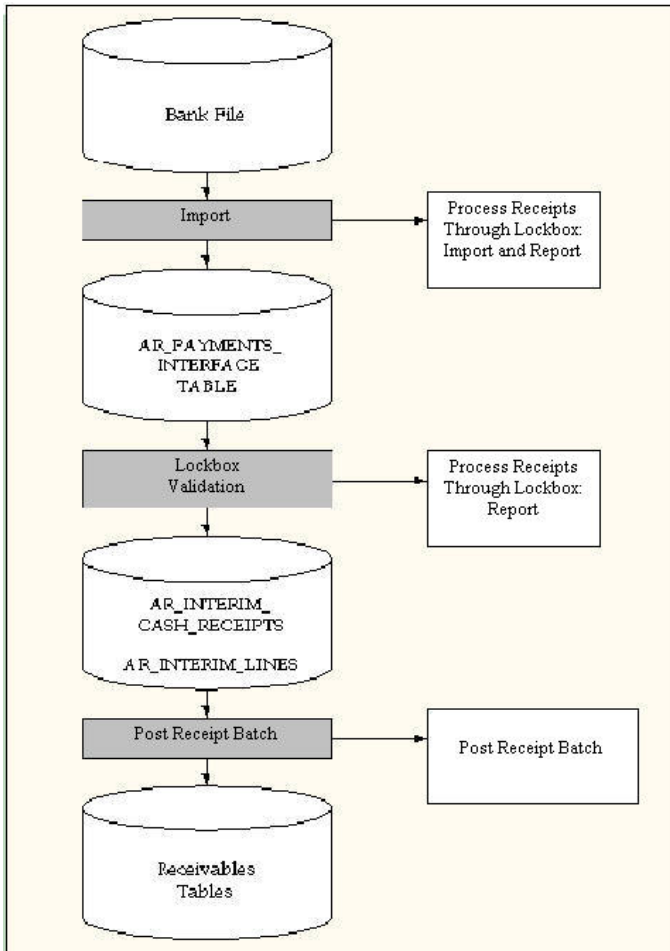
The lockbox process has three steps:

- **Import Data:** Lockbox reads and formats the data from your bank file into the interface table using an SQL\*Loader script.
- **Validate Data:** Receivables validates the data in the interface table for compatibility, then transfers the data to the receipts tables.
- **Post Receipts:** Receivables applies the receipts to transactions and updates the customer balances.

You can submit these steps individually or at the same time. After you post lockbox receipts, Receivables treats these receipts like any other receipts: you can reverse and reapply them, and apply any unapplied, unidentified, or on-account amounts.

 **Note:** You can load data to interface tables using predefined templates and the Load Interface File for Import scheduled process, which are both part of the External Data Integration Services for Oracle Cloud. For more information about file-based data import, see the File Based Data Import guide for your cloud services.

This diagram illustrates the lockbox process:



## Import

You use the Process Receipts Through Lockbox: Import program to read and format data from the bank file into the AR\_PAYMENTS\_INTERFACE\_ALL table using an SQL\*Loader script.

Before you can run the Process Receipts Through Lockbox Import program, you must ensure that you have completed these steps:

1. Review the environment properties file to get the directory path for the \$AR\_TOP and \$APPLBIN environment variables. \$APPLBIN is usually set up as APPLBIN=bin.  
The environment properties file is located at \$FA\_HOME/../../instance/ess/config/ where \$FA\_HOME is in the format //fusionapps/applications/.
2. Ensure that the directory path \$AR\_TOP/\$APPLBIN exists for the installation. If the \$APPLBIN subdirectory does not exist under \$AR\_TOP, then create this subdirectory and provide write access.
3. Place the SQL\*Loader control file (.ctl) in the \$AR\_TOP/\$APPLBIN subdirectory.
4. Enter the control file name in the **Control File** parameter of the Process Receipts Through Lockbox Import program. The fully qualified path is not necessary.
5. Enter the full path of the data file in the **Data File** parameter.

6. If you used the Load Interface File for Import process to load the data file, select Import Request ID.

Receivables uses the lockbox transmission format that you specify in the Process Receipts Through Lockbox: Import program submission to ensure that data is correctly transferred from the bank file into the AR\_PAYMENTS\_INTERFACE\_ALL table. The transmission format contains information such as the customer account number, bank account number, the amount of each receipt to apply, and transaction numbers to which to apply each receipt.

## Validate

The validation process includes these checks:

- No duplicate entries exist.
- Customer and receipt information is valid.
- Amounts to apply do not exceed the receipt amount.
- Columns in the AR\_PAYMENTS\_INTERFACE\_ALL table reference the correct values and columns in Receivables.

If the receipt and transaction currencies are different, Receivables also requires specific application information to determine the conversion rate between the two currencies.

Receivables transfers the receipts that pass validation to the AR\_INTERIM\_CASH\_RECEIPTS\_ALL and AR\_INTERIM\_CASH\_RCPT\_LINES\_ALL interface tables. At this point, you can optionally review receipts and change how they are to be applied before posting.

Use the Process Receipts Through Lockbox: Report to review all pass and fail validations for a lockbox transmission. Receipts that fail validation remain in the AR\_PAYMENTS\_INTERFACE\_ALL table until you manually correct the import errors. You can then resubmit the validation step for these receipts.

## Post

Use the Post Receipt Batch program to post lockbox receipt batches that have passed import and validation.

When you process a lockbox receipt batch, Receivables matches receipts to transactions and applies the receipts automatically. In cases where receipts are not applied automatically, Receivables generates a list of recommended transactions for receipt application to complete the process manually.

### Related Topics

- [Can one customer pay for another customer's transactions using lockbox?](#)
- [Why are there duplicate transactions in a lockbox?](#)
- [External Data Integration Services for Oracle Cloud: Overview](#)
- [File Based Data Import for Oracle Financials Cloud](#)

## Lockbox Interface Table AR\_PAYMENTS\_INTERFACE\_ALL

During the Import step of lockbox processing, Oracle Fusion Receivables stores receipt data from your bank file in the AR\_PAYMENTS\_INTERFACE\_ALL table. Each column in the AR\_PAYMENTS\_INTERFACE\_ALL table contains information needed to run lockbox successfully.

### TRANSMISSION\_RECORD\_ID

Receivables assigns a value to this column during the import process.

## Source

AR\_PAYMENTS\_INTERFACE\_S.NEXTVAL

## Validation

The import file must leave this column blank.

## Destination

None.

## CREATION\_DATE

Receivables assigns a value to this column during the import process.

## Source

Current system date.

## Validation

The import file must leave this column blank.

## Destination

None.

## CREATED\_BY

Receivables assigns a value to this column during the import process.

## Source

FND\_GLOBAL.WHO\_USER\_NAME

## Validation

The import file must leave this column blank.

## Destination

AR\_BATCHES.CREATED\_BY, AR\_INTERIM\_CASH\_RECEIPTS.CREATED\_BY, or  
AR\_INTERIM\_CASH\_RECEIPT\_LINES.CREATED\_BY.

## LAST\_UPDATE\_LOGIN

Receivables assigns a value to this column during the import process.

## Source

Unknown.

## Validation

The import file must leave this column blank.

## Destination

None.

## LAST\_UPDATED\_BY

Receivables assigns a value to this column during the import process.

## Source

FND\_GLOBAL.WHO\_USER\_NAME

## Validation

The import file must leave this column blank.

## Destination

None.

## LAST\_UPDATE\_DATE

Receivables assigns a value to this column during the import process.

## Source

Current system date.

## Validation

The import file must leave this column blank.

## Destination

None.

## RECORD\_TYPE

Enter the record type. For example, if this is a batch header record, and your bank uses the value 3 to identify batch headers, enter 3 in this column.

## Source

AR\_TRANS\_RECORD\_FORMATS.RECORD\_IDENTIFIER

## Validation

Find out from the bank what character they use to identify each record type. Keep in mind that not all banks use all of the record types. Assign values to identify the following types of records: TRANSMISSION HEADER, TRANSMISSION TRAILER,

LOCKBOX HEADERS, LOCKBOX TRAILERS, BATCH HEADERS, BATCH TRAILERS, PAYMENT RECORDS, PAYMENT OVERFLOW RECORDS, and SERVICE HEADER.

## Destination

None.

## STATUS

Enter the value AR\_PLB\_NEW\_RECORD\_INF for all records inserted into this table. The sample SQL\*Loader control files that Receivables provides populate this column.

## Source

FND\_MESSAGES.MESSAGE\_NAME

## Validation

The import file must leave this column blank.

## Destination

None.

## TRANSMISSION\_REQUEST\_ID

Receivables assigns a value to this column during the import process.

## Source

FND\_JOB.REQUEST\_ID

## Validation

The import file must leave this column blank.

## Destination

None.

## TRANSMISSION\_ID

Receivables assigns a value to this column during the import process.

## Source

AR\_TRANSMISSIONS.TRANSMISSION\_ID

## Validation

The import file must leave this column blank.

## Destination

None.

## DESTINATION\_ACCOUNT

Enter the account number of the sending bank.

## Source

Lockbox data file or lockbox transmission format.

## Validation

None.

## Destination

AR\_TRANSMISSIONS.DESTINATION

## ORIGINATION

Enter the transit routing number of the sending bank.

## Source

Lockbox data file or lockbox transmission format.

## Validation

If this value is included in a transmission header or trailer, you must have the same value here.

## Destination

AR\_TRANSMISSIONS.ORIGIN

## DEPOSIT\_DATE

Enter the date on which the transmission was deposited into your bank account. This date can be on any of the record types in the transmission.

Each unique deposit date determines a batch of transmission records. For example, if you enter two unique deposit dates for the transmission, lockbox divides the transmission into two batches of receipts.

## Source

Lockbox data file or lockbox transmission format.

## Validation

None.

## Destination

AR\_BATCHES.DEPOSIT\_DATE

## GL\_DATE

Receivables assigns a value to this column during the import process.

## Source

Derived from deposit date, import date, or entered date.

## Validation

The import file must leave this column blank.

## Destination

AR\_BATCHES.GL\_DATE, AR\_INTERIM\_CASH\_RECEIPTS.GL\_DATE, or AR\_CASH\_RECEIPT\_HISTORY.GL\_DATE.

## DEPOSIT\_TIME

Enter the time the deposit was made.

## Source

Lockbox data file or lockbox transmission format.

## Validation

None.

## Destination

None.

## TRANSMISSION\_RECORD\_COUNT

Enter the number of records in the import file. Include all of the types of records in the count: headers, trailers, receipts, and overflow records.

## Source

Lockbox data file or lockbox transmission format.

## Validation

If the transmission format includes the transmission header or trailer, lockbox counts all records in this transmission. The validated count includes all receipts and detail records transferred to the interim table.

## Destination

AR\_TRANSMISSIONS.COUNT



## TRANSMISSION\_AMOUNT

Enter the amount of the transmission.

### Source

Lockbox data file or lockbox transmission format.

### Validation

The sum of all of the receipt amounts within the transmission.

### Destination

AR\_TRANSMISSIONS.AMOUNT

## TRANSFERRED\_RECEIPT\_COUNT

Receivables assigns a value to this column during the import process.

### Source

Program counts the number of records transferred successfully.

### Validation

The import file must leave this column blank.

### Destination

AR\_TRANSMISSIONS.VALIDATED\_COUNT

## TRANSFERRED\_RECEIPT\_AMOUNT

Receivables assigns a value to this column during the import process.

### Source

Program totals the receipt amounts of records transferred successfully.

### Validation

The import file must leave this column blank.

### Destination

AR\_TRANSMISSIONS.VALIDATED\_AMOUNT

## LOCKBOX\_NUMBER

For lockbox header or trailer records, enter the lockbox name or number specified by the bank.

For batch header and trailer records, enter the lockbox number assigned to receipts in the batch.

For receipt records, enter the lockbox number assigned to receipts.

For overflow records, enter the number of the lockbox for the receipt.

## Source

Provided by the bank or entered by the user.

## Validation

This column is required on all lockbox headers and trailers. If the lockbox number is included in the lockbox transmission format, it must appear on every batch record. If the lockbox number is included in the lockbox transmission format and you don't have batch records, it must be entered for each receipt and overflow record.

## Destination

None.

## LOCKBOX\_BATCH\_COUNT

Enter the number of batches in the lockbox.

## Source

Lockbox data file or lockbox transmission format.

## Validation

None.

## Destination

None.

## LOCKBOX\_RECORD\_COUNT

Enter the number of payment records in the lockbox.

## Source

Lockbox data file or lockbox transmission format.

## Validation

Do not include payment overflow records.

## Destination

None.

## LOCKBOX\_AMOUNT

Enter the total value of the receipts in the lockbox.

### Source

Lockbox data file or lockbox transmission format.

### Validation

None.

### Destination

None.

## BATCH\_NAME

For batch header and trailer records, enter the name or number that the bank uses to identify the batch.

For receipt records, enter the batch name for this receipt.

For overflow records, enter the batch for this overflow record.

### Source

Lockbox data file or lockbox transmission format.

### Validation

This column is required for each batch header and trailer record. If the batch name is included in your format, it must be entered for every receipt record. Each unique batch name determines a batch of transmission records. For example, if you enter two unique batch names for your transmission, lockbox divides your transmission into two batches of receipts. If the batch name is included in your format, you must enter this name for each overflow record.

### Destination

AR\_BATCHES.LOCKBOX\_BATCH\_NAME

## BATCH\_AMOUNT

Enter the total value of all receipts in this batch.

### Source

Lockbox data file or lockbox transmission format.

### Validation

None.

## Destination

AR\_BATCHES.CONTROL\_AMOUNT

## BATCH\_RECORD\_COUNT

Enter the number of receipt records in this batch.

## Source

Lockbox data file or lockbox transmission format.

## Validation

None.

## Destination

AR\_BATCHES.CONTROL\_COUNT

## ITEM\_NUMBER

Enter a sequential number to indicate the location of each receipt or overflow record in the batch.

## Source

Lockbox data file or lockbox transmission format.

## Validation

This column is required, even if the lockbox transmission format doesn't have batch, lockbox, or transmission records. The item number must be unique within a batch, a lockbox (if batches are not provided), or within a transmission (if neither batches nor lockboxes are provided). All overflow records for a receipt must have the same item number as the receipt record. You must enter an item number for each overflow record to reference the receipt.

## Destination

None.

## CURRENCY\_CODE

Enter the currency for each receipt.

## Source

Lockbox data file or lockbox transmission format.

## Validation

None.

## Destination

AR\_BATCHES.CURRENCY\_CODE or AR\_INTERIM\_CASH\_RECEIPTS.CURRENCY\_CODE.

## EXCHANGE\_RATE

For receipts, enter the conversion rate to use for this currency.

## Source

Lockbox data file or lockbox transmission format.

## Validation

None.

## Destination

AR\_BATCHES.EXCHANGE\_RATE or AR\_INTERIM\_CASH\_RECEIPTS.EXCHANGE\_RATE.

## EXCHANGE\_RATE\_TYPE

Enter the conversion rate type to use for the receipt: Corporate, Spot, or User.

## Source

Derived from the lockbox definition or lockbox transmission format.

## Validation

None.

## Destination

AR\_BATCHES.EXCHANGE\_RATE\_TYPE or AR\_INTERIM\_CASH\_RECEIPTS.EXCHANGE\_RATE\_TYPE.

## REMITTANCE\_AMOUNT

Enter the value of each receipt in the batch.

## Source

Lockbox data file or lockbox transmission format.

## Validation

A value is required for each receipt record.

## Destination

AR\_INTERIM\_CASH\_RECEIPTS.AMOUNT

## TRANSIT\_ROUTING\_NUMBER

Enter the transit routing number from the receipt.

### Source

Lockbox data file or lockbox transmission format.

### Validation

This column is optional, but you must enter this number if you enter the account number. Receivables uses the transit routing number and account number together to identify the customer MICR number.

### Destination

AP\_BANK\_BRANCHES.BANK\_NAME, AP\_BANK\_BRANCHES.BANK\_BRANCH\_NAME, or  
AP\_BANK\_BRANCHES.BANK\_NUM

## ACCOUNT

Enter the bank account number from the receipt.

### Source

Lockbox data file or lockbox transmission format.

### Validation

This column is optional, but you must enter this number if you enter the transit routing number. Receivables uses the transit routing number and account number together to identify the customer MICR number.

### Destination

AP\_BANK\_ACCOUNTS.BANK\_ACCOUNT\_NUM

## CUSTOMER\_BANK\_ACCOUNT\_ID

Receivables assigns a value to this column during the import process.

### Source

AP\_BANK\_ACCOUNT\_USES.EXTERNAL\_BANK\_ACCOUNT\_ID

### Validation

The import file must leave this column blank.

### Destination

AR\_INTERIM\_CASH\_RECEIPTS.CUSTOMER\_BANK\_ACCOUNT\_ID

## ANTICIPATED\_CLEARING\_DATE

Date a receipt is expected to clear.

### Source

Lockbox data file or lockbox transmission format.

### Validation

None.

### Destination

AR\_INTERIM\_CASH\_RECEIPTS.ANTICIPATED\_CLEARING\_DATE

## CHECK\_NUMBER

Enter the check number printed on the receipt.

### Source

Lockbox data file or lockbox transmission format.

### Validation

A value is required for each receipt record.

### Destination

AR\_INTERIM\_CASH\_RECEIPTS.RECEIPT\_NUMBER or AR\_CASH\_RECEIPTS.RECEIPT\_NUMBER.

## SPECIAL\_TYPE

Receivables assigns a value to this column during the import process.

### Source

Program determines the type.

### Validation

The import file must leave this column blank.

### Destination

AR\_INTERIM\_CASH\_RECEIPTS.SPECIAL\_TYPE

## AUTOAPPLY\_FLAG

Indicates whether the AutoApply process is used during lockbox processing.

## Source

Lockbox data file or lockbox transmission format.

## Validation

None.

## Destination

AR\_INTERIM\_CASH\_RECEIPTS.AUTOAPPLY\_FLAG

# CUSTOMER\_NUMBER

Enter the customer account number.

## Source

Lockbox data file or lockbox transmission format.

## Validation

This column is optional.

## Destination

None.

# OVERFLOW\_INDICATOR

Receivables uses this column to indicate overflow records for the current receipt. You must enter a value for all overflow records.

## Source

Lockbox data file or lockbox transmission format.

## Validation

To identify the last overflow record, enter a value that is different from the overflow indicator in the transmission format. For example, in the BAI transmission format, 0 indicates an overflow record. You have three overflow records for a receipt, the first two records have 0 as the overflow indicator and the third record has 9. Since the third record is not 0, it is identified as the last overflow record.

## Destination

None.

# OVERFLOW\_SEQUENCE

Enter a sequential number to indicate the order of overflow records.



## Source

Lockbox data file or lockbox transmission format.

## Validation

Within each receipt, the Overflow Sequence usually begins with 1.

## Destination

None.

## CUSTOMER\_ID

Receivables assigns a value to this column during the import process.

## Source

Program determines the ID.

## Validation

The import file must leave this column blank.

## Destination

AR\_INTERIM\_CASH\_RECEIPTS.PAY\_FROM\_CUSTOMER or AR\_CASH\_RECEIPTS.PAY\_FROM\_CUSTOMER.

## BILL\_TO\_LOCATION

Enter the customer bill-to site for this receipt and include the bill-to site in the transmission format.

## Source

Lockbox data file or lockbox transmission format.

## Validation

If the **Require billing location for receipts** Receivables system option is enabled, you must enter a value in this column. If the **Require billing location for receipts** Receivables system option is not enabled, you only have to enter a value in this column if the **Require billing location** option on the lockbox record is enabled.

## Destination

None.

## CUSTOMER\_SITE\_USE\_ID

Receivables assigns a value to this column during the import process.

## Source

Program determines the ID.

## Validation

The import file must leave this column blank.

## Destination

AR\_INTERIM\_CASH\_RECEIPTS.SITE\_USE\_ID or AR\_CASH\_RECEIPTS.CUSTOMER\_SITE\_USE\_ID.

## RECEIPT\_DATE

For receipt records, enter the date that is written on the check.

## Source

Lockbox data file or lockbox transmission format.

## Validation

If you are using MICR numbers to identify customers, this date must be equal to or earlier than the date of the lockbox submission. Otherwise, the receipts are processed as unidentified.

## Destination

AR\_INTERIM\_CASH\_RECEIPTS.RECEIPT\_DATE, AR\_INTERIM\_CASH\_RECEIPTS.EXCHANGE\_DATE,  
AR\_CASH\_RECEIPTS.RECEIPT\_DATE, or AR\_CASH\_RECEIPTS.EXCHANGE\_DATE.

## RECEIPT\_METHOD

Enter the receipt method to associate with a receipt.

## Source

Lockbox data file or lockbox transmission format.

## Validation

Receipt methods contain information about the bank, bank account, and receipt distributions. The receipt method in this column must be the same as the receipt method assigned to the batch source for the lockbox.

## Destination

None.

## RECEIPT\_METHOD\_ID

Receivables assigns a value to this column during the import process.

## Source

Program determines the ID.

## Validation

The import file must leave this column blank.

## Destination

AR\_INTERIM\_CASH\_RECEIPTS.RECEIPT\_METHOD\_ID or AR\_CASH\_RECEIPTS.RECEIPT\_METHOD\_ID.

## INVOICE1-8

For receipt records and overflow records, optionally enter the invoice numbers that a receipt is applied to.

## Source

Lockbox data file or lockbox transmission format.

## Validation

You don't have to start with INVOICE1, nor use all eight of the INVOICE columns on a record before you create a receipt record or an overflow record. You can find a list of valid values in AR\_PAYMENT\_SCHEDULES.TRX\_NUMBER. You can supply invoice numbers without specifying the amount applied to each invoice.

## Destination

None.

## MATCHING1\_DATE - MATCHING8\_DATE

For matching dates.

## Destination

None.

## RESOLVED\_MATCHING\_NUMBER1-8

For resolved matching numbers.

## Source

Program determines the numbers.

## Destination

None.

## RESOLVED\_MATCHING1\_DATE - RESOLVED\_MATCHING8\_DATE

For resolved matching dates.

## Source

Program determines the dates.

## Destination

None.

## MATCH\_RESOLVED\_USING

For a match resolved.

## Source

Program determines the value.

## Validation

GD: No description or validation information is provided.

## Destination

None.

## RESOLVED\_MATCHING1\_INSTALLMENT RESOLVED\_MATCHING8\_INSTALLMENT

-

For resolved matching installments.

## Source

Program determines the value.

## Destination

None.

## INVOICE1\_STATUS - INVOICE8\_STATUS

Status of the invoices that a receipt is applied to.

## Source

Program determines the value.

## Validation

None.

## Destination

None.

## COMMENTS

For batch header and trailer records and for receipt records, enter any free-form comments.

## Source

Lockbox transmission format.

## Validation

For receipt records, Receivables stores these comments but doesn't display them on the receipts.

## Destination

AR\_BATCHES.COMMENTS or AR\_INTERIM\_CASH\_RECEIPTS.COMMENTS

## ATTRIBUTE\_CATEGORY

Enter the descriptive flexfield category information for this receipt.

## Source

Lockbox data file or lockbox transmission format.

## Validation

Descriptive flexfield category used to transfer additional information about a receipt.

## Destination

AR\_INTERIM\_CASH\_RECEIPTS.ATTRIBUTE1-15 or AR\_CASH\_RECEIPTS.ATTRIBUTE1-15.

## ATTRIBUTE1-15

Enter the descriptive flexfield attributes of the category designated in the ATTRIBUTE\_CATEGORY column.

## Source

Lockbox data file or lockbox transmission format.

## Validation

Use this column to transfer additional information about a receipt. For example, if the bank enters and transmits the customer name, use an attribute column to import this name.

## Destination

AR\_INTERIM\_CASH\_RECEIPTS.ATTRIBUTE1-15 or AR\_CASH\_RECEIPTS.ATTRIBUTE1-15.

## INVOICE1\_INSTALLMENT - INVOICE8\_INSTALLMENT

For receipt records and overflow records, enter the installment numbers of invoices with multiple payment schedules that a receipt is applied to.

### Source

Lockbox data file or lockbox transmission format.

### Validation

If you don't specify the installment number for an invoice with multiple payment schedules, then Receivables applies the receipt amount to the oldest payment schedule first. The installment number must be on the same record as the associated invoice number.

### Destination

None.

## CUSTOMER\_NAME\_ALT

The alternate name of the customer.

### Source

Lockbox data file.

### Validation

None.

### Destination

None.

## CUSTOMER\_BANK\_NAME

The name of the customer bank. Used for receipt records.

### Source

Lockbox data file.

### Validation

None.

### Destination

None.

## CUSTOMER\_BANK\_BRANCH\_NAME

The name of the customer bank branch. Used for receipt records.

### Source

Lockbox data file.

### Validation

None.

### Destination

None.

## REMITTANCE\_BANK\_NAME

The name of the bank that received the payment. Used for receipt records.

### Source

Program determines the name.

### Validation

None.

### Destination

None.

## BANK\_TRX\_CODE

The transaction code of the bank.

### Source

Program determines the code.

### Validation

None.

### Destination

None.

## AMOUNT\_APPLIED1-8

Enter the amount of the receipt to apply to the invoice. If the receipt currency and the transaction currency are different, enter the amount of the receipt to apply in the transaction currency.

## Source

Lockbox data file or derived from AMOUNT\_APPLIED\_FROM and EXCHANGE\_RATE.

## Validation

If you provide invoice numbers without specifying the amount applied to each invoice, Receivables applies the receipt to the invoices starting with the oldest receipt schedule first. The value of the AMOUNT\_APPLIED column must be on the same record as the invoice number to which it is applied. For example, you can't have all of the invoice numbers on the receipt record and all of the amounts applied on the overflow record.

## Destination

AR\_INTERIM\_CASH\_RECEIPTS\_ALL.AMOUNT\_APPLIED (if a single application) or  
AR\_INTERIM\_CASH\_RCPT\_LINES\_ALL.PAYMENT\_AMOUNT (if multiple applications).

## AMOUNT\_APPLIED\_FROM1-8

For receipt and overflow records, if the receipt currency and the transaction currency are different, enter the amount of the receipt to apply in the receipt currency.

## Source

Lockbox data file or derived from AMOUNT\_APPLIED\_FROM and EXCHANGE\_RATE.

## Validation

None.

## Destination

AR\_INTERIM\_CASH\_RECEIPTS\_ALL.AMOUNT (if a single application) or  
AR\_INTERIM\_CASH\_RCPT\_LINES\_ALL.AMOUNT\_APPLIED\_FROM (if multiple applications).

## INVOICE\_CURRENCY\_CODE1-8

For receipt and overflow records, if the receipt currency and the transaction currency are different, enter the currency of the transaction. This column is used for cross-currency receipt applications.

## Source

Lockbox data file or derived from AR\_PAYMENT\_SCHEDULES\_ALL.

## Validation

This column is optional. If null, lockbox derives this value from AR\_PAYMENT\_SCHEDULES\_ALL.

## Destination

AR\_INTERIM\_CASH\_RECEIPTS\_ALL.INVOICE\_CURRENCY\_CODE (if a single application) or  
AR\_INTERIM\_CASH\_RCPT\_LINES\_ALL.INVOICE\_CURRENCY\_CODE (if multiple applications).



## TRANS\_TO\_RECEIPT\_RATE1-8

For receipt and overflow records, if the receipt currency and the transaction currency are different, enter the conversion rate used to convert the receipt to the transaction currency.

### Source

Lockbox data file or derived from AMOUNT\_APPLIED\_FROM and EXCHANGE\_RATE.

### Validation

This value is used for cross-currency receipt applications when the receipt and transaction currencies do not have a fixed conversion rate.

### Destination

TRANS\_TO\_RECEIPT\_RATE.

## CUSTOMER\_REFERENCE\_1-8

For customer reference.


## CUSTOMER\_REASON1-8

For customer reason codes.

## Using a Custom Matching Rule with Lockbox: Explained

Oracle Fusion Receivables supplies the packaged procedure `arp_lockbox_hook.cursor_for_matching_rule`, which you can use to add your own custom matching rule with lockbox.

You can create a customer matching rule, for example, if you need to match numbers and dates passed to lockbox with numbers and dates in your own custom tables (`custom_table.custom_number` and `custom_table.custom_date`) instead of or in addition to standard matching options. You can also use this feature to match with other numbers and dates in the existing Receivables tables.

 **Note:** The packaged procedure `arp_lockbox_hook.cursor_for_matching_rule` is not available in Oracle Financials Cloud implementations.

### Using the Procedure

This procedure expects a row in the `AR_LOOKUPS` table with `lookup_type = ARLPLB_MATCHING_OPTION` and valid values for other columns required for using a customized matching rule. The master program `arp_process_lockbox` will fetch that row and, if it finds it to be one of the non-standard rows, it will pass the control to this procedure with the corresponding `lookup_code` in your database.


The procedure should return a string that Dynamic SQL can use to open and parse a cursor. You need to create this SQL string to replace the string named `p_cursor_string`.

Your string should have the following restrictions:

1. You should only use the following bind variables:
  - o `b_current_matching_number`: This retrieves a value of a `matching_number` passed in the overflow or payment record.
  - o `b_current_matching_date`: This retrieves a value of a `matching_date` passed in the overflow or payment record.
  - o `b_current_installment`: This retrieves a value for the installment number (if any) passed in the overflow or payment record.
  - o `b_customer_id`: If the customer is identified using a customer number or an MICR number, the program enforces that the `matching_number` is for the same customer (unless the value is Y in `b_pay_unrelated_customers`).
  - o `b_pay_unrelated_customers`: When you submit lockbox, the program prompts you to select whether to allow payments for unrelated customers. This variable retrieves a value Y or ' based on the value that you select.
  - o `b_lockbox_matching_option`: The value of this variable matches to the value of `ar_lookups.lookup_code`. It is also stored in `ar_customer_profiles.lockbox_matching_option` and in `ar_lockboxes.lockbox_matching_option`.
  - o `b_use_matching_date`: This variable is assigned a value NEVER, ALWAYS, or FOR\_DUPLICATES, depending upon the value of the Match on Corresponding Date option for your lockbox (in `ar_lockboxes`).
  - o `b_receipt_date`: This retrieves a value of `receipt_date` passed in the payment record.
  - o `b_cust_acct_rel_set_id`: This retrieves the set ID assigned to the reference group Customer Account Relationship for the business unit for which the lockbox is run.
2. If you are customizing lockbox using this procedure, be sure that this procedure returns a string that can create a valid cursor and that the SQL returns one and only one row (neither zero nor more than one).
3. The program expects three return values from the SQL statement in the following order:
  - a. `Customer_ID` (NUMBER(15))
  - b. `Invoice Number` (VARCHAR2(20))
  - c. `Invoice Date` (DATE)
4. The program expects that the combination of invoice number and invoice date is unique in `ar_payment_schedules`.
5. You don't have to use all the bind variables that are provided in your SQL statement. For example:

```
ct.trx_date ' ||  
'from custom_table ct ' ||  
'where ct.matching_number = :b_current_matching_number '  
||  
'and ct.matching_date = :b_current_matching_date  
';
```
6. If the SQL statement doesn't match with the given matching number and matching date (optional), the statement must return the following:

```
customer_id = -9999,  
trx_number = null,  
trx_date = null.
```
7. If the statement matches to multiple customers but with the same transaction numbers, it must return `customer_id = -7777`. In this case the procedure ignores `trx_number` and `trx_date`.

 **Note:** The program calling this procedure doesn't expect it to return any errors, because the definition of a cursor is a one-time procedure and, if done carefully, should not error.

Below is the packaged procedure `arp_lockbox_hook.cursor_for_matching_rule` that Receivables provides:

```
PROCEDURE CURSOR_FOR_MATCHING_RULE(p_matching_option IN
VARCHAR2,p_cursor_string OUT VARCHAR2) IS
BEGIN
arp_util.debug('arp_lockbox_hook.cursor_for_matching_rule()');
p_cursor_string := 'select -9999, NULL, NULL from dual';
arp_util.debug('arp_lockbox_hook.cursor_for_matching_rule()');
RETURN;
END cursor_for_matching_rule;
END arp_lockbox_hook;
COMMIT;
EXIT;
```

## Match Receipts By Method: Explained

During lockbox and manual receipt processing, Oracle Fusion Receivables uses the settings of the Match Receipts By rule to identify the document type to use to match receipts to transactions when customer information is not available.

### Using the Document Type Reference

The following six document types used to match receipts to transactions. During receipt processing, Receivables searches for one of these document types in this order:

- Transaction number
- Sales order number
- Purchase order number
- Balance forward billing number
- Shipping reference
- Contract number

### Using the Match Receipts By Rule

When Receivables finds a document type with the same number as the current search to match the receipt with a transaction, the process checks the locations where Match Receipts By rules are enabled in this order:

- Customer bill-to site
- Customer
- Lockbox (for lockbox processing)
- System options

Receivables looks for a rule that matches the document type of the number in the current search, and stops when a value is found. For example, if Receivables finds a matching transaction number in the first search, it checks the customer site for the Match Receipts By rule. If the rule is set to Transaction, Receivables matches the receipt with this transaction and applies the receipt.

If the Match Receipts By rule at the customer site is a document type other than Transaction, Receivables searches for a number that matches this document type.

If there are no values assigned at the customer site or customer level:

- For lockbox processing, Receivables uses either the Match Receipts By rule assigned to the lockbox or, if the **Use match criteria to determine customer** option is enabled, the entire document type hierarchy.
- For manual receipt processing, Receivables uses the Match Receipts By settings on the Receivables system options record assigned to the business unit.

If Receivables cannot find a match after searching each document type, the process applies the receipt using the AutoCash rule set defined for the customer.

If the AutoCash rule set is unable to apply the receipt, Receivables assigns the receipt a status of Unapplied. You must then manually apply the receipt.

## Matching Rules Examples

Here are two examples of using matching rules.

Example 1: During lockbox processing, a receipt record indicates that a receipt should be applied to open debit item 12345. Receivables first searches for a transaction (invoice, debit memo, chargeback) with this number. Receivables finds an invoice with this number, so the process checks the value of the Match Receipts By parameter at the customer site. The Match Receipts By rule is null for this customer site, so Receivables checks the setting in the customer profile. Match Receipts By is set to Transaction in the customer profile, so Receivables matches and applies the receipt to the invoice.

Example 2: Using the same receipt record information as Example 1, assume that Receivables fails to find a transaction with the number 12345. The process then searches for a sales order with this number. Receivables does not find a sales order with this number, so it now searches for and finds a purchase order with number 12345. Receivables then checks the Match Receipts By rule at the customer site. The Match Receipts By rule is null for this customer site, so Receivables checks the setting in the customer profile. The rule is also null in the customer profile, so Receivables checks the rule for the lockbox. The Match Receipts By rule is set to Purchase Order Number for this lockbox, so the process matches the receipt with this purchase order and applies the receipt to the transaction.

### Related Topics

- [Using AutoCash Rules: Examples](#)
- [What are exception trends?](#)

## FAQs for Define Lockbox

### Why can't I find a receipt source in the lockbox?

Receipt sources are created and assigned to a particular business unit. Lockboxes are created and assigned to a particular lockbox set. The receipt sources available for use with a given lockbox are limited to the receipt sources created and assigned to the business units that belong to the lockbox set.

To define a receipt source for use with lockbox, use these settings:

- Set **Receipt Source Type** to Manual.
- Provide both a default **Receipt Method** and a default **Bank Account**.
- Set **Batch Numbering** to Automatic.

### How can I calculate the lockbox batch size?

Use the **Batch Size** field to enter the number of receipts to assign to each receipt batch during lockbox processing. For example, if you have 991 receipts, and you set the batch size to 10, the lockbox process creates 99 batches with 10 receipts and one batch with one receipt.

If you do not want the lockbox process to separate your lockbox submission into multiple receipt batches, complete these steps:

- Enter a number that is larger than the number of receipts in the lockbox transmission for this lockbox.

- Enable the **Complete batches only** option when you submit your lockbox transmission.

## How can I use the accounting date source?

The value in the lockbox **Accounting Date Source** field determines the accounting date to apply to the receipts in the lockbox.

The accounting date source values for lockbox processing are:

- **Constant Date:** Use the accounting date that you enter in the lockbox transmission.  
If you do not enter an accounting date, the lockbox process does not validate your data.
- **Deposit Date:** Use the deposit date that you enter in the lockbox transmission. This is the date that your remittance bank deposits your receipts.  
If you do not enter a deposit date, the lockbox process displays an error and prompts for a deposit date to submit the lockbox.
- **Import Date:** Use the date you import receipts.

## How does AutoApply manage invalid transaction numbers?

If lockbox uses AutoApply, then the lockbox process attempts to match customers to receipts and receipts to transactions based on your setup.

If lockbox cannot fully apply a receipt due to invalid transaction numbers, then it manages the additional amounts according to the **Post Partial Amount as Unapplied** option. Receipts are applied to valid transactions, and the remaining receipt amounts are marked as Unapplied.

# Define Transmission Formats for Lockbox

## Validating the Lockbox File Transmission: How It Works

The first step in lockbox processing is validating the data imported from your bank file using the lockbox file transmission.

The lockbox process validates the data that you receive from the bank to ensure that:

- the entire file was received.
- there are no duplicate receipts within a batch.
- the customers and transactions are valid.

Lockbox also validates that all data is compatible with Oracle Fusion Receivables by ensuring that the columns in the AR\_PAYMENTS\_INTERFACE\_ALL table reference the appropriate values and columns in Receivables.

## Settings That Affect Lockbox Validation

Lockbox checks for duplicate receipts and transactions.

Duplicate receipts have the same receipt number, amount, currency, and customer account number. Lockbox doesn't allow duplicate receipts within the same receipt source for the same customer. This is the same validation Receivables performs when you manually enter receipts.

Transaction numbers are only required to be unique within a receipt source. A customer can have duplicate transaction numbers as long as they belong to different receipt sources. However, lockbox can't automatically apply a payment to these transactions.

If a customer has more than one transaction in the system with the same number, then lockbox can't determine to which transaction to apply the payment. In this case, the receipt is either left as Unapplied (if the customer account number or MICR number is provided) or Unidentified (if the customer account number or MICR number is not provided).

You can manually apply the receipt according to the transaction recommendations that Receivables presents according to your implementation.

## How a Lockbox Transmission Is Validated

When you import a bank file, lockbox completes the following validations:

- Transmission Level Validations
- Lockbox Level Validations
- Batch Level Validations
- Receipt Level Validations
- Overflow Level Validations
- Customer Validations
- Currency Validation

### Transmission Level Validations

Lockbox validates the lockbox transmission to ensure that transmission information corresponds to the transmission format. The following attributes are validated:

- Transmission format contains receipt records.
- Either the lockbox number is part of the transmission format, or you specify the lockbox number when you submit the lockbox.
- Accounting date is in an open accounting period.
- Total transmission record count and amount that you supply must match the actual receipt count and amount that is determined by lockbox. If the transmission format includes the transmission header or trailer, lockbox counts all records in this transmission. The validated count includes all receipts and detail records transferred to the interim table.
- Origination number is valid, if it is provided.

### Lockbox Level Validations

Lockbox validates the lockbox records to ensure that lockbox information corresponds to the transmission format. The following attributes are validated:

- If the transmission format includes the transmission header or trailer, ensure that the lockbox number is included and is valid.
- Lockbox batch count is correct, if it is provided.
- Lockbox amount is correct, if it is provided.
- Lockbox record count is correct, if it is provided.
- Origination number is valid, if it is provided.

- No duplicate lockbox numbers.

## Batch Level Validations

Lockbox validates the batch records to ensure that batch information corresponds to the transmission format. The following attributes are validated:

- Batch name exists on batch records.
- Batch name is unique within the transmission.
- Batch amount is correct.
- Batch record count is correct.
- Lockbox number exists on batch records, if this number is part of the transmission format.

## Receipt Level Validations

Lockbox validates the receipt records to ensure that receipt information corresponds to the transmission format. The following attributes are validated:


- Remittance amount is specified.
- Check number is specified.
- Item number is specified and is unique within a batch, a lockbox, or the transmission, depending on the transmission format.
- Lockbox number is specified (if this number is not part of the lockbox header or trailer of the transmission format) and batches are not imported.
- Batch name is specified, if either batch headers or trailers are part of the transmission format.
- Account number is specified, if transit routing number is part of the transmission format.
- Invoice1-8 are either valid or left blank.
- Installment1-8 are either valid installment numbers or are left blank.
- Invoice, debit memo, credit memo, on-account credit, or chargeback number derived from the matching number doesn't belong to a receipt.
- Transaction number is entered where an application amount is specified.
- Sum of all of the **Amount Applied** columns for a receipt doesn't exceed the remittance amount.
- Customer account number is valid.
- Customer account number and MICR number both reference the same customer, if both are provided.
- Receipt date is specified.
- Receipt method is valid.
- Currency is valid.

## Overflow Level Validations

Lockbox validates the overflow records to ensure that overflow information corresponds to the transmission format. The following attributes are validated:

- Batch name is specified, if either batch headers or trailers are part of the transmission format.
- Lockbox number is specified, if either the batch header or trailer is not specified and the transmission format includes the lockbox number.
- Item number is specified and matches a receipt record.

- Overflow indicator is specified, unless it is the last overflow record.
- Overflow sequence is specified.
- Invoice1-8 are either valid or are left blank.
- Installment1-8 are either valid installment numbers or are left blank.
- Transaction number derived is entered where an application amount is specified.

 **Note:** For Receipt and Overflow validations of Invoice1-8: If you are using matching numbers and a receipt record indicates that multiple transactions are to be paid by this receipt, lockbox assumes that all of the transactions are the same document type, such as invoices, sales orders, or purchase orders. For example, if the first 2 transactions are invoices, lockbox successfully matches them with this receipt. However, if the next transaction is not an invoice, lockbox either imports the remaining receipt amount as Unidentified or rejects the entire receipt, depending on the lockbox definition. If lockbox imports the remaining receipt amount as Unapplied, then Receivables retains the invalid matching numbers.

## Customer Validations

Lockbox can either validate customer data based on the following attributes or mark the receipt as Unidentified if no match is found:

- Customer account number is valid.
- MICR number is valid.
- Bill-to customer is from a matched invoice, if matching is enabled.

## Currency Validation

Receivables lets you process receipts in multiple currencies. If you pass the currency, conversion type, and receipt date, lockbox attempts to determine the conversion rate. If lockbox is unable to determine the conversion rate, the receipt will fail validation.

# Lockbox Transmission Formats

Transmission formats specify how data in a lockbox bank file is organized for import into the Oracle Fusion Receivables interface tables. The transmission format is used by the validation program to ensure that data is transferred correctly.

## Transmission Formats

Receivables provides five transmission formats. You can define new transmission formats based on the formats provided by Receivables.

You use an SQL\*Loader control file to import data from bank files to Receivables. If you define a different transmission format or edit the existing Default or Convert formats, you must edit the SQL\*Loader control file before you can import data into Receivables.

### Example (arxmpl.ctl)

This format contains an example of lockbox header information, receipt records, and overflow receipt records.

### Default (ardeflt.ctl)

This is a standard BAI (Bank Administration Institute) format used by most banks.



### **Convert (arconv.ctl)**

This format is used for transferring payment information from other systems.

### **Cross Currency (arxcrr.ctl)**

This format is used for importing cross-currency receipts.

### **Zengin (arzeng.ctl)**

This format is used to import bank files in the Japanese Zengin format.

## Lockbox Transmission Format Record Types

Enter the record types to include in a lockbox transmission format. Use the **Identifier** column to uniquely identify each record type.

Your bank file might not contain all of these record types. You should define your transmission format to only include the record types you actually use.

### Record Types

#### **Batch Header**

A Batch Header marks the beginning of a specific batch. Batch Headers usually contain information such as batch number, deposit date, and lockbox number.

#### **Batch Trailer**

A Batch Trailer marks the end of a specific batch. Batch Trailers usually contain information such as batch number, lockbox number, batch record count, and batch amount.

#### **Lockbox Header**

A Lockbox Header marks the beginning of a specific lockbox. Lockbox Headers usually contain information such as destination account and origination number.

#### **Lockbox Trailer**

A Lockbox Trailer marks the end of a specific lockbox. Lockbox Trailers usually contain information such as lockbox number, deposit date, lockbox amount, and lockbox record count.

#### **Overflow Receipt**

An Overflow Payment usually contains invoice information for a specific payment, such as batch number; item number; sequence number; overflow indicator; invoice, debit memo, or chargeback number; and debit item amounts. Receivables combines the overflow and payment records to create a logical record to submit payment applications.

#### **Receipt**

A Receipt usually contains payment information such as MICR number, batch number, item number, check number, and remittance amount.

### **Service Header**

Service Header records contain general information about the transmission.

### **Transmission Header**

A Transmission Header marks the beginning of a specific data file. Transmission Headers usually contain information such as destination account, origination number, deposit date, and deposit time.

### **Transmission Trailer**

A Transmission Trailer marks the end of a specific data file. Transmission Trailers usually contain information such as total record count.

## Lockbox Transmission Format Field Types

Enter the lockbox transmission field types to use to identify the characteristics of each lockbox transmission record type.

You specify the size, order, and format of each transmission record. The lockbox transmission program only validates the fields that you define in your transmission format. The transmission format must be fully compatible with how you organize data in your lockbox file.

### Field Types

#### **Account**

Your customer bank account. The bank account number and the transit routing number make up your customer MICR number.

#### **Alternate Name**

The alternate name for this customer.

#### **Amount Applied 1 to 8**

The amount applied to each invoice, debit memo, or chargeback. Each payment or overflow payment record can accommodate up to eight debit item numbers. For cross-currency applications, this is the amount to apply in the transaction currency.

#### **Amount Applied From 1 to 8**

Used for cross-currency receipt applications, this is the amount applied to each transaction in the receipt currency. Each payment or overflow payment record can accommodate up to eight debit item numbers.

#### **Attribute 1 to 15**

Use attributes to enter descriptive flexfield segments. Attributes can only be assigned to payment records.

#### **Bank Transaction Code**

A code defined for each account that is used by your bank to uniquely identify the kind of transaction in a bank statement (for example, debit, credit, void). This is also used by Oracle Fusion Cash Management to determine the receipt effective date.

**Batch Amount**

The total receipt batch amount for a specific bank batch.

**Batch Name**

The name of the batch for a specific bank batch.

**Batch Record Count**

The total number of payment records in a specific bank batch. The total number of all batch record counts equals the Lockbox Record Count. This doesn't include overflow payments, headers, or trailers.

**Billing Location**

Your bank transmits the billing location of the payment. You must only specify the field name and the field positions that the billing location occupies in the transmitted data file.

**Comment**

Any comments you want to associate with this transmission.

**Conversion Rate**

The conversion rate associated with this payment, if you are using lockbox to import foreign currency receipts.

**Conversion Type**

The conversion type used to convert a foreign currency receipt to the ledger currency.

**Currency Code**

The currency of the payment. For cross-currency payments, you can also enter the Invoice Currency Code. If you don't enter a value in this field, lockbox derives the currency from the information that is provided in the Amount Applied 1 to 8 and Amount Applied From 1 to 8 fields.

**Customer Bank Branch Name**

The name of the customer bank branch.

**Customer Bank Name**

The name of the customer bank.

**Customer Number**

The identification number of the customer who submitted a payment.

**Customer Reason 1 to 8**

The customer reason why a payment shows a discrepancy.

**Customer Reference 1 to 8**

Customer comments about this payment.

**Deposit Date**

The date the bank receives and deposits the customer payment.

**Deposit Time**

The time the bank receives and deposits the customer payment.

**Destination Account**

Your business bank account. Your business may have more than one bank account.

**Effective Date**

The date on which the bank determines a customer balance to apply interest (used by Cash Management).

**Invoice 1 to 8**

The invoices, debit memos, and chargebacks to which you apply your payment. Each payment or overflow payment record can accommodate up to eight debit item numbers.

**Invoice 1 to 8 Installment**

The installment number for this invoice.

**Invoice Currency Code 1 to 8**

The currency of the transaction. This field is used for cross-currency receipt applications. This field is optional.

**Item Number**

A sequence number that your bank assigns to a specific payment. This number associates an invoice with a receipt.

**Lockbox Amount**

The total payment amount in a specific lockbox.

**Lockbox Batch Count**

The total number of bank batches in a specific lockbox.

**Lockbox Number**

The identification number for a specific lockbox.

**Lockbox Record Count**

The number of payment records in a specific lockbox. This doesn't include overflow payments, headers, or trailers.

**Matching Date 1-8**

The dates to use to match receipts with transactions, if you are using the **Match on Corresponding Date** option for this lockbox.

### **Origination**

The bank origination number provided by your bank. This number uniquely identifies the bank branch that sends you lockbox information.

### **Overflow Indicator**

Indicates whether there are any additional overflow records for this payment.

### **Overflow Sequence**

A sequence number that your bank assigns to each overflow payment.

### **Receipt Method**

The receipt method associated with this lockbox.

### **Payment Number**

The identification number of a payment. For example, a check number.

### **Receipt Date**

The date your customer made a payment.

### **Record Identifier**

A number that identifies the kind of transmission record.

### **Remittance Amount**

The amount of a payment.

### **Remittance Bank Branch Name**

The name of the bank branch from which this payment originated.

### **Remittance Bank Name**

The name of the bank from which this payment originated.

### **Status**

The status of this payment.

### **Total Record Count**

The total number of transmission records in a bank file. This includes headers, trailers, payments, and overflow records.

### **Trans to Receipt Rate 1 to 8**

The conversion rate used to convert the receipt amount from the receipt currency to the transaction currency. This field is used for cross-currency receipt applications, when the receipt and transaction currencies don't have a fixed rate. If the currencies have a fixed rate, this field is optional (in this case, lockbox derives the rate to use).

### Transit Routing Number

The number that uniquely identifies your customer bank. The transit routing number and the customer bank account number make up the customer MICR number.

### Transmission Amount

The total amount of payments for a bank file.

## FAQs for Define Transmission Formats for Lockbox

### Can I reformat the transmission amount?

Yes, by using the **Format Amount** field setting. If you set this field to Yes, lockbox rounds the format amount to the same degree of precision and the same number of decimal places as your ledger currency.

For example, if your ledger currency is USD (precision = 2) and you set this option to Yes, a value of 50000 in the bank data file is formatted as 500.00. If you set the option to No, then this value is not formatted and would appear as 50000.

### What's an overflow record?

An overflow record stores additional receipt information that can't fit on the receipt record. This is typically additional invoice numbers and the amount of the receipt to apply to each invoice.

Each overflow record must have a receipt record as a parent. If there are multiple overflow records for a receipt record, each overflow record is assigned an overflow sequence.

## Define AutoMatch Rule Sets

### AutoMatch Recommendations: How They Are Calculated

During receipt processing, Oracle Fusion Receivables applies receipts to transactions based on the transaction information provided. In cases where the transaction information provided doesn't exactly match the transaction numbers on record, the AutoApply process attempts to find as close a match as possible and either apply the receipt to the transaction automatically or present one or more transactions as recommendations for manual receipt application.

In like manner, during lockbox processing it may happen that a lockbox contains incomplete or inaccurate customer information. The AutoApply process attempts to match a customer to a receipt and present one or more customers as recommendations for the receipt.


The AutoMatch rule set provides information that is used by the AutoApply process to complete the process of applying receipts to transactions. The settings in the AutoMatch rule set provide these recommendations:

- Customer recommendations: The matching process recommends customers for lockbox receipts that have invalid customer information.
- Transaction recommendations: The matching process recommends one or more transactions for receipt application for both lockbox and manual receipts.

## Settings That Affect AutoMatch Recommendations

Recommendations are based on matching threshold levels defined in the AutoMatch rule set. These threshold settings determine the percentage level necessary to consider a customer or transaction for receipt recommendation:

- Customer Recommendation Threshold: The qualifying percentage necessary to add customer information to a receipt. If the calculated score for a customer account number is above this threshold, then the AutoApply process adds this customer information to the receipt.
- Minimum Match Threshold: The qualifying percentage necessary to recommend a transaction for receipt application. If the calculated score for one or more transactions is above this threshold, the AutoApply process recommends the transactions for receipt application, in order of the highest percentage match.

 **Note:** The minimum match threshold must be less than the customer recommendation threshold and the combined weighted threshold.

- Combined Weighted Threshold: The qualifying percentage necessary for the AutoApply process to apply a receipt to a transaction automatically. This percentage is the sum of the qualifying percentages defined for the supplied customer information, transaction information, and actual transaction amount that is considered for receipt matching.
- Days of Closed Invoices Threshold: Determines which closed transactions to include in the AutoMatch process. All transactions that were closed on or before the number of days provided as the threshold value are considered for application or recommendation.

## How AutoMatch Recommendations Are Calculated

The threshold qualifying percentages defined in the AutoMatch rule set are compared to the resulting scores of each customer account number or transaction number that is analyzed by the matching process.

The matching process derives a recommendation in this way:

1. If applicable, remove characters and spaces from the number as defined by the AutoMatch rule set string handling.
2. Apply the formula (Levenshtein algorithm) to the resulting string to obtain the score. This formula is:  
$$1 - (\text{number of changes required to make the recommended string match the provided string} / \text{length of the larger string})$$
3. Compare the resulting score to the applicable threshold.

## AutoMatch Recommendation: Example

The transaction number 10010 is provided by lockbox for a receipt application. This number doesn't exist, but Receivables finds the number AR10001. The recommendation for this number is calculated in this way:

1. The AutoMatch rule set string handling settings indicate that the first two characters are to be removed from a string under consideration.  
Receivables removes AR, leaving the number 10001.
2. It will take one transposition for 10001 to match 10010. Therefore, the score for this match is  $(1 - 1/5) = 80\%$ .

3. The 80% score exceeds the Combined Weighted Threshold value of 70%, so the receipt is automatically applied to transaction AR10001.

#### Related Topics

- [When do I use days to AutoApply a receipt?](#)

## AutoMatch Combined Weighted Thresholds: Explained

When you define an AutoMatch rule set, use the weighted threshold fields to qualify the percentage necessary to allow for an automatic receipt application. This qualifying percentage considers the importance, or weight, that the AutoApply process should give to the three main components of a transaction: customer information, transaction information, and transaction amount.

### Using the Combined Weighted Thresholds

Enter the weight, as a percentage, that the AutoApply process should give to the successful matching of customer, transaction, and amount information, when considering a transaction for automatic receipt application.

The total of the three percentages must equal 100. In most cases, you would not give equal weight to each component of a transaction. That is, you would not enter 33, 33, 34 as the three weighted values. In order for AutoApply to apply a receipt to a transaction automatically, the weighted thresholds you enter should reflect your standard business practices.

Enter these weighted thresholds:

- **Customer Weight:** Enter the weight to give to matching the customer information on the transaction.
- **Transaction Weight:** Enter the weight to give to identifying the correct transaction, either by correct matching or by the Match Receipts By rule document reference.
- **Amount Weight:** Enter the weight to give to matching the open balance amount. Because the parts of a transaction amount can play a part in this decision, you can use the Amount Weight Exceptions section to provide additional granularity to the open balance amount considered.

The AutoApply process calculates the percentage score for each match between customer, transaction, and amount. It then derives a final score for each weighted threshold as a percentage of the weighted threshold values, and adds these results to obtain a final score. This final score is then compared against the combined weighted threshold value.

For example, you enter these weighted threshold values:

Weight	Value
Combined Weighted Threshold	75%
Customer	20%
Transaction	70%
Amount	10%

During lockbox processing, a transaction is presented for receipt application with these details:

- Customer account number: 1001



- Transaction number: 10010
- Amount due remaining : 127
- Amount after discount: 120.65
- Tax remaining: 20
- Freight Remaining: 7
- Unearned Discount: 6.35


Lockbox presents a receipt for application with these details:

- Customer account number: 1005
- Reference number: 1001
- Reference amount: 120.65

AutoApply performs these calculations:

- Customer match: The match between account number 1001 and 1005 has a calculated score of 75%  $[(1-1/4)*100]$ . Since the customer weighted threshold value is 20%, this provides a final score of 15%.
- Transaction match: The transaction match has a calculated score of 80%  $[(1-1/5)*100]$ . Since the transaction weighted threshold value is 70%, this provides a final score of 56%.
- Amount match: The transaction amount match is 100%. Since the amount weighted threshold value is 10%, this provides a final score of 10%.

The sum of the three final scores is 81%, which is greater than the Combined Weighted Threshold value of 75%. Therefore, AutoApply automatically applies the receipt to the transaction.

 **Note:** If the score were below the Combined Weighted Threshold value but above the Minimum Match Threshold value, then AutoApply would generate recommendations for user review. If the score were below the Minimum Match Threshold value, then AutoApply would not generate recommendations.

## AutoMatch Amount Weight Exceptions: Explained

When you define an AutoMatch rule set, the values you enter in the **Amount Weight Exceptions** section qualify the meaning of the open balance amount that is considered by the **Amount Weight** threshold.

### Using the Amount Weight Exceptions

Enter threshold percentages to qualify the weight to give to different portions of the open balance amount on a transaction.

Enter these amount weight exception thresholds:

- **Net of Tax Weight:** Enter the weight to give to the total transaction amount net of tax but including freight.
- **Net of Tax and Freight Weight:** Enter the weight to give to the total transaction amount net of tax and freight.
- **Net of Freight Weight:** Enter the weight to give to the total transaction amount net of freight but including tax.
- **Unearned Discount Weight:** Enter the weight to give to transaction amounts that include an unearned discount.

If an exact match exists between the receipt amount and transaction amount, AutoApply uses the **Amount Weight** threshold value without reference to the amount weight exceptions.

If an exact match doesn't exist between the receipt amount and transaction amount, AutoApply looks for the best match among all of the amount weight exceptions to derive a percentage score.

For example, you enter these amount weight exception threshold values:

Weight	Value
Net of Tax Weight	70%
Net of Tax and Freight Weight	70%
Net of Freight Weight	80%
Unearned Discount Weight	60%

During lockbox processing, a transaction is presented for receipt application with these details:

- Customer account number: 1001
- Amount due remaining: 127
- Amount after discount: 120.65
- Tax remaining: 20
- Freight Remaining: 7
- Unearned Discount: 6.35

Lockbox presents a receipt for application with these details:

- Customer account number: 1005
- Reference amount: 120

Because no exact match exists between the transaction amount and the receipt amount, AutoApply looks for the closest match among the amount weight exceptions to derive an amount weight. There is an exact match between the receipt amount and the transaction amount net of freight ( $127 - 7 = 120$ ). AutoApply therefore uses the **Net of Freight Weight** threshold value of 80% as the calculated score for the Amount Weight threshold.

## String Handling: How It Works

The **String Handling** section of the Create and Edit AutoMatch Rule Set pages provides rules that assist in the search for transaction matches against the Match Receipt By document reference used by AutoApply.

The rules indicate what part of a string to remove in order to compare the resulting stripped number against the matching numbers provided during receipt processing. Matching recommendations for the string are processed according to the weighted threshold values of the AutoMatch rule set.

You can define rules for transaction strings and remittance strings.

### Settings That Affect String Handling

Use the string handling settings to indicate what part of a string to remove before string comparison. This is typically an alphanumeric prefix or suffix, or a string of zeros used to pad numbers to equal length.

Enter these settings:

- String Location: Indicate whether to remove characters and digits from the Front or Back of the string.

- String Value: Indicate the type of characters and digits to remove: Zero, Empty Spaces, or Any. Any includes zeros, spaces, and any character, digit, or special character.
- Number of Characters: Indicate the number of places to remove.

## How String Handling Is Used

If the string handling settings for a transaction number are:

- String Location: Front.
- String Value: Any.
- Number of Characters: 5.

Then the string ABC: 10044 is converted to 10044. The string handling process removed the first 5 characters from the string: the alphanumeric characters ABC, the colon (:), and the space.

If the string handling settings for a transaction number are:

- String Location: Back.
- String Value: Zero.
- Number of Characters: 3.

Then the string:

- ABC: 10044000 is converted to ABC: 10044.
- 985660000 is converted to 985660.
- 9856600000000003 is not processed. This is because AutoApply looks for zeros at the end of the string but finds a number instead.

## Define Application Exception Rule Sets

### Exception Rules Conditions and Actions: Explained

Use the Exception Rules section of the Create Receivables Application Exception Rule Set page to indicate how to process each overpayment and underpayment condition.

Each exception rule consists of:

- A condition.
- The amount and percentage that applies to the condition.
- The action to take when the condition arises.

The **Action** field contains the Receivables activities that you have defined for Adjustment, Receipt Write-off, or Refund. The underpayment or overpayment amount is accounted for in the general ledger accounts belonging to the applicable Receivables activity.

The **User Review Required** option indicates whether the action is processed automatically or requires manual review and approval.

For example, the exception rule:

**Over Payment >= 100 Refund**

means that if a receipt application overpays a transaction by \$100 or more, then the customer should receive a refund.

The exception rule:

**Under Payment < 5 Write-off**

means that if a receipt application results in an underpayment of less than \$5, then the remaining amount can be written off.

You can use the **Percent** field with the **Amount** field to further refine the scope of a condition. If you use both fields, then both conditions must be met in order to apply the rule.

For example, the exception rule:

**Under Payment < 5 5% Write-off**

means that if a receipt application results in an underpayment that is both less than \$5 and also less than 5% of the open balance, then the remaining amount can be written off. In this case, if a \$10 open balance has a \$4 underpayment, the above rule doesn't apply, because \$4 is 40% of the open balance. If the \$4 underpayment were for an open balance of \$100, then the rule does apply, because \$4 is 4% of the open balance.

*Related Topics*

- [Approval Limits Document Types](#)
- [Recommendations for Receipt Application: How It Works](#)

## FAQs for Define Application Exception Rule Sets

### When do I use an application exception rule set?

Use an application exception rule set to manage remaining amounts after lockbox processing.

After lockbox processes and applies receipts, the AutoApply process uses the application exception rule set to determine how to manage over and under payments:


- If there is an overpayment, the application exception rule indicates whether to refund the amount to the customer, place the amount on account, write off the amount, or leave the amount unapplied.
- If there is an underpayment, the application exception rule indicates whether to allow write off of the remaining open balance amount on the transaction.

### What's the difference between the AutoMatch rule set and the application exception rule set?

During lockbox processing, the AutoMatch rule set provides recommendations for matching receipts to transactions based on the transaction information provided. The AutoApply process attempts to match receipts to transactions and either apply receipts automatically or present for manual processing transaction recommendations for receipt application.

If the AutoApply process is able to apply the receipt to all of the transaction references automatically, then it uses the details of the application exception rule set to refund overpayment to the customer, if there is one, or process underpayment according to the exception rules.

If AutoApply can't apply the receipt to all of the transaction references automatically, then the remaining transaction recommendations are presented, if available, for manual processing and the application exception rule set is not used.

 **Note:** If a receipt doesn't have a transaction reference, the application exception rule set for overpayments is not used.

## Define Customer Paying Relationship Assignments

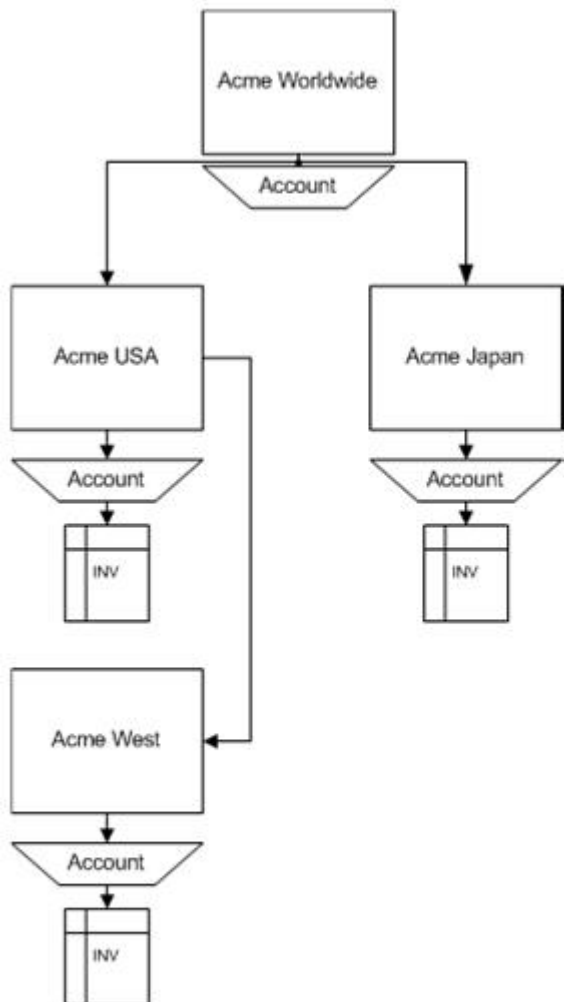
### Customer Hierarchies and Paying Relationships: How They Work Together

The customer hierarchy describes the structure of an enterprise or portion of an enterprise. Enterprise structures are defined and maintained in the FND\_HIER tables. There are three predefined hierarchies: Customer hierarchy, Trading Community party hierarchy, and Duns and Bradstreet hierarchy.

You assign a paying relationship to a hierarchy to indicate how the parties in the hierarchy are able to manage each other's customer payments. There are two paying relationships:

- **Pay Any:** Any party within the relationship can pay for the accounts of any other party within the relationship.
- **Pay Below:** A parent-child relationship, whereby each party can pay for its own transactions and the transactions of all parties that are lower in the hierarchy (children, grandchildren, and so on).

This diagram illustrates the customer hierarchy in Acme Corporation:



## Pay Any Paying Relationship

If this Acme Corporation hierarchy is assigned a Pay Any paying relationship, then:

- Acme Worldwide can pay for Acme USA, Acme Japan, and Acme West.
- Acme USA can pay for Acme Worldwide, Acme Japan, and Acme West.
- Acme Japan can pay for Acme Worldwide, Acme USA, and Acme West.
- Acme West can pay for Acme Worldwide, Acme USA, and Acme Japan.

## Pay Below Paying Relationship

If this Acme Corporation hierarchy is assigned a Pay Below paying relationship, then:

- Acme Worldwide can pay for Acme USA, Acme Japan, Acme West, and its own transactions.
- Acme USA can pay for Acme West and its own transactions.
- Acme Japan can pay for its own transactions.

### *Related Topics*

- [Party Hierarchy: Explained](#)

## FAQs for Define Customer Paying Relationships

### What's the difference between an account relationship and a paying relationship?

An account relationship is a flat relationship between two customer accounts only. In an account relationship, either one account can pay for the transactions of another account (one-way) or both accounts can pay for the transactions of each other (reciprocal).

A paying relationship makes use of a hierarchical structure within an enterprise to allow all corresponding accounts and transactions that are associated with one party to be accessible to other parties in the structure. In a paying relationship either one account can pay for the transactions of accounts lower in the hierarchy (Pay Below) or all accounts anywhere in the hierarchy can pay for the transactions of any other party (Pay Any).

### What happens if a customer hierarchy type is deactivated?

Then the active paying relationship assignments for this hierarchy are no longer valid. You must enter an end date for each related paying relationship assignment that is on or before the date the customer hierarchy type was deactivated.





# 5 Configure Payment System Connectivity

## Validations: Critical Choices

Validations are rules that ensure that transactions are valid before they are printed or submitted electronically to payment systems. You use validations to ensure that disbursement transactions, such as invoices, payments, and payment files meet specific conditions before they can be paid. You can assign validations to payment methods and payment formats. A validation can be executed at the document payable, payment, or payment file level.

In payment processing, it's critical that payment files sent to payment systems and financial institutions are valid and correctly formatted. If this is not done, the payment process is slowed, which results in additional time and cost due to problem resolution. Oracle Fusion Payments helps you achieve straight-through processing by ensuring that payment-related details are valid. To assign validations, you can select from the following options:

- Assigning validations
- Creating user-defined validations
- Selecting from a predefined library of validations

The following table shows the objects you can validate and when validations are executed for the applicable setup.

Object	Payment Method-Driven Validations are Enforced When...	Payment File Format-Driven Validations are Enforced When...
Document Payable	The invoice is saved in the source product.  The invoice installment is selected for payment.	The invoice installment is selected for payment.
Payment	The payment is created by building related documents payable together.	The payment is created by building related documents payable together.
Payment File	Not applicable.	The payment file is created.

## Assigning Validations

You can assign user-defined validations to any:

- Payment method
- Payment file format

You can assign a validation to whichever object drives the requirement for validation. For example, if your bank format requires a limited number of characters in a specific field, you can assign that validation to the bank format. By doing this, you ensure that the validation is enforced only when applicable. However, if you want to enforce a general validation that isn't specific to the payment method or format, you can consider timing in your decision.

Payments always validates as early as possible for a given object and setup. Document payable validations that are associated with payment methods are enforced earlier in the process than those associated with formats. If you want validation failures handled by the same person who is entering the invoice, you can associate the validation with the payment

method. This method is ideal for business processes where each person has full ownership of the items entered. However, if you want focused invoice entry with validation failures handled centrally by a specialist, you can associate the validation with the format. This method is ideal for some shared service centers.

## Creating User-Defined Validations

A user-defined validation explicitly specifies the object to which the validation applies:

- Document payable
- Payment
- Payment file

User-defined validations are basic validations that correspond to simple operations. These validations can be used as components, or building blocks, to build more complex validations. They enable you to validate, for example, the following conditions:

- Length of a value. Example: Payment Detail must be fewer than 60 characters for your bank-specific payment file format.
- Whether a field is populated. Example: Remit to bank account is required when payment method is Electronic.
- Whether content of a field is allowed. Example: Currency must be USD when using your domestic payment file format.

## Selecting From a Predefined Library of Validations

Payments provides a library of predefined validations. You can associate these predefined validations with any payment method or payment file format you create. Many of the payment formats provided by Oracle have predefined validations associated with them by default.

Predefined validations are groups of individual validations that are together for a specific purpose. Many of the predefined validations that you can associate with payment formats are country-specific. Predefined validations cannot be modified, although some have parameters you can set to define specific values.

## Setting Up Formats: Explained


Setting up formats is a mandatory task in Oracle Fusion Payments. A format is a disbursements or a funds capture data file to which an Oracle Business Intelligence Publisher (Oracle BI Publisher) template is applied. Oracle BI Publisher templates contain formatting attributes that format data files. Formatted outputs include printed checks, electronically transmitted payment files, settlement batches, and reports.

The purpose of setting up formats is to enable payment systems, financial institutions, or countries to understand your company's messages, given their specific formatting requirements for disbursements or funds capture transactions. Inbound messages come from a payment system or financial institution to your company. Outbound messages leave your company to your payment system or financial institution.

Setting up formats involves the following actions:

- Using Oracle BI Publisher templates
- Using data extracts
- Using the identity format
- Considering best practices

- Setting up formats
- Understanding associations between format entities
- Assigning validations to formats

 **Note:** Before you can set up formats, you must set up the corresponding templates in Oracle BI Publisher. For more information on setting up templates, see Oracle Fusion Middleware Report Designer's Guide for Oracle Business Intelligence Publisher at [http://docs.oracle.com/cd/E25054\\_01/bi.11111/e13881/toc.htm](http://docs.oracle.com/cd/E25054_01/bi.11111/e13881/toc.htm).

## Using Oracle BI Publisher Templates

Each Payments format corresponds to one Oracle BI Publisher template. BI Publisher templates specify exactly how formatted output is to be generated. BI Publisher templates can also be used to generate fixed-position, machine-readable files through BI Publisher's eText functionality.

## Using Data Extracts

Each disbursement or funds capture format is also associated with a disbursement or funds capture Payments' data extract. Each data extract contains transactional data. Oracle BI Publisher templates use data extracts to format transactional data. Transactional data is extracted from Payments' or Oracle Fusion Receivables' transactional tables.

For a disbursements extract, data comes from:

- Payments
- Payment files
- Documents payable tables

For a funds capture extract, data comes from:

- Funds capture transactions
- Settlement batches
- Receivables transactions

For more information on Payments' XML extracts, see How To Generate and View Fusion Payments XML Extract , Doc ID 1428249.1, on My Oracle Support.

## Using the Identity Format

The Identity format outputs the XML extract provided by Payments. It's intended for diagnostic purposes, but you can also use it to understand how extract fields are populated from transactional and setup entities. This is especially helpful if you intend to create complex customizations using other templates.

The Identity format is an Oracle BI Publisher template called IBY\_IDENTITY. It's part of the Funds Capture Authorization and Settlement report. If you want to use the Identity format for a disbursements report, you must download the RTF template and upload it as part of the intended disbursements report. Then, you can set up a customized format in Payments using the newly created format with a payment process profile or a funds capture process profile, and examine the XML output.

## Considering Best Practices

Before setting up formats, find out what payment formats your processing bank supports. Consider using standards-based payment formats that can be used with multiple banks:

- EDIFACT formats, such as PAYMUL, MT100, and MT103
- NACHA formats, such as Generic, CCD, or PPD

## Setting Up Formats

To set up a format, select a predefined format type on the Manage Formats page. Navigate to: **Setup and Maintenance > All Tasks > Search field: Task Lists and Tasks > Name field: Payments > Search > Define Invoicing and Payments Configuration > Configure Payment System Connectivity > Manage Formats > Go to Task > Manage Formats page.**

The format type you select specifies:

- Type of message that is created
- Oracle BI Publisher template that is used to format the data file
- Data extract that is used

On the Create Format page, associate an Oracle BI Publisher template with the format type you selected.

## Understanding Associations Between Format Entities

The following table shows the association between format types, templates, and data extracts.

Format Types	Oracle BI Publisher Template	Data Extracts
Disbursement Payment File Formats	Disbursement Payment File Formats	Disbursement Extract
Disbursement Positive Pay File Formats	Disbursement Positive Pay Formats	Disbursement Positive Pay Extract
Disbursement Separate Remittance Advice Formats	Disbursement Separate Remittance Advice Formats	Disbursement Extract
Disbursement Accompanying Letter Formats	Disbursement Accompanying Letter Formats	Disbursement Extract
Disbursement Payment Process Request Status Report Formats	Disbursement Payment Process Request Status Report	Disbursement Payment Process Request Extract
Disbursement Payment File Register Formats	Disbursement Payment File Register	Disbursement Extract
Funds Capture Settlement Format	Funds Capture Authorization And Settlement Formats	Funds Capture Extract
Funds Capture Accompanying Letter Formats	Funds Capture Accompanying Letter Formats	Funds Capture Extract

Format Types	Oracle BI Publisher Template	Data Extracts
Funds Capture Payer Notification Formats	Funds Capture Payer Notification Formats	Funds Capture Extract

## Assigning Validations to Formats

After you create a format, you can optionally assign predefined or user-defined payment validations to it on the Edit Format page. Validations ensure that disbursements or funds capture transactions execute according to specified conditions.


## Transmission Protocol: Explained

Computers use transmission protocols to communicate with each other across a network. To transmit data, such as payment files from Oracle Fusion Payments to a payment system, the implementor defines protocols that the payment system can receive.

Payments offers industry-standard transmission protocols, such as FTP, HTTP, and HTTPS, predefined. They are composed of the following:

- A code entry point, which the payment system servlet uses to accomplish transmission
- A list of parameters, such as network address and port, for which the transmission configuration must supply values
- Transmission protocol entry points, which are independent of payment servlets and may be called from the Payments engine

While the transmission protocol defines which parameters are expected in the communication, the transmission configuration defines what value is supplied for each parameter. Transmission configurations and payment systems are associated on the funds capture process profile for funds capture or on the payment process profile for disbursements.

 **Note:** This note applies only to on-premises customers, and never to Oracle Cloud customers. The preferred file-based transmission protocol is Secure File Transfer Protocol (SFTP). File Transfer Protocol (FTP) is unsecured and should only be used to meet legacy third-party requirements. FTP must be used over a secure link such as a virtual private network (VPN) or a leased line with data link level encryption enabled.

## Transmission Configuration: Explained

A transmission configuration is a group of specific transmission details. A transmission configuration defines a value for each parameter in a transmission protocol. The values in a transmission configuration are specific to one payment system or financial institution.

For example, suppose a transmission protocol requires parameter values for a Socket IP Address and a Socket Port Number. Your payment system that accepts that protocol will give you the values that it expects to receive for these parameters. You enter the applicable values in the **Socket IP Address** and **Socket Port Number** fields for the transmission configuration. The transmission configuration is then assigned to the payment system within the funds capture process profile for funds capture transactions or within the payment process profile for disbursement transactions.

To transmit files to your payment system, you can set up transmission configurations on the Create Transmission Configuration page. Navigate to: **Navigator > Directory > Setup and Maintenance > All Tasks tab > Search field:**

**Task Lists and Tasks > Name field: Payments > Search button > Define Invoicing and Payments Configuration > Configure Payment System Connectivity > Manage Transmission Configurations task > Go to Task > Manage Transmission Configuration page.**

#### Related Topics

- [Setting Up Transmission Configurations: Explained](#)

## Configuring Pretty Good Privacy (PGP) Encryption and Digital Signature for Outbound and Inbound Messages: Explained

You can secure both outbound and inbound messages using payload security. Payload security is the securing of payment files and other files using payment file encryption and digital signature based on the open PGP standard. You can update existing transmission configurations to use encryption and digital signature for your existing connectivity with banks


For outbound messages, Oracle Payments Cloud supports encryption and digital signature for:

- Payment files and positive pay files for disbursements
- Settlement batch files for funds capture

For inbound messages, the application supports decryption and verification of digitally signed encrypted files for:

- Funds capture acknowledgment files
- Bank statements

You can also secure payment data using secured transmission protocols, such as SFTP or HTTPS.

 **Note:** Oracle Applications Cloud supports decryption of payment files that are encrypted using version BCPG 1.45 or lower of the OpenPGP standard.

Configuring encryption and digital signature for outbound and inbound messages includes the following actions:

- Generating keys
- Setting up outbound transmission configuration
- Setting up inbound transmission configuration
- Uploading the bank-provided public key file
- Downloading the system-generated public key file

### Generating Keys

Encryption and digital signature verification requires a public key. Conversely, decryption and signing a digital signature requires a private key. The party who generates the public key and the private key, known as the key pair, retains the private key and shares the public key with the other party. You can generate or receive a public key subject to the agreement with your bank.

The following table provides typical generation details of the public and private key pair:

Key Pair Generated	Generates Outbound Messages from Payments	Generates Inbound Messages to Payments
PGP Public Encryption Key and PGP Private Signing Key	Bank	Deploying company
PGP Public Signature Verification Key and PGP Private Decryption Key	Deploying company	Bank

If you're generating the key pair, you can automatically generate them within Oracle Applications Cloud.

If you receive either the PGP Public Encryption Key or the PGP Public Signature Verification Key, you must import it into Oracle Applications Cloud using UCM.

## Setting Up Outbound Transmission Configuration

For outbound messages, such as payment files, positive pay files, and settlement batch files, you must:

- Encrypt your payment file using the bank-provided public encryption key.
- Optionally, sign the payment file digitally using the private signing key that you generate.

On the Create Transmission Configuration page, you can see the outbound parameters as shown in the following table.

Outbound Parameters	Description
PGP Public Encryption Key	<p>A key given to you by your bank that you use to encrypt your outbound payment file.</p> <p>To upload the bank-provided public encryption key, use UCM by navigating to Tools &gt; File Import and Export.</p> <p>Lastly, on the Create Transmission Configuration page for the PGP Public Encryption Key parameter, select the public encryption key file from the Value choice list.</p>
PGP Private Signing Key	<p>A key generated by you to digitally sign the outbound payment file.</p> <p>To generate the private signing key, select the Create option from the Value choice list for the PGP Private Signing Key parameter. The application:</p> <ul style="list-style-type: none"> <li>• Automatically generates the private signing key and links it to your transmission configuration.</li> <li>• Generates a public encryption key file that you can download from UCM and share with your bank. The bank uses your public encryption key file to verify the digital signature of the payment files that you transmit to the bank.</li> </ul>

## Setting Up Inbound Transmission Configuration

For inbound payment messages, such as acknowledgments and bank statements, you must:

- Verify the digital signature using the bank-provided public signature verification key.
- Decrypt the file using the private decryption key that you generate.


On the Create Transmission Configuration page, you can see the inbound parameters as shown in the following table.

Inbound Parameters	Description
PGP Public Signature Verification Key	<p>A key given to you by your bank that you use to validate the digital signature of inbound acknowledgment files or bank statements.</p> <p>To upload the bank-provided public signature verification key, use UCM by navigating to Tools &gt; File Import and Export.</p> <p>After you upload the bank-provided public signature verification key using UCM, you can select the key file from the Value choice list for the PGP Public Signature Verification Key parameter on the Create Transmission Configuration page. After you select the public signature verification key file, it's automatically imported.</p>
PGP Private Decryption Key	<p>A key generated by you to decrypt the inbound encrypted file. To generate the private decryption key, select the Create option from the Value choice list for the PGP Private Decryption Key parameter. The application:</p> <ul style="list-style-type: none"> <li>Generates the private decryption key and links it to your transmission configuration.</li> <li>Generates a public signature verification key file that you can download from UCM and share with your bank. The bank uses your public signature verification key file to encrypt acknowledgments and bank statements.</li> </ul>

## Uploading the Bank-Provided Public Key File

To upload, or import, either the bank-provided PGP Public Encryption Key or the PGP Public Signature Verification Key into Oracle Applications Cloud, perform the following steps:

1. Rename the bank-provided key file by including **\_public.key** as the suffix. Ensure that the key file name doesn't have any special characters other than the underscore.
2. Navigate to: **Navigator > Tools > File Import and Export**.
3. Import the bank-provided key file into account **fin/payments/import**.
4. Navigate to the Create Transmission Configuration page.
5. From the Value choice list for the applicable parameter, select the uploaded key file.


 **Tip:** The key name in the choice list is the same as the one you uploaded using UCM.

6. After you select the key and save the transmission configuration, the key is automatically imported into the Payments.

## Downloading the System-Generated Public Key File

To download the system-generated public key file from Payments to share with your bank, perform the follow steps:

1. On the Create Transmission Configuration page, select the Create option for the applicable parameter.
2. Click the **Save and Close** button.
3. Navigate to: **Navigator > Tools > File Import and Export**.
4. From the Account choice list, select **fin/payments/import** and search for the system-generated public key file.
5. Download the system-generated public key file.

 **Tip:** The file name is similar to the private key file that was generated and attached to the transmission configuration.



 **Note:** SSH (Secure Socket Shell) key-generation for SFTP two-factor authentication is generated by Oracle Support based on a service request.

## Testing the Transmission Configuration: Explained

The transmission configuration setup is used to transmit outbound payment files, settlement batch files, and positive pay files to your payment system or financial institution. It's also used to pull funds capture acknowledgment files. The setup captures various parameters, which may be different for different protocols. You can test your transmission configuration to confirm whether your setup of outbound and inbound transmission protocols is correct.

To confirm the accuracy of the setup, click the **Test** button on the Create or Edit Transmission Configuration page. The **Test** button is active only when the values associated with all the mandatory parameters are present. Typical transmission configuration parameters that are available to test include:

- Destination server URL
- Destination server IP address
- Destination server port number
- Remote file directory
- User credentials

Testing the transmission configuration setup includes reviewing return messages.

### Reviewing Return Messages

The **Test** button action contacts the destination server with the specified parameters, which results in a return message. The return message is a combination of functional text and raw message text. This occurs so both functional and technical users can benefit from the message. For example, suppose the remote file directory is invalid. The return message is: Incorrect remote directory (IBY\_Trans\_Test\_Remote\_Dir\_Fals).

The following table describes transmission configuration connection tests and test results with their associated return messages.

Test	Test Result	Return Message
Whether the connection is correct.	<ul style="list-style-type: none"> <li>• Connection is successful.</li> <li>• Remote file directory is present.</li> </ul>	Success (raw message)
Whether the remote file directory is valid.	<ul style="list-style-type: none"> <li>• Connection is successful.</li> <li>• Remote file directory isn't present or incorrect.</li> </ul>	Incorrect remote directory (raw message)
Whether user credentials are correct.	<ul style="list-style-type: none"> <li>• Connection is unsuccessful.</li> <li>• Destination IP address and port is correct.</li> <li>• Incorrect login credentials.</li> </ul>	Incorrect user credentials (raw message)
Whether IP address or port is correct.	<ul style="list-style-type: none"> <li>• Connection is unsuccessful.</li> <li>• Incorrect IP address or port.</li> </ul>	Incorrect destination server details (raw message)

Test	Test Result	Return Message
Whether two factor key file-based authentication is successful.	<ul style="list-style-type: none"> <li>• Connection is unsuccessful.</li> <li>• Unsuccessful key file-based authentication.</li> </ul>	Unsuccessful authentication (raw message)
Whether the destination server is responsive.	Destination server is down.	Destination server is not responding (raw message)
Not applicable.	Any other failure.	(raw message)

## Configuring a Communication Channel to a Payment System: Explained

To transmit payment information to or receive payment information from a payment system, you must configure the channel used to communicate with the payment system. This topic discusses the following concepts:

- Proxy server
- Secure sockets layer (SSL) protocol
- Unsecured protocols

### Proxy Server

Payments must communicate through a proxy server to its payment system. You can use the standard Java networking proxy system properties that are in the configuration files when Oracle Fusion Applications were initially set up. Alternatively, you can specify proxy settings for a transmission protocol on the Create Transmission Configuration page by entering a value for the **Proxy Host** parameter.

### Secure Sockets Layer Protocol

When Payments communicates with a payment system, the information exchanged may be sensitive information, such as credit card numbers. If the communication isn't secure, it poses a security risk. The security risk increases when communication to the payment system is across a public network, such as the Internet.

To set up a payment system servlet with a secure sockets layer, enable HTTPS on the application server where the servlet resides. If funds capture process profiles aren't defined for the payment system, change the **Transmission Servlet Base URL** on the Edit Payment System page to start with HTTPS. If funds capture process profiles are defined, change the value for the **Destination URL** parameter on the Edit Transmission Configuration page to start with HTTPS.

For secure communication with the payment system, ensure that the most current certification authority (CA) certificates are in the Fusion secure sockets trust store file. If Payments rejects the payment system's secure sockets certificate, you may have to insert the certificates manually.

The following table shows security actions that are applicable to Cloud and on-premise.

Action	Cloud	On-premise
Set up a payment system servlet with a secure sockets layer.	Not applicable.	Enable SSL on the servlet's application server.
Insert certification authority certificates.	File a service request with the help desk.	Enter a value in the <b>Value</b> field for the Wallet Location parameter on the Create or Edit Transmission Configuration page.

## Unsecured Protocols

The preferred file-based transmission protocol is Secure File Transfer Protocol (SFTP). File Transfer Protocol (FTP) is unsecured and should only be used to meet legacy third-party requirements. FTP must be used over a secure link such as a virtual private network (VPN) or a leased line with data link level encryption enabled.

## Setting Up a Payment System: Explained

In Oracle Fusion Payments, setting up a payment system is mandatory if your company wants to transmit electronic payments or funds capture transactions to a payment system or a bank. The payment system can be the bank where your company has its bank accounts or it can be a third-party processor or gateway that connects your company to a financial network. The purpose of setting up a payment system is to define the external organization that Payments uses to process your funds capture and disbursement transactions.

Payment systems are not required for printed disbursement payments, such as checks, but may be required for related services, such as a positive pay report.

Setting up a payment system includes the following actions:

- Selecting a gateway or processor payment system
- Considering best practices
- Defining a payment system
- Specifying payment system settings
- Understanding payment system accounts

## Selecting a Gateway or Processor Payment System

Payments supports both gateway and processor payment systems. A gateway is a service provider that acts as an intermediary between a first party payee and a payment processor. A processor is a service provider that interacts directly with banks and card institutions to process financial transactions. Examples of payment processors include Visa, MasterCard, and American Express.

Your choice of integrating with a gateway or a processor payment system is generally determined by your:

- Type of business
- Number of transactions per day
- Your bank

The following table shows the differences between gateway and processor payment systems.

Factors	Gateways	Processors
Connectivity and Security	Provide easy connection, often using SSL-based internet connectivity.	Provide more rigorous security, connectivity, and testing requirements.
Additional Fees	Charge additional fees, including per-transaction fees, beyond what processors charge.	Not applicable.
Volume of Transactions	Favor lower-volume merchants or merchants who are willing to pay a per-transaction premium for easier setup and connectivity.	Often favor higher-volume merchants who are willing to exert more effort for processor connectivity.
Online or Offline	Takes all transactions online.	<p>Allows authorizations in real-time and follow-up transactions, such as settlements and credits offline.</p> <ul style="list-style-type: none"> <li>• Offline transactions must be batched together and sent as a single request to the payment system.</li> <li>• All transactions other than authorizations are, by default, performed offline.</li> <li>• Offline transactions are sent when the next settlement batch operation is attempted.</li> </ul>

## Considering Best Practices


Before you set up a payment system, leverage your current banking or processing relationship. Determine whether your bank or processor can process transactions or has a partnership with a processor.

## Defining a Payment System

To define a payment system, navigate to: **Navigators > Tools > Setup and Maintenance > Search: Tasks field: Manage Payment Systems task > Go to Task > Manage Payment Systems page > Create button > Create Payment System page.**


When you set up a payment system on the Create Payment System page, specify the following:

- Types of payment instruments the payment system supports for funds capture transactions
- Data file formats and transmission protocols accepted by the payment system
- Settings required by the payment system
- Settings required by the tokenization provider if your company has enabled tokenization

 **Note:** Setting up a payment system may be required, even for transmitting a payment file offline by downloading it to your local drive and then e-mailing it to your payment system or bank. The payment system and payment system account setup capture several attributes, which are passed in the payment file or settlement batch message. Without these attributes, a payment file is invalid and rejected by bank.

## Specifying Payment System Settings


In the Settings Required by Payment System section, specify the settings that the payment system requires from each internal payer or payee. These settings can be used to identify the internal payer or payee as a client of the payment system or to provide other processing information. You can specify the type of data required for each setting and decide whether it is advisable to secure the setting's values by masking.

 **Tip:** The payment system generally provides the values for the payment system settings, which you enter as part of the payment system account.

## Understanding Payment System Accounts

You define your company's account with the payment system on the Edit Payment System Accounts page. The payment system account contains a value for each of the attributes required by the payment system. For example, the payment system may require a Submitter ID and Submitter Password to be included in any message sent to it.

You can configure a secure payment system account by entering a password. For secured settings, the values captured in the payment system account are masked.

 **Tip:** You can set up multiple payment system accounts in Payments for a single payment system.

## Payment System Account: Explained

A payment system account is an account identifier that is composed of values for parameters. The payment system provides you with the values that it requires to identify each payment or settlement batch file. Stored in the payment system account are values for settings and your company's identifiers. Your company can have multiple payment system accounts with a single payment system.

Payment system accounts are associated with the following setup objects:

- Internal payees
- Funds capture process profiles
- Payment process profiles

The following table shows setup objects and the action they perform relative to the payment system.

Setup Object	Setup Object Action
Payment system	<ul style="list-style-type: none"><li>• Tells Payments where to send the funds capture transactions.</li><li>• Tells Payments where to send the disbursements transaction.</li></ul>
Payment system account	Tells Payments how to identify itself to the payment system
Transmission configuration	Tells Payments how to transmit the transaction to the payment system

## Internal Payees

You can set up routing rules that are assigned to an internal payee. Routing rules specify which payment system account a transaction is transmitted to, based on the values of various transaction attributes.

If you don't need granular routing rules to determine which payment system account is the right one for a transaction, or if you want a fallback value should none of the routing rules apply, you can set up one default payment system account on each internal payee for each payment method.

## Funds Capture Process Profiles

The funds capture process profile tells Oracle Fusion Payments how to process a funds capture transaction and how to communicate with the payment system. A funds capture process profile is specific to one payment system and its payment system accounts.

For each payment system account that is enabled on the funds capture process profile, you can select up to three transmission configurations, one each for authorization, settlement, and acknowledgment.

## Payment Process Profiles

The payment process profile tells Payments how to process a disbursement transaction and how to communicate with the payment system to transmit a payment file or a positive pay file. When an electronic transmission is required, a payment process profile is specific to one payment system and its payment system accounts. For each payment system account that is enabled on the payment process profile, you select a transmission configuration.

## Auditing Payments Business Objects

You can audit specific business objects and attributes in Oracle Fusion Payments to monitor user activity and data changes. Auditing includes recording and retrieving information pertaining to the creation, modification, or removal of business objects. All actions the user performs on an audited business object and its attributes are recorded by the application and logged in a table. From the table, you can run a report that indicates changes made to attributes by users. For example, you can enable auditing on sensitive information, such as a customer bank account number and its bank account name and currency. Changes made by any user to the audit-enabled attributes are recorded and retrievable.

To enable auditing on Payments business objects, navigate to: **Setup and Maintenance > Search Tasks: Manage Audit Policies > Manage Audit Policies page > Configure Business Object Attributes button > Configure Business Object Attributes page > Application choice list: Payables.**

## Business Object: IBY\_SECURITY\_CONFIGURATION

Attribute name on the Configure Business Object Attributes page: SystemSecurityOptionsVO. System security options set to secure the application.

The IBY\_SYS\_SECURITY\_OPTIONS table stores the system security options that are used for funds capture and disbursements processing. The following table shows the fields that you can enable for auditing.

Audit Attribute	Table Column	Description	Audit-Enabled by Default
TokenProfileId	TOKEN_ PROFILE_ID	Payment system that is providing the tokenization services	No
Wallet File Location	SYS_ KEY_ FILE_LOCATION	Location of the system password file.	Yes
SecKeyHashUpd	SYS_ KEY_ HASH_UPG	Upgrade system key hash value.	No
Encryption Status	CC_ ENCRYPTION_ MODE	Encryption status of the credit card.	Yes
External Bank Account Masking Setting	EXT_ BA_ ENCRYPTION_ MODE	Encryption status of the external bank accounts.	Yes
Instrument Security Code	INSTR_ SEC_ CODE_ ENCRYPTION_ MODE	Security code encryption mode for credit card instruments.	Yes
Encryption Status	EXT_ BA_ ENCRYPTION_ MODE	Encryption status of the external bank account.	Yes
Credit Card Masking Setting	CREDIT_ CARD_ MASK_SETTING	Setting for credit card masking.	Yes
Number of Digits to Display	CREDIT_ CARD_ UNMASK_LEN	Number of digits exposed for masked credit card numbers.	Yes
External Bank Account Masking Setting	EXT_ BA_ MASK_SETTING	Setting for external bank account masking.	Yes
Number of Digits to Display	EXT_ BA_ UNMASK_LEN	Number of digits exposed for external bank account numbers.	Yes
Maximum Number of Uses	SUBKEY_ USE_MAXIMUM	Maximum use count for security subkeys.	Yes
Maximum Age in Days	SUBKEY_ AGE_MAXIMUM	Maximum age for security subkeys.	Yes

## Business Object: DisbursementSystemOptionAM

Attribute name on the Configure Business Object Attributes page: DisbursementEnterpriseWiseOptionVO. Enterprise-level options.

The IBY\_INTERNAL\_PAYERS\_ALL table stores payment processing and supplier-level defaulting attributes. Enterprise-level options are applied by default to all business units. The table also stores business unit-level attributes. You can override the

payment method defaulting and allow payee bank account override options at the business unit level. You can optionally configure business unit-level overrides.

The following table shows the fields that you can enable for auditing.

Audit Attribute	Table Column	Description	Audit-Enabled by Default
Payment Method Default Basis	PAYMENT_ METHOD_ AT_ PAYEE_ FLAG	Indicates whether the payment method defaults based on the payment method defaulting rules only or on the supplier-level payment method setup.	No
Review proposed payments after creation	REQUIRE_ PROP_ PMTS_ REVIEW_ FLAG	Indicates whether Payments stops payment creation to allow an administrator to review and modify proposed payments before grouping them into payment files.	No
Allow payee bank account override on proposed payments	ALLOW_ EXT_ ACCT_ OVERRIDE_ FLAG	Indicates whether the payee bank account can be overridden during proposed payment review.	No
Pay each document alone	EXCLUSIVE_ PAYMENT_ FLAG	Indicates whether each payable invoice is paid separately.	No
Save formatted payment file in database	SAVE_ PAYMENT_ FILE_ IN_ DB	Indicates whether Payments saves a copy of the payment file in the database.	Yes

Attribute name on the Configure Business Object Attributes page: DisbursementBusinessUnitWiseOptionVO. Business unit-level options set to override the enterprise-level options.

The IBY\_INTERNAL\_PAYERS\_ALL table stores payment processing and supplier-level defaulting attributes. Enterprise-level options are applied by default to all business units. You can optionally configure business unit-level overrides. The following table shows the fields that you can enable for auditing.

Audit Attribute	Table Column	Description	Audit-Enabled by Default
Allow Payee Bank Account Override on Proposed Payments	ALLOW_ EXT_ ACCT_ OVERRIDE_ FLAG	Indicates whether the payee bank account can be overridden during proposed payment review.	No
Payment Method Default Basis	PAYMENT_ METHOD_ AT_ PAYEE_ FLAG	Indicates whether the payment method defaults based on the payment method defaulting rules only or on the supplier-level payment method setup.	No



## Business Object: IBY\_PAYMENT\_SYSTEM

Attribute name on the Configure Business Object Attributes page: PaymentSystemInfoAuditVO Payment system information.

The IBY\_BEPINFO table stores information about payment systems. The following table shows the fields that you can enable for auditing.

Audit Attribute	Table Column	Description	Audit-Enabled by Default
Transmission Servlet Base URL	BASEURL	URL that navigates to the payment system.	No
Code	SUFFIX	Three-letter suffix that identifies each payment system.	No
Active Status	ACTIVESTATUS	Indicates whether the payment system is active or inactive.	No

## Business Object: IBY\_PAYMENT\_SYSTEM

Attribute name on the Configure Business Object Attributes page: PaymentSystemFormatVO Payment system formats used.

The IBY\_PMT\_SYS\_FORMATS table links payment systems to the payment formats they support. When you create a payment process profile, the IBY\_PMT\_SYS\_FORMATS table stores the formats that the payment system accepts. The following table shows the fields that you can enable for auditing.

Audit Attribute	Table Column	Description	Audit-Enabled by Default
Name	PAYMENT_SYSTEM_ID	Payment system identifier	No

## Business Object: IBY\_PAYMENT\_SYSTEM

Attribute name on the Configure Business Object Attributes page: PaymentSystemTransmissionVO Payment system transmission information.

The IBY\_PMT\_SYS\_TRANSMISSIONS table links payment systems to the transmission protocols they support. When you create a payment process profile, this table holds the transmission protocols that are supported by a particular payment system.


The following table shows the fields that you can enable for auditing.

Audit Attribute	Table Column	Description	Audit-Enabled by Default
Name	TRANSMIT_ PROTOCOL_CODE	Transmission protocol identifier	No
BEPID	BEPID	Payment system identifier	No

## Business Object: IBY\_EXTERNAL\_BANK\_ACCOUNT

Attribute name on the Configure Business Object Attributes page: ExternalBankAccountVO. External bank account information.

The IBY\_EXT\_BANK\_ACCOUNTS table defines the external bank account entity. The following table shows the fields that you can enable for auditing.

 **Caution:** To ensure security, it's inadvisable to translate any of the columns in the IBY\_EXT\_BANK\_ACCOUNTS table.

Audit Attribute	Table Column	Description	Audit-Enabled by Default
Country	COUNTRY_CODE	Country code of the bank account.	Yes
Branch Identifier	BRANCH_ID	Identifier of the bank branch to which this bank account belongs.	Yes
Bank Identifier	BANK_ID	Identifier of the bank to which this bank branch belongs.	Yes
Account	BANK_ ACCOUNT_NUM	Account number.	No
Currency	CURRENCY_CODE	Currency code.	Yes
IBAN	IBAN	International Bank Account Number.	No
Check Digits	CHECK_DIGITS	Check digits.	No
Account Type	BANK_ ACCOUNT_TYPE	Type of bank account.	No
Account Suffix	ACCOUNT_SUFFIX	Account suffix.	No
Allow international payments	FOREIGN_ PAYMENT_ USE_FLAG	Indicates whether this bank account is used for foreign payments or only for domestic payments.	No

Audit Attribute	Table Column	Description	Audit-Enabled by Default
Account Name	BANK_ ACCOUNT_NAME	Bank account name.	No

## Business Object: IBY\_EXTERNAL\_BANK\_ACCOUNT

Attribute name on the Configure Business Object Attributes page: ExternalBankAccountOwnerVO External bank account owner information.

The IBY\_ACCOUNT\_OWNERS table stores the joint account owners of a bank account. The following table shows the fields that you can enable for auditing.

Audit Attribute	Table Column	Description	Audit-Enabled by Default
Primary	PRIMARY_FLAG	Indicates whether the owner is the primary owner of the bank account.	Yes
To Date	END_DATE	Date the party's ownership of the account becomes inactive.	Yes


### Related Topics

- [Configuring Audit Business Object Attributes: Points to Consider](#)
- [Managing Audit Policies: Explained](#)

## Importing a Security Credential File: Procedures

To secure your electronic transmissions, you can upload, import, and assign a security credential file to transmission configurations. A security credential file is a digital file that stores your security key, which the application uses to encrypt or authenticate data transmitted to remote third-party systems such as banks. Once your security credential file with its secret key is assigned to a specific transmission configuration, any future process that runs and references this transmission configuration will use the file with its key for transmission security. The application understands which credential files are used by which protocols and displays only the appropriate ones in the setup pages.

Payments supports a variety of security-related credential files, including wallet files, trust store files, and digital certificates.

 **Note:** This procedure is applicable to Oracle Cloud implementations only.

Before you can import a security credential file with its key into Payments, you must first create a Payments master encryption key.


## Creating a Wallet File and a Master Encryption Key Automatically

To create a wallet file and a master encryption key automatically, perform the following steps:

1. Navigation: **Tools > Setup and Maintenance**.
2. Search on Manage System Security Options.
3. In the Search Result section, click the **Go to Task** icon to open the Manage System Security Options page.
4. Click the **Edit Master Encryption Key** button to open the **Edit Master Encryption Key** dialog box.
5. Select the **Automatically create wallet file and master encryption key** check box.
6. Click the **Save and Close** button.
7. When you see the Warning message, click **Yes**.

## Uploading the Wallet Security Credential File

Before you can import the security credential file, you must first upload it to Payments.

 **Caution:** Ensure the credential file is password-protected when you create it and that it is immediately deleted from Oracle Fusion Applications upon completion of the import process.

1. Navigation: Tools > File Import and Export.
2. Click the **Upload** icon to open the **Upload File** dialog box
3. Browse to the file you created and stored locally.
4. From the **Account** choice list, select **fin/payments/import**.
5. Click the **Save and Close** button.

## Importing the Wallet Security Credential File

To import security-related credential files, such as wallets and private keys for use in advanced security features, you can use the Import Security Credential Job process.

1. Navigation: Tools > Scheduled Processes to open the Scheduled Processes page.
2. Click the **Schedule New Process** button to open the **Schedule New Process** dialog box.
3. Search and select the case sensitive Import Security Credential Job to open the **Process Details** dialog box.
4. From the **Credential File Type** choice list, select the appropriate file type for your credential.
5. In the **Security Credential Name** field, enter a name for the credential file.
6. From the **UCM (Universal Content Management) File Name** choice list, select the file you previously uploaded.
7. Click the **Submit** button.

A confirmation message indicates the process ran successfully.

8. Click the **Close** button.
9. Click the **Refresh** icon. The Import Security Credential Job appears in the Search Results section with a status of **Succeeded**.

## Assigning the Wallet File to a Transmission Configuration

To assign the credential file to a transmission configuration, perform the following steps:

1. Navigation: Tools > Setup and Maintenance.
2. Search on Transmission Configuration.

- 3. Click the **Go to Task** icon to open the Manage Transmission Configurations page.
- 4. In the Search section, select your protocol from the **Protocol** choice list and click the **Search** button.
- 5. In the Search Results section, click the applicable configuration link to open the Edit Transmission Configuration page.
- 6. In the **Value** field for your protocol's applicable parameter, select the file you created, uploaded, and imported.

The name of the specific parameter used to import a security credential file depends upon the protocol. You can now securely transmit electronic files using this transmission configuration.

## FAQs for Configure Payment System Connectivity

### What's a format type?

A type or categorization that indicates what a format is used for. Examples of format types include payment file, remittance advice, and bank statement. Each format that you create in Oracle Fusion Payments is associated with a format type so the application knows how the format is used. Format types are either disbursement formats that relate to payment files or funds capture formats that relate to settlements or reports.

The following table shows several examples of format types.

Disbursement Format Types	Funds Capture Format Types
Disbursement separate remittance advice	Funds capture authorization and settlement
Disbursement positive pay file	Funds capture accompanying letter
Disbursement payment process request status report	Funds capture payer notification

The format type you associate with a format specifies the following:

- Type of message that is created
- Data extract that is used by the template to format the transaction data

#### Related Topics


- [Using Oracle BI Publisher Enterprise to Modify Templates for Use with Formats: Explained](#)



## 6 Define Funds Capture

### Funds Capture Process Profile: Explained

A funds capture process profile is a key setup entity that contains all the rules necessary for processing funds capture transactions. It tells Oracle Fusion Payments how to communicate with a specific payment system, and includes the payment system accounts to be used for processing transactions. During processing, the funds capture process profile is derived from transaction routing rules, which are created during the setup of internal payees.

 **Note:** Credit card services are currently not available in Oracle Cloud implementations.

A funds capture process profile contains rules that control each of the following steps of the funds capture process:

- Formatting messages
- Building settlements into a settlement batch
- Transmitting messages to the payment system

### Formatting Messages

When the processing type is Bank Account, a Verification section is displayed on the Create Funds Capture Process Profile page. When the processing type is Credit Card, an Authorization section is displayed instead of the Verification section. In either case, you select the format in which your payment system expects to receive the online message. This outbound format instructs Oracle Fusion BI Publisher how the message should look. You also select the format in which you expect to receive an inbound response from the payment system.

The Settlement section on the Create Funds Capture Process Profile page enables you to select a settlement message format that your payment system accepts. The settlement is online for a gateway-type payment system and in a batch for a processor-type payment system. Online settlement transactions are typically transmitted in a group, although they are processed as individual transactions. You also select the format in which you expect to receive an inbound response from the payment system.

You can select formats for contacting your payment system to retrieve an acknowledgment. You can also select other formats for receiving responses from the payment system that specifies whether a transaction succeeded or failed. Acknowledgments can be pushed by the payment system to your company, or your company can retrieve acknowledgments from the payment system.

In the Notification to Payer section on the Create Funds Capture Process Profile page, you select a notification format and delivery method to the payer. A payer notification is a message sent to each payer after the settlement or settlement batch is transmitted. The message notifies each payer of a funds capture transaction that charges their credit card or bank account.

### Building Settlements into a Settlement Batch

In the Creation Rules tab on the Create Funds Capture Process Profile page, you select settlement grouping attributes. When a specific grouping attribute is enabled, Payments ensures that settlements within one settlement batch all share the same value. Settlements with disparate attribute values trigger the creation of as many settlement batches as there are unique value combinations. For example, if you select Business Unit as a grouping rule, then only settlements with the same business unit are grouped in a settlement batch.

You can also limit the size of the settlement batch by amount or number of settlements.

## Transmitting Messages to the Payment System

A funds capture process profile is specific to one payment system and its payment system accounts. In the Accounts tab on the Create Funds Capture Process Profile page, you specify payment system accounts for use with the funds capture process profile. For each payment system account you enable, you select up to three transmission configurations, one each for authorizations or verifications, settlements, and acknowledgments. The payment system account identifies Payments to the payment system, and the transmission configurations tell Payments how to transmit transactions to the payment system.

### Related Topics

- [Payment System Account: Explained](#)
- [Transmission Configuration: Explained](#)

## Settlement Grouping Rules: Example

Settlement grouping is configured by selecting one or more check boxes in the Settlement Grouping Rules section on the Create Funds Capture Process Profile page. Selection of settlement grouping attributes specifies that settlements with the same attribute are included in a unique settlement batch when that profile is used. The following scenarios illustrate how settlement grouping rule options are used to group settlements into settlement batches using a specific funds capture process profile.

## Funds Capture Process Profile 1

In this example, Funds Capture Process Profile 1 has the following settlement grouping options selected:

- Business unit
- First-party legal entity
- Settlement date

## Create Settlement Batches

During funds capture transaction processing, the Create Settlement Batches program selects the following settlements:

Settlement	Amount	External Payer	Business Unit that Owns the Transaction	First-Party Legal Entity that Owns the Transaction	Settlement Date
A	1000 USD	Customer 1	California	North America	February 1, 2012
B	250 USD	Customer 2	California	North America	February 1, 2012
C	500 USD	Customer 3	Oregon	North America	February 1, 2012



Settlement	Amount	External Payer	Business Unit that Owns the Transaction	First-Party Legal Entity that Owns the Transaction	Settlement Date
D	750 USD	Customer 4	California	North America	March 1, 2012

The Create Settlement Batches program then groups the settlements into the following settlement batches:

## Settlement Batch 1

Settlement Batch 1 contains Settlements A and B because both settlements have the same settlement grouping attributes as follows:

- Business unit = California
- First-Party legal entity = North America
- Settlement date = February 1, 2012

## Settlement Batch 2

Settlement Batch 2 contains Settlement C because it has the following settlement grouping attributes:

- Business Unit = Oregon
- First-Party legal entity = North America
- Settlement date = February 1, 2012

## Settlement Batch 3

Settlement Batch 3 contains Settlement D because it has the following settlement grouping attributes:

- Business Unit = California
- First-Party legal entity = North America
- Settlement date = March 1, 2012

## Routing Rules: Explained

Routing rules route a funds capture transaction to the correct payment system account. Based on attributes of the transaction, the routing rules determine which funds capture process profile to use. For example, Oracle Fusion Payments compares attributes of a transaction, such as payment method, business unit, amount, and currency against the conditions in a routing rule. When all conditions are met, the specified payment system account and funds capture process profile are used for the transaction. The funds capture process profile is used to define how the transaction is processed.

You assign routing rules to internal payees during the setup of internal payees. Each payee can have prioritized routing rules. The routing rule with the highest priority is evaluated by Payments first. If the values in the requested funds capture transaction match the conditions in the routing rule, that transaction is routed to the applicable payment system account for processing.

To create a routing rule, navigate to: **Navigator > Tools > Setup and Maintenance > Search: Tasks pane: Define Receivables > right arrow > Define Receivables Configuration > Define Funds Capture > Manage Internal Payees task > Go to Task > Manage Internal Payees page > Payee field: enter a payee > Search button > Manage Routing Rules button > Manage Routing Rules page > Payment Method choice list: select a payment method > Create button > Create Routing Rule page.**

If the attributes in the requested funds capture transaction do not match the conditions in the routing rule, the routing rule is disregarded and Payments evaluates the next routing rule. If all routing rules are evaluated and none apply, Payments looks for a payment system account and funds capture process profile specific to the payment method type entered for the payee in the Default Routing section on the Set Rules page.

The payment system account and funds capture process profile that are used for an authorization are automatically used for additional actions, including settlement and any subsequent refunds.

## Using Oracle BI Publisher Enterprise to Modify Templates for Use with Formats: Explained

Each format in Oracle Fusion Payments corresponds to one Business Intelligence Publisher Enterprise (BI Publisher Enterprise) template. Payments uses BI Publisher Enterprise templates to format funds capture and funds disbursement transactions according to the formatting requirements of financial institutions and payment systems. Each template contains prescribed formatting attributes, such as data location. Banks, payment systems, and countries have specific electronic formatting requirements for payment files and settlement batches.

You can use existing BI Publisher Enterprise templates or modify them with minimal effort by using a standard text editor, such as Microsoft Word. For example, when a payment system requires a change to its payment file format, you can quickly make the change by modifying the appropriate template.

Whether you modify an existing template or create a new one, determines whether you also create a new format and a new payment process profile. Each payment process profile is associated with a format. The table lists two template scenarios and indicates the resulting action you take.

	Scenario 1	Scenario 1
Action by user.	Create a new template.	Modify an existing template.
Naming the template.	Rename the template.	Keep the same name.
Where to save the new or modified template.	Payments folder under the Custom folder or Payments folder under the Financials folder.	Payments folder under the Custom folder.
Create a new format.	Yes	No
Create a new payment process profile.	Yes	No


To modify a template, you can:

- Download a copy of the applicable template.
- Upload a copy of the modified template.

## Download a Copy of the Applicable Template

To download a copy of a predefined template, perform the following steps:

1. Sign in to Oracle BI Publisher Enterprise.
2. On the Home tab, click the **Catalog Folders** link. The Catalog tab appears with a hierarchy of folders.
3. Expand the Financials folder.
4. Expand the Payments folder.
5. Locate the predefined template type that you want to modify and click the **More** link.
6. From the menu, select **Customize**. All the templates that are associated with the predefined template type that you want to modify are copied to a folder under the Custom folder.
7. You can now download the files from the Custom folder and modify them according to the steps below or you can continue with step 8.


 **Note:** Do not modify predefined templates. When you apply a new patch or a new release, it overrides any changes you made to the predefined template. You can, however, copy a predefined template and then modify it.

8. On the Data Model tab, to copy a predefined template and save it to your local drive as a .rtf file, click the **Edit** link of the applicable template. Then click the **Save** button.
9. Navigate to the location where you want to save the copy of the template and click the **Save** button.
10. Navigate to the saved .rtf file and open it.

## Upload a Copy of the Modified Template

To upload a copy of a modified template, perform the following steps:

1. Using a text editor, modify the .rtf file on your local drive.
2. Save as **Other Formats**, change the file name, click the **Save** button, and close the file.
3. To upload a copy of your modified template to Oracle BI Publisher Enterprise, navigate to the applicable tab, and click the **Add New Layout** link.
4. Click the **Upload** icon. The **Upload Template File** dialog box appears.
5. In the **Layout Name** field, enter a name for the template you modified.
6. In the **Template File** field, browse to the location of the modified template on your local drive and click the **Open** button.
7. From the Type choice list, select **RTF Template**.
8. From the Locale choice list, select the language for the modified template.
9. Click the **Upload** button. The modified template appears on the Data Model tab of the applicable tab.

 **Note:** The modified template is also copied to the Payments folder that is within the Custom folder.

10. To open the modified template, click the **Edit** link.
11. To confirm that the modified template is saved, click the **Catalog** link. The Catalog tab appears with a hierarchy of folders.
12. Navigate as follows: **Custom folder > Financials folder > Payments folder**.

13. Select the Payments folder.
14. For the applicable template, click the **Edit** link. Your modified template appears.

## FAQs for Define Funds Capture

### What's a payment code?

Oracle Fusion Payments enables you to specify payment codes that are required by financial institutions. Payment codes can provide details to banks or payments systems about transaction handling, bank charges, or payment reasons for regulatory reporting purposes.

Payment code types include:

- Bank instruction codes
- Delivery channel codes
- Payment reason codes

### What's a bank instruction code?

Bank instruction codes are values that contain instructions that must be passed to a bank or financial institution at the payment file level. Up to two bank instructions can be entered on a payment process profile. When that payment process profile is used during the creation of a payment file, the bank instruction values are copied directly to it. The values are made available to the formatting process by the extract. If the payment format specifies the use of bank instructions, the values are passed to the bank in the header level of the payment file.

Oracle Fusion Payments provides many predefined bank instruction codes.

### What's a delivery channel code?

Delivery channels are instructions that tell the bank how to make the payment to the payee. A default delivery channel value can be set on the supplier, supplier address, or supplier site. A value defaults from the lowest of these levels with a value populated, onto the invoice in Oracle Fusion Payables. On the invoice, it's displayed with the installments and can be manually overridden there.

When an installment is paid, the delivery channel is copied from the document payable to the payment when payment documents have the same delivery channel. If you select delivery channel as a grouping rule on the profile, all documents that share the same delivery channel are grouped into a payment.

Oracle Fusion Payments provides many predefined delivery channel codes.

### What's a payment reason code?

Payment reason codes are generally country-specific identifiers provided by a country's government or central bank. These codes provide the payment system or bank with additional details about the reason for the payment for regulatory reporting

purposes. The purpose of entering payment reason codes required by a country's payment system or central bank is to specify the reason for the payment.

Oracle Fusion Payments provides many predefined payment reason codes.

## What's an online verification?

Before capturing funds from a payer's bank account, you can send an online message to the payment system requesting that it contact the bank and verify that the payer's bank account is valid. This process does not put a hold on any funds in the account, which is done with a credit card authorization.

You enable online verification in a funds capture process profile by selecting an outbound format and an inbound response format for verification.

 **Note:** Online payer verification is an optional feature and is not offered by all payment systems.



# 7 Define Payments Security

## System Security Options: Critical Choices

You can implement application security options on the Manage System Security Options page as part of a complete security policy that's specific to your organization. Security options can be set for encryption and tokenization of credit cards and bank accounts, as well as for payment instrument masking. Security options are used for both funds capture and disbursement processes.

To secure your sensitive data, consider the following security questions:

- Which security practices do you want to employ?
- Do you want to tokenize your credit card data?
- Do you want to encrypt your bank account data?
- Do you want to encrypt your credit card data?
- How frequently do you want to rotate the master encryption key and the subkeys?
- Do you want to mask credit card and bank account numbers, and if so, how?

To set up application security options, search for and select the **Manage System Security Options** task from the Setup and Maintenance work area.

## Best Security Practices

The following actions are considered best security practices for payment processing:

- Comply with the Payment Card Industry Data Security Standard (PCI DSS). PCI DSS is the security standard that is required for processing most types of credit cards.
  - Comply with all requirements for accepting credit card payments.
  - Minimize the risk of exposing sensitive customer data.
  - Work with a PCI DSS auditor to ensure compliance with the required security standards and to avoid potential violations.
- Before importing or entering data into Payments, encrypt and mask the following:
  - Customer credit card numbers
  - Supplier bank account numbers
  - Cardholder names
- Create a wallet.
  - Store the wallet file in a secure file location with limited access.
  - Rotate the master encryption key periodically.

## Implementation Process of Wallet File, Master Encryption Key, and Encryption

Before you can enable encryption for credit card or bank account data, you must automatically create a wallet file. The wallet file exists on the file system of the Oracle Enterprise Storage Server. A wallet file is a digital file that stores your master encryption key. The application uses your master encryption key to encrypt your sensitive data.

Automatic creation of the wallet file ensures that the wallet file is created in the proper location and with all necessary permissions.

## Credit Card Tokenization

If you tokenize your credit card data, you are complying with Payment Card Industry Data Security Standard (PCI DSS) requirements. PCI DSS requires companies to use payment applications that are PA DSS compliant.

Tokenization is the process of replacing sensitive data, such as credit card data, with a unique number, or token, that isn't considered sensitive. The process uses a third-party payment system that stores the sensitive information and generates tokens to replace sensitive data in the applications and database fields. Unlike encryption, tokens can't be reversed mathematically to derive the actual credit card number.


You can set up your tokenization payment system by clicking the Edit Tokenization Payment System button on the Manage System Security Options page. Then, to activate tokenization for credit card data, click the Tokenize button in the Credit Card Data section.

## Credit Card Data Encryption

You can encrypt your credit card data to assist with your compliance of cardholder data protection requirements with the following:

- Payment Card Industry (PCI) Data Security Standard
- Visa's PCI-based Cardholder Information Security Program (CISP)

Credit card numbers entered in Oracle Fusion Receivables and Oracle Fusion Collections are automatically encrypted. Encryption is based on the credit card encryption setting you specify on the Manage System Security Options page.


 **Note:** If you bring card numbers into Payments through import or customization, it's advisable to run the Encrypt Credit Card Data program immediately afterward.

## Bank Account Data Encryption

You can encrypt your supplier and customer bank account numbers.

Bank account encryption doesn't affect internal bank account numbers. Internal bank accounts are set up in Oracle Fusion Cash Management. They are used as disbursement bank accounts in Oracle Fusion Payables and as remit-to bank accounts in Receivables.

Supplier, customer, and employee bank account numbers entered in Oracle applications are automatically encrypted. Encryption is based on the bank account encryption setting you specify on the Manage System Security Options page.

 **Note:** If you bring bank account numbers into Payments through import or customization, it's advisable to run the Encrypt Bank Account Data program immediately afterward.




## Master Encryption Key and Subkey Rotation


For payment instrument encryption, Payments uses a chain key approach. The chain key approach is used for data security where A encrypts B and B encrypts C. In Payments, the master encryption key encrypts the subkeys and the subkeys encrypt the payment instrument data. This approach allows easier rotation of the master encryption key.

The master encryption key is stored in the wallet. The wallet is an Oracle Applications program module that protects stored data in an encrypted format. The master encryption key can be rotated, or generated, which also encrypts subkeys, but doesn't result in encrypting the credit card or bank account numbers again.

If your installation has an existing master encryption key, you can automatically generate a new one by clicking the **Rotate** button.

 **Note:** To secure your payment instrument data, you're advised to annually rotate the master encryption key or rotate it according to your company's security policy.

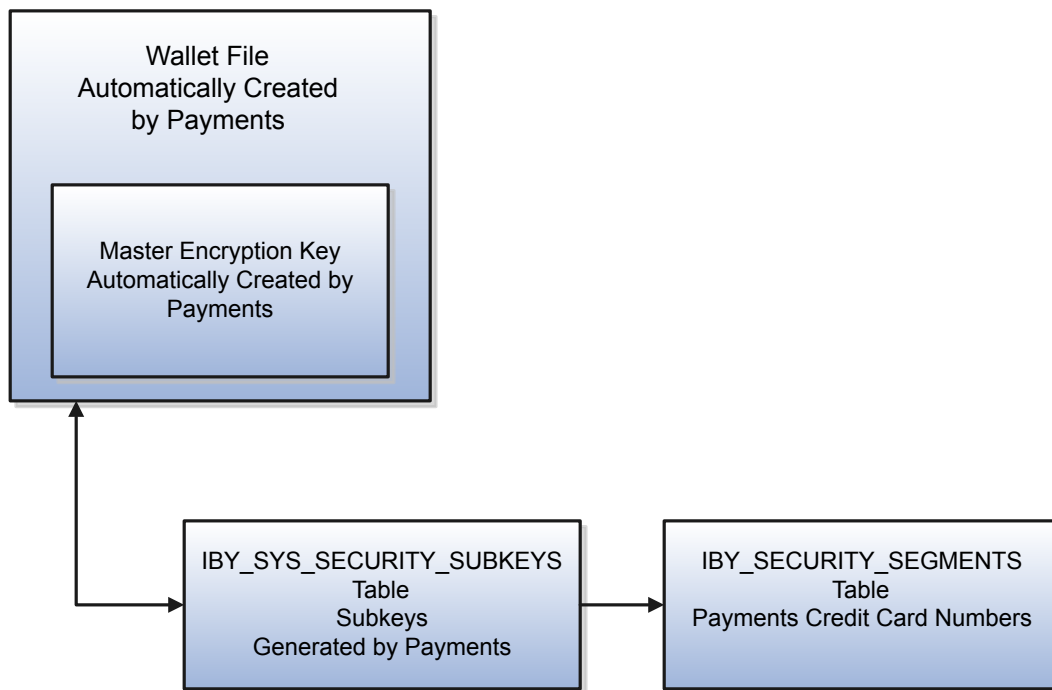
You can also select the frequency with which new subkeys are automatically generated, based on usage or on the maximum number days. To specify a subkey rotation policy, click the **Edit Subkey Rotation Policy** button.

 **Note:** To secure your payment instrument data, you are advised to schedule regular rotation of the subkeys.

The security architecture for credit card data and bank account data encryption is composed of the following components:

- Oracle Wallet
- Payments master encryption key
- Payments subkeys
- Sensitive data encryption and storage

The following figure illustrates the security architecture of the wallet, the master encryption key, and the subkeys.



## Credit Card and Bank Account Number Masking

Payments serves as a payment data repository on top of the Oracle Fusion Trading Community Architecture (TCA) model. TCA holds customer and supplier information. Payments stores all of the customer and supplier payment information and their payment instruments, such as credit cards and bank accounts. Payments provides data security by allowing you to mask payment instrument numbers.

On the Manage System Security Options page, you can mask credit card numbers and external bank account numbers. To do it, select the number of digits to mask and display. For example, a bank account number of XXXX8012 displays the last four digits and masks all the rest. These settings specify masking for payment instrument numbers in the user interfaces of multiple applications.

## Enabling Encryption of Sensitive Payment Information: Procedure

Financial transactions contain sensitive information, which must be protected by a secure, encrypted mode. To protect your credit card and external bank account information, you can enable encryption. Encryption encodes sensitive data, so it can't be read or copied. To enable encryption, you must create a wallet file. A wallet file is a digital file that stores your master encryption key, which the application uses to encrypt your sensitive data.

To secure your credit card or bank account data, navigate to the Setup and Maintenance work area, search for the Manage System Security Options task and perform the following steps:

1. Open the Manage System Security Options page.

2. Click **Apply Quick Defaults**.
3. Select all the check boxes:
  - Automatically create wallet file and encryption key
  - Encrypt credit card data
  - Encrypt bank account data
4. Click **Apply**.



# 8 Define Customer

## Define Customer Account

### Customer Account Relationships: Explained

Use customer account relationships to manage the sharing of billing and shipping services and payment activities between two accounts.

Account relationships let you identify customer accounts that can receive shipments or invoices for another account, as well as accounts that can pay for the open debit items of other accounts.

#### Using Customer Account Relationships

Use the Relationships tab of the Edit Account page to create and maintain account relationships for a particular customer account.

A customer account relationship is either a one-way relationship or a reciprocal relationship. By default an account relationship is one-way, meaning that the parent account can apply receipts to open debit items of the related account, but receipts in the related account can't be applied to open debit items of the parent account. If you want to allow the two accounts to pay for each other's open debit items, then enable the **Reciprocal** option for the relationship definition.

Enable the **Bill To** and **Ship To** options for relationships that involve billing and shipping services. In a one-way relationship, the parent account can process invoices and receive shipments for the related account. In a reciprocal relationship, either account can manage these services for the other account.

You can use the **Allow payment of unrelated transactions** Receivable system option to help manage customer account relationships. You must not enable this option if you want to restrict the sharing of receipt application and billing and shipping services to the account relationships that you define for each customer account.

If you do enable the **Allow payment of unrelated transactions** Receivables system option, then any customer account can pay for the open debit items of any other customer account. You can still use account relationships to manage billing and shipping services.

#### *Related Topics*

- [What's the difference between an account relationship and a paying relationship?](#)

### Customer Account Uses: Points to Consider

Use the customer account record to maintain detailed information about each of your customer accounts.

There are these points to consider when entering and updating customer account information:

- Payment Details
- Communication Information
- Account Relationships

- Account Profiles

## Payment Details

Use the Payment Details section to maintain active receipt methods and payment instruments for use by this customer account. During transaction and receipt entry, Oracle Fusion Receivables assigns the primary receipt method and payment instruments as the default. You can override these defaults at the transaction or receipt level with another active receipt method or payment instrument.

- Receipt Methods: Assign automatic and manual receipt methods to the customer account. Start and end dates for each receipt method must not overlap.

Assign automatic receipt methods for use with automatic receipts. Automatic receipt methods determine the required processing steps for automatic receipts, including confirmation, remittance, and reconciliation.

Assign manual receipt methods to indicate which form of receipt to use to collect payment for transactions belonging to this account.

- Payment Instruments: Assign credit cards and bank accounts to this customer account. The payment information that you create here is stored in Oracle Fusion Payments for use during funds capture processing.

You assign bank accounts for use with automatic receipts. The bank accounts you specify are used by the automatic receipt process to transfer funds from these accounts to your remittance bank accounts.

You can define multiple customer bank accounts in different currencies and assign bank accounts to customer addresses. The primary bank account for a particular currency is used as the default account during automatic receipt processing. You can define multiple, non-primary accounts in the same currency, even if the date ranges overlap.

## Communication Information

Use the Communication section to maintain customer contact persons for a customer account. Information that you can maintain for each contact includes:

- Name and job title.
- Contact points, such as phone and fax numbers, e-mail and instant message addresses, and URLs.
- Job responsibilities.
- Addresses. You can either enter new addresses specific to the contact, or you can enter addresses of account sites belonging to the customer account.

You can create a new contact person or add an existing contact person for a customer account or account site. If you are adding an existing contact person, you must define a party relationship at the customer level for each party and assign the party the Contact role type.

## Account Relationships

Use customer account relationships to manage the sharing of billing and shipping services and payment activities between two accounts.

## Account Profiles

The account profile is the profile class record assigned to this customer account for a specified time period. You use the profile class record to organize your customer accounts into categories that reflect the needs of your enterprise. After you assign a profile class, you can modify the profile according to the needs of a given customer account.

## Customer Addresses: Points to Consider

Enter and maintain address information for your customer records in Oracle Fusion Receivables.

When you create a customer, you must enter account and address information. This address becomes an account site record of the customer.

When you create additional customer accounts for the customer, you can either enter new address information to create a new account site, or you can use an existing account site. If you assign an existing account site to the new account, you can modify the address information itself, but you will receive a warning if this address is shared by other parties.

There are these points to consider when entering and updating customer address information:

- Reference Data Set for Account Sites
- Update Sharing
- Account Address Details
- Address Business Purposes
- Reference Accounts

### Reference Data Set for Account Sites

Before creating or importing customers, you must create a reference data set for use with customer account sites. You can share this reference data set across one or more business units, according to your requirements.

To access the new reference data set on customer account sites, provision the appropriate data role to the designated users in order to grant them access to the job role for the given business units and reference data sets.

### Update Sharing

If you update an address due to incorrect or missing information, these updates are shared with all account sites that use this address.

Update sharing applies to address information only, and not to account address details or business purposes.

### Account Address Details

Use the Account Address Details subsection to identify these values for the account site:

- Customer Category Code: Used to classify this customer according to the needs of your enterprise. You define these category codes under the Oracle Fusion Trading Community Model.
- Translated Customer Name: Used to enter the customer name in another language. The translated name replaces the customer name on external documents.

### Address Business Purposes

Assign address business purposes to account sites to identify the functions performed by each particular customer account site. You can designate one account site as Primary for each address business purpose.

Common address business purposes include:

- Bill-to: Assign the bill-to business purpose to account sites designated to receive and process bills. You must create at least one bill-to site to process transactions for a customer.

- **Ship-to:** Assign the ship-to business purpose to account sites designated to receive goods purchased by the customer account.

If you create a ship-to business purpose, you must indicate the related bill-to site that processes the bills for the shipped goods.

Bill-to sites available for association with ship-to sites either belong to the same customer account or to related customer accounts.

- **Deliver-to:** Assign the deliver-to business purpose to sites that receive all or part of goods sent to a ship-to site.
- **Bills of lading:** Assign the bills of lading business purpose to sites that manage contracts for carriers that ship customer goods.
- **Dunning:** Assign the dunning business purpose to sites that receive dunning letters from you.
- **Late Charge:** Assign the late charge business purpose to the site that is assigned the late charge policy for this customer account.

If the **Statement, Dunning, and Late Charges Site Profiles Used** profile option is set to Yes, then Receivables uses the late charge policy assigned to the late charge site to calculate late charges on transactions. If the profile option is set to No, then receivables uses the late charge policy assigned to the bill-to site assigned to the transaction.

## Reference Accounts

Assign distributions for revenue, freight, receivable, AutoInvoice clearing, tax, unbilled receivable, and deferred revenue accounts to customer bill-to sites. If AutoAccounting depends on Bill-to Site, then Receivables uses the bill-to site of the transaction to determine the related segment for these distributions.

### Related Topics

- [Reference Data Sets: Explained](#)

## Related Contact Sites: Explained

By default the Manage Customers page displays in the Sites section the account sites assigned to the selected customer account. If you enable the **Show related contact sites** option, the Sites section may display additional sites.

Additional sites may belong to related parties that, through some action of the party, were used as account sites. For example, an employee placed an order on behalf of the customer account, but used his or her personal address as the shipping address. This created an account site for the customer account with the employee address as the account site address.

Additional sites may also be part of data migration, such as existing account sites from other systems that are linked to the customer account.

### Related Topics

- [Data Import Objects: Explained](#)

## Manage Receivables Customer Profile Classes



## Profile Classes: Explained

Use profile classes to organize your customer accounts into categories that reflect the needs of your enterprise. The profile class record contains generic options that you can set in different ways to group your customers into broad categories, such as, for example, by industry, location, size, creditworthiness, business volume, or payment cycles.

For example, you might define three profile classes that reflect customer payment behavior: one for customers who pay promptly; one for late paying customers with high late charge rates; and a third for customers who for the most part pay on time, with discount incentives for early payment.

### Working with Profile Classes

When you create a customer account, Oracle Fusion Receivables assigns the profile class DEFAULT. Use the Edit Account page to modify this profile class or assign another profile class to the customer account.

When you create a customer site, Receivables does not assign either the DEFAULT profile class or the profile class of the customer account. Instead you must use the Create Site Profile page to assign the first profile class to the site. Use the Edit Site page either to update the existing profile or to assign a new profile class to the site.

After you assign a profile class to a customer account or site, you can customize details of the profile class to meet specific requirements for that account or site. For example, a particular site may transact business in a separate currency, or the site may be subject to additional late charges or penalty charges. These updates apply only to that particular account or site and do not affect the profile class record itself.

Profile class updates and assignments are managed using effective date ranges. Each profile class that you assign or update supersedes the previous profile class for the given date range. In this way you can manage over time the changes that take place in customer behavior or customer requirements. The date of a given transaction or receipt determines which account or site profile applies to the activity. The profile history for an account or site provides details of when a given version of a profile class is active.

## Updating and Versioning Profile Classes: Explained

After you update an existing profile class, Oracle Fusion Receivables presents you with three options for how to save your updates. You can update the profile class record only, or you can update the profile class record along with updating in different ways the profiles that make use of this record.

The three options for updating profile class records are:

- Apply changes to new profiles only.
- Apply changes to uncustomized profiles and version existing uncustomized profiles
- Apply changes to all profiles and version all existing profiles.

This update process is managed by the Propagate Customer Profile Class Update program.

### Apply changes to new profiles only

This option updates the profile class record only. It doesn't apply your updates to any account or site profile that uses this profile class.

The changes you make to this profile class will apply to new account or site profiles that are assigned this profile class.

## Apply changes to uncustomized profiles and version existing uncustomized profiles

This option applies versioning to profiles that don't have customized settings. The update process for this option is:

- Updates the profile class record.
- Applies your updates to uncustomized account and site profiles that use this profile class.
- Creates a new version of account and site profiles that use this profile class, using the system date as the effective start date.
- Assigns the previous version of account and site profiles that use this profile class an effective end date of the day before the system date. This previous version remains valid within its effective date range.

Use the Profile History tab of customer account and site records to view the versions of a profile.

## Apply changes to all profiles and version all existing profiles

This option applies versioning to all profiles. The update process for this option is:

- Updates the profile class record.
- Applies your updates to all account and site profiles that use this profile class.
- Creates a new version of account and site profiles that use this profile class, using the system date as the effective start date.
- Assigns the previous version of account and site profiles that use this profile class an effective end date of the day before the system date. This previous version remains valid within its effective date range.

Use the Profile History tab of customer account and site records to view the versions of a profile.

# Setting Up for Customer Credit Checks: Explained

The Credit Check service in the quote-to-cash process evaluates credit authorization requests according to the credit check settings in the customer account profile.

Set up the Credit and Collections section of each customer account profile with the settings that you want to use to determine credit authorization for the customer.

The settings used for credit authorization are:

- **Credit Limit** field: Enter the total credit limit that the customer has available in the given currency.
- **Tolerance** field: Enter the percentage that a customer can exceed their credit limit.
- **Conversion Rate Type** field: Enter the conversion rate type to use to convert transaction amounts to the credit limit currency.
- **Include in credit check** option:
  - If you enable this option, then the application compares the available credit in the applicable currency to the order amount.
  - If you do not enable this option, then the customer account is excluded from the credit check process and the order is approved.
- **Expiration Offset Days** field: Enter the number of days to use in the calculation of the expiration date of a credit authorization.

## Deriving the Credit Authorization Expiration Date

Receivables derives the expiration date of a credit authorization depending on the available date information for the transaction and the value in the **Expiration Offset Days** field, if there is one.

The logic for deriving the credit authorization expiration date is as follows:

- If the **Expiration Offset Days** field contains a value, then:
  - If the fulfillment/shipment date is provided, then:  $\text{Expiration Date} = \text{Fulfillment/Shipment date} + \text{Offset Days}$ .
  - If fulfillment/shipment date is not provided, but the sales order date is provided, then:  $\text{Expiration Date} = \text{Sales Order Date} + \text{Offset Days}$ .
  - If neither the fulfillment/shipment date nor the sales order date is provided, then:  $\text{Expiration Date} = \text{Credit Authorization Date} + \text{Offset Days}$ .
- If the **Expiration Offset Days** field does not contain a value, then:
  - If the fulfillment/shipment date is provided, then:  $\text{Expiration Date} = \text{Fulfillment/Shipment date} + 5$ .
  - If fulfillment/shipment date is not provided, but the sales order date is provided, then:  $\text{Expiration Date} = \text{Sales Order Date} + 5$ .
  - If neither the fulfillment/shipment date nor the sales order date is provided, then:  $\text{Expiration Date} = \text{Credit Authorization Date} + 5$ .

## FAQs for Manage Receivables Customer Profile Classes

### When are profile class defaults used?

The customer profile class shares these default settings with other parts of Oracle Fusion Receivables: Match Receipts By; AutoMatch rule set; AutoCash rule set; AutoInvoice grouping rule; payment terms; and tax printing options. Each of these settings is used according to the hierarchy required by the transaction or receipt process.

For receipt processing:

- Receivables searches for a Match Receipt By value to use for receipt and transaction comparison in the order: customer site profile, customer account profile, lockbox (for lockbox processing), and Receivables system options.
- Receivables searches for an AutoMatch rule set to determine how receipts are applied automatically and transactions recommended for manual application in the order: customer site profile, customer account profile, lockbox (for lockbox processing), and Receivables system options.
- If Receivables can't match receipts to transactions, the application searches for an AutoCash rule set to use to apply receipts in the order: customer site profile, customer account profile, and Receivables system options.

For transaction processing:

- During AutoInvoice import, Receivables searches for an AutoInvoice grouping rule to use to group transaction lines into transactions in the order: transaction source, customer site profile, customer account profile, and Receivables system options.
- Receivables searches for the payment terms to use on a transaction in the order: customer site profile, customer account profile, and transaction type. If the payment terms come from the site profile or account profile, and the **Override terms** option is enabled, then you can update the default payment terms on the transaction.

- Receivables searches for the tax printing option to use to print tax on transactions in the order: customer site profile, customer account profile, and Receivables system options. If no value is found, then Receivables uses Total Tax Only as the default value to print transactions.

## Manage Customers

### Customer and Party: Explained

You create customers to properly record and account for sales transactions, as well as to identify other attributes of the selling relationship. Recording a sales transaction requires that a customer, stored as a party in Oracle Fusion Trading Community Model, has both a customer account and customer site with a bill-to business purpose.

To understand the role of a customer in the context of the trading community model, it is helpful to understand a few related concepts. The key concepts related to customers and customer activities are:

- Party
- Customer
- Customer Account
- Site
- Relationship
- Contact

#### Party

A party is an entity that can enter into a business relationship with another party, such as buying and selling goods or offering services. A party can be either an organization or a person. A party exists separately from any business relationship that it enters into with another party.

#### Customer

A customer is a party, either an organization or a person, with whom you have a selling relationship. This selling relationship can result, for example, from the purchase of products and services or from the negotiation of terms and conditions that provide the basis for future purchases.

#### Customer Account

A customer account represents the attributes of the business relationship that a party can enter into with another party. The customer account has information about the terms and conditions of doing business with the party.

For example, you can create a commercial account for purchases made by a company for internal use, and a reseller account for purchases made by the same company for sales of your products to end users.

You can create multiple customer accounts for a party to maintain information about different categories of business activities. For example, to track invoices for different types of purchases, you can maintain an account for purchasing office supplies and another account for purchasing furniture.

You can also maintain multiple customer accounts for a customer that transacts business with more than one line of business in your organization.

You can share information about a party, such as profile, addresses, and contacts, across the customer accounts of the party. In addition, you can also maintain separate profiles and contacts, along with contact addresses and contact points, for each customer account.

## Site

A site is a point in space described by an address. A party site is the place where a party is physically located.

A customer site is a party site that is used in the context of a customer account. A single customer account can have multiple sites.

A customer address is a site that is used for billing, shipping, or other purposes that are part of the selling relationship.

An identifying address is the party site address that identifies the location of the party. Every party has only one identifying address, but a party can have multiple party sites.

## Relationship

A party relationship is the role of the party in the context of another party. Examples include affiliate, subsidiary, partner, employee, or contact.

An account relationship between different customer accounts of a party allows for the sharing of billing, shipping, and payment information.

## Contact

A contact is a person who communicates for or acts on behalf of a party or customer account. A contact can exist for a customer at the account or site level.

A person usually acts as a contact for an organization, but can also act as a contact for another person. For example, an administrative assistant is often the contact for an executive.

### *Related Topics*

- [Customer Hierarchies and Paying Relationships: How They Work Together](#)

## Party Relationships: Explained

Use party relationships to model your customer records according to the way you conduct your business. Party relationships can help you better understand and make decisions about the customers that you do business with.

Define party relationships to use with your customer records. A party relationship record contains the name of the party that relates to the customer, the way in which the party relates to the customer (relationship role), and the period of time that this particular relationship exists. You can then add these parties as contacts belonging to the accounts or account sites of a given customer.

A party relationship represents the way two entities interact with each other, based on the role that each entity takes with respect to the other. For example, the employment relationship between a person and an organization is defined by the role of the person as the employee and the organization as the employer.

A relationship can be reciprocal. Each entity is either the subject or object, depending on the perspective, or direction. The party that you define relationships for is the subject, and the party that you relate to is the object. For example, if Joe is the employee of Oracle, then Joe is the subject and Oracle is the object. Oracle as the employer of Joe, which reverses the subject and object, still describes the same relationship.

## Tax Profiles: Explained

Use the tax profile to maintain optional tax information about your customers and customer sites. You create third-party tax profiles under Oracle Fusion Tax. You can update the tax profile record in Oracle Fusion Receivables for the applicable customer or customer site.

Receivables doesn't populate a customer site record with the tax profile belonging to the customer record. You must define a separate tax profile for each customer site, according to the requirements of the site. Tax profile records defined at the customer site level take precedence over the tax profile record defined at the customer level.

For example, a customer level tax profile might not include tax exemptions. However, one of the related customer sites may be located in a developing country or region, where tax incentives for businesses include tax exemptions on specific products or services. In this case, you should define and maintain a separate tax profile for this customer site.

### *Related Topics*

- [Party Tax Profiles: Explained](#)

## Merging Customers: Explained

If you have Oracle Fusion Data Quality installed, the Oracle Fusion Trading Community Model manages the merging of duplicate and redundant customer records.

You can manually merge two customer records from the Manage Customers page in this way:

1. Search for the customer or customers that you want.
2. Select a customer.
3. Select the **Merge Record** command from the Actions menu to open the Merge window.
4. In the Merge window, select the master and duplicate records, and complete the merge request.
5. Repeat steps 2 to 4 for each customer that you want.

### *Related Topics*

- [Merging Records Manually: Worked Example](#)

## Preparing Customer Data for Upload: Points to Consider

Use the Upload Customers from Spreadsheet process to upload customer data in bulk. The single upload process performs all the operations of generating a batch, transferring the customer data in the spreadsheet template to the interface tables, and importing the data from the interface tables into Oracle Fusion Applications.

You can download a customer spreadsheet template to use to prepare your customer data. The template contains an instruction sheet and sample data to help guide you through the process of entering your customer information.

### Setting Up Related Customer Information

Set up the business objects you need in advance of the customer data upload.

This can include:

- Account address sets: Set up the reference sets you need for your customer account sites.

- Customer profile classes: Set up one or more profile classes for your customer records.
- Reference accounts: Set up general ledger accounts that you intend to use as reference accounts for customers.
- Customer bank accounts: Set up banks and bank account information.
- Tax information: Set up tax registration numbers and tax rate codes in Oracle Fusion Tax.
- Descriptive flexfields.

## Using the Spreadsheet Template

Enter data in the designated columns in each of the four worksheets: Customers, Contacts, Reference Accounts, Customer Bank Accounts.

These rules apply to entering data in columns:

- Column labels with an asterisk (\*) denote required columns.
- Use the **Show Extensible Attributes** and **Hide Extensible Attributes** buttons to show or hide additional columns.
- Don't move or delete existing columns, and don't insert new columns.
- Enter data in the correct format. In most cases, the columns will format the data that you enter according to the requirements of the upload.
- Each customer must have a unique combination of these values:
  - Customer number.
  - Customer account number.
  - Customer site number.
- Each customer contact must have a unique person number.

## Validating Unique Values in the Customer Spreadsheet Upload: Examples

During upload processing, the Upload Customers from Spreadsheet process checks for unique values in certain columns of the Customers worksheet and Contacts worksheet. If the values are unique, then the record is created. If the values are not unique, then the record fails with an upload error.

The columns with this validation are:

- Customers worksheet:
  - Customer Number
  - Account Number
  - Site Number
- Contacts worksheet:
  - Person Number

The sections below provide examples of the validation process. The assumption in these examples is that all records have the same Source System value.

## Customers Worksheet: Customer Number Validation

The Customer Number validation looks for a unique combination of values across the Customer Number, Customer Source Reference, and Customer Name columns.

The records in the table below fail the uniqueness validation on the customer number, because, for the same customer name, there are two different customer numbers and customer source references.

Customer Number	Customer Source Reference	Customer Name
ICS-B 256113	ICS-B 256113	Pfizer Inc
ICS-B 256114	ICS-B 256114	Pfizer Inc

The records in the table below also fail the uniqueness validation on the customer number, because, for the same combination of customer number and customer source reference, there are two different customer names.

Customer Number	Customer Source Reference	Customer Name
ICS-B 256113	ICS-B 256113	Pfizer Inc
ICS-B 256113	ICS-B 256113	Pfizer Inc 1

The records in the table below also fail the uniqueness validation on the customer number, because, for the same customer number, there are two different customer source references.

Customer Number	Customer Source Reference	Customer Name
ICS-B 256113	ICS-B 256113	Pfizer Inc
ICS-B 256113	ICS-B 256114	Pfizer Inc

In like manner, the Account Number validation looks for a unique combination of values across the Account Number, Account Source Reference, and Account Description columns. The Site Number validation looks for a unique combination of values across the Site Number, Site Source Reference, and Site Name columns.

## Contacts Worksheet: Person Number Validation

The Person Number validation looks for a unique combination of values across the Person Number, Person Source Reference, and First Name and Last Name columns.

The records in the table below fail the uniqueness validation on the person number, because, for the same combination of person number and person source reference, there are two different first name and last name combinations.

Person Number	Person Source Reference	First Name	Last Name
1000119901	1000119901	Matthew	Gorman



Person Number	Person Source Reference	First Name	Last Name
1000119901	1000119901	John	Gorman

The records in the table below also fail the uniqueness validation on person number, because, for the same person number, there are two different person source references.

Person Number	Person Source Reference	First Name	Last Name
1000119901	1000119901	Matthew	Gorman
1000119901	1000119902	Matthew	Gorman

The records in the table below pass the uniqueness validation. The validation process allows a combination of two different person numbers and person source references with the same first name and last name combination. This is because two different people may have the same name.

Person Number	Person Source Reference	First Name	Last Name
1000119901	1000119901	Matthew	Gorman
1000119902	1000119902	Matthew	Gorman

## Customer Listing Report

Use the Customer Listing Report to review your customer information. You can review existing customer information, as well as customer information entered manually, uploaded using the Upload Customers from Spreadsheet process, or imported using the Trading Community Model customer upload process.

Because customer records can contain large amounts of information, you may find that you need to review customer information that is not present in the predefined report, or filter the displayed information differently. You can copy and modify the full dashboard or individual reports of the dashboard using features of Oracle Transactional Business Intelligence. This allows you to replace or add columns to the report that are available from the Receivables Customer Real Time subject area.

The Customer Real Time subject area makes available many additional data attributes for display, including, for example, customer relationships, customer creditworthiness, preferred billing practices, calculation of late charges and penalty charges, and payment preferences.

Use this report when entering new customer information or updating existing customer information.

### Selected Report Parameters

Enter a value in at least one of the required parameters to select customer data.


Use additional parameters to refine your customer selections. Additional parameters are available on each tab of the dashboard to accommodate narrowing the selected data based on the type of data, such as customer contacts, customer addresses, and payment information.

## Last Update Date

Use this parameter to review uploaded data.

Enter the submission date of a customer upload:

- Enter a submission date in the first field and leave the second field blank to include all customer data from the submission date to the system date.
- Enter the same date in both fields to see the results of customer uploads for a given day.

 **Tip:** If you run the report at the end of the day, and if there were multiple uploads in the course of the day, the report will include all of the customer data uploaded. For large volumes of data, you may need to use additional parameters, such as a range of customer names, to filter the data according to your needs.

Because customer records are subject to manual update at any time, it is recommended to run the report as soon as the submission of the customer upload has completed.

## Customer Name

Enter a range of customers to display. The report retrieves and displays customer names in alphanumeric order.

To display one customer only, enter the same name in both fields. If you enter a customer name in the first field only, the report displays all customers in alphanumeric order beginning with the first name you enter.

## Customer Registry ID

Enter the related party identifier for the customer that you want to display. Use this parameter, for example, to ensure that all accounts for a customer are selected, which in some cases may be more reliable than the customer name.

## Customer Class

Display customers belonging to one classification only, for example, Commercial or Government.

## Report Output

The predefined report displays customer information in a format similar to the customer upload spreadsheet template.

There are five tabs:

- Customer General Information: Customer name, customer class, customer profile class, related identifiers (customer registry ID, DUNS, Taxpayer ID, customer account number, site number).
- Customer Address Information: Address of each customer site and all site-related information.
- Customer Contact Information: Contact names for each customer account, with full contact details.
- Customer Reference Account Information: General ledger reference accounts for each customer bill-to site: revenue, freight, receivable, AutoInvoice clearing, tax, unbilled receivable, and deferred revenue.

When AutoAccounting depends on Bill-to Site, Receivables uses the bill-to site of the transaction to determine the related segment values for the transaction distributions.

- Customer Payment Information: Bank account information for each customer account.

The predefined report displays all records for a customer account site, even if a record contains no information, such as contacts or payment information. This can help, for example, to identify data that may have been missing from a customer

upload. If you do not need to display records with no values, you can modify the individual reports to display only the sites that have the information you need by using the standard functionality of Oracle Transactional Business Intelligence.

You can export information from the report in PDF, Excel, or CSV format for viewing offline.

#### *Related Topics*

- [Receivables Subject Areas, Folders, and Attributes: Explained](#)

## FAQs for Manage Customers

### Does a new customer belong to a customer account?

No, creating a customer person or organization only establishes this party as having some kind of selling relationship with you. You also need to create at least one account and one account site with a bill-to purpose under the customer record in order to properly record and account for both sales transactions with the party and other attributes of the selling relationship with the party.

When you create a customer manually, the customer account is assigned the profile DEFAULT. When you import customers, there is no default profile assigned to the customer account. You must designate a customer account profile in the RA\_CUSTOMER\_PROFILES\_INT\_ALL table by specifying information for the CUST\_ORIG\_SYSTEM and CUST\_ORIG\_SYSTEM\_REFERENCE attributes, and leaving blank the CUST\_SITE\_ORIG\_SYSTEM and CUST\_SITE\_ORIG\_SYS\_REF attributes.

### What happens if I enter customer information that already exists?

If you have Oracle Fusion Data Quality installed, the Oracle Fusion Trading Community Model manages the control of duplicate and redundant party information.

If you enter party information resembling or matching existing information, then depending on your setup you are prompted in different ways to manage this information.

This can include:

- Prevention of data entry.
- Inclusion with confirmation of data entry.
- Review and selection of matching or similar data.

### How can I update other customer information?

Use the Edit Account pages to update information related to the accounts created for this customer.

Use the Edit Site pages to update information related to the sites created under a customer account. This includes address information and the tax profile for the related account site.

### Why can't I upload the customer spreadsheet?

You can only upload the .zip file that contains all of the .csv files. You can't select an individual .csv file or the completed .xlsm file template itself.

## Why are there errors in the customer upload batch?

Either because of invalid data or duplicate records.

Invalid data includes missing information or information in an incorrect format. You can correct some information directly in the Correct Customer Data spreadsheet. If the invalid data is related to the customer profile class, payment method, reference account, or customer bank account, then you must import the correct data using a new spreadsheet upload.

To avoid duplicating existing records, ensure that each customer in the Customers worksheet has a unique combination of these values:

- Customer number
- Customer account number
- Customer site number

Ensure that each contact in the Contacts worksheet has a unique person number.

## 9 Define Bills Receivable

### Updating System Options for Bills Receivable: Explained

Set Receivables system options to help you manage the processing of bills receivable transactions and receipts.

These system options relate to bills receivable:

- **Allow payment of unrelated transactions** option: Enable this option if you want to apply transactions of unrelated customers to a bill receivable.  
If you don't enable this option, then you can only apply transactions that belong to the customer drawee and its related customers.
- **Enable bills receivable** option: You must enable this option to use bills receivable.
- **Bills Receivable Transaction Source**: You can optionally select a default bills receivable transaction source. The transaction source that you select appears by default on all bills receivable transactions.  
You must set up bills receivable transaction sources before you can set this option.
- **Allow factoring of bills receivable without recourse** option: Enable this option if you want to create factored bills receivable remittances without recourse.  
Enabling this option lets you disable the **Factored with recourse** option on the Bills Receivable Remittance Batch pages. If you don't enable this option, then the **Factored with recourse** option is enabled and read only.

#### *Related Topics*

- [What's a drawee?](#)
- [Bills Receivable Risk: Explained](#)

### Setting Up a Bills Receivable Creation Receipt Method: Procedure

Set up bills receivable creation receipt methods to create bills receivable from transactions automatically using the Create Bills Receivable Batch program. The receipt method designates the transaction type, maturity date, bill number, and if necessary the minimum and maximum bill amounts, and determines how transactions are grouped into bills receivable.

After you set up bills receivable creation receipt methods, you must assign both a bills receivable creation receipt method and a paying customer defined as drawee to transactions that you want to automatically batch for bills receivable.

To set up a new bills receivable creation receipt method:

1. Navigate to the Create Receipt Class and Methods page.
2. Name the receipt class and, in the **Creation Method** field, select **Bills Receivable**.
3. Create a receipt method.
4. Navigate to the Bills Receivable tab.
5. If applicable, enable the **Bills receivable inherit transaction numbers** option to use the transaction number as the bill number.

This only applies if there is one transaction in the bill receivable. If the bill receivable contains more than one transaction, then Oracle Fusion Receivables assigns the bill number according to the settings in the bills receivable transaction source.

6. In the **Number of Bills Receivable Rule** field, select the rule to use to group transactions into bills receivable:
  - a. **One per Customer:** Group all transactions belonging to a single customer drawee into one bill receivable.
  - b. **One per Customer Due Date:** Group all transactions belonging to a single customer drawee that have the same payment schedule due date into one bill receivable.

If a transaction uses the payment terms Immediate, then this transaction is included in the first available bill receivable that can accommodate the transaction amount, without regard to the due date. In this case the effective grouping rule is the same as **One per Customer**.

- c. **One per Invoice:** Group all payment schedules for a single transaction into one bill receivable.
  - d. **One per Payment Schedule:** Create a separate bill receivable for each transaction payment schedule (no grouping).
  - e. **One per Site:** Group all transactions for a single customer drawee site into one bill receivable.
  - f. **One per Site Due Date:** Group all transactions for a single customer drawee site that have the same payment schedule due date into one bill receivable.

If a transaction uses the payment terms Immediate, then this transaction is included in the first available bill receivable that can accommodate the transaction amount, without regard to the due date. In this case the effective grouping rule is the same as **One per Site**.

7. In the **Bills Receivable Maturity Date Rule** field, select **Earliest** or **Latest** to indicate whether to derive the maturity date for a bill receivable from the earliest or latest due date of all transactions applied to the bill.
8. Enter the bills receivable transaction type to use with this receipt method. All bills receivable created from transactions use the settings of this transaction type.
9. In the **Lead Days** field, enter the number of days before the transaction due date that a transaction can be applied to a bill receivable.
10. Use the **Minimum Amount** and **Maximum Amount** fields to enter minimum and maximum amounts for bills receivable created from transactions.
  - o If you enter a minimum amount, then a bill receivable is not created for the applicable transactions unless their sum is greater than this amount.
  - o If you enter a maximum amount, then the total amount of the bill receivable must match this amount exactly.

After applying a transaction to a bill receivable, a partial transaction amount that exceeds the maximum amount of the bill can remain unapplied. For example, an invoice of \$1000 applied to a bill receivable with a maximum amount of \$900 leaves an open amount of \$100 on the transaction.

#### Related Topics

- [Batching Transactions for Bills Receivable: Points to Consider](#)

## Preparing Transactions for Bills Receivable Batches: Critical Choices

If you plan to generate bills receivable from transactions using the Create Bills Receivable Batch program, you must prepare transactions to make them eligible for inclusion in a bills receivable batch.

To prepare transactions for bills receivable batches, review and, if necessary, complete tasks related to each of the following activities:

- Preparing Customers
- Assigning Bills Receivable Creation Receipt Methods
- Selecting Transactions for a Bills Receivable Batch
- Using the Customer Drawee Bank Account

## Preparing Customers

Prepare customers for bills receivable transactions. You must define customers as drawees, and assign drawees to the applicable transactions.

Perform these two tasks:

1. Define customers as drawees.

Define customer account sites as drawee sites for bills receivable. You must define a drawee site for each customer account for which you plan to create bills receivable.

For each applicable account site of the customer account:

- a. Navigate to the Edit Site page: Site Details tab.
- b. Assign the business purpose Drawee.
- c. Enter the bills receivable reference accounts: bills receivable, unpaid bills receivable, remitted bills receivable, and factored bills receivable.

2. Assign drawees to transactions as paying customers.

Assign a customer drawee to each applicable transaction as the paying customer. The drawee, as the party who is responsible for paying for the goods or services on the transaction, must be designated as the paying customer.

## Assigning Bills Receivable Creation Receipt Methods

Assign a bills receivable creation receipt method to each transaction that you want to apply to bills receivable. The bills receivable creation receipt method designates the transaction type, maturity date, bill receivable number, and minimum and maximum bill amounts for each bill receivable, and determines how transactions are grouped into bills receivable.

You can assign bills receivable creation receipt methods to transactions manually or automatically.

To assign bills receivable creation receipt methods automatically:

1. Open the Edit Account or Edit Site page for a customer drawee.
2. In the Payment Details tab, add one or more bills receivable creation receipt methods.
3. Set one of these receipt methods as **Primary**.

Oracle Fusion Receivables automatically assigns the primary receipt method to transactions belonging to this customer drawee that you enter manually or import using AutoInvoice.

To assign a bills receivable creation receipt method manually:


1. Open the transaction in the Edit Transaction page.
2. Open the Payment tab and select the receipt method that you want in the **Receipt Method** field. The **Receipt Method** field choice list contains all receipt methods assigned to the customer drawee.

**Important:** If the bills receivable creation receipt method uses the **Number of Bills Receivable Rule** of **One Per Customer** or **One Per Customer Due Date**, then you must also set **Drawee** as the primary business purpose for the applicable customer drawee sites in order to generate bills receivable automatically. This is because the customer drawee is the defining element of transactions.

## Selecting Transactions for a Bills Receivable Batch

When you run the Create Bills Receivable Batch program, the program selects all eligible transactions from customer drawees belonging to the business unit and the parameter values that you enter for the program run.


This table describes rules that apply to selecting transactions for a bills receivable batch.

Feature	Rule
Ineligible Transactions	You cannot assign disputed transactions or other bills receivable to a bill receivable using the Create Bills Receivable Batch program.
Currency	<p>If you enter a value in the <b>Currency</b> parameter, the program only selects transactions in this currency to create bills receivable.</p> <p>If you don't enter a value in the <b>Currency</b> parameter, the program creates bills receivable in different currencies, according to the currencies of the transactions selected.</p> <p>The program generates bills receivable from all eligible transactions with the same currency and conversion rate.</p>
Debit Memos and Credit Memos	<p>If you are batching debit and credit memos with the payment term Immediate, Receivables includes these debit and credit memos in the first available bill receivable, without regard to the due date grouping rule assigned to the bill.</p> <p> <b>Note:</b> The amounts of debit and credit memos must still respect the maximum amount of the bill receivable, if there is one. If the sum of debit and credit memos applied to a bill exceeds the designated maximum amount of the bill, then the program excludes one or more debit memos or credit memos in order to bring the bill to the proper amount.</p>

## Using the Customer Drawee Bank Account

The currency, conversion rate, paying customer bank account, and the settings assigned to the bills receivable creation receipt method determine how transactions are grouped into bills receivable.

The customer drawee bank account on eligible transactions acts as an additional rule for grouping transactions into bills receivable. The Create Bills Receivable Batch program creates bills receivable from transactions with the same customer drawee (paying customer) bank account.

 **Note:** The customer drawee bank account must be in the same currency as the transactions applied to the bill receivable.



To automatically assign the customer drawee bank account to transactions that you enter manually or import using AutoInvoice:

1. In the Edit Site page: Site Details tab, for each applicable customer drawee site, assign the additional business purpose **Bill-to**.
2. In the Edit Site page: Payment Details tab: Bank Accounts section, enter the customer drawee bank account for the bill-to site.
3. Set this bank account as **Primary**.

Bills receivable created by the Create Bills Receivable Batch program inherit the bank account on the transaction if both of these conditions apply:

- The customer drawee site is also a bill-to site.
- The bills receivable creation receipt method does not use the **Number of Bills Receivable Rule** of **One Per Customer** or **One Per Customer Due Date**.

If the **Number of Bills Receivable Rule** on the bills receivable creation receipt method is **One Per Customer** or **One Per Customer Due Date**, then bills receivable created by the Create Bills Receivable Batch program inherit the bank account on the transaction only if both of these conditions apply:

- The bank account is assigned to the customer drawee site marked as **Primary**.
- The primary drawee site is also a bill-to site.

If the bank account on eligible transactions is not assigned to the primary drawee site, then the program creates bills receivable without a drawee bank account.

#### *Related Topics*

- [Why can't I create bills receivable from transactions?](#)
- [Batching Transactions for Bills Receivable: Points to Consider](#)

## Setting Up a Bills Receivable Remittance Receipt Method: Explained

Set up bills receivable remittance receipt methods to use with bills receivable remittance batches. A bills receivable remittance batch processes all eligible bills receivable and prepares them for remittance to your bank.

These settings on the Create and Edit Receipt Class and Methods pages apply to bills receivable remittance receipt methods:

- **Creation Method** field: Select **Bills Receivable Remittance**.
- **Remittance Method** field: Select **Standard** for standard bills receivable and **Factoring** for factored bills receivable with or without recourse.
- **Debit Memos Inherit Receipt Numbers** option: Enable this option if you want debit memo reversals of receipts applied to a bill receivable that are remitted with this receipt method to inherit the receipt number.
- **Receipts inherit bills receivable number** option: Enable this option if you want receipts created for bills receivable remitted with this receipt method to inherit the bill numbers.
- **Remittance Bank Accounts** tab: Click this tab and create a remittance bank account to use with this receipt method. The remittance bank contains both the bank account information to use for remittances and the accounting for remitted bills receivable and the subsequent receipt.

### Related Topics


- [Bills Receivable Remittance: Overview](#)
- [Creating a Bills Receivable Remittance Batch: Explained](#)
- [Creating and Clearing Receipts for Bills Receivable Remittances: Explained](#)

## Setting Up a Remittance Bank Account for Bills Receivable: Explained

Set up a remittance bank account to use with bills receivable remittance receipt methods. The remittance bank account determines how bills receivable are grouped and remitted, and provides accounting information for both bills receivable transactions and receipts.

These settings on the Create and Edit Remittance Bank Account pages apply to bills receivable remittance bank accounts:

- **Primary** option: Enable this option if this is the primary bank account for this receipt method. If you are creating more than one remittance bank account in the same currency, then you must set one bank account as **Primary**.
- **Override bank** option: Enable this option if you want to be able to change the remittance bank information on a bills receivable remittance batch after it is created.
- **Clearing Days** field:
  - If this receipt method is for standard remitted bills receivable, enter the number of days it will take for the bank to clear the customer drawee receipt.
  - If this receipt method is for bills receivable factored with recourse, enter the number of days it will take for the bank to clear the cash advance (short-term debt) receipt.
- **Risk Elimination Days** field: If this receipt method is for bills receivable factored with recourse, enter the number of days after the maturity date to clear the short-term debt.
- **Remitted Bills Receivable** and **Factored Bills Receivable** fields: Enter the remitted bills receivable and factored bills receivable accounts used for bills receivable transactions.
- **Collection Days** field: If this receipt method is for standard remitted bills receivable, enter the minimum number of days that the remittance bank uses to collect on a bill remitted after the maturity date.
- **Short Term Debt** field: Enter the short-term debt account to use for factored bills receivable with this receipt method.

 **Note:** The **Factoring** field is not used for bills receivable. Bills receivable receipts are not factored, but rather created as remitted.

### Related Topics

- [Creating a Bills Receivable Remittance Batch: Explained](#)
- [Creating and Clearing Receipts for Bills Receivable Remittances: Explained](#)

## FAQs for Define Bills Receivable

### How can I create a customer drawee?

By creating at least one customer site for the customer account with the business purpose **Drawee**. You must create a customer drawee site for each customer account for which you plan to use bills receivable.

If you plan to create bills receivable from customer transactions, then you must also assign the customer drawee as the paying customer for these transactions.

### What's the difference between a signed and unsigned bill receivable?

A signed bill receivable requires acceptance by the customer drawee before you can complete the bill and make it available for remittance.

You can initiate an unsigned bill receivable without requiring acceptance by the customer drawee. The unsigned bill receivable itself becomes the document necessary to claim repayment.

To create an unsigned bill receivable transaction type, leave both the **Signed** and **Issued by drawee** options unchecked.

### When is a bill receivable issued by the drawee?

When the customer drawee wants to create a bill receivable with a promise to repay a specific amount on a specific date. This is also called a promissory note. Promissory notes are usually guaranteed by the bank that issues the note.

### How can I present late charges on bills receivable?

You can only present late charges on bills receivable as a debit memo or invoice. You can't present late charges as an adjustment, because you can't make adjustments to bills receivable transactions.

If you select Adjustment as the late charge type, then late charges are generated for all overdue transactions except bills receivable.



# 10 Define Revenue Management Configuration for Receivables

## Evaluating Revenue Policy: Points to Consider

Use the Manage Revenue Policies page to specify revenue policies for each applicable business unit. Oracle Fusion Receivables uses the revenue policy definition to make automatic revenue recognition decisions for manually entered and imported transactions.

Receivables compares each transaction against the revenue policy, and assigns revenue contingencies to the transaction or transaction lines that deviate from the policy definitions.

There are these points to consider for each revenue policy definition:

- Credit Classification
- Refund Policy Threshold
- Payment Terms Threshold

### Credit Classification

Use credit classifications to identify your high risk, noncreditworthy customers. You can assign up to three levels of risk. Receivables compares these risk levels to the credit classification assigned to the customer profile.

When you enter or import a transaction for a customer with a credit classification that matches one of the credit classifications in the revenue policy, Receivables:

- Assigns the Customer Creditworthiness contingency to the transaction.
- Defers revenue on the entire transaction.
- Recognizes revenue on the transaction only to the extent of payments received.

### Refund Policy Threshold

Use the **Refund Policy Threshold** column to enter the standard refund period in days that you typically offer to your customers.

When you enter or import a transaction with a line that is associated with a contract, Receivables analyzes the contract details. If the contract offers a refund period that exceeds the refund policy, Receivables:

- Assigns the Refund contingency to the transaction line.
- Defers revenue on the transaction line.
- Recognizes revenue on the transaction line only after the refund period on the transaction line expires.

### Payment Terms Threshold

Use the **Payment Terms Threshold** column to enter the maximum time period in days before payment terms become extended.

When you enter or import a transaction with payment terms or an installment schedule that exceeds the payment terms policy, Receivables:

- Assigns the Extended Payment Term contingency to the transaction.
- Defers revenue on the entire transaction.
- Recognizes revenue on the transaction only to the extent of payments received.

For example, you enter a payment terms threshold of 180 days on your revenue policy, and you later enter or import an invoice with payment terms that have four installments:

- Net 60
- Net 90
- Net 120
- Net 200

Receivables defers the entire revenue amount on the invoice because the last installment exceeds the 180-day threshold by 20 days.


#### *Related Topics*

- [Recognizing Revenue: Explained](#)
- [Payment Terms: Explained](#)

## Event-Based Revenue Management: How It Works

Oracle Fusion Receivables automates the timing of revenue recognition for both manually entered transactions and transactions imported using AutoInvoice. This automated revenue management process helps you to comply with the strict revenue recognition requirements mandated by US GAAP and International Accounting Standards.


The event-based revenue management process evaluates each transaction and decides whether to immediately recognize revenue, or temporarily defer revenue to an unearned revenue account based on the contingencies assigned to the transaction. Revenue is subsequently recognized according to the removal event assigned to each contingency.

 **Note:** Even if you set up for automated revenue recognition, you can still manually adjust revenue on transactions. Once you manually adjust revenue, Receivables discontinues the automatic monitoring of contingencies.

## Settings That Affect Event-Based Revenue Management

These settings affect event-based revenue management:

- **Require salesperson** Receivables system option: You must enable the **Require salesperson** system option to use revenue recognition.
- AR\_INTERFACE\_CONTS\_ALL table: You can use the AR\_INTERFACE\_CONTS\_ALL table to assign revenue contingency IDs to billing lines, before importing transactions using AutoInvoice.

 **Note:** When importing parent and child transaction lines, AutoInvoice automatically copies any contingencies from the parent line to the child lines.

- **Revenue Contingencies:** The revenue contingencies assigned to transactions, and their corresponding removal events, determine what revenue is deferred and for how long.
- **Revenue Policy:** Your revenue policy may trigger the assignment of contingencies to transactions.
- **Revenue Contingency Assignment Rules:** Your active revenue contingency assignment rules may trigger the assignment of contingencies to transactions.
- **Revenue Scheduling Rules:** If a revenue scheduling rule is assigned to the transaction, then revenue is recognized according to the revenue scheduling rule details and rule start date.

## How Event-Based Revenue Management Is Calculated

The event-based revenue management process for deferring and later recognizing revenue on transactions follows these steps:

1. Receivables evaluates a transaction either entered manually or imported using AutoInvoice for revenue recognition.
2. If one or more contingencies exist, Receivables defers the corresponding revenue to an unearned revenue account and records the reason for the deferral.
3. Receivables monitors the contingencies until an event occurs that can remove the contingency and trigger revenue recognition.
4. When a removal event occurs, Receivables recognizes the appropriate amount of unearned revenue on the transaction.  
The revenue is recognized either according to the revenue scheduling rule or the last contingency removal date.

*Related Topics*

- [Updating System Option Records: Critical Choices](#)

## Revenue Contingencies: Explained

You can use predefined revenue contingencies to assign to your customer transactions. You can define your own contingencies based on the predefined contingencies, and you can define revenue contingency assignment rules to control which contingencies are assigned to which transactions.

This table describes the predefined revenue contingencies and their corresponding contingency removal events.

Contingency Name	Contingency Removal Event
Cancellation	Contingency expiration date or expiration period
Customer Creditworthiness	Receipt application
Delivery	Proof of Delivery

Contingency Name	Contingency Removal Event
Doubtful Collectibility	Receipt application
Explicit Acceptance	Customer acceptance
Extended Payment Terms	Receipt application
Forfeitures	Contingency expiration date or expiration period
Installation	Customer acceptance
Prior Billing Acceptance	Invoicing
Refund	Contingency expiration date or expiration period

## Payment-Based Contingencies: Explained

Revenue recognition of payment-based revenue contingencies depends upon payments made against open transactions. You use payment-based contingencies for customers, or for certain transactions, where you want to recognize revenue only when you receive payment.

If a transaction or transaction line contains payment-based contingencies, these rules apply:

- Oracle Fusion Receivables initially defers revenue on the sum of all line balances, excluding taxes, freight, and late charges.

If the payment-based contingency is on one line only, then under either of these circumstances Receivables defers revenue for the relevant line only:

- Payment-based contingency was assigned by default to all the lines of the transaction, and the contingency was expired manually on certain lines.
  - Payment-based contingency was manually applied to one or more lines.
- During AutoInvoice import, full or partial receipt application on an imported transaction can trigger automatic recognition of previously deferred revenue. In such cases, Receivables initiates the distribution of revenue in the amount of the applied receipt from an unearned revenue account to the appropriate earned revenue account.
- When you apply a receipt to a transaction with payment-based contingencies, the total amount of revenue that is recognized can never exceed the original amount due on the transaction, less any applicable credit memo.
- If you later need to reverse a receipt, then Receivables automatically moves the amount of the reversed receipt back to the unearned revenue account.



## Payment-Based Contingencies

Payment-based revenue management occurs when deferred revenue exists on a transaction due to any of these revenue contingencies: Creditworthiness, Extended Payment Terms, Doubtful Collectibility.

- **Creditworthiness:** You can assign a credit classification that indicates noncreditworthiness to a customer or site profile. You can also assign up to three such credit classifications to your revenue policy.

If Receivables can't associate the customer on the transaction with any of these credit classifications, then the customer is presumed to be creditworthy. If Receivables can associate a customer with one of these credit classifications, then Receivables assigns the Creditworthiness contingency to all transaction lines and defers the entire transaction amount.

- **Extended Payment Terms:** You define the payment terms threshold on your revenue policy.

If the payment terms on a transaction exceeds the stated threshold, then Receivables assigns the Extended Payment Terms contingency to all transaction lines and defers the entire transaction amount.

- **Doubtful Collectibility:** Collectibility of late charges is typically in doubt, and shouldn't be considered earned revenue until payment is received.

Decisions about doubtful collectibility are typically made in feeder systems, before AutoInvoice import occurs.

Deferred revenue can exist on a transaction due to a combination of these contingencies, as well as time-based contingencies. In such cases, applied payments initiate revenue recognition only if time-based contingencies have expired.

The following types of receipt application have no impact on revenue recognition:

- You apply a miscellaneous receipt. Only standard receipts have potential revenue recognition implications.
- You apply a receipt against a transaction that had revenue manually deferred.
- You apply a receipt against a transaction that had revenue deferred due to unexpired time-based contingencies.

In this case, Receivables keeps the revenue amount for the transaction or transaction line in the unearned revenue account, but marks the transaction as revenue that is pending recognition until after the contingency expires.

## Removal Events: Explained

The event-based revenue management process in Oracle Fusion Receivables manages the recognition of revenue on transactions with revenue contingencies. If a transaction has one or more revenue contingencies, Receivables defers revenue to an unearned revenue account until the contingencies expire.

The extent of the revenue deferral, and the subsequent timing of revenue recognition, depends on the nature of the contingency:

- Time-based contingencies must expire before the contingency can be removed and revenue recognized.
- Payment-based contingencies require payment before the contingency can be removed and revenue recognized.
- Post-billing customer acceptance clauses must expire (implicit acceptance), or be manually accepted (explicit acceptance), before the contingency can be removed and revenue recognized.
- Prior billing customer acceptance clauses require the recording of customer acceptance in the feeder system, or its expiration, before importing into Receivables for invoicing. Customer acceptance or its expiration must occur before the contingency can be removed, and the order can be imported into Receivables for invoicing.

Each contingency that Receivables provides has a corresponding removal event. The removal event determines the action or event necessary to remove the contingency from a transaction or transaction line.

This table lists the revenue contingencies with their corresponding removal events:

Contingency Name	Contingency Removal Event
Cancellation	Contingency expiration date or expiration period
Customer Creditworthiness	Receipt application
Delivery	Proof of delivery
Doubtful Collectibility, due to conditions such as late charges and other fees	Receipt application
Explicit Acceptance	Customer acceptance
Extended Payment Terms	Receipt application
Installation	Customer acceptance
Prior Billing Acceptance	Invoicing
Refund	Contingency expiration date or expiration period

## Using Revenue Contingency Assignment Rules: Examples

Use the Revenue Contingency Assignment Rules pages to define rules for automatically assigning revenue contingencies to transactions.

A revenue contingency assignment rule consists of:

- Rule name and optional description.
- Revenue contingency to assign to transactions.
- One or more parameters that contain the conditions under which the contingency is assigned.

The rule parameters available for rule definition include:

- Revenue Scheduling Rule
- Transaction Source
- Bill-to Customer
- Bill-to Site
- Customer Profile Class
- Memo Lines

- Business Unit
- Ship-to Customer
- Ship-to Site
- Transaction Type

You can define multiple rules with different matching criteria that return the same contingency, and you can also define multiple rules with the same matching criteria that return multiple contingencies.

After you create an assignment rule, you can test the rule to ensure that all expressions are accurate and the rule functions as expected.

## Scenario

Revenue contingency assignment rules let you automate the assignment of contingencies to transactions in a way that reflects the practical needs of your enterprise. You typically use these rules to support the contingency assignment requirements that are not supported by your revenue policy setup.

For example, you may want to require Customer ABC to manually accept an item before recognizing revenue on the related transactions. To make this a mandatory process:

1. Create a new revenue contingency assignment rule.
2. Select the revenue contingency Explicit Acceptance for this rule.
3. Select Bill-to Customer as the rule parameter.
4. Enter the related customer condition.

With this defaulting rule in place, Receivables assigns the Explicit Acceptance contingency to all transaction lines for Customer ABC, and defers revenue until the customer acknowledges acceptance of each line item.

## FAQs for Define Revenue Management Configuration for Receivables

### What's a revenue contingency?

A revenue contingency is the terms and conditions in a sales contract or business agreement that prevents revenue from being immediately recognized, based on the revenue recognition requirements mandated by US GAAP and International Accounting Standards.

Typical contingencies that can delay revenue recognition include customer creditworthiness, nonstandard payment terms, and nonstandard refund policies.

## When do I use a revenue policy with a contingency?

You can assign a contingency to a transaction based on the revenue policy of your enterprise.

You have these options:

- **Credit Classification:** The contingency is assigned to the transaction if the applicable customer has a credit classification that matches one of the credit classifications defined in your revenue policy.
- **Payment Terms:** The contingency is assigned to the transaction if its payment terms exceed the payment terms threshold of your revenue policy.
- **Refund:** The contingency is assigned to the transaction if it includes a refund policy that exceeds the refund policy threshold of your revenue policy.

Select None if you don't want to consider any details of your revenue policy for the contingency.

## When do I create revenue contingency assignment rules?

You must create revenue contingency assignment rules if you want to automatically assign contingencies to transactions. For each rule that you define, you specify one or more matching criteria. Whenever the rule criteria match, Oracle Fusion Receivables assigns the specified contingency to the applicable transaction lines.

There are two cases where you don't need to create revenue contingency assignment rules:

- **Revenue policy violations:** Receivables automatically assigns the appropriate revenue contingencies to transactions whenever any revenue policy is violated.
- **Deferred revenue scheduling rules:** Transactions assigned a deferred revenue scheduling rule have all revenue deferred until either the related contingency expires or you manually schedule revenue.

# 11 Define Cash Management and Banking Configuration

## Bank, Branch, and Account Components: How They Work Together

Banks, branches, and accounts fit together on the premise of the Bank Account model.

The model enables you to define and keep track of all bank accounts in one place and explicitly grant account access to:

- multiple business units
- functions
- users

This eliminates the redundant duplicate bank account setup under different business units when these business units share the same bank account.

### Banks

Creating a bank is the first step in the bank account creation. You can:

- Search for existing banks to view and update
- Create a new bank from an existing party

Consider the following:

- The option to create from an existing party is implicitly implemented by the matching option.
- The option is available only after the existing party has been found with the same bank.
- If you select the matching option, the page repopulates the information from the matched party.

### Branches

Once you have created your bank, the next step is creating a branch or branches associated to the bank. The matching option is also available when creating branches. To create a new branch without using the matching option, manually enter the required information. You can also define other branch- related attributes in the same page.

If you don't use the matching option when an existing party is found, a branch with the same party name is created.

### Accounts


The four areas associated with defining an account are:

- General information
- Control of the account
- Security and access to the account

- Business unit assignment

Once the bank and branch are created, proceed to the bank account setup by doing the following:

- Select the bank branch you want to associate to your bank account.
- Assign the owner of the bank account.

 **Note:** To create a bank account for Payables or Receivables, add the Business Unit Access first for the business units to use the bank account.

Consider the following:

- The Oracle Fusion Account Payables or Receivables accounts are identified by the business unit.
- The Oracle Fusion Payroll accounts are identified by the legal entity.

## Creating Accounts: Points to Consider

Banks, branches and accounts fit together on the premise of the Bank Account model. The Bank Account model enables you to define and keep track of all bank accounts in one place.

The Bank Account Model can explicitly grant account access to multiple business units, functions, and users. Consider the following when you set up bank accounts:

- Assign a unique general ledger cash account to each account, and use it to record all cash transactions for the account. This facilitates book to bank reconciliation.
- Grant bank account security. Bank account security consists of bank account use security, bank account access security, and user and role security.

## Account Use

Account Use refers to accounts created for:


- Oracle Fusion Payables
- Oracle Fusion Receivables
- Oracle Fusion Payroll

Select the appropriate use or uses when creating an account in one or more of these applications.

## Account Access


Payables and Receivables account access is secured by business unit. Before the bank account is ready for use by Payables or Receivables, you must:

1. Select the appropriate use for the application.
2. Grant access to one or more business units.

 **Note:** You can only assign access to the business units that use the same ledger as the bank accounts owning the legal entity,

## User and Role Security

You can further secure the bank account so that it can only be used by certain users and roles. The default value for secure bank account by users and roles is No. For Payables and Receivables, you must have the proper business unit assigned to access a bank account even if the secure bank account by users and roles is No. If the secure bank account by users and roles is set to Yes, you must be named or carry a role assigned to the bank account to use it.

 **Note:** You must assign the security duty role Cash Management Administration to the Cash Manager job role to provide access for setting up banks, branches, and accounts.

## Parse Rule Sets: Overview

Oracle Fusion Cash Management supports parse rule sets to transform data during the bank statement import process. Parse rules are used to move data from one field to another. The parse rule set is associated to a bank account in the bank account setup. The parse rule set is most commonly used to parse data from the statement line addenda field into more specific statement line fields. Each parse rule within a parse rule set consists of the following fields:

- **Sequence:** Determines the order in which to process the rules.
- **Transaction Code:** The code used to determine the statement line type.
- **Source Field:** The interface table field that contains the data to be parsed.
- **Target Field:** The statement line field that the data is to be parsed to.
- **Rule:** Contains the syntax for determining the data within the source field to be parsed.
- **Overwrite:** Used to control whether to overwrite existing data in a target field or skip parsing the data.

The parse rule syntax is described below:

[LITERAL](<[MATCHING TOKEN],[START-END]>)[LITERAL]

Where

LITERAL represents a string or character value represented by an identifier that should match the source data exactly.

MATCHING TOKEN represents a token (or set of tokens) which describes the data to extract. The following table lists the valid tokens with their descriptions:

Token	Description
N	Extract a valid number
.	Decimal position
X	Extract an alphanumeric
~	Extract everything in the source field from the parse position to either the end of the data or up to the next literal.
START	A position to begin extracting data, offset by the parse position. It must be a valid numeric.

Token	Description
END	A position to stop extracting data. END can be either a valid numeric or the ~ token.

Listed in the table below are some examples:

Description	Source Data	Rule	Target Data
Extract numeric rate data from a source field	EST/TRX RTE 3.76 USD/LIBOR CPTY: PRU	RTE (N.NN)	3.76
Extract value date from a source field	Dt.01/01/2011? Receipt	Dt.(1-10)? Receipt	01/01/2011
Extract check number from a source field	Account Number 1005	Account Number.(X~)	1005
Extract currency from a source field	\$^EUR:Dt	\$(1-3):Dt.	EUR
Extract the counterparty of an unknown string length from the same source field	EST/TRX RTE 3.76 USD/LIBOR CPTY:PRU	CPTY: (X~)	PRU
Extract the currency from the same source field using positional matching	PRU EST/TRX RTE 3.76 USD/ LIBOR CPTY PRU	RTE(7-9)	USD
Extract Contract ID from Additional Entry Information	TXT:AR:Receipt Num: CEF-1: For: 2010\$^USD:Dt01/01/2011Receipt Method: CE-Foreign: Receipt Type: Standard: BU:Vision Operations: Customer: World of Business: Account No.1001	Account Number(NNNN)	1001
Extract Transaction ID from Customer Reference	CustRef # A.23@orlc.com	CustRef (X~).com	# A.23@orlc

## Transaction Type Mapping: Overview

The transaction type mapping enables you to associate a cash transaction type to an application transaction.

The following must be created to associate and mapped to cash transaction types:

- Oracle Fusion Account Payables payment methods
- Oracle Fusion Account Receivables payment methods



- Oracle Fusion Payroll payment types

Assigning cash transaction types to application transactions result in a more efficient bank statement reconciliation process.

Bank statement lines are also associated with cash transaction types and matching rules can be created using this common attribute.

## Tolerance Rules: Overview

Tolerance rules enables you to specify date and amount tolerances that prevent or warn you when reconciliation would be a breach of a defined tolerance.

- **Amount tolerances** are most often used when reconciling foreign currency transactions where there may be differences due to rounding or fluctuations in the conversion rate. They can also be used if a bank includes a processing fee in the bank statement line amount.
- **Date tolerances** are primarily used for checks that may be issued on one day and not clear the bank until days or weeks later.

Consider the following when defining your tolerance rules:

- Applying tolerances you can automate the reconciliation and accounting for these types of transactions.
- If no date or amount tolerance is defined within a rule, it requires an exact match.
- For manual reconciliation, a tolerance rule can optionally be assigned to a bank account.
- For automatic reconciliation, a tolerance rule can be associated with a matching rule in the Rule Set setup and can be applied if the matching rule matches on date and amount or both.
- The one exception occurs when you assign a tolerance rule that includes amount tolerances to a non one to one match type matching rule. In this non one to one match, the amount tolerance is ignored and amounts must match exactly.

## Date Tolerance

Reconciliation date tolerances are defined as day ranges. The date tolerances are to validate that the source transaction date or dates are within a certain number of days before and after the bank statement line date or dates.

In manual reconciliation, if a date tolerance is specified in the tolerance rule assigned to the bank account it applies to all matching scenarios. In the event of a date tolerance breach, a warning message is displayed, but the user is allowed to reconcile the statement line or lines and the transaction or transactions. If no date tolerance is assigned or specified it is required to be an exact date match and a warning message is displayed.

In automatic reconciliation, a tolerance rule that includes date tolerances can be associated with a matching rule. If the matching rule matches on the date, then the date tolerance is applied. In this scenario a date tolerance breach prevents reconciliation.

## Amount Tolerance

Reconciliation amount tolerances can only be used in one to one matching scenarios for both manual and automatic reconciliation. No reconciliation amount tolerances are allowed in one to many, many to one, or many to many matching scenarios. In these scenarios the amount of the bank statement line or lines must be equal to the amount of the transaction or transactions. Reconciliation amount tolerances can be defined as percentage or amount ranges or both. If both percentages and amounts are applied, the application uses the most conservative tolerance depending upon the statement line amount.

For example, if the amount tolerance equals plus or minus \$5, the percentage tolerance equals plus or minus 1%, and the statement line amount is \$100, the application first calculates the percentage amount (1% of \$100 dollars = \$1). It then compares this to the \$5 amount and uses the smaller amount. In this case it is \$1 dollar, so to reconcile a transaction to this line it must be between \$99 and \$101.

In automatic reconciliation, a tolerance rule that includes percentage, amount, or both types of tolerance ranges can be associated with a matching rule. But the tolerance can only be applied if the matching rule is a one to one match type rule. In this scenario of a one to one type match, any amount difference within tolerance is automatically created as an external transaction in cash management.

## Reconciliation Matching Rules: Explained

Reconciliation Matching rules help you match bank statement lines and system transactions to minimize the need for manual intervention.

Define bank statement automatic reconciliation matching rules and assign them to bank statement automatic reconciliation rule sets. After you assign the rule sets to the bank account, the Autoreconciliation process picks up the reconciliation matching rules to achieve a higher match rate.

Specify the following for each matching rule:

- Transaction Sources: Payables, Receivables, Payroll, or External.
- Matching Type: One to One, One to Many, Many to One, or Many to Many. The following table explains the different matching types available in Oracle Fusion Cash Management:

Matching Type	Description
One to One	A bank statement line is matched with a system transaction and reconciled against each other
One to Many	A bank statement line is reconciled against many system transactions
Many to One	Many bank statement lines are grouped and reconciled against a system transaction
Many to Many	Many statement lines are grouped and reconciled against many system transactions

- Grouping Attributes: Used to group bank statement lines and system transactions based on the matching type you select. The combination of the attributes you select also determine what you can use as the matching criteria.

You can use date, transaction type, and reconciliation reference as matching criteria only after you select these as grouping attributes. The following table displays the required grouping attributes for a selected matching type:

Matching Type	Statement Line Grouping Attributes	System Transaction Grouping Attributes
One to One	Not applicable	Not applicable
One to Many	Not applicable	Grouping attributes required
Many to One	Grouping attributes required	Not applicable
Many to Many	Grouping attributes required	Grouping attributes required

In Many to One matching the grouping attributes are used to group bank statement lines. In One to Many matching the grouping attributes are used to group system transactions.


The following is a list of common grouping attributes that can be used to group bank statement lines:

- Transaction date
- Transaction currency
- Transaction type
- Reconciliation reference
- Transaction code
- Counterparty bank account

The following is a list of common grouping attributes that can be used to group system transactions:

- Bank deposit number
- Transaction type
- Counterparty bank account
- Counterparty name
- Payment file reference
- Payment method
- Receipt batch number
- Receipt class
- Reconciliation match date
- Reconciliation reference
- Remittance batch number
- Transaction currency
- Transaction date
- Transaction source
- Business unit

- **Matching Criteria:** Includes a list of commonly used matching attributes. You can simply select the attributes to include them in the matching rule you selected. The selected attributes define the matching conditions between the bank statement lines and the system transactions to be matched successfully when they're reconciled.

 **Note:** On the Create Reconciliation Matching Rule page the delivered setting for the matching type is One to One, and the check boxes for **Reconciliation Reference**, **Date** and **Transaction Type** are enabled. When you change the matching type to One to Many or Many to One or Many to Many, the check boxes are disabled.

The matching criteria attributes are:

- Amount
  - Date
  - Reconciliation reference
  - Transaction type
- **Advanced Matching Criteria:** Enables you to specify additional matching logic or filtering conditions that must be true for the bank statement lines and system transactions to match successfully. Consider the following:
  - You have the option to enable or disable the **Case Sensitive Comparison** check box while creating a condition.
  - The data type of the left-hand side attribute and the right-hand side attribute must be the same if they're selected to match in the criteria. For example, if Statement.Booking date is selected on the left-hand side of the criteria, then Transaction.Transaction date can be selected as the matching criteria on the right-hand side.
  - For literal expression type, the operand value should match the database value. For example: Statement.Transaction Type equals ACH

The list of statement and transaction attributes available in the Create Condition page differs according to the matching type you select. The following table lists some of the common statement and transaction attributes:

Statement Attributes	Transaction Attributes
Statement. Account servicer reference	Transaction. Bank deposit number
Statement. Additional entry information	Transaction. Business unit
Statement. Booking date	Transaction. Counterparty name
Statement. Check number	Transaction. Counterparty site
Statement. Clearing system reference	Transaction. Party bank account
Statement. Contract ID	Transaction. Payment file reference
Statement. Counterparty bank account	Transaction. Receipt batch number
Statement. Customer reference	Transaction. Reconciliation reference

Statement Attributes	Transaction Attributes
Statement.End to End ID	Transaction. Remittance batch number
Statement. Instruction ID	Transaction. Transaction currency
Statement. Reconciliation match amount	Transaction. Transaction date
Statement. Reconciliation reference	Transaction. Transaction source
Statement. Transaction ID	Transaction. Transaction type
Statement. Transaction currency	Transaction. Transaction method
Statement. Transaction type	Transaction. Receipt class
Statement. Value date	Transaction. Reconciliation match date

You can select one or multiple transaction sources in a rule. Consider the following:

- If multiple sources are selected in a one to one or many to one matching rule, the autoreconciliation program looks for a matching transaction across the selected sources.
- If multiple sources are selected in a one to many or many to many matching rule, the program first finds all available transactions across the selected sources and then applies grouping rule to the whole data pool. This means that the statement lines can be reconciled to a group that includes transactions across the different sources.
- If you want transactions included in a group to be from the same transaction source then you can specify Transaction Source as a grouping attribute.

#### Related Topics

- [Automatic Reconciliation: Explained](#)

## Reconciliation Rules Sets: Overview

Bank statement reconciliation rule sets are a group of matching rules and tolerance rules. They are assigned to a bank account and used to reconcile bank statement lines with transactions.

Consider the following when creating your rules:

- Build the rule set and the rule set detail as a parent-child relationship.
- Each rule set consists of one or more matching rules that can be prioritized or sequenced.
- The rules should be ordered to achieve a greater reconciliation success rate. It is strongly recommended that one to one rules be sequenced above rules of other types.
- To provide an optimum reconciliation rate, you should change the sequence number depending on how accurately the given rule is likely to reconcile against the correct bank transactions.

For example, transactions from sources for which the bank provides you a reference ID are likely to have a higher reconciliation rate. These rules should be placed at the top with a lower sequence number. Conversely, transactions with no reference ID are likely to have duplicates or lower reconciliation rates, and you should place them at the bottom with a higher sequence number.

## Bank Statement Transaction Codes: Overview

Bank statement transaction codes are the internal codes that are used on a bank statement line to identify the type of transaction being reported. These are also referred to as:

- Transaction codes
- Statement codes

Oracle Fusion Cash Management maintains a single set of these codes and transform externally reported transaction codes from other formats into this single normalized set. This configuration is done through Oracle Fusion Payments code map setup.

## Bank Statement Transaction Creation Rules: Overview

Bank Statement Transaction Creation Rules are used by Oracle Fusion Cash Management to identify an unreconciled bank statement line or lines and create and account for a transaction.

Configure **Bank Statement Transaction Creation Rules** by specifying some of the attributes and characteristics of the created transactions. Consider the following when configuring your rules:

- Create as a separate business object.
- Assign to a bank account in the Manage Bank Account page.
- Arrange in order and group to be processed sequentially.

The group of sequenced rules on the bank account constitutes the bank accounts rule set that is used when running the Bank Statement Transaction Creation program.

Process the **Bank Statement Transaction Creation Rules** by running the Bank Statement Transaction Creation program to create transactions from unreconciled bank statement lines. The program is used to create transactions and account for first notice items such as bank charges, fees, or interest. You must perform the following prior to running the program.

- Run autoreconciliation for the bank statement.
- Perform any manual reconciliation on the bank statement.

This avoids creating external transaction from bank statement lines that already have transactions recorded in the application.

## Create Banks, Branches, and Accounts in Spreadsheet

## Cash Management Rapid Implementation: Overview

Use Microsoft Excel templates to rapidly implement the following setup objects:

- Banks
- Bank Branches
- Bank Accounts

## Functional Setup Manager Tasks

The following are the Functional Setup Manager tasks that are required to be performed to rapidly create the setup objects data:

- Create Banks, Branches, and Accounts in Spreadsheet: Downloads the rapid implementation excel spreadsheet template. Enter the bank, branch, and bank account data in this spreadsheet, and generate the data file to be loaded.
- Upload Banks, Branches, and Accounts: Launches the Upload Banks, Branches, and Accounts process with the data file to be uploaded as the parameter. You must upload the data file generated from the previous task.

## Preparing Data

Prepare your bank, branch, and account information to enter into the spreadsheet template.

- Bank information requires the country, name, and number.
- Branch information requires name, number, BIC code, and alternate name.
- Account information requires name, number, currency, legal entity, type, and IBAN.

After you finish preparing the data in the spreadsheet, click the Generate Banks, Branches, and Accounts File button. Save the generated XML file.

## Loading Data

Use the following steps to load your data.

- From Functional Setup Manager, search for and select Upload Banks, Branches, and Accounts task. This task launches the Upload Banks, Branches, and Accounts process.
- Select the XML file you have saved earlier and submit the process.
- Verify in the process monitor that the process completed successfully.
- Review the banks, branches, and accounts created.

## Best Practices

The following are recommended best practices:

- Determine the Legal Entity for each bank account. The Legal Entity must be associated to a primary ledger.
- Determine the use for each bank account: Payable, Receivable, or both.
- Determine the Cash and Cash Clearing account for each bank account. Enter the entire account combination based on your chart of accounts, for example 01-000-1110-0000-000.

### Related Topics

- Processing Electronic Bank Statements: Explained

## Setting Up Cash Positioning and Forecasting

### Setting up Oracle Fusion Payments for Cash Management: Worked Example

To make ad hoc payments you must do the following:

- Create payee in Oracle Fusion Cash Management.
- Review Payment Methods under the tab; Usage Rules, for Cash Management in Oracle Fusion Payments.
- Review the payment method defaulting rules in Oracle Fusion Payments and prioritize the Cash Management Payment Method accordingly.

Creating a payee in Cash Management is a separate task than setting up suppliers in procurement. The setup done in Cash Management is strictly used for making ad hoc payments from the application. You must also review and edit the set ups in Payments to successfully make payments.

### Creating a Payee in Cash Management

- Create the payee in Cash Management. Enter the following payee information:

Field	Required	Description
Name	Yes	Name of the payee.
Tax Registration Number	No	Description of the payee.
Tax Registration Number	No	Unique identifier assigned to a payee by a tax authority.
Active	No, but recommended.	Check box indicating if the payee is active or inactive.

- Create the bank account information. Enter the following bank account information:

Field	Description
Country	Country of the bank where the bank account belongs.
Account Number	Bank account number of the payee bank account.
Currency	Currency of the payee bank account



Field	Description
Account Type	Type of payee bank account. For example, checking or savings.
Check Digit	The account number validation.
Account Name	Name of the bank account holder.
Secondary Account Reference	Additional account reference such as the Building Society Role Number in the UK.
Bank	Name of the payee bank.
Bank Branch	Name of the payee bank branch.
Routing Transit Number	The routing transit number for electronic transfers.
BIC Code	The code used to SWIFT to identify the bank or bank branch.
Active	Flag to indicating if the bank account is active. The default is set to active.

3. Click the Save and Close button to save your information.

## Setting Up Usage Rules in Oracle Fusion Payments for Cash Management

1. Navigate to Oracle Fusion Payments and the Create Payment Methods page.
2. Enter the following required fields: Name, Code, and From Date.
3. Select the Usage Rules tab.
4. Select the Cash Management tab
5. Select the check box Enable for use in Cash Management.
6. Determine select All or Specific.
7. Review the delivered Payment Process Transaction Types for Cash Management. Apply the appropriate payment types. The valid values are:
  - o Bank Account Transfer
  - o Ad Hoc Payment
8. Review the payment method defaulting rules in Oracle Fusion Payments and prioritize the Cash Management Payment Method accordingly.

### Related Topics

- [Payment Methods: Explained](#)
- [Payment Method Defaulting: Explained](#)
- [Usage Rules: Explained](#)

## Setting Up Cash Positioning and Forecasting: Explained

Use the following to set up your cash positioning and forecasting reporting requirements:


- Specify Cash Positioning and Forecasting Options
- Manage Cash Positioning and Forecasting Transaction Grouping

### Specify Cash Positioning and Forecasting Options

Use the options page to define the extraction period used to transfer data to the Essbase cube. Transactions with transaction dates within that period are extracted to the cube. You can also select different GL accounting calendars to lay out the time dimension structure in the cubes.

Configure the following options:

Field Name	Available Values	Default Value
Extraction Duration	<ul style="list-style-type: none"> <li>• Last 3 months</li> <li>• Last 6 months</li> <li>• Last 1 year</li> <li>• Last 2 years</li> <li>• Last 3 years</li> </ul>	Last 2 years
Reporting Currency	List of currencies defined in the application	USD - US Dollars
Balance Code	Internal Balance Codes lookups (LOV)	Closing booked
Balance Date Threshold Days	Number days defined before a missing bank statement is reported	2
Transaction Calendar	List of transaction calendars defined in the application, if not defined, everyday (7) is considered a business day.	No default
Time Periods	List of accounting calendars defined in the application	No default

 **Note:** Once the cube is created and locked, the update is disabled in this page. You can't update the cube until you submit the Cash Position Data Deletion program to clear the details in the cube.


### Manage Cash Positioning and Forecasting Transaction Grouping

You create or edit custom dimensions for cash positioning and forecasting from this page. You must have the Manage Cash Positioning and Forecasting Transaction Grouping privilege to access the Create or Edit Cash Position Dimension page:

- Search for the custom dimensions defined in the application.
- Create custom dimensions to meet company-specific requirements
- Modify and edit custom dimension to meet reporting requirements.
- Entering a description is optional but recommended.

The following table contains the fields to be completed:

Field Name	Description
Name	Required and must be unique
Application	Required and valid values are Oracle Fusion Applications or Other.
Source	Required and the following are possible values: <ul style="list-style-type: none"><li>• Payables invoices</li><li>• Payables payments</li><li>• Receivables receipts</li><li>• Receivables transactions</li><li>• Bank statement</li><li>• External cash transactions</li></ul>
Source Table	List of tables from the selected Application and Source
Source Column	List of the columns from the selected Source Table.

 **Note:** Once the cube is created and locked, you can't update this page. You must submit the Cash Position Data Deletion program to clear the details in the cube to do an update.

#### Related Topics

- [Cash Positioning and Forecasting: Explained](#)
- [Cash Positioning: Explained](#)
- [Cash Forecasting: Points to Consider](#)

## Bank Account Validation

## Bank Account Validation by Country: Andorra to Guadeloupe

This outlines the country specific bank account validation rules performed in Oracle Fusion Cash Management.

The following countries have country specific validations:

- Andorra
- Australia
- Austria
- Belgium
- Bosnia and Herzegovina
- Brazil
- Bulgaria
- Canada
- Columbia
- Croatia
- Cyprus
- Czech Republic
- Denmark
- Estonia
- Finland
- France
- French Guiana
- Germany
- Gibraltar
- Greece
- Guadeloupe

When entering bank accounts, different countries can have certain rules governing the format and content of the following related fields:

1. **Bank Code**
2. **Branch Number**
3. **Account Number**
4. **Check Digit**
5. **IBAN**

Use the **Disable Country Specific Bank Validations** profile option to disable the country-specific validations pertaining to the bank code, branch number, account number, check digit, and IBAN. You can set this profile option at the site, product, or user level. The profile is predefined with a default value of **No** at the site level. If the profile is set to **Yes**, these validations

are not performed. The checks for unique banks, branches, accounts, and the mandatory requirement of bank account number are not affected by this profile.

## Andorra

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Optional, if entered, the below rules apply.</li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 24 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Australia

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, the length should be either 2 or 3 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>The combined length of the Branch Number and Bank Code should be 6 numeric characters. Hence, the valid length values (3,4,6) depend upon the Bank Code (3,2,0).</li> <li>This field is labeled as <b>Bank State Branch</b>.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be between 5 to 10 characters.</li> <li>If the account currency is Australian Dollar, account number should be numeric. For foreign currencies, alphanumeric values are allowed</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Optional, if entered, the below rules apply.</li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> </ul>

Field	Rule
	<ul style="list-style-type: none"> <li>The third and fourth characters are numbers.</li> </ul>

## Austria

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>Length should be of 5 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>Length should be of 5 numeric characters.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be between 4 to 11 numeric characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 20 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Belgium

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be of 12 numeric characters.</li> <li>It should be in the format 999-9999999-99.</li> <li>A check algorithm is applied on the Account Number.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>

Field	Rule
	<ul style="list-style-type: none"> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 16 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

#### Check Algorithm for **Account Number**

- The entered check digit **CD1**, is the last two digits of the Account Number
- The calculated check digit **CD2**, is derived by concatenating the first two sections of the Account Number and calculating the remainder on dividing this by 97. If the remainder is equal to **0**, then the calculated check digit is taken to be **97**.
- If the entered check digit (CD1) and calculated check digit (CD2) are equal, then the Account Number is valid, else the check has failed.
- Additionally, if the entered check digit (that is, the last section) is '00', then the Account Number is invalid because the calculated check digit can never be **00** as per the 3rd point.

Example using account number 123-4567890-78

- The entered check digit (CD1) is '78'. The concatenation of the first two sections gives '1234567890'
- Divide the result by '97'.  $1234567890 / 97 = 12727504$
- Derive the remainder.  $1234567890 - (12727504 * 97) = 2$  Therefore CD2 = 2
- Here CD1  $\neq$  CD2, therefore the Account Number is not valid.

In this case, a valid Account Number would be '123456789-02'.

## Bosnia and Herzegovina

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 20 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> </ul>

Field	Rule
	<ul style="list-style-type: none"> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Brazil

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 3 numeric characters.</li> <li>If the length is less than 3, then it is converted to a 3 digit number by prefixing it with as many leading zeroes as is necessary.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 5 numeric characters.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Company Code</b>	<ul style="list-style-type: none"> <li>Optional.</li> <li>This is entered in the Account Creation form.</li> <li>If entered, length should be a maximum of 15 numeric characters</li> </ul>
<b>Secondary Account Reference</b>	<ul style="list-style-type: none"> <li>This field is labeled as <b>Company Code</b>.</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Optional, if entered, the below rules apply.</li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Bulgaria

### Validation Rules



The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 22 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Canada

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>This field is labeled as <b>Routing Transit Number</b>.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Optional, if entered, the below rules apply.</li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Columbia

For Colombia, there are no validations for Bank Code, Branch Number, Account Number, or Check Digit fields

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Tax Payer ID</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>Length should be a maximum of 15 numeric characters 14 digits for Tax Payer ID plus the last digit for check digit.</li> <li>It is unique within the country.</li> <li>Cross Validations of Tax Payer ID in Customers, Suppliers, and Companies. If the Tax Payer ID is used by a Customer, Supplier, or a Company, then the Customer name, Supplier name, or the Company name should match with the Bank name.</li> <li>A check digit is applied on the Tax Payer ID.</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Optional, if entered, the below rules apply.</li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

Check Algorithm for Tax Payer ID

The first 15 digits are multiplied by the associated factor.

Digit	Factor
1st	71
2nd	67
3rd	59
4th	53
5th	47
6th	43
7th	41

Digit	Factor
8th	37
9th	29
10th	23
11th	19
12th	17
13th	13
14th	7
15th	3

1. These 15 products are added and the sum is divided by 11.
2. If the remainder is 1 or 0, then the Check Digit should be 1 or 0 respectively.
3. If the remainder is not 1 or 0, then the remainder is subtracted by 11 and that should be the Check Digit.

## Croatia

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>• Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>• Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>• Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>• The module-97 rule is used to calculate the validity of the IBAN.</li> <li>• Length should be 21 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>• The first 2 characters are letters.</li> <li>• The third and fourth characters are numbers.</li> </ul>

## Cyprus

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 28 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Czech Republic

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 24 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Denmark

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Length should be a maximum of 10 numeric characters</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 18 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Estonia

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 20 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Finland

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, it should be 6 numeric characters.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be between 8 to 14 numeric characters.</li> <li>A check algorithm is applied on the Account Number.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, it should be 1 numeric digit.</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 18 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

If 1st digit of Account Number is:	Check Value Method
1	1
2	1
3	1
4	2
5	2
6	1
7	2
8	1
9	1

## Method 1

The check is formed in the following two parts:

- The first part of the check is formed from the first 6 digits of the Account Number. To illustrate, if the account number is 123456789, then the first part of check would be created as 123456.
- The second part of check is formed as an eight digit value, comprising the 8th to 15th digits of the Account Number. If the length is less than 8, then it is converted to an 8 digit number by prefixing it with as many leading zeroes as is necessary. Using the same example, the second part of check would be created as 00000089. check is then formed by concatenating the two parts. So, in our example the check is formed as 12345600000089.

## Method 2

The check is formed in the following three parts:

- The first part of the check is formed from the first 6 digits of the Account Number. To illustrate, if the account number is 123456789, then the first part of check would be created as 123456.
- The second part of check is formed as the 8th digit of the Account Number. Using the same example, the second part of check would be created as 8.
- The third part of check is formed as a seven digit value, comprising the 9th to 15th digits of the Account Number. If the length is less than 7, then it is converted to a 7 digit number by prefixing it with as many leading zeroes as is necessary. Using the same example, the second part of check would be created as 0000009. The check is then formed by concatenating the three parts. So, in our example the check is formed as 12345680000009.

A computed sum is then calculated based on the value of the check. Different calculations are performed depending on the first two digits of the formed check value.

If the first two digits of the check are '88', then:

- The Finnish government provides the following factor table. The 8th to 13th digits of the check number are multiplied by the associated factor. The computed sum is then calculated by summing the totals.

Digit	Factor
8th	1
9th	3
10th	7
11th	1
12th	3
13th	7

Example using check number 88345600000089: Multiply the given digits with the given factor.

Digit	Value	Factor	Result
8th Digit	0	1	0

Digit	Value	Factor	Result
9th Digit	0	3	0
10th Digit	0	7	0
11th Digit	0	7	0
12th Digit	0	3	0
13th Digit	8	7	56
<b>Total</b>			56

So the computed sum for this example is 56.

The test fails unless either of the following applies:

- The 14th digit of the check should equal the value of 10 minus the last digit of the computed sum. For the check value is '88345600000089', the last digit of the computed sum is 6. So  $10 - 6 = 4$ . So, the 14th digit of the check should equal 4. The test fails here as the 14th digit is 9.
- Both the 14th digit of the check and the last digit of the computed sum are 0. Using the same example, the test fails here as both values are not 0.

If the first two digits of the check are NOT '88', then the computed sum is calculated for each of the first 13 digits by adding the even numbered digits to the following calculated sum for each odd numbered digit :

- Multiply the digit by 2.
- Divide the result by 10.
- From the result add the integer to the remainder.

**Example using account number 123456800000089:**

Digit	Value	Multiply (a)	Divide (b)	Integer	Remainder	Result
1st	1	2	0.2	0	2	2
2nd	2					2
3rd	3	6	0.6	0	6	6
4th	4					4
5th	5	10	1	1	0	1
6th	6					6
7th	0	16	1.6	1	6	0



Digit	Value	Multiply (a)	Divide (b)	Integer	Remainder	Result
8th	0					0
9th	0	0	0	0	0	0
10th	0					0
11th	0	0	0	0	0	0
12th	0					0
13th	8	16	1.6	1	6	7
<b>Total</b>						28

The computed sum is then converted using the following process, before being used to see if the Account Number is valid:

1. Computed sum is added to 9.
2. The result is divided by 10.
3. The integer result is multiplied by 10.
4. The result is subtracted by the original computed sum.

So the computed sum '282 is converted to '2' as:

1.  $28 + 9 = 37$
2.  $37/10 = 3.7$ . Integer result therefore = 3
3.  $3 * 10 = 30$
4.  $30 - 28 = 2$

This number is then compared to the 14th digit of the Account Number. If it matches, then the test is passed, else it is failed.

In our example, the test fails as the 14th digit of the account number is 9. If the 14th digit had been 2, then the test would have been passed.

## France

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• Length should be a maximum of 5 numeric characters.</li> <li>• If the length is less than 5, then it is converted to a 5 digit number by prefixing it with as many leading zeroes as is necessary.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• Length should be a maximum of 5 numeric characters.</li> </ul>

Field	Rule
	<ul style="list-style-type: none"> <li>If the length is less than 5, then it is converted to a 5 digit number by prefixing it with as many leading zeroes as is necessary.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 11 numeric characters</li> <li>Special characters and spaces are not allowed</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, length should be a maximum of 2 numeric characters.</li> <li>A check algorithm is applied on the check digit.</li> </ul>
<b>Account Type</b>	<ul style="list-style-type: none"> <li>This field is labeled as <b>Deposit Type</b>.</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 27 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

#### Check Algorithm for Check Digit

A check digit is calculated (CD1) from the Account Number, Bank Code, and Branch Number in the following manner. This is then used as the basis for the check digit validity test.

#### CDI

For the check algorithm, the digits of the Account Number entered as characters A to Z. are converted to numeric values, the French government provides the following conversion table:

Value	Conversion
A, J	1
B, K, S	2
C, L, T	3
D, M, U	4
E, N, V	5
F, O, W	6
G, P, X	7
H, Q, Y	8

Value	Conversion
I, R, Z	9

Example using account number A1234567890:

The letter A is converted by applying the above table to 1, so the account number becomes 11234567890.

A value for CD1 is formed by joining together the bank fields in the following way:

- The Bank Code is concatenated with Branch Number concatenated to the converted Account Number. To illustrate with the Bank Code as 12345, the Branch Number as 67890 and the converted Account Number as 11234567890. Then CD1 is created as 123456789011234567890.
- To this concatenated value, 00 is added as a suffix and the resulting value is divided by 97. The remainder obtained as result of this division is then subtracted from 97. The result of this subtraction is the calculated check digit.
- In our example, suffixing 00 gives 12345678901123456789000. Dividing by 97 and deriving the remainder.  $\text{Mod}(12345678901123456789000, 97) = 86$  Subtract from 97.  $97 - 86 = 11$
- If the user entered Check Digit is equal to this calculated value, then the validation is successful.

In the given example, as the user entered check digit is not 11, the check is not valid.

## French Guiana

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>• Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>• Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>• Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>• The module-97 rule is used to calculate the validity of the IBAN.</li> <li>• Length should be 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>• The first 2 characters are letters.</li> <li>• The third and fourth characters are numbers.</li> </ul>

## Germany

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be 8 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be 8 numeric characters.</li> <li>If the Bank Code and Branch Number are entered, then both values must match.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 10 numeric characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If a value is entered for the check digit, then it must be a single digit and must match the last digit of the Account Number.</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 22 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Gibraltar

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 23 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Greece

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be of 3 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be of 4 numeric characters.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be between 8 to 16 alphanumeric characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If a value is entered, then it must be one numeric character.</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 27 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Guadeloupe

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 34 characters. Leading and trailing spaces are ignored. There should be no spaces in the middle. .</li> <li>The first 2 characters are letters.</li> </ul>

Field	Rule
	<ul style="list-style-type: none"><li>The third and fourth characters are numbers.</li></ul>

## Bank Account Validation by Country: Hungary to Norway

This outlines the country-specific bank account validation rules performed in Oracle Fusion Cash Management.

The following countries have country-specific validations:

- Hungary
- Iceland
- Ireland
- Israel
- Italy
- Japan
- Kuwait
- Latvia
- Liechtenstein
- Lithuania
- Luxembourg
- Malta
- Martinique
- Mauritius
- Mayotte
- Mexico
- Monaco
- Montenegro
- Netherlands
- New Zealand
- Norway

When entering bank accounts, different countries can have certain rules governing the format and content of the following related fields:

- Bank Code**
- Branch Number**
- Account Number**
- Check Digit**
- IBAN**

Use the **Disable Country Specific Bank Validations** profile option to disable the country-specific validations pertaining to the bank code, branch number, account number, check digit, and IBAN. You can set this profile option at the site, product, or user level. The profile is predefined with a default value of **No** at the site level. If the profile is set to **Yes**, these validations are not performed. The checks for unique banks, branches, accounts, and the mandatory requirement of bank account number are not affected by this profile.

## Hungary

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 28 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Iceland

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be of 4 numeric characters.</li> <li>If the length is less than 4, then it is converted to a 4 digit number by prefixing it with as many leading zeroes as is necessary.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be of 4 numeric characters.</li> <li>If the Bank Code and Branch Number are entered, then both values must match.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 18 numeric characters.</li> <li>If the length is less than 18, then it is converted to an 18 digit number by prefixing it with as many leading zeroes as is necessary.</li> </ul>

Field	Rule
	<ul style="list-style-type: none"> <li>A check algorithm is applied on the Account Number.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If a value is entered for the check digit, then it must be a single digit and must match the seventeenth digit of the Account Number.</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 26 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

### Check Algorithm for Account Number

1. Check algorithm is performed against the Account Number (from digit 9 to 16). Each of these digits is multiplied with the factors as given in the following table:

Digit	Factor
9th	3
10th	2
11th	7
12th	6
13th	5
14th	4
15th	3
16th	2

These products are added and the sum is divided by 11. The remainder obtained as a result of this division is subtracted from 11 to obtain the calculated check digit. If remainder is 0, then calculated check digit is taken as 0.

This calculated check digit should match the entered check digit (seventeenth digit of the Account Number), else the Account Number is not valid.

## Ireland

### Validation Rules



The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be of 6 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be of 6 numeric characters.</li> <li>If the Bank Code and Branch Number are entered, then both values must match.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 8 numeric characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 22 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Israel

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If entered, the length should be a maximum 2 numeric characters</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be 3 numeric characters.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be 9 numeric characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Optional, if entered, the below rules apply.</li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Italy

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• Length should be a maximum of 5 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• Length should be a maximum of 5 numeric characters.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• Length should be a maximum of 12 alphanumeric characters.</li> <li>• If the length is less than 12, then it is converted to a 12 digit number by prefixing it with as many leading zeroes as is necessary.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>• Optional</li> <li>• If entered, length should be a single alphabetic character and a check algorithm is applied on the Check Digit.</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>• The module-97 rule is used to calculate the validity of the IBAN</li> <li>• Length should be 27 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>• The first 2 characters are letters.</li> <li>• The third and fourth characters are numbers.</li> </ul>

### Check Algorithm for Check Digit

The check digit is used to validate against the Bank Code, Branch Number, and Account Number. These are concatenated to obtain a 22 character string.

Each character is assigned a value depending upon whether the character is in an odd position or an even position in the string as given in the following table:

Even Position Values	Odd Position Values
A/0 = 0	A/0 = 1
B/1 = 1	B/1 = 0
C/2 = 2	C/2 = 5
D/3 = 3	D/3 = 7
E/4 = 4	E/4 = 9

Even Position Values	Odd Position Values
F/5 = 5	F/5 = 13
G/6 = 6	G/6 = 15
H/7 = 7	H/7 = 17
I/8 = 8	I/8 = 19
J/9 = 9	J/9 = 21
K = 10	K = 2
L = 11	L = 4
M = 12	M = 18
N = 13	N = 20
O = 14	O = 11
P = 15	P = 3
Q = 16	Q = 6
R = 17	R = 8
S = 18	S = 12
T = 19	T = 14
U = 20	U = 16
V = 21	V = 10
W = 22	W = 22
X = 23	X = 25
Y = 24	Y = 24
Z = 25	Z = 23

The first character on the left is an odd position. The values assigned are added up and the sum is divided 26.

The remainder obtained as a result of this division is converted into an alphabet as given in the following table:

#### Transformation Algorithm

Calculation	Calculation	Calculation
0 = A	9 = J	18 = S
1 = B	10 = K	19 = T
2 = C	11 = L	20 = U
3 = D	12 = M	21 = V
4 = E	13 = N	22 = W
5 = F	14 = O	23 = X
6 = G	15 = P	24 = Y
7 = H	16 = Q	25 = Z
8 = I	17 = R	

This value should be the same as the user entered check digit or else the Check Digit validation fails.

## Japan

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be 4 numeric characters</li> </ul>
<b>Alternate Bank Name</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be 3 numeric characters.</li> </ul>
<b>Alternate Branch Name</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
Account Type	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>

Field	Rule
	<ul style="list-style-type: none"> <li>This field is labeled as <b>Deposit Type</b>.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Optional, if entered, the below rules apply.</li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Kuwait

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, the length should be a maximum of 4 characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 21 characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Latvia

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 21 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Liechtenstein

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 21 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Lithuania

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 20 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Luxembourg

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be 3 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be 3 numeric characters.</li> <li>If the Bank Code and Branch Number are entered, then both values must match.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be 13 alphanumeric characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be 2 numeric characters</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 20 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Malta

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 31 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Martinique

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>



## Mauritius

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 30 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Mayotte

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Mexico

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
Secondary Account Reference	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered: <ul style="list-style-type: none"> <li>Should be of 18 digits</li> <li>Should be numeric</li> </ul> </li> </ul>

## Monaco

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Montenegro

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 22 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Netherlands

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Two types of account numbers are validated:</li> <li>If the bank account number is numeric and consists of one of the following then bank account will be considered as Post or Giro Account. <ul style="list-style-type: none"> <li>length is 7 digits or less, or</li> <li>prefixed with 000, or</li> <li>prefixed with P or G</li> </ul> </li> </ul> <p>There is no check digit validation for Post or Giro accounts.</p>

Field	Rule
	<ul style="list-style-type: none"> <li>For other account numbers, the length should be between 9 and 10 numeric characters. A check algorithm is applied on the Account Number.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 18 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

### Check Algorithm for Non-Post or Giro Account Number

1. If the length is less than 10, then it is converted to a 10 digit number by prefixing it with as many leading zeroes as is necessary.
2. The Netherlands government provides the following factor table for each of the 10 digits:

Digit	Factor
1st	10
2nd	9
3rd	8
4th	7
5th	6
6th	5
7th	4
8th	3
9th	2
10th	1

These are multiplied and the sum of the products is calculated 4.

If the result so obtained is perfectly divisible by 11 (that is, no remainder on division by 11), then the test is successful, otherwise the account number entered is not valid.

## New Zealand

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• Length should be 2 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• Length should be 4 numeric characters.</li> <li>• This field is labeled <b>Bank State Branch</b>.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>• Optional</li> </ul>
<b>Description</b>	<ul style="list-style-type: none"> <li>• This field is labeled <b>Reference</b>.</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>• Optional, if entered, the below rules apply.</li> <li>• The module-97 rule is used to calculate the validity of the IBAN.</li> <li>• Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>• The first 2 characters are letters.</li> <li>• The third and fourth characters are numbers.</li> </ul>

## Norway

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>• Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>• Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• Length should be of 11 numeric characters.</li> <li>• A check algorithm is applied on the Account Number, if the 5th and 6th digits of the account number are not 00.</li> </ul> <p>For example, for Account Number, <b>1234001234</b>, the check algorithm will not be applied but for Account Number 02056439653, the check algorithm will be applied as outlined below.</p>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>• Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> </ul>

Field	Rule
	<ul style="list-style-type: none"> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 15 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

#### Check Algorithm for Account Number

- The check digit is set as the last (that is, the 11th digit) of the Account Number. For example, if the account number is 02056439653, then the check digit is set to 3.
- The Norwegian government provides the following factor table:

Digit	Factor
1st	5
2nd	4
3rd	3
4th	2
5th	7
6th	6
7th	5
8th	4
9th	3
10th	2

The first ten digits of the account number are multiplied by the associated factor. The computed sum is then calculated by summing the totals.

- Example using account number 02056439653:

Multiply each digit with the given factor.

Digit	Value	Factor	Result
1st	0	5	0

Digit	Value	Factor	Result
2nd	2	4	8
3rd	0	3	0
4th	5	2	10
5th	6	7	42
6th	4	6	24
7th	3	5	15
8th	9	4	36
9th	6	3	18
10th	5	2	10
<b>Total</b>			163

So the computed sum for this example is 163.

4. The computed sum is then added to the check digit. In the above example,  $163 + 3 = 166$ .
5. Divide the result by 11.  $166 / 11 = 15 \text{ R } 6$ .
6. Derive the remainder.  $166 - (11 * 15) = 1$ .
7. If the remainder is '0', then the validation is successful, else the check fails.
8. In the given example, the check fails the Account Number as the remainder is 1. If the 11th digit of the Account Number was 2 (that is, the check digit would be 2), then the remainder would be  $165 - (11 * 15) = 0$  and the check on the Account Number would be successful.

## Bank Account Validation by Country: Poland to the United States

This outlines the country specific bank account validation rules performed in Oracle Fusion Cash Management.

The following countries have country specific validations:

- Poland
- Portugal
- Reunion
- Romania
- Saint Barthelemy
- Saint Martin
- Saint Pierre and Miquelon
- Saudi Arabia
- Serbia
- Serbia and Montenegro
- Singapore
- Slovakia
- Slovenia
- Spain
- Sweden
- Switzerland
- The Former Yugoslav Republic of Macedonia
- Tunisia
- Turkey
- United Arab Emirates
- United Kingdom
- United States

When entering bank accounts, different countries can have certain rules governing the format and content of the following related fields:

- 1. Bank Code**
- 2. Branch Number**
- 3. Account Number**
- 4. Check Digit**
- 5. IBAN**



Use the **Disable Country Specific Bank Validations** profile option to disable the country-specific validations pertaining to the bank code, branch number, account number, check digit, and IBAN. You can set this profile option at the site, product, or user level. The profile is predefined with a default value of **No** at the site level. If the profile is set to **Yes**, these validations are not performed. The checks for unique banks, branches, accounts, and the mandatory requirement of bank account number are not affected by this profile.

## Poland

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, the length should be of 8 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, the length should be of 8 numeric characters.</li> <li>If the Bank Code and Branch Number are entered, then both values must match</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 16 alphanumeric characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 28 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Portugal

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be of 4 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be of 4 numeric characters.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 11 numeric characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>Length should be of 2 numeric characters.</li> </ul>

Field	Rule
	<ul style="list-style-type: none"> <li>If entered, a check algorithm is applied on the Check Digit.</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 25 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

### Check Algorithm for Check Digit

- A check digit is formed (CD1) from the Bank Code, Branch Number, and Account Number by concatenating the three numbers.
- For example, using Bank Code 1234, Branch Number 5678, and Account Number 12345678901. Then CD1 is set as 1234567812345678901.
- The Portuguese government provides the following factor table:

Digit	Factor
1st	73
2nd	17
3rd	89
4th	38
5th	62
6th	45
7th	53
8th	15
9th	50
10th	5
11th	49
12th	34
13th	81

Digit	Factor
14th	76
15th	27
16th	90
17th	9
18th	30
19th	3

The nineteen digits of the created check digit (CD1) are multiplied by the associated factor. The multiple sum is then calculated by summing the totals.

Example using the above value for CD1:

Digit	Value	Factor	Result
1st	1	73	73
2nd	2	17	34
3rd	3	89	267
4th	4	38	152
5th	5	62	310
6th	6	45	270
7th	7	53	371
8th	8	15	120
9th	1	50	50
10th	2	5	10
11th	3	49	147
12th	4	34	136
13th	5	81	405

Digit	Value	Factor	Result
14th	6	76	456
15th	7	27	189
16th	8	90	720
17th	9	9	81
18th	0	30	0
19th	1	3	3
<b>Total</b>			3794

- Divide the result by 97.  $3794 / 97 = 39$
- Derive the remainder.  $3794 - (39 * 97) = 11$
- CD1 is then derived by subtracting the remainder from 97.  $97 - 11 = 86$ . So for this example CD1 = 86
- If the calculated value for CD1 is not the same as the user entered check digit, then the check digit fails the validation. In the given example, unless the user entered check digit is 86, the validation will fail.

## Reunion

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>• Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>• Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>• Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>• The module-97 rule is used to calculate the validity of the IBAN</li> <li>• Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>• The first 2 characters are letters.</li> <li>• The third and fourth characters are numbers.</li> </ul>

## Romania

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 24 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Saint Barthelemy

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Saint Martin (French Section)

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Saint Pierre and Miquelon

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Saudi Arabia

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be a maximum of 4 characters</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 25 characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Serbia

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Optional, if entered, the below rules apply.</li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 22 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Serbia and Montenegro

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Singapore

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be 4 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be 3 numeric characters.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 12 numeric characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Optional, if entered, the rules below apply.</li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>



## Slovakia

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length cannot be more than 24 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Slovenia

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length cannot be more than 19 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Spain

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• Length should be a maximum of 4 numeric characters.</li> <li>• If the bank code is less than 4 digits, then it is converted to a 4 digit number by prefixing it with as many leading zeroes as is necessary.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• Length should be a maximum of 4 numeric characters.</li> <li>• If the bank code is less than 4 digits, then it is converted to a 4 digit number by prefixing it with as many leading zeroes as is necessary.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• Length should be 10 numeric characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>• Optional</li> <li>• If entered, length should be a maximum of 2 numeric characters.</li> <li>• A check algorithm is applied on the Check Digit.</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>• Mandatory</li> <li>• If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>• The module-97 rule is used to calculate the validity of the IBAN</li> <li>• Length should be 24 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>• The first 2 characters are letters.</li> <li>• The third and fourth characters are numbers.</li> </ul>

### Check Algorithm for Check Digit

Two check digits are calculated, CD1 from the Bank Code and Branch Number and CD2 from Account Number in the following manner; these are then used as the basis for the check digit validity test:

#### CD1

1. For the Bank Code, the Spanish government provides the following factor table:

Digit	Factor
1st	4
2nd	8
3rd	5
4th	10

The four digits of the Bank Code are multiplied by the associated factor. The computed sum is then calculated by summing the totals.

Example using Bank Code '1234':

Multiply each digit with the given factor.

Digit Value Factor Result	Digit Value Factor Result	Digit Value Factor Result	Digit Value Factor Result
1st	1	4	4
2nd	2	8	16
3rd	3	5	15
4th	4	10	40
<b>Total</b>			75

So the computed sum for this example is 75.

2. For the Branch Number, the Spanish government provides the following factor table:

Digit	Factor
1st	9
2nd	7
3rd	3
4th	6

The four digits of the Branch Number are multiplied by the associated factor. The computed sum is then calculated by summing the totals.

Example using Branch Number '5678':

Multiply each digit with the given factor.

Digit	Value	Factor	Result
1st	5	9	45
2nd	6	7	42
3rd	7	3	21

Digit	Value	Factor	Result
4th	8	6	48
<b>Total</b>			156

So the computed sum for this example is 156.

**3.** The computed sums from both the Bank Code and Branch Number calculations are then summed up. From the above example, it is  $75 + 156 = 231$ .

**4.** Divide the result by 11.

$$231 / 11 = 21$$

**5.** Derive the remainder

$$231 - (11 * 21) = 0.$$

**6.** CD1 is then derived by subtracting the remainder from 11. If difference is 11, then CD1 is 0 and if difference is 10, then CD1 is 1  $11 - 0 = 11$ . So for this example, **CD1 = 11 = 0**.

## CD2

**1.** For the Account Number, the Spanish government provides the following factor table:

Digit	Factor
1st	1
2nd	2
3rd	4
4th	8
5th	5
6th	10
7th	9
8th	7
9th	3
10th	6

The ten digits of the bank number are multiplied by the associated factor. The computed sum is then calculated by summing the totals.

Example using account number '1234567890':

Multiply each digit with the given factor.

Digit	Value	Factor	Result
1st	1	1	1
2nd	2	2	4
3rd	3	4	12
4th	4	8	32
5th	5	5	25
6th	6	10	60
7th	7	9	63
8th	8	7	56
9th	9	3	27
10th	0	6	0
<b>Total</b>			280

So the computed sum for this example is 280.

**2.** Divide the result by 11

$$280 / 11 = 25$$

**3.** Derive the remainder.

$$280 - (11 * 25) = 5$$

**4.** CD2 is then derived by subtracting the remainder from 11.  $11 - 5 = 6$ . So for this example **CD2 = 6**.

Check Digit Validity Test

The value in the user entered check digit field is compared to the calculated CD1 and CD2 using the following checks, if both of the checks are true, then the validation is unsuccessful.

Check	Description
1	CD1 is compared to the first digit of the entered check digit field.
2	CD2 is compared to the second digit of the entered check digit field.

Check	Description
-------	-------------

Example of the test using the previously calculated CD1 and CD2:

Where CD1 = 0 and CD2 = 6 and suppose the user entered Check Digit Value is '05'. As CD2 does not match, the check digit is invalid.

## Sweden

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be between 4 to 5 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be between 4 to 5 numeric characters.</li> <li>If the Bank Code and Branch Number are entered, then both values must match.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 16 numeric characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>Length should be a single numeric character.</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 24 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Switzerland

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be between 3 to 5 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be between 3 to 9 numeric characters.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>

Field	Rule
	<ul style="list-style-type: none"> <li>Length should be a maximum of 17 numeric characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Type</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 21 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## The Former Yugoslav Republic of Macedonia

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed: <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN</li> <li>Length should be 19 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Tunisia

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Optional, if entered, the below rules apply.</li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 22 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## Turkey

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed, <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 26 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## United Arab Emirates

### Validation Rules



The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, the length should be a maximum of 4 characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 21 characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed.</li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length should be 23 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The first 2 digits is AE, followed by 21 alphanumeric digits (format: AE + 21 digits).</li> </ul>

## United Kingdom

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, then the length should be 6 numeric characters.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>It is unique within the country.</li> <li>Length should be a maximum of 6 numeric characters.</li> <li>If the length is less than 6, then it is converted to a 6 digit number by prefixing it with as many leading zeroes as is necessary.</li> <li>This field is labeled as <b>Sort Code</b>.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>Length should be a maximum of 8 numeric characters.</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>Secondary Account Reference</b>	<ul style="list-style-type: none"> <li>Optional</li> <li>If entered, length should be a maximum of 18 characters.</li> <li>This field is labeled as <b>Building Society Roll Number</b>.</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Mandatory</li> <li>If the IBAN is not entered, a warning message is displayed, <b>IBAN has not been entered. This bank account is defined in a country that requires IBAN for payment processing.</b></li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> </ul>

Field	Rule
	<ul style="list-style-type: none"> <li>Length should be 22 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

## United States

### Validation Rules

The fields are checked for validity by adopting the following rules:

Field	Rule
<b>Bank Code</b>	<ul style="list-style-type: none"> <li>Optional.</li> </ul>
<b>Branch Number</b>	<ul style="list-style-type: none"> <li>This field is labeled as <b>Routing Transit Number</b>.</li> <li>Length should be a maximum of 9 numeric characters.</li> <li>If the length is less than 9, then it is converted to a 9 digit number by prefixing it with as many leading zeroes as is necessary.</li> <li>Note that on padding the number to 9 digits, the first 8 digits cannot be all zeroes.</li> <li>For example, 001 and 000007 are invalid Routing Transit Numbers because on padding to 9 digits, they become - 000000001, 000000007, and thus having 8 leading zeroes.</li> <li>A check algorithm is applied on the Routing Transit Number.</li> </ul>
<b>Account Number</b>	<ul style="list-style-type: none"> <li>Mandatory</li> </ul>
<b>Check Digit</b>	<ul style="list-style-type: none"> <li>Optional</li> </ul>
<b>IBAN</b>	<ul style="list-style-type: none"> <li>Optional, if entered, the below rules apply.</li> <li>The module-97 rule is used to calculate the validity of the IBAN.</li> <li>Length cannot be more than 34 characters. Spaces are removed from the left and right. Spaces in the middle are not removed.</li> <li>The first 2 characters are letters.</li> <li>The third and fourth characters are numbers.</li> </ul>

### Check Algorithm for Routing Transit Number

- The ninth digit of the Number field is used to represent the Check Digit.
- A calculated Check Digit is computed from the remaining 8 digits using Modulus 10 algorithm.
- Multiply each digit in the Routing Transit Number by a weighting factor. The weighting factors for each digit areas given in the following table:

Digit	1st	2nd	3rd	4th	5th	6th	7th	8th
Factor	3	7	1	3	7	1	3	7

- The digits of the Routing Transit Number are multiplied by the associated factor. The computed sum is then calculated by summing the totals.

- Subtract the sum from the next highest multiple of 10. The result is the calculated Check Digit. This should be the same as the 9th digit of the Branch Number or Routing Transit Number; otherwise the Branch Number or Routing Transit Number is invalid.

For Example:

Routing Number	0	7	6	4	0	1	2	5	Total
Multiply by	3	7	1	3	7	1	3	7	
Sum	0	49	6	12	0	1	6	35	= <b>109</b>

So the Check Digit = 1 (110 minus 109).

In this example, the Routing Transit Number 076401251 passes validation.



# 12 Define Advanced Collections Configuration

## Manage Collections Preferences

### Setting Up Collections Preferences: Points to Consider

Oracle Fusion Advanced Collections provides a user interface to guide you through the collection preferences. The two required setup regions are: Global Preferences and Preferences. You answer questions and make decisions about how the application functions, such as date ranges and defaults. Consider the following decisions before defining preferences:

Decision	Is This Fusion Applicable?
How many employees, locations and organizations to create?	Yes, depends on the setup of the enterprise structure. Review and verify before setting up Advanced Collections.
Which employees to create?	Yes, determine the number of employees who are involved with the collection process.
Assign collectors?	Yes, how are the collectors going to be assigned; by customer account, account, site or other criteria?
Set up Oracle Fusion Accounts Receivables?	Yes, how is Receivables being set up and what is the impact on Advanced Collections.
Enable AR Transactions Summary Tables?	Yes, a profile option set in Receivables to update activities applied to transactions.
Set up Oracle Fusion Payments?	Yes, set up to use credit cards and automatic fund transfers to apply to transactions.
Set up Units of Measure?	Yes, a Receivables set up needed for transactions.
Set up Security and Users?	Yes, set up through Oracle Fusion Identity Management to grant access to collections.
Set up Notes?	Yes, is an Oracle Fusion Common Application Components (CAC) feature allowing collectors to comment on interactions with customers.
Set up Tasks?	Yes, is a CAC feature allowing collectors or managers to assign follow-up tasks.
Set up Oracle Business Intelligence Publisher?	Yes, but must be verified that Business Intelligence Publisher is working to run collection reports.

## Global Preferences

Selections made in the Global Preference region impact the view the collectors see from the Collections Customer Work Area and Collections Dashboard. Global Preferences define the following:

- Default transaction class that appears
- Display of open transactions
- Display of closed transactions
- Number of days for prior and future transactions to be displayed
- Number of days to reschedule work on the dashboard
- Default aging method
- Delimiter used to separate data and the number of characters required to do a search
- Sender or return E-mail address for dunning correspondence
- Maximum grace days before a promise is broken
- Maximum number of days for a promise due date
- Enable bankruptcy

## Preferences

Selections made in the Preference region impact the defaults the collector encounters when going through the collection process. Preferences define the following

- Preference set
- Collections business level
- Summarize or age credits
- default conversion rate
- Send method of dunning
- Default contact for unknown dunning recipients

## Correspondence

Selections made in the Correspondence table impact the defaults the collector encounters when sending correspondence during the collection process. Correspondence defines the following:

- Preference Set
- Send Dispute Notice
- Dispute Template
- Send Payment Notice
- Payment Template
- Send Promise to Pay Notice
- Promise to Pay Template

## Defining Collections Preferences: Worked Example

This example demonstrates how to set preferences for Oracle Fusion Advanced Collections.

You have been tasked to set up the two sections for preferences: Global and Preferences. Configure your preferences based on the following requirements:

- Your company requires collectors to review transaction that are 90 days past due and current ones 30 days into the future.
- Collectors review customers by account and send out past due notices by E-mail.
- Notices are sent to the accounts payable manager if no contact information is available.
- Reviewing a customer's history requires viewing current and closed transactions.
- Your company allows up to two days for rescheduling work on the dashboard.
- Adjustments and disputes are handled within 1 business day of being recorded.
- Credits on an account are summarized.
- Aging is displayed by a 5 bucket aging method.
- The conversion rate is based on a rate set by the corporation.
- The delimiter symbol used to separate data is a pipe.
- A three-character minimum is required and recommended to perform a search on customer accounts.

Use the following tables to define the two sections for preferences:

### Configuring Collections Preferences

#### 1. Defining **Global Preferences**

Field	Value
Select the default Transaction Type for the Transactions tab	Select Invoice, other transaction choices are all, credit, or debit memos.
Automatically display closed transactions on Transaction tab	Yes is select to view closed transactions as part of the customers history.
Automatically display current transactions on Transaction tab	Yes is selected to view current or open transaction the customer has pending.
Enter the number of days before the current date the transaction date range should start	In this example, 90 days gives the collector 3 months of history. 1 to 9999 is the range that can be used.
Enter the number of days after the current date the transaction date range should end	30 days of current or open transactions are displayed. 1-9999 is the range that can be used.
Enter the maximum number of days work can be rescheduled on the dashboard	Two days is the allowable change in schedule for work. 1-9999 is the range that can be used.

Field	Value
Enter the number of days after submitting an adjustment or dispute that an activity is generated	One day generates the disputes and adjustments. 1-9999 is the range that can be used.
Select the default Aging Method	5 Bucket Aging, several aging buckets can be defined and may be available.
Enter the delimiter used to separate customer, account and site on the Collections Dashboard	The pipe symbol, other available choices are the greater than, dash, and colon symbols.
Enter the minimum number of characters required to perform a search	Three characters is the Oracle recommended number for a search. 1-9999 is the range that can be used.
Enter the sender or return E-mail for dunning correspondence	The default return E-mail for customers to respond

## 2. Defining Preferences

Preferences Table Column Name	Field Value
Collections Preference Set	Select the value; Common Set.
Select the Collections Business Level	Account, other choices are customer and site.
Select whether open credits should be aged or summarized by default	Summarized, credits can be displayed as aged.
Select the conversion rate type for converting multicurrency transactions	Corporate, several choices can be defined and are displayed.
Select the default method to send collections notifications	Select E-mail, other methods are fax or print.
Enter the default contact for unknown dunning recipients	Accounts Payable Manager. User defined if no contact is listed.

Correspondence Table Column Name	Field Value
Collections Preference Set	Select the value; Common Set.
Send Adjustment Notice	Select yes to notify the customer of adjustments made to their account.
Adjustment Template	Select the delivered template.
Send Dispute Notice	Select yes to notify the customer of disputes made to their account.



Correspondence Table Column Name	Field Value
Dispute Template	Select the delivered template.
Send Payment Notice	Select yes to notify the customer of payments made to their account.
Payment Template	Select the delivered template.
Send Promise Notice	Select yes to notify the customer of promises to pay on their account.
Promise Template	Select the delivered template.

3. **Save** or **Save and Close** to enable your preferences.

## FAQs for Manage Collections Preferences

### What's the difference between Global Preferences and Preferences?

Global preferences affect the Collections Customer Work Area, such as display of closed or open transactions and setting date range parameters.

Preferences can be set uniquely for a specific business unit. Preferences impact the default settings, such as preference set and default method to send notifications.

### What's the difference between customer, account, and site at the Business Level?

Transactions set at the customer level to view or modify, encompasses all of the transaction activity associated with the customer. For example, a dunning letter at the customer level is inclusive of transactions at the customer and account level.

Transactions set at the account level to view or modify, are for a particular account and include transactions for all the bill-to sites under that account.

Transactions set at the site level to view or modify, are specific to the bill-to location.

### What's the difference between minimum dunning amount and minimum dunning invoice amount?

Minimum dunning amount is the total amount set for all overdue transactions to have correspondence generated. For example, if your minimum dunning amount is set to \$100 and you have three overdue transactions of \$25, \$35, and \$55 totaling \$105, a dunning letter is generated listing the three invoices as overdue.

Minimum dunning invoice amount is the amount set for an overdue transaction to generate correspondence. For example, if your minimum dunning invoice amount is set to \$25 and you have three overdue transactions of \$15, \$50, and \$75, a dunning letter is generated for the \$50 and \$75 transactions.

## Manage Aging Methods

## Aging Method: Explained

Aging methods are periods of time used to group receivables, debit and credit items to achieve an understanding of a customer's delinquency profile.

Grouping transactions by buckets of time creates an aging view of the customer. Aged information:

- Enables the collector to be effective when trying to recover delinquent debt.
- Allows a collections organization to develop more effective policies to assign work and streamline efficiencies.

Advanced Collections delivers preconfigured aging methods and allows you to create new aging methods based on company requirements.

### Aging Method Features

Aging is the concept of calculating a customer's past due and current transactions. Oracle Fusion Advanced Collections groups overdue debt by time and assigns work based on aging buckets of debt items. The following features of the aging methods include:

- Showing the amounts owed to the company by its customers and includes the length of time the amounts have been outstanding. For example, aging categorizes receivables into buckets such as; current, 30 days, 60 days, 90 days, and 120 days and over.
- Functionality consisting of an Aging Methods Setup screen and a separate **Aging** tab.
- The use of aging methods to be used during the creation of dunning plans.
- The viewing of transactions or receivables data that helps a collector understand the overdue amounts viewed over time. This understanding enables the collector to take a payment or create an adjustment or dispute.

#### Related Topics

- [Dunning: Overview](#)

## Creating an Aging Method: Worked Example

This example demonstrates how to create an aging method not predefined in Oracle Fusion Advanced Collections.

Your company requires an aging method that is not a collections predefined aging method. You are to create a new aging method. Use the default values except where indicated. Create an aging method named 1 to 120 Days Method, with the following segments:

Sequence	Bucket Type	Aging Days From	Aging Days To	Aging Bucket Heading
1	Current	-9999	0	Current
2	Past Due	1	45	45 Past Due
3	Past Due	46	60	60 Past Due
4	Past Due	61	90	90 Past Due

Sequence	Bucket Type	Aging Days From	Aging Days To	Aging Bucket Heading
5	Past Due	91	120	120 Past Due
6	Past Due	121	9999	121+ Past Due

## Creating an Aging Method

1. Start from the Manage Aging Methods page.
2. From the **Aging Method** region, click the green plus sign to create an aging method.
3. Complete the fields, as shown in this table.

Field	Value
<b>Aging Method Name</b>	1 to 120 Aging Day Method
<b>Aging Type</b>	Statement Aging; 4 bucket and 7 bucket aging are options, depends on the number of buckets for your aging method.
<b>Enable</b>	Yes
<b>Aging Method Set</b>	Common Set
<b>Aging Method Description</b>	Current to 120 Day Aging method.

4. Click **Save**.
5. From the 1 to 120 Aging Day Method: Details, complete the fields, as shown in this table.

Field	Value
<b>Sequence</b>	1
<b>Bucket Type</b>	Past Due
<b>Aging Days From</b>	1
<b>Aging Days To</b>	45
<b>Aging Bucket Heading</b>	45 Past Due; Click the <b>Create</b> icon to create the row for the next sequence.
<b>Sequence</b>	2
<b>Bucket Type</b>	Past Due
<b>Aging Days From</b>	46
<b>Aging Days To</b>	60

Field	Value
<b>Aging Bucket Heading</b>	60 Past Due

6. Repeat the steps for sequences 4 to 6 and increase the Aging Days From and Aging Days To accordingly.
7. Click the **Save and Close** to complete the creation of a new aging method.

The following validation rules are in place when creating or modifying an aging method:

- A bucket must contain at least 2 bucket lines and at most 7 lines.
- A Bucket Type passes from the user interface as a fixed constant value.
- A bucket must have Aging Days From less than Aging Days To.
- All buckets must cover the range from -9999 to 9999 and there is no way to update a field with the value of -9999 or 9999.
- Aging buckets cannot have any gaps between -9999 to 9999.

## FAQs for Manage Aging Methods

### How can I modify a predefined aging method?


Predefined aging methods cannot be modified, however they can be copied, are renamed and allow modifications made to the duplicate. From the Manage Aging Methods page, select one of the delivered aging methods and select the **Duplicate** button. The method is renamed as Copy in front of the name of the duplicated aging method. For example a 5 Bucket Aging becomes Copy 5 Bucket Aging, and allows you to make your modifications. Saving your changes creates the modified aging method.

## Manage Collectors

### Setting Up a Collector: Points to Consider

A collector is an individual or a group of individuals, assigned to a customer to conduct various collections work. Tasks include:

- Sending correspondence
- Reviewing customer history
- Collecting payment from customers

 **Note:** Prior to creating a collector, the individual must be set up as an employee in Oracle Fusion Human Resources and as a resource in Oracle Fusion Customer Relationships Management (CRM) applications.

## Setting Up Collectors

Consider the following when setting up individuals as collectors:

- A collector can individually be assigned to one or more customers or can be a group member that collects from one or more customers. Evaluate the most appropriate collection needs for your organizations.
- Collectors can be assigned at the customer, account or site level. Determine how your organization interacts with customers.
- Research how your organization divides the work and tasks among collectors.
- Collections organization structures can be created in several ways based on the number of customers. For example, you can group customers according to size, small to large or divide customers regionally or by the monetary volume you do with a customer.

## Creating Collectors: Worked Example

This example demonstrates creating collectors and assigning them as an employee assignment and a group assignment.

Your company wants to create five individuals as collectors. Set up your collectors based on the following:

- The Collections Department collects on a regional basis, north, south, east and west.
- Acme Corporation is a large customer and they want to assign one collector to this account.
- All five individuals have been created as a Person Party and Employee, a prerequisite to creating a collector.
- The regions have also been created as groups.

The following information is required for each individual:

Field	Action
Name	Employee Name
Description	Optional; detail information
Correspondence	Name used on sent correspondence
Telephone Number	Contact number
Employee Name	Active employee list
Group	Uses group from setup feature
Active or Inactive	Collector status

## Creating an Employee Assignment

1. Start on the **Collectors Setup** page in Functional Setup Manager.
2. You can search or create a new collector from this page.

3. Click the **Add or Create Collector** icon.
4. Enter the **Collector Name**.
5. Enter the **Type** of collector as employee.
6. Select the name of the employee from the drop down list from the **Employee or Group** column.
7. Set the status as **Enabled**.
8. Select the appropriate **Collector Set**.
9. Click the **Save and Close** button.

## Creating a Collector Group

1. Start on the **Collectors Setup** page in Functional Setup Manager.
2. You can search or create a new collector from this page.
3. Click the **Add or Create Collector** icon.
4. Enter the **Collector Name**.
5. Enter the **Type** of collector as group.
6. Select the appropriate group from the **Employee or Group** drop down list. In this example each would be assigned to one of the regions; north, south, east or west.
7. Set the status as **Enabled**.
8. Select the appropriate **Collector Set**.
9. Save your work by clicking **Save and Close**.

### Related Topics

- [Can I create an employee or contingent worker resource?](#)

## Collectors Descriptive Flexfield: Overview

You can use the Collectors Descriptive Flexfield to add custom attributes for collector information such as a geographical region and collections experience.

For example you can create a descriptive flexfield to identify the collector location that can be used to assign them to customers in or near their location. Tracking the experience of a collector can be used to determine which customer accounts get assigned to a collector.

One descriptive flexfield is available in Oracle Fusion Advanced Collections.

Descriptive Flexfield	Description
Collectors	15 segment available to display additional information about collectors.

### Defining Descriptive Flexfield Segments

Use the Manage Descriptive Flexfields task to define a segment or segments for a descriptive flexfield. You can add more information related to collectors based on your collection needs and level of detail information about them.

### Activating Descriptive Flexfields

Activate a descriptive flexfield after you have defined value sets and segment values by deploying the flexfield. You must sign out and sign in to the application to see your descriptive flexfield.

 **Note:** The Global segments for the Collectors Descriptive Flexfield are not available.

Related Topics

- [Descriptive Flexfields: Explained](#)
- [Managing Descriptive Flexfields: Points to Consider](#)

## Setting Up the Collectors Descriptive Flexfield: Worked Example

The Manage Collectors descriptive flexfield provides 15 segments to capture additional information. These segments are made available to you as individual fields in the Manage Collectors page. This example illustrates how to set up the descriptive flexfield using the context-sensitive segments, based on the following scenario

### Scenario

You have been tasked with setting up collectors and want to capture more information about the collectors geographical location and collections experience. You want to be able to search on this information and associate this information to each collector you create. Set up the additional fields using the following information

Segment Name	Value
Region	East
	West
	Central
Experience	Novice
	Junior
	Senior

### Setup Steps

1. Navigate to the **Set Up and Maintenance** in the Functional Setup Manager.
2. Select the **All Tasks** tab.
3. Search for the task **Manage Value Sets** Manage Value Sets.
4. Click the **Go to Task** icon.
5. Search on the **Module** field for **Advanced Collections**.
6. Click **Search**.
7. In the **Search Results** region click on the **Create** icon
8. In the **Create Value Set** page, enter the following information:

Create Value Set	Value
Value Set Code	IEX_REGION

Create Value Set	Value
Description	Optional
Module	Advanced Collections
Validation Type	Independent
Value Data Type	Character

Definition Region	Value
Value Subtype	Text
Maximum Length	20
Minimum Value	Leave Blank
Maximum Value	Leave Blank
Uppercase only	Unchecked
Zero fill	Unchecked

9. Click **Save and Close** button.
10. **Search** on the Value Set Code that you just created. Your set code appears in the search results.
11. Click the **Manage Values** button in the search results.
12. On the Manage Values page, in the **Search Results** region, click the **Create** icon.
13. In the **Create Values** page create the East, West, and Central values.
14. Enter the following information:

Field or Check Box	Value
Required Value	East
Description	Optional
Enabled	Checked
Start Date	Optional
End Date	Optional
Sort Order	Optional

15. Click the **Save and Close** button.



16. Repeat for the Central and West regions. Click the **Create** icon
17. Once you have completed entering each of the regions, click the **Save and Close** button.
18. Click the **Save and Close** button.
19. Click the **Done** button.
20. Click the **Done** button, a second and final time.
21. From the **Manage Value Sets** task, click on the **Go to Task**.
22. Click on the **Create** icon.
23. Enter in the following information from **Create Value Set** page:

Value Set Field	Value
Value Set Code	IEX_EXPERIENCE
Description	Optional
Module	Advanced Collections
Validation Type	Independent
Value Data Type	Character

Definition Region Field or Check Box	Value
Value Subtype	Text
Maximum Length	20
Minimum Value	Leave Blank
Maximum Value	Leave Blank
Uppercase only	Unchecked
Zero fill	Unchecked

24. Click the **Save and Close** button.
25. **Search** on the Value Set Code that you just created. Your set code appears in the search results.
26. Click the **Manage Values** button.
27. On the Manage Values page in Search Results region click the **Create** icon
28. In the **Create Values** page create the Novice, Junior and Senior values.

Field or Check Box	Value
Required Value	Novice
Description	Optional

Field or Check Box	Value
Enabled	Checked
Start Date	Optional
End Date	Optional
Sort Order	Optional

29. Repeat the information for Junior and Senior values.
30. Click the **Save and Close** button.
31. Click the **Save and Close** button. a second time.
32. Click the **Done** button.
33. Click the **Done** button a second time.
34. From the **All Task** tab, search for the Manage Descriptive Flexfields.
35. Click the **Go to Task** icon
36. In the **Name** field enter Collectors.
37. Click the **Search** button
38. In the Search Results region for Collectors, click the **Edit** icon.
39. In the **Context Segment** region in the **Prompt** field, enter **Additional Information** as the name for the Prompt to be displayed on the **Manage Collectors** page.
40. Click the **Manage Contexts** button.
41. From the Manage Contexts page click the **Create** icon and enter the following:

Field	Value
Display Name	Additional Information
Context Code	IEX ADD INFO (uppercase recommended)

42. Click the **Save** button.
43. In the **Context Sensitive** region click the **Create** icon and enter the following:

Field	Value
Name	Region
Code	IEX REGION
Description	Optional

44. In the **Column Assignment** region enter the following:

Field	Value
Data Type	Text

Field	Value
Table Column	Attribute 1

45. In the **Validation** region enter the following:

Field or Check Box	Value
Value Set	IEX_Region
Description	Optional
Range Type	Not required
Required	Unchecked
Initial Default	Not required

46. In the Display Properties region enter the following:

Field or Check Box	Value
Prompt	Collectors Region
Display Type	List of Values
Required	Unchecked

47. Click the **Save and Close** button

48. In the **Context Sensitive** region click the **Create** icon and enter the following:

Field	Value
Name	Experience
Code	IEX EXPERIENCE
Description	Optional

49. In the **Column Assignment** region enter the following:

Field	Value
Data Type	Text
Table Column	Attribute 2

50. In the **Validation** region enter the following:

Field or Check Box	Value
Value Set	IEX_Experience
Description	Optional
Range Type	Not required
Required	Unchecked
Initial Default	Not required

51. In the Display Properties region enter the following:

Field or Check Box	Value
Prompt	Collector Experience
Short Prompt	Experience
Display Type	List of Values
Required	Unchecked

52. Click the **Save and Close** button.
53. click the **Save and Close** button, a second time.
54. Click the **Done** button.
55. Click the **Save and Close** button.
56. In the **Manage Descriptive Flexfields** page, search in name field for **Collector**.
57. In the **Search Results** region, the Descriptive Flexfield appears.
58. Click the **Deploy Flexfield** button.
59. Click **OK** in the pop-up window.
60. Activate the Descriptive Flexfield by signing out and signing back into the environment.
61. Navigate to Set Up and Maintenance. Under the **All Tasks** tab and search on **Manage Collectors**.
62. Click the **Go to Task** icon.
63. Verify the **Additional Information** field appears in the **Search** region with a List of Values of Region and Experience.
64. **Create** a new collector and **Expand** the row. Verify the Additional Information with the 2 segments Region and Experience appear with the appropriate values for each.

#### Related Topics

- [Descriptive Flexfields: Explained](#)
- [Managing Descriptive Flexfields: Points to Consider](#)

## FAQs for Manage Collectors

### What's the difference between an employee assignment and a group assignment?

An employee assignment is a collector assigned to one customer. You must create individuals as employees before you can set them up as users, resources, or collectors.

A group assignment is created to assign work and customers to a group of collectors. You can have multiple employees or collectors in one group.

## Manage Dunning Configuration

### Aged or Staged Dunning: Explained

The following two dunning methods can be used to set up customer correspondence:

1. Aged
2. Staged

The dunning methods are supported at the following levels:

- Customer
- Account
- Bill-to Site

### Aged Dunning Method

The **Aged Dunning** method can be used by companies sending collection letters to their customers based on the oldest aged transaction. Delinquent transactions are identified automatically when the Delinquency Identification concurrent process is run. Preconfigured dunning notice templates are available, or the deploying company can create their own dunning notice templates. As the oldest aged transaction moves into the next aging bucket, the content of the dunning letter can change. A single consolidated letter can be sent to eliminate sending multiple aged dunning letters to the same customer who has more than one delinquent transaction. The templates are associated to aging buckets in the dunning configuration process.

### Staged Dunning Method

The **Staged Dunning** method is based on the number of days since the last dunning letter was sent, rather than the number of days transactions are past due. A single consolidated letter can be sent to eliminate sending multiple staged dunning letters to the same customer who has more than one delinquent transaction. Delinquent transactions are identified automatically when the Delinquency Identification concurrent process is run. In cases where transactions have installment due dates, the staged dunning letter includes each overdue installment date in the body of the consolidated letter. Staged dunning letters are sent at delayed intervals defined in the Dunning Configuration process. These intervals control the timing of each letter sent. Use preconfigured dunning templates or create new ones.

## Customer Dunning Configuration: Points to Consider

Configure dunning plans and letters to facilitate regular correspondence with your customers.

All of the following options except rerun the dunning process are created in the Correspondence Configuration in Functional Setup Manager.

- Dunning letter options
- Delivery options for dunning letters
- Dunning configuration set
- Dunning transactions
- Rerun the dunning process

### Dunning Letter Options

Use the preconfigured aged or staged dunning letters delivered by the application or create new dunning letters to meet unique requirements. Configure dunning to run in one of two methods:

1. **Aged Dunning** is based on the oldest past due debit item.
2. **Staged Dunning** is based on the number of days since the previous letter was sent.

### Delivery Options for Dunning Letters

Delivery of dunning letters to customer is:

- E-Mail
- Fax
- Regular mail

### Dunning Transactions

Select the transactions and charges to include in the dunning letters.

- The Dunning Letter Generate program can include current invoices, disputed invoices, and credit memos.
- The dunning letters can include current transactions.
- Finance charges can be included as a line item.
- When you select the Include Disputed Invoices, the transactions awaiting dispute resolution are displayed and included in the total.

### Dunning Configuration Set

Dunning can be configured by one business unit or multiple business units. Consider the following when defining sets:

- A Set defines a subset of a master list of business objects, such as suppliers, customers, or in this case a Dunning Configuration Set.
- A Set Determinant is a business object, usually a type of organizational unit, that has a 1 to 1 relationship to Set. When Set Determinant is visible, Set can be hidden.
- A Set of business objects can be associated with an organizational unit, such as Business Unit or Legal Entity, to enforce data security. Then, when you are associated with organizational units, they automatically see only the subset of business objects appropriate for their organizational units.

- A Set or Set Determinant field appears to you as a lookup code that determines which subset of a master list of objects their organizational unit uses. For example, if you are performing a Purchasing transaction, the Set or Set Determinant value would determine which Suppliers can be used.

## Rerun the Dunning Process

Rerun the dunning process to correct errors or regenerate dunning letters:

- Rerun the dunning process should there be an error during any phase of the program.
- Regenerate a previous dunning letter if a customer request another copy.

Keep customers from receiving dunning letters by indicating them as Exclude from dunning on the Profile tab.

## Post Installation for Dunning Configuration

Oracle Fusion Advanced Collections Dunning feature utilizes Oracle Business Intelligence Publisher to distribute dunning letters to customers via E-mail, fax or print. To use this feature, you must configure Oracle Business Intelligence Publisher to connect to the deploying company's internal E-mail, or the print or fax servers.

Configuring Oracle Business Intelligence Publisher: Configure Oracle Business Intelligence Publisher by adding an E-mail server. Refer to the Oracle Fusion Middleware Administrator's Guide for Oracle Business Intelligence Publisher (Oracle Fusion Applications Edition).

## Adding the E-Mail Server

To add an E-mail server:

1. From the **Admin** page select E-mail. This displays the list of servers that have been added. Select **Add Server**.
2. Enter the Server Name, Host, and Port for the E-mail server.
3. Select a **Secure Connection** method to use for connections with the E-mail server. The options are: None or SSL. Use Secure Socket Layer. TLS (Transport Layer Security). Use TLS when the server supports the protocol; SSL is accepted in the response. TLS Required. If the server does not support TLS, then the connection is not made.
4. Optionally enter the following fields if appropriate: General fields; Port Security fields User name and Password.

## FAQs for Manage Dunning Configuration

### What's the difference between an aged and staged dunning method?

Aged dunning sends dunning letters based on the age of the oldest transaction. Managers identify the dunning letter to be sent for each aging bucket.

Staged dunning sends dunning letters based on the age of the oldest transaction and the number of days since the last letter was sent. Managers specify how long to wait before executing the next stage.

For both methods, managers decide whether or not to manually schedule a follow-up call if the delinquency isn't resolved.

## Define Scoring

## Understanding Scoring Formulas: Explained

Scoring formulas are the foundation for your collections activities.

### Scoring Formulas

Oracle Fusion Advanced Collections uses scoring in two ways:

1. Determines transaction status as current, delinquent, or predelinquent. Collections scores invoices and debit memos from Oracle Fusion Accounts Receivables. A customer is delinquent when they have past due transactions.
2. Determines the collectability of each customer at the customer, account, or bill-to site level. Use data points about the customer to determine collectability.

Scoring formulas consist of:

- High and low scores
- One or more weighted data points that must sum to 1
- Data point details you define to cover all possible values
- Status used only in the case of scoring transactions
- A segment which enables you to run against a subset or segment of the database
- A review where you can test your formula

A Scoring formula provides you an effective way to manage your collections resources and strategies.

Collections delivers a set of scoring formulas you can apply or copy and modify to your specific business needs.

### Considerations and Recommendations

Consider the following:

- You should test all data points and scoring formulas in a test environment using a portion or all of your production data
- Once you are satisfied with the scoring results and performance, you can move your tested scoring formulas to your production environment

## Creating a Scoring Formula: Worked Example

This examples shows you how to create a scoring formula.

Create a simple scoring formula based on the following data points:

1. Account Level Delinquency Count
2. Accounted Amount of Delinquencies

 **Note:** If these do not exist you must create them first before creating a scoring formula.



## Create a Scoring Formula

**Navigation:** Setup and Maintenance > All Tasks > Define Collections > Manage Collections Scoring > Go to Task icon > Manage Scoring Formulas

1. From the Manage Scoring Formulas page, click **Create** icon.
2. Enter the following information using the table below:

Field	Value
Name: Required	Simple Scoring Formula
Type: Required	Account: Determines the type of data points can be added to the formula.
Description: Optional	Number and Amount of Delinquencies.
From Score: Required	1: The lowest possible score the formula calculates.
To Score: Required	100: The highest score the formula calculates.
Enabled: Activates the formula	Check

3. From the Data Points sections, you add data points to the formula. Each data point must be mapped across the high and low range of the Scoring Formula.
4. Click **Add Row** icon.
5. In the Name field search for Account Level Delinquency Count data point.
6. Highlight the data point and click **OK**.
7. Enter a value of 1 in the weight field.
8. Enter the following information under the Data Point Details section populating 4 rows using the table below:

From Range	to Range	Value
0	10	100
11	50	75
51	100	50
101	999,999,999	1

9. Click **Save**.
10. Click **Add Row** icon.
11. In the Name field search for Accounted Amount of Delinquencies data point.
12. Highlight the data point and click **OK**.
13. Enter the following information under the Data Point Details section populating 4 rows using the table below:

From Range	to Range	Value
0	5,000.00	100

From Range	to Range	Value
5,000.01	25,000.00	75
25,000.01	50,000.00	50
50,000.01	999,999,999.00	1

14. Adjust the weights of the 2 data points using the following table:

Data Point	Weight
Account Level Delinquency Count	0.25
Accounted Amount of Delinquencies	0.75
<b>Total</b>	1.00

15. Click **Save** button.  
16. Click **Next** button. Enter the following information in the Scoring Formula Segment page using the table below:

Segment	Value
View Name Select the grouping for the customers, accounts, sites, or transaction that applies to the scoring formula.	IEX_F_ACCOUNTS_V
Identifier Select the unique attribute identifying each customer, account, site or transaction.	CUST_ACCOUNT_ID

17. Click **Next** button.  
18. Click **Save** button.  
19. Move to the Scoring Formula Review page.  
20. In the Sample Score section, test your formula using the information in the table below:

Field	Value
Account Number: Select an account that has delinquent or current transactions.	Valid account number
Business Unit: Select a business unit that contains those transactions.	Valid business unit

21. Click **Score** button.

## Scoring Formula Results

The following are displayed in the Data Points table:

1. Component Score: the actual number of delinquencies and total amount of delinquencies for the test account.
2. Mapped Component Score: how that number translates to each data point detail.
3. Weighted Score: the Mapped Component Score multiplied by the data point weight.

The total score displays the sum of the weighted scores for all data points. This is the score assigned to the account.

## Applying the Formula

Your formula is now activated and you want to apply it by submitting the Collections Scoring as Job program.

**Navigation:** Navigator > Scheduled Process > Collections Scoring as Job

1. Click **Schedule New Process** button.
2. Select from the list of values: Collections Scoring as Job.
3. Enter the Parameters according to the table below:

Parameter	Value
Business Unit: Required	Select a business unit that contains the account.
Scoring Formula: Required	Simple Scoring Formula

4. Click **Submit** button.
5. Verify by navigating to the Collections Work Area. Query for the customer, account, or bill-to site and review. The score appears in the Customer Hierarchy task pane.

## Editing a Scoring Formula: Worked Example

This example shows how to take an existing formula and use the duplicate feature to make edits to the formula.

Oracle Fusion Advanced Collections delivers a set of formulas you can use to score customers. You can modify the existing formulas to meet your business requirements by copy and modifying them.

## Editing an Existing Formula

You are using the formula, Customer Scoring and want to change the weight in the existing formula to give customers who are 91+ days overdue lower scores.

**Navigation:** Setup and Maintenance > All Tasks > Define Collections > Manage Collections Scoring > Go to Task icon > Manage Scoring Formulas

1. Query the existing formula. In the **Query by Example** field for Name enter Customer Scoring and **Enter**.
2. Select the row and click **Duplicate** icon.
3. Edit the formula information using the table below:

Field	Value
Name	91+ Customer Scoring
<ul style="list-style-type: none"> <li>Required</li> <li>Replace Copy of Customer Scoring. This must be a unique name</li> </ul>	
Description: Optional	Modified copy of the Customer Scoring formula. Change in weighted values.
Start Date: Required	Current Date
End Date: Optional	Leave Blank: Use an end date if the formula has limited availability.
Type: Determines the type of data points you can add to the formula.	Customer
Description: Optional	
From Score: Required	1: The lowest possible score the formula calculates.
From Score: Required	100: The highest possible score the formula calculates.
Enabled: Activates the formula.	Yes

4. Alter the weights according to the following table:

Data Point	Weight
The Percent of Delinquencies 1-30 Days Overdue	0.1 Lower value to increase the other data point
The Percent of Delinquencies 31-60 Days Overdue	0.1 Lower value to increase the other data point
The Percent of Delinquencies 61-90 Days Overdue	0.1 Lower value to increase the other data point
The Percent of Delinquencies 91+ Days Overdue	0.7 Lower value to increase the other data point
Total Must equal 1.00	1.00

- Click **Next** button.
- Keep the Customer Segments listed. Click **Next** button.
- From the Create Scoring Formula: Review, click **Save** button.
- In the Sample Score section test your formula using the information in the table below:

Field	Value
Account Number	Select an account known to have 91+ delinquent transactions.
Business Unit	Select a Business Unit that contains those transactions.

9. Click **Score** button.

The following are displayed in the Data Points table:

- a. Component Score: the actual number of delinquencies and total amount of delinquencies for the test account.
- b. Mapped Component Score: how that number translates to each data point detail.
- c. Weighted Score: the Mapped Component Score multiplied by the data point weight.

The Total Score displays the sum of the weighted scores for all data points.

## Assigning the Formula

Your formula is now activated and you want to apply it by doing the following:

**Navigation:** Navigator > Scheduled Process > Collections Scoring as Job

1. Click **Schedule New Process** button.
2. Select from the list of values: Collections Scoring as Job.
3. Enter the parameters according to the table below:

Parameter	Value
Business Unit	Select a Business Unit that contains the account you tested previously.
Scoring Formula	91+ Customer Scoring

4. Click **Submit** button.
5. Verify by navigating to the Collections Work Area. Query for the customer, account, or bill-to site and review. The score appears in the Customer Hierarchy task pane.

## Understanding Scoring Formula Data Points: Explained

A data point is a component of a scoring formula.

### Data Points

You must first define your data points prior to creating a scoring formula.

 **Note:** Cloud customers must submit a Service Request to create, edit, or modify scoring data points used in formulas.

Data points can be one of the following two types:

1. A select statement

2. A database function, an example would be asking for the total number of delinquent transactions or the total delinquent amount of customer delinquencies

 **Note:** You must know how to create PL/SQL statements or functions for either type.

The following are criteria for data point details:

- They must return a single numeric value
- They must be created for a certain element which is the database entity to be scored
- Can be positive or negative
- Must account for the range of numbers from -999,999,999 to 999,999,999
- Must be contiguous and do not overlap

The values calculated by a data point are weighted and used in the scoring formula. Collections deliver formulas where a higher score indicates a better chance of collectability and a lower score indicates a lower chance of collectability. You can configure formulas where higher scores result in lower chance of collectability and lower scores a higher chance of collectability.

## Creating Scoring Data Points: Worked Example

This example shows you how to create data points used in a scoring formula.

Create two data points based on the following information:

1. Configure the first data point for the number of delinquencies for a given customer account
2. Configure the second for the total amount delinquent. This data point uses accounted amount and allows summing of different currencies.

 **Note:** You must create data points before you can create a formula.

## Creating Data Points

 **Note:** Cloud customers must submit a Service Request to create, edit, or modify scoring data points used in formulas.

**Navigation:** Setup and Maintenance > All Tasks > Define Collections > Manage Collections Scoring Data Points > Go to Task icon

1. Navigate to the Manage Collections Scoring Data Points page, click **Create** icon.
2. Use the table below to create your first data point:

Field	Value
Name: Required	Account Level Delinquency Count: Use a unique name.
Type: Required	Account Other choices are: <ul style="list-style-type: none"> <li><input type="radio"/> Customer</li> <li><input type="radio"/> Bill-to</li> </ul>

Field	Value
	<input type="radio"/> Transaction
Description: Optional	Total number of delinquencies for a customer account.
Function: Refers to a database function	No: This example uses a PL/SQL statement.
Enabled: Activates the data point to be used in formulas.	Check
SQL Statement: The ability to write PL/SQL statements is required for this task	Select count(1) from iex_ delinquencies where cust_ account_ id=:cust_ account_id and status<>'CURRENT'

3. Click **Save and Add Again** button.

4. Use the table below to create your second data point:

Field	Value
Name: Required	Accounted Amount of Delinquencies: Use a unique name.
Type: Required	Account Other choices are: <ul style="list-style-type: none"> <li><input type="radio"/> Customer</li> <li><input type="radio"/> Bill-to</li> <li><input type="radio"/> Transaction</li> </ul>
Description: Optional	Total amount of the delinquencies in the accounted amount that enables summing of different currencies.
Function: Refers to a database function	No: This example uses a PL/SQL statement.
Enabled: Activates the data point to be used in formulas.	Check
SQL Statement: The ability to write PL/SQL statements is required for this task	Select NVL(sum(ar. acctdamount_ due_remaining)0) from ar_ payment_ schedules ar, iex_ delinquencies del where ar.payment_ schedule_id = del.payment_ schedule_id AND del.status in ('DELINQUENT', 'PREDELINQUENT') AND del.cust_ account_id = :cust_ account_id AND del.creation_ date >= sysdate -365

5. Click **Save and Close** button.

## FAQs for Define Scoring

### What's a scoring formula status?

A scoring formula status is applicable only when the scoring formula type value is Transaction. Assign one of the following statuses:

- Current
- Delinquent
- Predelinquent

### What's a scoring formula segment?

A segment is a subset of the objects being scored.

For example, if you want to score customers in US based on different criteria than customers based in Japan, you can create two different formulas and attach two different segments to those formulas.

The segment is a database view and must be created by your Database Administrator.

## Define Strategy

## Processing Strategies: Explained

A series of processes must occur before using strategies.

The following is the process flow for strategies:

- Score collections objects by running the Collections Scoring program
- Aggregate customer data by running the Update Collections Summary Data program
- Run the Strategy Management concurrent program to create a strategy for each delinquent object
- Each strategy is assigned based on the score applied to it
- This score determines how aggressively or softly the strategy treats delinquent customers

### Processing a Strategy

You have created delinquencies and scored collections objects, such as 30 day and 60 day past due invoices.

Two strategies exist with the following information:

Strategy Name	Tasks
Soft	<ol style="list-style-type: none"><li>1. Send reminder letter</li><li>2. Make call</li><li>3. Follow-up call</li><li>4. Send dunning letter</li></ol>
Hard	<ol style="list-style-type: none"><li>1. Send dunning letter</li><li>2. Make call</li></ol>



Strategy Name	Tasks
	3. Call escalation by supervisor

## Results

The following scores result after the Strategy Management program runs:

Score	Results
50	Higher Score results in a softer strategy
30	Lower score results in a harder strategy

## Analysis

The following occurs:

- If the program cannot find a strategy to match the exact score, it assigns a more aggressive strategy
- If the delinquency score returned by the scoring engine is 35, then the selection module in the program looks for strategies ranked 35, and if not found, looks for 34, then 33, until a valid score is found. In this example, the program assigns the Hard Strategy to the delinquent object

### Related Topics

- [How can I assign a strategy to a customer?](#)

# Collections Strategies: Explained

Oracle Fusion Advanced Collections strategies are a series of manual or automated activities, known as tasks that are linked together in the order they are to be executed.

Each task is a collection activity used to collect a delinquency. Examples of tasks include:

- Calling the customer to collect
- Sending a letter, fax, or E-mail
- Escalating to a supervisor
- If a task is escalated then the manager will receive an escalation notification automatically, and the task remains assigned to the collector.

 **Note:** You can create several tasks and reuse them in many strategies.

Collections strategies support activities based on the following:

- Your business requirements
- Scoring rules
- Alternative to dunning plans
- To collect on delinquent transactions
- Simple to complex collection activities

 **Note:** You can use either strategies or dunning plans to manage your delinquencies, but not both methods.

Use strategies in the following way:

- Associate to customers, accounts, or bill-sites depending on the data level you do business
- Leverage information from the Advanced Collections scoring engines that identifies delinquent transactions and score delinquent objects

## Setting Up Strategies

Setup your strategies in the following order:

1. Create collections tasks
2. Create collections strategies
3. Assign collections method name
4. Assign task to strategies

## Strategy Processing and Results

Assign appropriate strategies to the collections objects by running the Strategy Management concurrent program for all business units at once.

The strategies are based on the scoring engine results. The concurrent program results in the following:

- Oracle SOA BPEL initiates and manages tasks and sends notifications to the designated person
- Oracle BI Publisher sends correspondence

### *Related Topics*

- [How can I assign a strategy to a customer?](#)
- [How do scoring engines and strategies work together?](#)

## Creating a Collections Strategy: Worked Example

This exercise show you how to create a strategy, assign tasks, and assign them sequentially.

Strategies are a series of manual or automated tasks and linked together in the order they are to be executed. A strategy consists of the following:

- Create Strategy Template Group: Used to define your general information about the strategy
- Customer Segment: Identifies the grouping and unique attribute for this strategy
- Strategies: Lists the various stages and scores for your strategy
- Tasks: Lists the tasks in sequential order and defines the details for each

## Create a Strategy

**Navigation:** Setup and Maintenance > All Tasks > Manage Collections Strategies> Go to Task icon > Manage Strategies

1. From the Manage Collections Strategies section click **Create** icon.
2. Enter the following information:

**Create Strategy Template Group page:**

Field	Value
Group Name	US Customers
Strategy Level	Account
Change Strategy per Tolerance	No
Enabled	Yes
Start Date	Select the current date from the pick list.
End Date	Blank

### Customer Segment section

Field	Value
View Name	Account
Identifier	ID

### Strategies section

Field	Value
Method Name	USA Strategy Good
Lowest Applicable Score	50
Method Name (Add row icon)	USA Strategy Bad
Lowest Applicable Score	0

### Tasks section: Select the USA Good Strategy

Field	Value
Order	10
Name	Friendly Phone Call
Order	20

Field	Value
Name	Friendly Phone Call
Order	30
Name	Friendly Phone Call

3. Click **Save and Close** button

#### Related Topics

- [How can I assign a strategy to a customer?](#)

## Creating a Task: Worked Example

This example shows you how to create a manual task.

Each strategy is made up of one or more tasks. A task executed manually or automatically is driven by a workflow. The workflow notifies the collector to perform a task or initiates the automated process. Every task has an associated workflow.

Tasks are grouped by one of two work types:

1. **Manual:** a task to be completed by a collector or specialist. When you complete the task it is closed from the queue. Manual tasks examples are a personal visit, phone call, or contact salesperson.
2. **Automatic:** a task completed by an automated process. This includes tasks such as sending e-mails, faxes, or sending documents to be printed.

### Create a Manual Task

**Navigation:** Setup and Maintenance > All Tasks > Manage Collections Strategy Tasks > Go to Task icon > Manage Strategy Tasks

1. From the Strategy Tasks section click **Create** icon.
2. Enter the following information:

Field	Value
Enabled	Checked
Name	Friendly phone call
Description	Customer is slightly past due, make friendly call
Type	Manual
Category	Phone Call
How long will strategy wait until it executes this task	0 days

Field	Value
How long will strategy wait after it executes this task	0 days
Optional	Y
Days	1 day

- Click **Save and Close** button.

#### *Related Topics*

- How can I assign a strategy to a customer?



# 13 Define Revenue Management

## Source Document Types: Explained

The source document type represents the source of the originating document that is imported into Revenue Management. Document types can include sales orders, service contracts, receivable invoices, and more.

Source document types are used to:

- Identify the external applications and Oracle Cloud applications from which the data is imported. The downstream Revenue Management processing uses the internal identifier to identify the original source.
- Assign a satisfaction measurement model.
- Assign a satisfaction plan when the satisfaction measurement model is set to Period.
- Assign the default satisfaction measurement model, revision intent, and performance satisfaction plan modeling attributes.

Revenue Management contains predefined source document types for the Oracle Applications Cloud and Oracle E-Business Suite Applications applications that can be integrated with Revenue Management. You can also define additional source document types. For example, you can use a source document type to identify documents that come from your order management application.

## System Options: Explained

Revenue Management system options define key configuration, integration and accounting options at the application level, including currency conversion rate types, source document and transaction source extraction start dates, and revenue accounting and thresholds.

On the Manage Revenue Management System Options page, you can define system options for the following:

- Currency conversion
- Integration
- Revenue accounting and thresholds

## Currency Conversion

Use the following system option to define the currency conversion rate:

System Option	Description
Conversion Rate Type	Defines the conversion rate type to be used by Oracle Fusion General Ledger to convert the local currency line transaction amounts into ledger currency amounts.


## Integration

Use the following system options to specify the source applications that are integrated with Revenue Management:

System Option	Description
Source Document Type	<p>Defines the type of source documents to be imported into the application, along with:</p> <ul style="list-style-type: none"> <li>The revenue clearing account to be used when creating the revenue clearing journal entry for the source document.</li> <li>The start date of integration for a source document type. This is the date data will begin being imported into Revenue Management.</li> </ul>
Oracle Fusion Receivables Transaction Sources	<p>Defines the Receivables transaction source to be imported into Revenue Management. The start date of integration for a transaction source is the date on which data will begin being imported into Revenue Management.</p>

## Revenue Accounting and Thresholds

Use the following system options to specify accounting and thresholds.

System Option	Description
Accounting	<p>Defines the general ledger's contract liability, contract asset, price variance, contract discount account, and revenue clearing account to be used when creating journal entries for Revenue Management.</p>
Threshold Amounts	<p>Define the lower limit amount to be applied in determining the transaction price, discount allocation exemptions, and whether a contract's transaction price requires manual review:</p> <ul style="list-style-type: none"> <li>Transaction Price Exemption: Used to automatically exempt a customer contract from transaction price allocation if the contract's total transaction price is less than or equal to the threshold amount.</li> <li>Discount Exemption: Used to automatically exempt a customer contract from transaction price allocation if the discount amount is less than or equal to the threshold amount.</li> </ul> <p> <b>Note:</b> The discount amount is the absolute amount of the difference between the billing amount and the allocated amount of the performance obligation.</p> <ul style="list-style-type: none"> <li>Transaction Price Review: Used to automatically set the contract status to a status of Review Required when the contract's total transaction price is equal to or greater than the threshold amount. When applied, the contract will be allocated only after manual review is completed by the revenue analyst or manager, and the status of the contract is Mark as Reviewed.</li> </ul>



## Contract Identification Rules: Explained

Use contract identification rules to create contracts based on source data imported from multiple source applications. Configure rules to determine which source document types should be processed by the rule and how the data should be grouped into an accounting contract. For example, data can be grouped by customer party, time frame, customer purchase order number, or other common identifiers such as opportunity ID or salesperson.

Revenue Management provides three predefined contract identification rules:

Rule	Description
Identify customer contract based on party (and time duration)	This rule identifies all source documents created for a party within a 30-day time frame as a customer contract.
Identify customer contract based on source document	This rule groups all lines of a source document into a single accounting contract.
Identify customer contract based on source document line	This rule identifies each source document line as an individual accounting contract.

Rules are executed in the order of priority.

To ensure that the Identify Customer Contracts program can process all lines correctly, all lines in the unprocessed pool and every source document line are part of a customer contract. These contracts can be either newly identified customer contracts or existing customer contracts.

 **Note:** You cannot deactivate the predefined rules.

## Performance Obligation Identification Rules: Explained

A performance obligation is a distinct promise in the contract to transfer goods or services to the customer. Revenue recognition is based on the satisfaction of a contract's performance obligations rather than billing. It is important to correctly identify and create performance obligations for revenue recognition.

Revenue Management uses performance identification rules to determine how to automatically group source document lines into performance obligations within an accounting contract. The rules specify the matching attributes that provide a common link for grouping the source document lines into performance obligations. The rules also determine how revenue recognition is handled, including the satisfaction method and allocation exemption that will be used.

Revenue Management provides three predefined performance obligation identification rules:

Rule	Description
Identify performance obligation based item	This rule groups source document lines as a performance obligation based on the item number on the line.

Rule	Description
Identify performance obligation based on item classification	This rule will group source document lines as a performance obligation based on the item classification on the line.
Identify performance obligation based on source document line	This rule identifies each source document line as performance obligation.

Rules are executed in the order of priority.


## Creating Pricing Dimension Structures and Value Sets: Explained

You can segregate standalone sales into sets of similar transactions called pricing dimensions. You identify the pricing dimensions by analyzing the pricing policies of your goods and services. The pricing policy takes into account multiple factors when determining customers' discount eligibility, such as the deal size and customer type. Pricing dimensions are used to categorize the standalone selling prices.

After you identify your pricing dimensions, you group them as segments of a pricing dimension flexfield structure. Define as many pricing dimension structures as you need by creating unique groups of pricing dimensions. Pricing dimension structures are shared by all of the goods and services that use the pricing dimensions included in the structure.

The pricing dimension values are included by default in all sales transactions imported into Revenue Management. The combination of pricing dimension values assigned to a transaction is called a pricing dimension combination. Pricing dimension combinations are used to categorize standalone sales. The transactions are segregated in the standalone sales pool using the pricing dimension combinations. All transactions in the sales pool that have the same pricing dimension combination are grouped into one set for calculation and establishing the standalone selling price.

When determining the standalone selling price for a transaction that is part of a performance obligation, Revenue Management first assigns the pricing dimension combination to the transaction and then derives the standalone selling price based on the pricing dimension combination.

 **Note:** A good or service can have multiple standalone selling prices if the standalone sales pool of the good or service contains multiple pricing dimension combinations.

Example:

The following table shows an example of a pricing dimension structure with three pricing dimensions:

Geographic Region	Customer Type	Deal Size	Standalone Selling Price
North America	Government	1,000,000 USD to 4,999,999 USD	17%
North America	Government	5,000,000 USD to 9,999,999 USD	21%
North America	Commercial	1,000,000 USD to 4,999,999 USD	17%

Geographic Region	Customer Type	Deal Size	Standalone Selling Price
North America	Commercial	5,000,000 USD to 9,999,999 USD	21%
North America	Commercial	10,000,000 USD or more	32%

In the example above, each pricing dimension must have a set of pricing dimension values. The table above shows that the Customer Type pricing dimension has two unique values: Commercial and Government. The standalone selling price is established using a 17% discount for the following pricing dimension combination:

- Geographic region: North America
- Customer type: Government
- Deal size: 1,000,000 USD to 4,999,999 USD

Similarly, standalone selling prices are established for the other four pricing dimension combinations.

## Creating a Pricing Dimension Structure and Value Sets: Worked Example

The Pricing Dimension Structure in Revenue Management allows organizations to model pricing dimensions as a flexfield structure for use in establishing standalone selling prices.

The price of a product can vary based on factors like:

- Geographic location
- Selling quantity
- Type of customer

These factors typically influence how the company determines or sets the selling price of the product. An organization with multiple lines of business may use different sets of dimensions for each line of business.

An organization should analyze how products are priced in each of its lines of business and compile the list of unique groups of pricing dimensions used. Each unique group of pricing dimensions should be defined as a pricing dimension structure.

Consider the following when determining the structures you need:

- Analyze pricing policies across all your products and services and identify all of the pricing dimensions being used.
- Identify the price bands you need for the pricing dimensions.
- Identify distinct groupings of price dimensions.
- Define a pricing dimension flexfield structure for each distinct grouping of pricing dimensions.
- Assign segment labels to the dimensions that have price bands.
- Create only one structure instance for each pricing dimension flexfield structure.

Identify Source Document Attributes:

- Use pricing dimension combinations to help determine standalone selling prices for a source document line.

- Use pricing dimension assignments to help you to enter the pricing dimension values automatically for the document lines imported into Revenue Management.
- Identify the document attributes that capture the pricing dimension values.
- Assign the document attributes to the pricing dimensions for each source document.

Your organization has three tiers of pricing for hardware drives and software licenses:

- Tier 1 hardware and software products that are available to non-US customers.
- Tier 2 hardware and software products that are only available to customers in the US.
- Tier 3 software products that are specific to only Japan and the United Kingdom.

## Defining Value Sets and Assigning Values

The table below provides an example of how you would configure the pricing dimension structure and assign the value sets:

Pricing Structure Name	Value Set Name	Segment Order
WG_VRM_Structure	Customer Classification	1
	Quantity Band	2
	Geographic Region	3
	Country	You assign values but do not assign it as a segment.

1. Based on the table, create the value set Customer Classification.
2. Assign the following values for Customer Classification:
  - Tier 1
  - Tier 2
  - Tier 3
3. Create the value set Quantity Band.
4. Assign the following values for Quantity Band:
  - 1-100
  - 101-500
  - 500-999,999
5. Create the value set Geographic Region.
6. Assign the following values for Geographic Region:
  - Europe
  - North America
  - Asia Pacific
7. Create the value set Country
8. Assign the following values for Country:
  - US

- Canada
  - UK
  - Japan
9. Create a structure, name it WG\_VRM\_Structure, and assign the following segments:
    - Segment 1: Customer Classification
    - Segment 2: Quantity Band
    - Segment 3: Geographic Region
  10. Compile the structure and click Save and Close
  11. Create Pricing Dimension Assignments using the table below for each of the following document:

Document	Segment 1	Segment 2	Segment 3
Projects	Customer Classification	Line Quantity	Bill-to Customer
Sales Orders	Customer Classification	Line Quantity	Bill-to Customer
Service Orders	Customer Classification	Line Quantity	Bill-to Customer
Receivables	Customer Classification	Line Quantity	Bill-to Customer

12. Create Pricing dimension bands for the following:
  - Quantity band for hardware
  - Quantity band for software
  - Set band and select countries and region
13. Create the following items:
  - Hard drives
  - Licenses
14. Create the standalone selling price profile and select the following:
  - Structure
  - Representation Type (discount percentage)
  - Pricing Dimension Assignments
  - Band Assignments:
    - For item licenses assign software quantity band
    - For item hard drives assign hardware quantity band

#### Related Topics

- [Value Sets: Explained](#)
- [Planning Value Sets: Points to Consider](#)
- [Key Flexfield Structures: Explained](#)
- [Flexfields and Value Sets: How They Work Together](#)

## Pricing Dimensions: Explained

Organizations use multiple pricing models to sell their goods and services. Each good or service has a specific pricing model, applied consistently to all sales of that good or service. The goods or services are categorized by different dimensions for the purpose of pricing. These categorization are called pricing dimensions.

You can, for instance, categorize transactions based on:

- Country
- Customer type
- Quantity sold
- Total deal value

The price of the same good or service varies based on the category. Categorization parameters are not fixed. They vary from organization to organization, and within an organization, they vary from good to good or service to service.

For the purposes of standalone selling price analysis (either calculated standalone selling price or estimated selling price), pricing consistency is checked within each pricing dimension value. Therefore, the standalone selling price for a good or service also needs to be established separately for each pricing dimension value.

## Pricing Dimension Structure

Pricing dimensions are implemented using the key flexfield infrastructure. You can define multiple structure definitions for the Pricing Dimension key flexfield. Pricing dimensions are captured as segments in the pricing dimension structure.

Pricing dimension structures:

- Can have up to 30 segments
- Are assigned to a standalone selling price profile
- Can vary from item to item

Each segment of the key flexfield represents a pricing dimension.

## Standalone Selling Prices

Standalone selling prices for a good or service are calculated for each combination of segment values of the pricing dimension structure. For example, if a pricing dimension structure assigned to the good or service has three segments and each segment has five values, then the good or service can have up to 125 standalone selling prices.

Values for pricing dimensions can be individual values or a range of values. For example, a pricing dimension named Country will have individual values such as:

- United States
- Canada
- France
- Germany

A pricing dimension named Quantity Sold will typically have different value ranges, such as:

- 0 to 100
- 101 to 500

- 501 to 1000

Each range is a value for the segment. Pricing dimension structures support both individual values and ranges of values to determine the standalone selling price. Pricing dimensions flexfield segments with either individual values or a range of values use a value set of type Independent. In the second case, each range is defined as a value in the value set, and the range details are stored separately as pricing dimension bands.

## Assigning Pricing Dimension Values

When you run the Identify Customer Contracts process, it extracts source data from the source applications, then automatically runs the Assign Pricing Dimension subprocess. This subprocess analyzes source data and assigns values for the pricing dimensions.

Based on the item in the transaction line, the pricing dimension structure is derived from the standalone selling price profile. The pricing dimension values are analyzed and attached to this line.

Pricing dimension value assignments are based on the user defined attribute mapping. The assignment of values is required for each applicable segment of the pricing dimension structure for every transaction line. If a value for a segment is not available, then a default value, if defined, is assigned.

For pricing dimensions that are of type Range, Revenue Management reads the detail value provided and assigns the applicable parent value to the pricing dimension.

You can review and update the pricing dimension values assigned by the Assign Pricing Dimension subprocess.

## Understanding Pricing Dimension Bands: Explained

Price bands are assigned as pricing dimension segment values to the document lines and help in categorizing the sale for the purposes of establishing standalone selling prices.

Revenue Management provides the following three types of pricing bands:

- Quantity band
- Amount band
- Set band

### Quantity Band

Use the quantity band when the price of the product depends on the number of products sold.

The following example shows a sample pricing policy that has four quantity bands and the corresponding list price per unit.

Quantity Band	List Price Per Unit
0 - 99	USD 1000
100-499	USD 900
500-999	USD 800
1000 or more	USD 700

Quantity Band	List Price Per Unit
---------------	---------------------

You should also use the quantity band when the discount percentage is based on the quantity sold, rather than on the list price.

Revenue Management uses the quantity of the transaction line to determine the band in which the quantity falls, and automatically assigns the band name as the pricing dimension value.

## Amount Band

The amount band is similar to the quantity band, except that the amount band is used when the price of the product depends on the total value sold rather than the quantity sold.

The following example shows a sample pricing policy that has four amount bands and the corresponding list price per unit.

Amount Band	List Price Per Unit
0-9,999	USD 100
10,000-49,999	USD 90
50,000-99,999	USD 80
100,000 or more	USD 70

You should also use the amount band when the discount percentage is based on the total value sold, rather than on the list price.

Revenue Management uses the line amount of the transaction line to determine the band in which the amount falls, and automatically assigns the band name as the pricing dimension value.

## Set Band

Use the set band type of pricing dimension when you want to assign set names as the pricing dimension value instead of individual values within the set. For example, you can use the set band to assign pricing dimension values by geographic region.

You can define sets and their individual values. You can configure your setup to automatically determine the set name for a transaction using any attribute value of the transaction as the detail value.

The following example shows a set band that is being used for geographic regions:

Document Attribute	Document Attribute Value	Geography Dimension Value
Bill-to Customer	Germany	Europe
Bill-to Customer	France	Europe
Bill-to Customer	Canada	North America



Document Attribute	Document Attribute Value	Geography Dimension Value
Bill-to Customer	United States	North America

In the example above, the Bill-to Customer document attribute stores the host country name of the customer who is buying your products. In this case, you configure Revenue Management to use the Bill-To Customer Country attribute value as the basis to determine the value of the Geography pricing dimension. For example, if the bill-to customer country for a transaction is Germany, then Revenue Management automatically assigns Europe as the value of the Geography pricing dimension.

## Item Groups: Explained

If you have large numbers of items that share common pricing policies, then you can group those items into item groups and establish standalone selling prices at the item group level. All of the items in a group share the value established at the group level.

You can group items into an item group if they satisfy two conditions:

- All items in the group share a common pricing policy.
- The standalone selling price for each item is usually established in the form of a percentage. For example, a discount percentage.

For example, InFusion Company has the following hardware systems price list for its commercial customers:

Product Description	Model	Price Per Unit
Big Data Appliance	1187	\$1,200,000
Super Cluster	1902	\$175,000
Database Appliance	1243	\$250,000
Virtual Computer Appliance	1150	\$55,000
Storage Appliance	1102	\$125,000

InFusion's discount policy states that they offer automatic discounts based on the total value purchased in a single order, as shown in the table below:

Order Size Range	Discount
Less than 1,000,000	8 percent
1,000,000 to 4,999,999	10 percent
5,000,000 to 9,999,999	12 percent

Order Size Range	Discount
10,000,000 or more	15 percent

In the example above, all of the items in the list:

- Use the discount pricing policy
- Have the same eligibility criterion for discounts
- Use the Discount Percentage of Unit List Price as the revenue price representation type

Because the hardware products satisfy all of these conditions, you can group them into an item group and establish the standalone selling price at the item group level.

## Standalone Selling Price Profiles: Explained

Standalone selling price profiles are user-configurable profiles. The profiles contain the details Revenue Management uses to derive the standalone selling price for the item, item group, memo line, or performance obligation template.

Use standalone selling price profiles to define the following for the item, item group, memo line, or performance obligation template:

Name	Description
Pricing dimension assignments	Specifies the pricing dimension assignment to which the standalone selling profile is attached.
Minimum standalone sales	Assesses whether the observed standalone selling price calculated by the application is valid, based on whether the minimum number of standalone sales is met.
Item classifications	Defines the category of the item, memo line, or performance obligation template.
Price tolerances	Specifies the standalone selling price tolerance usage, which can be Median, High, or Low.
Coverage	Specifies the number of months covered for the calculation of observed standalone selling price.
Effective period range	Specifies the effective start and end period for the item, item group, memo line, or performance obligation template that is assigned to the standalone selling price profile.

## Standalone Sales Pool Exclusion Rules: Explained

Revenue Management uses standalone sales pool exclusion rules to automatically exclude standalone sales lines from the sales pool used to calculate observed standalone selling price values.

The Calculate Observed Standalone Selling Prices program uses these rules to determine if any sales pool lines are to be excluded from the calculation of the standalone selling price.

For example, a supplier may want to single out some older versions of computer accessories, such as printers and headsets, and place them for sale at a low price. The supplier may decide not to include these sales in the calculation of observed standalone selling prices.

## FAQs for Define Revenue Management

### What's an item group?

An item group identifies a group of items that share the same pricing policies and a common standalone selling price.

Items inherit the standalone selling price from the item group if they are assigned to an item group. Usage of item groups is recommended when the representation type is Discount Percentage.

Item groups allow you to establish standalone selling prices for groups of items, which allow you to substantially reduce the number of standalone selling prices you need.



# Glossary

## **document payable**

An item that is ready to be paid. Equivalent to an installment in Oracle Fusion Payables.

## **extract**

An XML file that contains the superset of data relevant to a payment file.

## **format**

A key setup entity in Oracle Fusion Payments, which ties together formatting attributes, such those used by Oracle Fusion Business Intelligence Publisher (BI Publisher) templates and validations to execute during transaction processing.

## **FTP**

Acronym for File Transfer Protocol. A system for transferring computer files, generally by the Internet.

## **funds capture payment profile**

A key setup entity that holds rules for funds capture processing.

## **HTTP**

Acronym for Hypertext Transfer Protocol. A request and response standard typical of client-server computing. In HTTP, web browsers or spiders act as clients, while an application running on the computer hosting the web site acts as a server. The client, which submits HTTP requests, is also referred to as the user agent. The responding server, which stores or creates resources such as HTML files and images, may be called the origin server. In between the user agent and origin server may be several intermediaries, such as proxies, gateways, and tunnels.

## **HTTPS**

Acronym for Hyper Text Transfer Protocol Secure. A protocol primarily developed for secure, safe Internet transactions. HTTPS allows secure e-commerce transactions, such as online banking.

## **installment**

Any of several parts into which a debt or other sum payable is divided for payment at successive fixed times.

## **internal payee**

The deploying company or any of its business units that receive funds from their customers, the payers. Payees receive funds from their customers by credit card payments, debit card payments, direct debits to bank accounts, or bills receivable transactions sent to banks.

### **payment process profile**

A setup entity which drives processing behavior for each document payable, payment, and payment file.

### **payment system**

An external organization that provides financial settlement services. The payment system can be the bank at which the deploying company has its bank accounts or it can be a third-party processor that connects companies and financial networks.

### **payment system account**

In the funds capture flow, a representation of the relationship between a payment system and the internal payee, that allows an authorization, settlement, or settlement batch to be transmitted to a payment system. In the disbursement flow, a representation of the relationship between a payment system and the internal payer that allows a payment file or positive pay file to be transmitted to a bank or other payment system.

### **servlet**

A Java programming language class used to extend the capabilities of servers that host applications accessed via a request-response programming model. Although servlets can respond to any type of request, they are commonly used to extend the applications hosted by Web servers. For such applications, Java Servlet technology defines HTTP-specific servlet classes.

### **settlement batch**

A group of transactions, typically settlements and credits, that are sent to the payment system together in a file. Settlement batches are generally used with a processor-model payment system.

### **transmission configuration**

Configuration for transmitting files such as payment files.

### **transmission protocol**

A method used to electronically transmit data, such as FTP and Secure HTTP.