

Oracle® Communications Billing and Revenue Management

Calculating Taxes



Release 12.0

E51036-07

July 2022

The Oracle logo, consisting of a solid red square with the word "ORACLE" in white, uppercase, sans-serif font centered within it.

ORACLE®

Oracle Communications Billing and Revenue Management Calculating Taxes, Release 12.0

E51036-07

Copyright © 2017, 2022, Oracle and/or its affiliates.

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

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

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

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

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

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

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

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

Contents

Preface

Audience	v
Documentation Accessibility	v
Diversity and Inclusion	v

1 About Calculating Taxes

About Calculating Taxes	1-1
Impacts of Tax Calculation	1-2

2 Calculating Taxes for Accounts Receivable Actions

Tax Calculation for Adjustments, Disputes, and Settlements	2-1
About Adjustment Tax and Billing Cycles	2-2
About Dispute and Settlement Taxes and Billing Cycles	2-3
Configuring Tax Calculation for Adjustments, Disputes, and Settlements	2-3
Configuring the Default Tax Treatment for Customer Center	2-5
Configuring the Default Tax Method for Account Adjustments	2-5

3 Creating Tax Codes

About Creating Tax Codes (Patch Set 6 and Later)	3-1
Creating Tax Codes for a Flat Tax or Custom Implementation (Patch Set 6 and Later)	3-2
Creating Tax Codes for Vertex (Patch Set 6 and Later)	3-4
About Creating Tax Codes (Patch Set 5)	3-5
Creating Tax Codes for Flat Taxes and Custom Implementations (Patch Set 5)	3-6
Creating Tax Codes for Vertex (Patch Set 5)	3-8
About Creating Tax Codes (Patch Set 4 and Earlier)	3-9
Creating Tax Codes for a Flat Tax or Custom Implementation	3-9
Creating Tax Codes for Vertex	3-11

4	Creating Tax Suppliers	
	About Tax Suppliers	4-1
	Providing the Tax Supplier Data with Each Transaction	4-2
	Allowing Your Sales and Use Tax Software to Determine Tax Supplier Data	4-4
	Specifying Divisions for Tax Suppliers	4-6
	Defining a Default Ship-From Locale	4-6
5	Configuring Tax Calculation Options	
	Configuring Tax Calculation for Account Groups	5-1
	Validating Customer Addresses	5-2
	Reporting Zero Tax Amounts	5-3
	Reporting Taxes of Zero Percent	5-3
	Enabling and Disabling Taxation Globally	5-4
	Configuring Taxation in Pricing Center	5-5
	Configuring Taxation in loadpricelist	5-5
	Configuring Tax Calculation in ECE	5-6
6	Configuring Vertex Manager	
	About Implementing Vertex Tax Calculation	6-1
	Installing the Vertex Software	6-1
	Configuring the Sales Tax Q Series Shared Libraries	6-2
	Specifying Storage Manager Shared Library for Sales Tax Q Series	6-2
	Configuring the Communications Tax Q Series Shared Libraries	6-3
	Specifying Storage Manager Shared Library for Communications Tax Q Series	6-4
	Configuring the Vertex DM for Sales Tax Q Series	6-4
	Configuring the Vertex DM for Communications Tax Q Series	6-5
	Configuring the CM for Vertex Tax Calculation	6-6
	Specifying Vertex DM Connection Entries in CM Configuration File	6-6
	Specifying whether to Validate ZIP Codes	6-6
	Itemizing or Summarizing Taxes for Each Jurisdiction Level	6-7
	Verifying That Taxes Are Being Calculated	6-8
7	Tax Calculation Utilities	
	load_pin_ar_taxes	7-1
	load_tax_supplier	7-1

Preface

This guide describes how to configure taxation in your Oracle Communications Billing and Revenue Management (BRM) system.

Audience

This guide is intended for those who configure BRM tax calculation.

Documentation Accessibility

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

Access to Oracle Support

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

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1

About Calculating Taxes

Learn how Oracle Communications Billing and Revenue Management (BRM) calculates taxes for your customers' purchases.

Topics in this document:

- [About Calculating Taxes](#)
- [Impacts of Tax Calculation](#)

About Calculating Taxes

BRM calculates taxes for charges and discounts. To calculate taxes, you create tax codes. Tax codes define the percentage of tax to apply to a charge, the jurisdiction (state or local), and how to calculate the charge. When you create product offerings, you can specify the tax code to apply to charges, discounts, and chargeshares. For example, you can apply one tax code to telephone handsets and another tax code to online service subscriptions.

When configuring charges, you can also specify when to calculate the taxes. You can calculate taxes at these times:

- **During rating.** In this configuration, taxes are calculated when the event is rated, and they are added to the customer's account balance. This way, you always have an accurate reading of a customer's account balance at any time in the accounting cycle.
- **During billing.** In this configuration, taxes are calculated at billing time. Deferring tax calculation to the billing process reduces rounding errors because all events of the same type are calculated together. For example, taxes are calculated on the total amount of usage fees rather than on individual usage events.
- **During billing, but using the event time.** In this configuration, taxes are calculated at billing time using the tax rate at the time the event occurred. This provides the benefits of billing-time taxation while allowing you to change tax rates in the middle of a billing cycle. For example, assume a recurring charge's tax rate changes from 2% to 3% on May 15. If a billing cycle ends on May 30, BRM would use the 2% rate to calculate the charge's April 30 through May 14 taxes and the 3% rate to calculate the charge's May 15 through May 29 taxes.

You can use the following methods to calculate taxes:

- Use tax codes alone to apply simple flat taxes.
- Use Vertex tax calculation software to apply complex taxation for multiple jurisdictions. With this method, you run the Vertex Data Manager to communicate with the Vertex tax database.

When you use Vertex to calculate taxes, you configure tax suppliers. A tax supplier is the company or corporate division responsible for collecting taxes for a given transaction. Tax suppliers typically include your corporate headquarters and branch offices.

You can also use tax calculation opcodes to create a custom tax calculation application. See "Using Custom Tax Rates" in *BRM Opcode Guide*.

Impacts of Tax Calculation

Tax calculation affects how other BRM features work:

- You can specify whether to perform an adjustment, dispute, or settlement with or without taxes, provided the original item or event was taxable. Taxed adjustments, disputes, and settlements effectively reverse whatever tax was levied on the original item. See "Accounts Receivable" in *Billing Care Online Help*.
- You can record taxes as general ledger data. To do so, create a general ledger ID (G/L ID) for each tax code. See "Setting Up Your General Ledger" in *BRM Collecting General Ledger Data*.
- When rerating taxed events, BRM calculates taxes on any deferred taxable amount in the rerated events during the subsequent bill run. The rerated tax appears on the invoice for the subsequent bill run. You can configure BRM to apply deferred taxes during rerating. This enables BRM to include the deferred tax amounts on corrected invoices.
- You can define tax exemptions for customers in Billing Care or Customer Center. For example, you can specify if an account is exempt from city taxes. You can also specify the percentage of the exemption, such as 10% of the amount is not taxed or 100% for a total exemption.

You can apply tax exemptions only after the account is created. Therefore, tax exemptions do not apply to charges incurred when you create the account.

Vertex Communications Tax Q Series does not support partial tax exemptions in percentage. It supports only full exemption (100%) or no exemption (0%). If you set up an account to have partial tax exemption, BRM sends the transaction for this account to the tax package as 100% tax exempt.

2

Calculating Taxes for Accounts Receivable Actions

Learn how Oracle Communications Billing and Revenue Management (BRM) calculates taxes for adjustments, disputes, and settlements.

Topics in this document:

- [Configuring Tax Calculation for Adjustments, Disputes, and Settlements](#)
- [About Adjustment Tax and Billing Cycles](#)
- [About Dispute and Settlement Taxes and Billing Cycles](#)
- [Configuring Tax Calculation for Adjustments, Disputes, and Settlements](#)

Tax Calculation for Adjustments, Disputes, and Settlements

You can configure how BRM calculates taxes for adjustments, disputes, and settlements. For example, whether tax reversals occur when account adjustments are created or at billing time.

You can specify whether to perform an adjustment, dispute, or settlement with or without taxes at the bill, item, and event level. For taxable account, bill, and item adjustments, BRM uses information supplied by a tax configuration file to calculate the tax reversal for the account.

If you include taxes in an A/R action and billing has already occurred, taxes are always calculated when the adjustment, dispute, or settlement is created. If billing has not occurred, the calculated amount is applied only if the item has real-time taxes. If the item has deferred taxes, only the adjustment, dispute, or settlement amount is applied; the tax amount is calculated and applied at billing time.

Note:

To ensure tax calculations are performed correctly for adjustments, disputes, settlements, and write-offs:

- Configure a separate item for taxes exclusively. To configure a separate item for taxes, see "Cumulative Custom Item for Taxes" in *BRM Configuring and Running Billing*.
- All events associated with an item must use the same tax method: tax now or tax deferred. Nontaxable events are valid in either taxation scenario.

BRM calculates tax as a percentage of the net amount (amount without taxes) of the original item. For example, consider a \$110 item that has a 10% tax. The net amount is \$100, and the tax amount is \$10. If you perform a 50% adjustment with tax, the net adjustment amount is \$50 and the tax reversal amount is \$5 (10% of 50).

When BRM computes tax, it compares the calculated tax amount with the cycle tax's due amount:

- If the calculated tax amount is greater than the cycle tax due amount, the cycle tax due amount is transferred to the tax item.
- If the calculated tax amount is less than or equal to the cycle tax due amount, the calculated tax amount is used.

 **Note:**

For untaxed adjustments, BRM adjusts the amount but does not calculate or apply a tax reversal for the adjustment amount.

To calculate adjustment, dispute, or settlement tax, BRM acquires tax information from one of two sources:

- For taxable adjustments, disputes, and settlements at the account, bill, and item level, BRM uses information supplied by a tax configuration file to calculate the tax reversal.

 **Note:**

This tax configuration file supports only a single tax code, that is, for account adjustments, bill adjustments, and item adjustments. The tax reversal would happen according to the single tax code mentioned in the configuration object.

- For taxable adjustments, disputes, and settlements at the event level, BRM obtains tax information from the event itself if BRM is configured for the tax now method. Otherwise, BRM obtains the taxes from the **!event/billing/cycle/tax** object.

In either case, this information typically includes the event type, tax supplier, and tax code. BRM uses this information in concert with the tax locale determined at bill run time to calculate the adjustment tax. For information on tax suppliers and tax codes, see "About Calculating Taxes".

For BRM to correctly apply adjustment, dispute, and settlement taxes, you must configure tax information for events, and define characteristics such as whether BRM should defer taxes. For more information, see "[Configuring Tax Calculation for Adjustments, Disputes, and Settlements](#)".

About Adjustment Tax and Billing Cycles

If the adjustment occurs at the account, subscription service, or member service level, BRM uses the total amount of the adjustment, without consideration of the specific events that are included in the adjustment.

For bill, item, and event adjustments, BRM distinguishes between items and events that need real-time taxes and those that need deferred taxes. If some adjusted items or events are nontaxable, BRM omits these from the deferred amount that it will

transfer for taxation during billing. For partial adjustments, BRM also proportions the deferred amount according to how much of an adjustment was actually granted.

About Dispute and Settlement Taxes and Billing Cycles

For bill and item disputes and settlements, real-time taxes are applied at the time the dispute or settlement is created. When processing a dispute or settlement with deferred taxes, BRM determines whether the original item has been billed. If billing has not yet occurred, BRM has not levied the tax for the original item. In this case, there is no need apply the taxes separately because the dispute or settlement will have acted on the balance of the original item before BRM ever calculates the taxes for that item.

However, if the dispute or settlement is opened against an item that has already been billed, the customer has already been charged tax for the item. In this case, BRM must specifically accommodate dispute or settlement tax. BRM defers applying the tax reversal amount until billing is run.

For partial disputes and settlements, BRM proportions the deferred amount in accordance with the relative size of the dispute or settlement.

Configuring Tax Calculation for Adjustments, Disputes, and Settlements

Depending on tax law, jurisdictions, and other factors, you may need to specify whether to apply a tax reversal when performing certain adjustments, disputes, and settlements. For example, the law may require that you reverse taxes for any adjustment you apply to a charge for a call placed outside the country, but not require you to reverse taxes for adjustments to prepaid balances.

As a prerequisite for this configuration, you must load a **/config/ar_taxes** object that defines tax treatment parameters for each event type. BRM uses this object to enrich the adjustment, dispute, and settlement opcode flists with the tax information needed to calculate the tax reversal.

After loading the **pin_config_ar_taxes.xml** file, you have the following options:

- If you use Customer Center for accounts receivable operations, you can configure a default tax treatment for adjustments, disputes, and settlements. Customer Center uses this configuration to determine whether these activities always have a tax reversal, never have a tax reversal, or have a tax reversal at the CSR's discretion. See "[Configuring the Default Tax Treatment for Customer Center](#)".
- If you don't use Customer Center to make adjustments, you can edit the CM **pin.conf** file to configure BRM to specify that tax reversals occur at the time account adjustments are created rather than deferring them until billing. See "[Configuring the Default Tax Method for Account Adjustments](#)".

If you use Customer Center, the tax reversal entry in the **pin.conf** file is ignored. Instead, you use the Customer Center Configurator to set up the tax treatment.

Before you load the tax configuration, you must first load the tax suppliers. See "[About Tax Suppliers](#)".

The **pin_config_ar_taxes.xml** file provides information on the tax supplier and tax code for each adjustment, dispute, and settlement event type.

- The tax suppliers you specify in the **pin_config_ar_taxes.xml** file must be listed in the **tax_supplier_map** file.
- The tax code you use in the **pin_config_ar_taxes.xml** file must be present in the **taxcodes_map** file. The tax code is mapped to a package that calculates tax for that charge offer type, and can be a numeric or descriptive string.

For information on tax suppliers, tax codes, and the associated map files, see "[Creating Tax Suppliers](#)".

The **pin_config_ar_taxes.xml** file must follow all standard XML formatting rules. The following example shows a typical **pin_config_ar_taxes.xml** file.



Note:

The **<Name>** field in the sample identifies the tax supplier.

```
<?xml version="1.0" encoding="UTF-8" ?>
<AccountReceivablesConfiguration>
  <TaxConfigurationList>
    <TaxConfiguration>
      <Event>/event/billing/adjustment/item</Event>
      <Name>ABC Inc - California branch</Name>
      <TaxCode>usage</TaxCode>
    </TaxConfiguration>
    <TaxConfiguration>
      <Event>/event/billing/dispute/item</Event>
      <Name>ABC Inc - California branch</Name>
      <TaxCode>usage</TaxCode>
    </TaxConfiguration>
    <TaxConfiguration>
      <Event>/event/billing/settlement/item</Event>
      <Name>ABC Inc - New Jersey branch</Name>
      <TaxCode>usage</TaxCode>
    </TaxConfiguration>
    <TaxConfiguration>
      <Event>/event/billing/adjustment/account</Event>
      <Name>ABC Inc - Illinois branch</Name>
      <TaxCode>cycle</TaxCode>
    </TaxConfiguration>
  </TaxConfigurationList>
</AccountReceivablesConfiguration>
```

To load the tax configuration:

1. Edit the sample **pin_config_ar_taxes.xml** file in the *BRM_home/sys/data/config* directory.

You can name the file anything you want and place it anywhere you like.

2. Save the **pin_config_ar_taxes.xml** file.
3. Use the following command to run the **load_pin_ar_taxes** utility:

```
load_pin_ar_taxes -f input_file.xml
```

where *input_file.xml* is the name and path to the XML file.

4. Stop and restart the Connection Manager (CM) and, if necessary, your client application.

To verify that the `pin_config_ar_taxes.xml` file was loaded, you can display the `/config/ar_taxes` object by using the Object Browser, or use the `robj` command with the `testnap` utility. See "Reading an Object and Writing Its Contents to a File" in *BRM Developer's Guide*.

Configuring the Default Tax Treatment for Customer Center

If you are using Customer Center as your CRM application, you can configure a default tax treatment for adjustments, disputes, and settlements. Customer Center uses this configuration to determine whether these activities always have a tax reversal, never have a tax reversal, or have a tax reversal at the CSR's discretion. Configuring a default tax treatment can help you enforce uniform tax treatments for a given location and eliminate the need for a decision by the CSR.

You configure the tax treatment on the **Balance** tab in the Customer Center Configurator. This configuration overrides any tax treatment specification in the `pin.conf` file. To set up tax treatment for adjustments, disputes, and settlements, select one of the following options under **Tax treatment** on the Configurator **Balance** tab:

- **Include tax:** Always perform a tax reversal for the adjustment, dispute, or settlement.
- **Exclude tax:** Never perform a tax reversal for the adjustment, dispute, or settlement.
- **None:** Allow the CSR to choose whether to include taxes.

For information on using the Customer Center Configurator **Balance** tab and the implications of these three options, see "Balance Configurator" in *BRM Developer's Guide*.

Configuring the Default Tax Method for Account Adjustments

You can configure BRM to calculate the tax for account adjustments using one of these methods:

- **Billing-time taxation:** BRM defers tax calculation for all account adjustments until the end of the billing cycle. This is the default.
- **Adjustment-time taxation:** BRM calculates taxes when an account adjustment is created.
- **Dynamic taxation:** BRM defers tax calculation for all account adjustments until the end of the billing cycle. However, it calculates the tax using the tax rate at the time of the account adjustment. For example, if an account creates an adjustment on June 15 and has a billing date of June 30, BRM calculates the tax on June 30 using the tax rate from June 15. This provides the benefits of billing-time taxation while allowing you to change tax rates in the middle of a billing cycle.

To configure the default tax method for account adjustments:

1. Open the Connection Manager configuration file (`BRM_home/sys/cm/pin.conf`).
2. Modify the `tax_now` parameter:

```
-fm_ar tax_now value
```

where `value` is set to one of the following:

- **0:** Billing-time taxation. This is the default value.
- **1:** Adjustment-time taxation.

- 2: Dynamic taxation.

 **Note:**

To ensure tax calculations are performed correctly, all events associated with an item must use the same tax method.

3. Save and close the file.

You do not need to restart the CM to enable this entry.

3

Creating Tax Codes

In Oracle Communications Billing and Revenue Management (BRM), you use tax codes to assign taxes to the products and services that you sell to your customers. For example, you use tax codes to apply taxes differently for usage charges and charges for physical goods.

Topics in this document:

- [About Creating Tax Codes \(Patch Set 6 and Later\)](#)
- [About Creating Tax Codes \(Patch Set 5\)](#)
- [About Creating Tax Codes \(Patch Set 4 and Earlier\)](#)

About Creating Tax Codes (Patch Set 6 and Later)

A tax code indicates which tax to apply based on a BRM charge offer. For example, a telephone handset uses a different tax code than an online service subscription. When a customer is charged, the tax code identifies the tax to apply.

Before BRM 12.0 Patch Set 6, you created tax codes in BRM and synchronized them to Pricing Design Center (PDC). In Patch Set 6 and later releases, you create tax codes in PDC and publish them to BRM and Elastic Charging Engine (ECE) using the **ImportExportPricing** utility.

To implement tax codes in BRM 12.0 Patch Set 6 and later releases:

1. Define your tax codes in an XML file. You can base your tax codes on those provided in the **Sample_Tax_Code.xml** file available in the *PDC_home/apps/Samples/Examples* directory, where *PDC_home* is the directory in which the PDC software is installed. The sample file contains examples of tax codes for calculating taxes based on a custom implementation or Vertex. For more information about XML elements specific to Vertex or custom tax codes, see:
 - [Creating Tax Codes for Vertex \(Patch Set 6 and Later\)](#)
 - [Creating Tax Codes for a Flat Tax or Custom Implementation \(Patch Set 6 and Later\)](#)
2. Import the XML file into PDC using the **ImportExportPricing** utility:

```
ImportExportPricing -import -config MyTaxCodes.xml
```

where *MyTaxCodes* is the name of the XML file where you defined the tax codes.

You can also modify existing tax codes using the **-ow** parameter.

See "Importing Pricing and Setup Components" and "ImportExportPricing" in *PDC Creating Product Offerings* for more information about the utility's syntax and parameters.

The **ImportExportPricing** utility also publishes the tax codes to the real-time rating engine (RRE) and ECE.

 **Note:**

You cannot publish tax codes to the batch rating engine (BRE).

Creating Tax Codes for a Flat Tax or Custom Implementation (Patch Set 6 and Later)

In BRM 12.0 Patch Set 6 and later, you implement simple flat taxes by defining them in an XML file that you import into PDC. In the file, you enter a tax code and then assign it to one or more tax rates, which are differentiated by criteria such as validity date and jurisdiction.

The following XML example shows custom tax codes:

```

<taxCodes>
  <name>CustomTaxCode</name>
  <priceListName>Default</priceListName>
  <obsolete>>false</obsolete>
  <code>CustomTaxCode</code>
  <taxPackageType>CUSTOM</taxPackageType>
  <taxCodeValidityPeriods>
    <validFrom>20220101</validFrom>
    <validTo>20230120</validTo>
    <taxCodeMaps>
      <description>TRS1</description>
      <percent>4.0</percent>
      <taxJurisdictionLevel>FEDERAL</taxJurisdictionLevel>
      <taxJurisdictions>US</taxJurisdictions>
      <taxRuleType>TAX</taxRuleType>
    </taxCodeMaps>
  </taxCodeValidityPeriods>
  <taxCodeValidityPeriods>
    <validFrom>20240101</validFrom>
    <validTo>20300120</validTo>
    <taxCodeMaps>
      <description>TRS2</description>
      <percent>8.0</percent>
      <taxJurisdictionLevel>LOCATION</taxJurisdictionLevel>
      <taxJurisdictions>EU</taxJurisdictions>
      <taxRuleType>INCLUSIVE</taxRuleType>
    </taxCodeMaps>
    <taxCodeMaps>
      <description>TRS3</description>
      <percent>8.0</percent>
      <taxJurisdictionLevel>LOCATION</taxJurisdictionLevel>
      <taxJurisdictions>ASIA</taxJurisdictions>
      <taxRuleType>NCS</taxRuleType>
    </taxCodeMaps>
  </taxCodeValidityPeriods>
</taxCodes>

```

When the value of the **<taxPackageType>** element is **CUSTOM**, one of the following is required under the **<taxCodeMaps>** element:

- **<percent>**: If this element is provided, the elements described in [Table 3-1](#) are required as well.
- **<transType_CategoryCode_Rate>**: If this element is provided, no other elements are supported in the tax code map.

[Table 3-1](#) describes the important XML elements specific to custom tax codes.

Table 3-1 XML Elements in the Tax Code File for Custom Tax Codes

Element	Description
code	The tax code. A unique alphanumeric value that defines categories with different tax treatments.
taxPackageType	The package type. For flat taxes and custom implementations, this is always CUSTOM .
validFrom	Start date of the validity period for the tax rate in <i>yyymmdd</i> format. Note: You can use multiple validity periods to create different rates for a single tax code, as shown in the example, but the validity periods must not overlap.
validTo	End date of the validity period for the tax rate in <i>yyymmdd</i> format.
description	A brief description of the tax code.
percent	The tax rate in percent. For example, 4.25 means 4.25%. For prepaid purchase events that grant negative currency balances, the corresponding tax associated with it should also be negative. For example, -4.25 .
taxJurisdictionLevel	Jurisdiction level for which this rate is applicable. Values are FEDERAL, STATE, COUNTY, CITY, LOCATION .
taxJurisdictions	The jurisdiction where the rate applies. Similar to a nexus for the corresponding jurisdiction level. For example, if the taxJurisdictionLevel value is STATE , the taxJurisdictions values must be state-level jurisdictions.
taxRuleType	Determines how taxes will be computed. Values are: <ul style="list-style-type: none"> • STANDARD: Standard tax computation. Taxes are computed based on the taxable amount and are then added to the total. • TAX: "Tax on tax" computation. Taxes are computed based on previous taxable amounts and taxes, and are then added to the total. For example, if tax1 = 10%, tax2 = 20%, and charge = 100.00, taxes are computed as follows: <ul style="list-style-type: none"> – tax1 = 10% @ 100.00 = 10.00 – tax2 = 20% @ (100.00 + 10.00) = 22.00 • INCLUSIVE: Inclusive tax computation. Taxes are already included in the total. For example, if the total is \$110 and the tax is 10%, the tax amount is \$10 and the charge is \$100. • NCS: Noncumulative standard tax computation. Taxes are computed based on the taxable amount but are not added to the total. • NCT: Noncumulative "tax on tax" computation. Taxes are computed based on the taxable amount but are not added to the total. For example, if tax1 = 10%, tax2 = 20%, and charge = 100.00, taxes are computed as follows: <ul style="list-style-type: none"> – tax1 = 10% @ 100.00 = 10.00 – tax2 = 20% @ 100.00 = 20.00

Creating Tax Codes for Vertex (Patch Set 6 and Later)

In BRM 12.0 Patch Set 6 and later, you specify to calculate taxes using Vertex by mapping tax codes to Vertex product codes in an XML file that you import into PDC. The Vertex product codes are used to apply the tax rate.

The following XML example shows the two types of Vertex tax code. The three elements under the **taxCodeMaps** element are the only ones supported for Vertex tax codes.

```
<taxCodes>
  <name>VQTaxCode</name>
  <priceListName>Default</priceListName>
  <obsolete>>false</obsolete>
  <code>VQTaxCode</code>
  <taxPackageType>VERTEX_QUANTUM</taxPackageType>
  <taxCodeValidityPeriods>
    <taxCodeMaps>
      <transType_CategoryCode_Rate>01</
transType_CategoryCode_Rate>
      <transSubType_ServiceCode>01</transSubType_ServiceCode>
      <salesIndicator>SALE</salesIndicator>
    </taxCodeMaps>
  </taxCodeValidityPeriods>
</taxCodes>
<taxCodes>
  <name>VCTaxCode</name>
  <priceListName>Default</priceListName>
  <obsolete>>false</obsolete>
  <code>VCTaxCode</code>
  <taxPackageType>VERTEX_COMMTAX_21</taxPackageType>
  <taxCodeValidityPeriods>
    <taxCodeMaps>
      <transType_CategoryCode_Rate>4</
transType_CategoryCode_Rate>
      <transSubType_ServiceCode>2</transSubType_ServiceCode>
      <salesIndicator>SALE</salesIndicator>
    </taxCodeMaps>
  </taxCodeValidityPeriods>
</taxCodes>
```

Table 3-2 describes the important XML elements specific to Vertex tax codes.

Table 3-2 XML Elements in the Tax Code File for Vertex

Element	Explanation
code	The tax code. A unique alphanumeric value that defines categories with different tax treatments.
taxPackageType	The tax package you use: <ul style="list-style-type: none"> VERTEX_QUANTUM: Vertex Communications Tax Q Series (telecommunications). VERTEX_COMMTAX_21: Vertex Sales Tax Q Series (sales and use).

Table 3-2 (Cont.) XML Elements in the Tax Code File for Vertex

Element	Explanation
transType_CategoryCode_Rate	<p>The field value is determined by the tax package you use.</p> <ul style="list-style-type: none"> For Vertex Communications Tax Q Series, this field is the category code. See the Communications Tax Q Series documentation for more information about its category codes. For Vertex Sales Tax Q Series, this field is the transaction type. See the Sales Tax Q Series documentation for more information on transaction types.
transSubType_ServiceCode	<p>The field value is determined by the tax package you use.</p> <ul style="list-style-type: none"> For Vertex Communications Tax Q Series, this field is the service code. Refer to the Communications Tax Q Series documentation for more information on service codes. For Vertex Sales Tax Q Series, this field is the transaction subtype. Refer to the Sales Tax Q Series documentation for more information on transaction subtypes.
salesIndicator	<p>For Vertex Communications Tax Q Series and Vertex Sales Tax Q Series, this is the resale flag field, which indicates if the charge offer is for sale (SALE) or resale (RESALE). The default is SALE.</p>

See your tax package documentation for information about product codes, service codes, and service indicators.

When configuring Vertex tax calculation, you also need to configure tax suppliers. See "[Creating Tax Suppliers](#)".

About Creating Tax Codes (Patch Set 5)

A tax code indicates which tax to apply based on a BRM charge offer. For example, a telephone handset uses a different tax code than an online service subscription. When a customer is charged, the tax code identifies the tax to apply.

To implement tax codes in BRM 12.0 Patch Set 5, you do the following:

1. Define your tax codes in the *BRM_home/sys/data/config/config_taxcodes_map.xml* file. The XML file contains different content depending on how you calculate taxes:
 - If you calculate taxes based on a flat tax or a custom implementation, the file defines the tax rate, such as 4%. See "[Creating Tax Codes for Flat Taxes and Custom Implementations \(Patch Set 5\)](#)".
 - If you calculate taxes by using Vertex, the file does not include the tax rate. Instead, it maps BRM tax codes to Vertex product codes. The Vertex product codes are used to apply the tax rate. See "[Creating Tax Codes for Vertex \(Patch Set 5\)](#)".
2. Load the XML file into the */config/taxcodes_map* object in the BRM database by using the **load_config** utility:

```
load_config config_taxcodes_map.xml
```

See "load_config" in *BRM Developer's Guide* for more information about the utility's syntax and parameters.

 **Note:**

In Patch Set 5, you cannot use the **SyncPDC** utility to synchronize tax codes from BRM to Pricing Design Center (PDC). Instead, you must manually re-create the BRM tax codes in PDC. For more information, see "Moving Existing Tax Codes from BRM to PDC for Patch Set 5" in *BRM Patch Set Release Notes*.

Creating Tax Codes for Flat Taxes and Custom Implementations (Patch Set 5)

In BRM 12.0 Patch Set 5, you implement simple flat taxes by defining them directly in the **config_taxcodes_map.xml** file. In the file, you enter a tax code and then assign it to one or more tax rates, which are differentiated by criteria such as validity date and jurisdiction.

[Table 3-3](#) describes the elements in the **config_taxcodes_map.xml** file for configuring flat taxes.

Table 3-3 Tax Code Map Elements for Flat Taxes

Element	Description
<TAX_CODE >	The tax code. A unique alphanumeric value that defines categories with different tax treatments. Note: If you create multiple instances of the same tax code, they must all use the same start and end dates.
<TAXPKG_T AX_CODE>	The package code. For flat taxes and custom implementations, this is always U (<i>user-defined</i>).
<PERCENT>	The tax rate in percent. For example, 4.25 means 4.25%. For prepaid purchase events that grant negative currency balances, the corresponding tax associated with it should also be negative. For example: -4.25 .
<VALID_FROM STR>	The start date for the validity period in <i>mm/dd/yy</i> format. Note: If you create multiple instances of the same tax code, they must all use the same start date.
<VALID_TO STR>	The end date for the validity period in <i>mm/dd/yy</i> format. Note: If you create multiple instances of the same tax code, they must all use the same end date.
<TAX_JURIS DICTION>	The tax jurisdiction: <ul style="list-style-type: none"> • Fed: Federal level • Sta: State level • Cou: County level • Cit: City level • Loc: Location level

Table 3-3 (Cont.) Tax Code Map Elements for Flat Taxes

Element	Description
<TAX_JURISDICTION_LIST>	The list of jurisdiction values applicable for this rate. This is similar to a nexus for the corresponding jurisdiction level. For example, if the tax jurisdiction is Sta , the list of values must be state-level jurisdictions. Separate list values by a semicolon (;). To make the value applicable to all, use an asterisk (*).
<DESCR>	A brief description of the tax code.
<RULE_TYPE>	The rule for calculating the tax amount: <ul style="list-style-type: none"> • Std: Standard tax computation. Taxes are computed based on the taxable amount and then added to the total. • Tax: "Tax on tax" computation. Taxes are computed based on previous taxable amounts and are then added to the tax total. For example, if tax1 = 10%, tax2 = 20%, and charge = 100.00, taxes are computed as follows: tax1 = 10% @ 100.00 = 10.00 tax2 = 20% @ (100.00 + 10.00) = 22.00 • Inc: Tax inclusive. The line amount for an item includes the tax for the given item. • NCS: Noncumulative standard tax computation. Taxes are computed based on the taxable amount but are not added to the total. • NCT: Noncumulative "tax on tax" computation. Taxes are computed based on the taxable amounts and taxes but are not added to the total. For example, if tax1 = 10%, tax2 = 20%, and charge = 100.00, taxes are computed as follows: tax1 = 10% @ 100.00 = 10.00 tax2 = 20% @ 100.00 = 20.00

The following sample content for the **config_taxcodes_map.xml** file creates two sample tax codes, which are both valid from February 1, 2022 through January 31, 2023:

- Federal VAT tax of 4.5% for Great Britain
- County sales tax of 1.25%

```
<ConfigObjectconfigName="/config/taxcodes_map"configMode="recreate">
  <DESCR>Taxcodes Map Configuration</DESCR>
  <NAME>taxcodes_map</NAME>
  <TAXES elem="1">
    <TAX_CODE>VAT</TAX_CODE>
    <TAXPKG_TAX_CODE>U</TAXPKG_TAX_CODE>
    <PERCENT>4.5</PERCENT>
    <VALID_FROM_STR>02/01/22</VALID_FROM_STR>
    <VALID_TO_STR>01/31/23</VALID_TO_STR>
    <TAX_JURISDICTION>Fed</TAX_JURISDICTION>
    <TAX_JURISDICTION_LIST>GB</TAX_JURISDICTION_LIST>
    <DESCR>VAT-GB</DESCR>
    <RULE_TYPE>Std</RULE_TYPE>
  </TAXES>
  <TAXES elem="2">
    <TAX_CODE>purchase</TAX_CODE>
    <TAXPKG_TAX_CODE>U</TAXPKG_TAX_CODE>
    <PERCENT>1.25</PERCENT>
    <VALID_FROM_STR>02/01/22</VALID_FROM_STR>
    <VALID_TO_STR>01/31/23</VALID_TO_STR>
  </TAXES>
</ConfigObject>
```

```

    <TAX_JURISDICTION>Cou</TAX_JURISDICTION>
    <TAX_JURISDICTION_LIST>*</TAX_JURISDICTION_LIST>
    <DESCR>Sales</DESCR>
    <RULE_TYPE>Std</RULE_TYPE>
  </TAXES>
</ConfigObject>

```

Creating Tax Codes for Vertex (Patch Set 5)

In BRM 12.0 Patch Set 5, you specify to calculate taxes using Vertex by mapping tax codes to Vertex product codes in the **config_taxcodes_map.xml** file. The Vertex product codes are used to apply the tax rate.

Vertex product codes are numeric and therefore not descriptive. When you create your tax codes, you can use descriptive names, such as *usage*, *purchase*, and so on.

[Table 3-4](#) describes the elements in the **config_taxcodes_map.xml** file for configuring taxation with Vertex.

Table 3-4 Tax Code Elements for Vertex Taxation

Element	Description
<TAX_CODE >	The tax code. A unique alphanumeric value that defines categories with different tax treatments.
<TAXPKG_T AX_CODE>	The tax package you use: <ul style="list-style-type: none"> C: Vertex Communications Tax Q Series (telecommunications). Q: Vertex Sales Tax Q Series (sales and use).
<CODE1>	The field value is determined by the tax package you use. <ul style="list-style-type: none"> For Vertex Communications Tax Q Series, this field is the category code. See the Communications Tax Q Series documentation for more information about its category codes. For Vertex Sales Tax Q Series, this field is the transaction type. See the Sales Tax Q Series documentation for more information about transaction types.
<CODE2>	The field value is determined by the tax package you use. <ul style="list-style-type: none"> For Vertex Communications Tax Q Series, this field is the service code. See the Communications Tax Q Series documentation for more information about service codes. For Vertex Sales Tax Q Series, this field is the transaction subtype. See the Sales Tax Q Series documentation for more information about transaction subtypes.
<SALES_IND ICATOR>	The sales indicator. For Vertex Communications Tax Q Series and Vertex Sales Tax Q Series, this is the resale flag field, which indicates if the charge offer is for sale (S) or resale (R). The default is S .

The following sample content for the **config_taxcodes_map.xml** file maps two tax codes to Vertex product codes:

```

<ConfigObjectconfigName="/config/taxcodes_map"configMode="recreate">
  <DESCR>Taxcodes Map Configuration</DESCR>
  <NAME>taxcodes_map</NAME>
  <TAXES elem="1">
    <TAX_CODE>qt_usage</TAX_CODE>
    <TAXPKG_TAX_CODE>Q</TAXPKG_TAX_CODE>
    <CODE1>01</CODE1>
  </TAXES>
</ConfigObject>

```

```

        <CODE2>01</CODE2>
        <SALES_INDICATOR>S</SALES_INDICATOR>
    </TAXES>
    <TAXES elem="2">
        <TAX_CODE>qt_cycle</TAX_CODE>
        <TAXPKG_TAX_CODE>Q</TAXPKG_TAX_CODE>
        <CODE1>01</CODE1>
        <CODE2>01</CODE2>
        <SALES_INDICATOR>S</SALES_INDICATOR>
    </TAXES>
</ConfigObject>

```

When configuring Vertex tax calculation, you also need to configure tax suppliers. See ["Creating Tax Suppliers"](#).

About Creating Tax Codes (Patch Set 4 and Earlier)

A tax code indicates which tax to apply based on the BRM charge offer. For example, a telephone handset uses a different tax code than an online service subscription. When a customer is charged, the tax code identifies the tax to apply.

To implement tax codes in BRM 12.0 through BRM 12.0 Patch Set 4, you define them in the *BRM_home/sys/cm/taxcodes_map* file. The file contains different content depending on how you calculate taxes:

- If you calculate taxes based on a flat tax, the **taxcodes_map** file defines the tax rate, such as 4%. See ["Creating Tax Codes for a Flat Tax or Custom Implementation"](#).
- If you calculate taxes by using Vertex, the **taxcodes_map** file does not include the tax rate. Instead, it maps the BRM tax codes to Vertex product codes. The Vertex product codes are used to apply the tax rate.

Vertex product codes are numeric and therefore not descriptive. When you create your tax codes, you can use descriptive names, such as *usage*, *purchase*, and so on. See ["Creating Tax Codes for Vertex"](#).

After you edit the file, you do not need to load the **taxcodes_map** file into the BRM database.

If you change the name or location of the **taxcodes_map** file, edit the following line in the *BRM_home/sys/cm/pin.conf* file:

```
- fm_rate taxcodes_map location
```

where *location* is the name and path of the file.

Creating Tax Codes for a Flat Tax or Custom Implementation

You implement simple flat taxes by defining the tax rates directly in the **taxcodes_map** file. You enter one or more tax rates for each tax code. You can use several criteria, such as validity dates and jurisdictions, to specify more than one rate for each tax code.

[Table 3-5](#) provides descriptions of what you can enter in the each of the columns in the **taxcodes_map** file.

Table 3-5 taxcodes_map Columns

Column	Explanation
Taxcode	The tax code. A unique alphanumeric value that defines categories with different tax treatments.
Pkg	Package code. For flat taxes and custom implementations, this is always U , for <i>user-defined</i> .
Rate	The tax rate in percent. For example, 4.25 means 4.25%. For prepaid purchase events that grant negative currency balances, the corresponding tax associated with it should also be negative. See the following example: PREPAID_PURCH_TAX : U : -4.5 : 02/01/02 : 01/31/08 : Fed : GB : VAT-GB : Std
Start	Start date of the validity period for the tax rate in <i>mm/dd/yy</i> format.
End	End date of the validity period for the tax rate in <i>mm/dd/yy</i> format.
Lvl	Jurisdiction level for which this rate is applicable. Values are: Fed : Federal level Sta : State level Cou : County level Cit : City level
List	List of jurisdiction values applicable for this rate. Similar to a nexus for the corresponding jurisdiction level. For example, if the Lvl value is Sta , the List values must be state-level jurisdictions. Separate list values by a semicolon (;). To make the value applicable to all, use an asterisk (*).
Descr	Text description of the tax.
Rule	Determines how taxes will be computed. Values are: Std : Standard tax computation. Taxes are computed based on the taxable amount and are then added to the total. Tax : "Tax on tax" computation. Taxes are computed based on previous taxable amounts and taxes and are then added to the total. For example, if tax1 = 10%, tax2 = 20%, and charge = 100.00, taxes are computed as follows: tax1 = 10% @ 100.00 = 10.00 tax2 = 20% @ (100.00 + 10.00) = 22.00 NCS : Noncumulative standard tax computation. Taxes are computed based on the taxable amount but are not added to the total. NCT : Noncumulative "tax on tax" computation. Taxes are computed based on the taxable amount but are not added to the total. For example, if tax1 = 10%, tax2 = 20%, and charge = 100.00, taxes are computed as follows: tax1 = 10% @ 100.00 = 10.00 tax2 = 20% @ 100.00 = 20.00

The following shows the contents of the **taxcodes_map** file for a flat tax.

```
# Taxcode   Pkg Rate   Start   End     Lvl   List   Descr
Rule
```

```
# -----
usage      : U : 4.0 : 02/01/16 : 01/31/17 : Fed : US      : USF      : Std
cycle     : U : 3.5 : 02/01/16 : 01/31/17 : Fed : US      : Excise   : Std
purchase  : U : 8.25 : 02/01/16 : 01/31/17 : Sta : CA      : Sales    : Std
toll      : U : 2.0 : 02/01/16 : 01/31/17 : Fed : US      : TRS      : Tax
toll      : U : 1.5 : 02/01/17 : 01/31/18 : Sta : CA      : 911      : Tax
toll      : U : 2.0 : 01/01/18 : 01/01/19 : Sta : CA      : B&O      : Tax
toll      : U : 3.15 : 01/01/16 : 01/01/17 : Cit : Cupertino : Deaf     : Tax
VAT       : U : 5.0 : 02/01/16 : 01/31/17 : Fed : GB;FR   : VAT-EU   : Std
VAT     : U : 4.5 : 02/01/16 : 01/31/18 : Fed : GB    : VAT-GB : Std
VAT       : U : 4.0 : 02/01/16 : 01/31/18 : Fed : FR      : VAT-FR   : Std
```

Creating Tax Codes for Vertex

The **taxcodes_map** file for Vertex maps BRM tax codes to Vertex product codes.

Table 3-6 lists the columns in the **taxcodes_map** file for Vertex:

Table 3-6 taxcodes_map Columns for Vertex

Column	Explanation
Taxcode	The tax code. A unique alphanumeric value that defines categories with different tax treatments.
Pkg	The tax package you use: <ul style="list-style-type: none"> C: Vertex Communications Tax Q Series (telecommunications). Q: Vertex Sales Tax Q Series (sales and use).
Code 1	The field value is determined by the tax package you use. <ul style="list-style-type: none"> For Vertex Communications Tax Q Series, this field is the category code. See the Communications Tax Q Series documentation for more information about its category codes. For Vertex Sales Tax Q Series, this field is the transaction type. See the Sales Tax Q Series documentation for more information on transaction types.
Code 2	The field value is determined by the tax package you use. <ul style="list-style-type: none"> For Vertex Communications Tax Q Series, this field is the service code. Refer to the Communications Tax Q Series documentation for more information on service codes. For Vertex Sales Tax Q Series, this field is the transaction subtype. Refer to the Sales Tax Q Series documentation for more information on transaction subtypes.
Si	Sales indicator. For Vertex Communications Tax Q Series and Vertex Sales Tax Q Series, this is the resale flag field, which indicates if the charge offer is for sale (S) or resale (R). The default is S.

See your tax package documentation for information about product codes, service codes, and service indicators.

The following is an example of a **taxcodes_map** file for Vertex:

```
# Taxcode      Pkg  Code1  Code2  Si
# -----
# usage       : T  : 85000 : 85000 : S
# cycle      : T  : 61000 : 61000 : S
# t_shirt    : T  :   899 :   899 :
# modem_rent : T  :   234 :  1234 : R
# usage2     : Q  :    01 :    01 : S
# direct     : C  :    01 :    01 : S
# toll       : C  :    04 :    02 : S
```



```
# direct2      : B : 01 : 04 : S
# toll2       : B : 01 : 01 : S
# user_code    : U : 10.0
monthlyCard   : Q : 03 : 03 : S
installVertex : Q : 03 : 03 : S
interCommtax  : C : 04 : 01 : S
chet_is_cool  : C : 04 : 01 : S
intraCommtax  : C : 01 : 02 : S
ipUsage       : Q : 03 : 03 : S
```

When configuring Vertex tax calculation, you also need to configure tax suppliers. See ["Creating Tax Suppliers"](#).

4

Creating Tax Suppliers

In Oracle Communications Billing and Revenue Management (BRM), the tax supplier data defines the location of your business. Vertex uses this data to determine which tax rate to apply to a given product offering.

Topics in this document:

- [About Tax Suppliers](#)
- [Providing the Tax Supplier Data with Each Transaction](#)
- [Allowing Your Sales and Use Tax Software to Determine Tax Supplier Data](#)
- [Defining a Default Ship-From Locale](#)

About Tax Suppliers

A tax supplier is the company or corporate division responsible for collecting taxes for a given transaction. Tax suppliers typically include your corporate headquarters and branch offices. Vertex uses the tax supplier to calculate taxes.

Tax supplier data can include the following information:

- The name of the tax supplier.
- The address to use as the ship-from locale.
- The tax nexus state. You can have a tax nexus in any state where your company has a substantial presence, as defined by federal tax laws.
- The VAT certificate number.

Tax software typically calculates taxes by using four locales:

- **Shipped-to:** The address of the customer who made the purchase. BRM obtains the shipped-to address from the customer's account. For telecommunications taxation, this is the termination number.
- **Shipped-from:** The address of the company supplying the product. For telecommunications taxation, this is the origination number.
- **Bill origin:** The location from which the product was shipped (also called *point of origin*); for example, a warehouse. This is not used for telecommunications taxation.
- **Bill approval:** The location where the order for the product was taken (also called *point of acceptance*); for example, the location of a customer service representative who creates accounts. For telecommunications taxation, this is the charge-to number.

For sales and use taxes, BRM does not differentiate between ship from, bill origin, and bill approval. Therefore, when providing the tax software with data, BRM uses the same value for all three locales. This value is derived from the tax supplier's address. For telecommunications taxation, the numbers are derived from call details records (CDRs).

 **Note:**

For telecommunications taxation, the shipped-to, bill origin, and bill approval locales are not used and are therefore stored in BRM with a NULL value.

You specify how to supply tax supplier data to your tax software in one of the following ways:

- Provide the tax supplier data with each transaction that you send to the tax software. For example, each transaction might include the ship-from locale based on the tax supplier address and the applicable VAT certificate. See "[Providing the Tax Supplier Data with Each Transaction](#)" for more information.
- Allow the tax software to determine the correct tax supplier based on the transaction data plus the company and business locations you give the tax software. You provide information to the tax software by running the tax software's toolkits. The tax software then determines which ship-from locale to use, if there is a tax nexus presence, and which VAT certificate to use. See "[Allowing Your Sales and Use Tax Software to Determine Tax Supplier Data](#)" for more information.

You can also configure a default ship-from address in case BRM cannot find any tax supplier. See "[Defining a Default Ship-From Locale](#)".

Providing the Tax Supplier Data with Each Transaction

If your business uses only one tax supplier, you can define a default tax supplier to use for all charge offers. You do not need to enter any tax supplier information when creating charge offers.

You can specify multiple tax suppliers. Use PDC or Pricing Center to assign each charge offer to a tax supplier. When an event is rated, the tax supplier defined in the charge offer is sent to the tax software.

For example, if you have different charge offers for different states, you can indicate the state in which the charge offer was sold by selecting the tax supplier for that state.

To use this method of defining tax suppliers, you edit the *BRM_home/sys/data/config/pin_tax_supplier.xml* file. You load tax suppliers into the database by running the **load_tax_supplier** utility.

You cannot add or change tax suppliers individually. Each time you run **load_tax_supplier**, you overwrite existing data with the entire contents of the *pin_tax_supplier.xml* file. To modify or remove existing tax suppliers, edit or remove their information in the file before loading it.

By default, the *pin_tax_supplier.xml* file is located in *BRM_home/sys/data/config*.

Add a **TaxSupplierElement** child element to the TaxSupplierConfiguration parent element for each tax supplier.

A **TaxSupplierElement** looks like this:

```
<TaxSupplierElement>
  <Name>supplier_name</Name>
  <Description>supplier_description</Description>
  <Address>supplier_address</Address>
```

```

<NexusInfo>nexus_values</NexusInfo>
<RegulatedFlag>flag_value</RegulatedFlag>
<DefaultFlag>flag_value</DefaultFlag>
<VATInfo>
  <CanonCountry>country_code</CanonCountry>
  <VATCertificate>certificate_number</VATCertificate>
</VATInfo>
</TaxSupplierElement>

```

Table 4-1 describes the `pin_tax_supplier.xml` file.

Table 4-1 `pin_tax_supplier.xml` File

Entry	Description
Name	Enter the tax supplier name. This name is the company ID in the <code>tax_supplier_map</code> file. For Vertex, this name is the company ID in the TDM or company file. If you use the TDM, the entry will be a four-character code. The name must match exactly.
Description	Enter a description for the tax supplier.
Address	Enter the address of the tax supplier. Use the following format: <i>city;state;zip_code;country</i>
NexusInfo	Enter a semicolon-delimited list of states in which the tax supplier has a tax nexus. If you use Vertex taxation software, enter an asterisk (*) to allow Vertex to determine the tax nexus. See " Configuring Vertex Manager " for more information.
RegulatedFlag	For Communications Tax Q Series, enter whether the utility is regulated (1) or unregulated (0).
DefaultFlag	Enter 1 for default the tax supplier; enter 0 for all other tax suppliers. If you have multiple tax suppliers, only one can be the default.
CanonCountry	Enter the country name. See the Vertex documentation for a list of valid values.
VATCertificate	Enter the VAT certificate number associated with the tax supplier. BRM currently does not use this entry. This entry can be used in a custom implementation of tax calculation.

The following `pin_tax_supplier.xml` file contains two tax suppliers; the first is the default tax supplier.

```

<TaxSupplierConfiguration>
  <TaxSupplierElement>
    <Name>TS_1</Name>
    <Description>Tax Supplier 1</Description>
    <Address>Cupertino;CA;95014;US</Address>
    <NexusInfo>*</NexusInfo>
    <RegulatedFlag>0</RegulatedFlag>
    <DefaultFlag>1</DefaultFlag>
    <VATInfo>
      <CanonCountry>US</CanonCountry>
      <VATCertificate>vat_cert_US</VATCertificate>
    </VATInfo>
  </TaxSupplierElement>
</TaxSupplierConfiguration>

```

```

        <CanonCountry>UK</CanonCountry>
        <VATCertificate>vat_cert_UK</VATCertificate>
    </VATInfo>
</TaxSupplierElement>
<TaxSupplierElement>
    <Name>TS_2</Name>
    <Description>Tax Supplier 2</Description>
    <Address>New York;NY;10013;US</Address>
    <NexusInfo>*</NexusInfo>
    <RegulatedFlag>0</RegulatedFlag>
    <DefaultFlag>0</DefaultFlag>
    <VATInfo>
        <CanonCountry>US</CanonCountry>
        <VATCertificate>vat_cert_US</VATCertificate>
    </VATInfo>
    <VATInfo>
        <CanonCountry>UK</CanonCountry>
        <VATCertificate>vat_cert_UK</VATCertificate>
    </VATInfo>
</TaxSupplierElement>
</TaxSupplierConfiguration>

```

To load tax suppliers:

1. Enter the following command to load the tax supplier data:

```
load_tax_supplier pin_tax_supplier.xml
```

Note:

If you are not working in the same directory as the `pin_tax_supplier.xml` file, include the complete path to the file. For example:

```
load_tax_supplier BRM_home/sys/data/config/pin_tax_supplier.xml
```

2. Stop and restart the CM.

Allowing Your Sales and Use Tax Software to Determine Tax Supplier Data

If your business has customers in multiple countries or operates under other conditions where complex tax laws are involved, you should allow your sales and use tax software to choose the correct tax supplier. The disadvantage is that you need to include every BRM charge offer in the `tax_supplier_map` file, and you need to update the file when you add or delete charge offers.

The `tax_supplier_map` file enables your tax package to determine the correct tax supplier company ID and business location, based on the charge offer and the ship-to locale.

For Sales Tax Q Series, the `tax_supplier_map` file also maps BRM tax supplier names to your company IDs stored in the company profile, or Sales Tax Q Series TDM file. BRM uses this information to maintain database integrity.

The following example shows part of a `tax_supplier_map` file:

Product	ShipTo	Company ID	Business loc	Ship From	Reg
Item_One	::;CA;US	: Tax supplier1	: loc_1	: Cupertino;CA;95014;US	: 1
Sub_One	::;US	: Tax supplier1	: loc_1	: Cupertino;CA;95014;US	: 1
Sub_One	::;FR	: Tax supplier2	: loc_2	: Paris;;;FR	: 0

Table 4-2 provides a list of the entries in the **tax_supplier_map** file.

Table 4-2 Entries in tax_supplier_map File

Entry	Description
Product	The name of the charge offer for which taxes are calculated. See "Creating Pricing Setup Components" in <i>BRM Creating Product Offerings</i> .
ShipTo	The address the product is shipped to (your customer's location). Enter the address in this format: <i>city;state;ZIP;country</i> You can leave portions of this string blank, but you must include the semicolons for each address element. Examples: <ul style="list-style-type: none"> • ::;US • ;CA;95003;US In most cases, you do not need to enter the city. See the Vertex documentation for correct state and country entries.
Company ID	The company ID specified in the Vertex TDM (Tax Decision Maker) or company file. This name is also entered in the PIN_FLD_NAME field when you define tax suppliers. This entry must match the name exactly as entered in the TDM or company file. If you use the TDM, the entry will be a four-character code.
Business Loc	The business location specified in the Vertex TDM file or company file. This value indicates which location should be considered when taxes are calculated.
ShipFrom	The address the product is shipped from. Enter the address in this format: <i>city;state;ZIP;country</i> You can leave portions of this string blank, but you must include the semicolons for each address element. Examples: <ul style="list-style-type: none"> • ::;US • ;CA;95003;US The ship-from address that you define in the tax_supplier_map file can be different from the address used when you defined the default tax supplier. In this case, your tax package obtains the address from the tax_supplier_map file. This enables you to create multiple ship-from addresses for a single tax supplier. See the Vertex documentation for correct state and country entries.
Regulated	For telecommunications tax only, this flag identifies the utility or company that is billing as regulated (0) or unregulated (1). Some taxes apply to regulated or unregulated utilities only; this flag determines which types of taxes are applied.

To specify the location of the **tax_supplier_map** file:

1. Open the CM configuration file (*BRM_home/sys/cm/pin.conf*).
2. Add the **tax_supplier_map** entry. For example:


```
- fm_rate tax_supplier_map BRM_home/sys/cm/tax_supplier_map
```
3. Save and close the file.
4. Stop and restart the CM.

Specifying Divisions for Tax Suppliers

Some companies operate divisions with locations that can impact the tax calculation. Use the **Business Loc** column in the **tax_supplier_map** file to add information about divisions for tax suppliers. The data in this column is mapped to the **Division** field in Sales Tax Q Series. This data should match what is stored in the Sales Tax Q Series TDM database.

For example, suppose the tax supplier Acme has two divisions, one in California and one in Illinois. You can configure the **tax_supplier_map** file to assign customers to the appropriate location. For example:

```
# Product      ShipTo      Company Id   Business Loc   ShipFrom
Regulated
# -----
-----
electrical : ;;CA;US : Acme       : West        : Cupertino;CA;95014;US : 1
electrical : ;;TX;US : Acme       : Central     : Oak Brook;IL;60523;US : 1
```

With this configuration, for customers in California, BRM sends **West** as the division to Sales Tax Q Series. The tax supplier's address will be Cupertino, CA. Similarly, customers in Texas will be assigned to the **Central** division with the Oak Brook, IL address.

Defining a Default Ship-From Locale

This option provides a ship-from locale in case a tax supplier cannot be found. You need to define at least one tax supplier.

To define a default ship-from locale:

1. Open the CM configuration file (*BRM_home/sys/cm/pin.conf*).
2. Edit the **provider_loc** entry. For example:


```
- fm_rate_pol provider_loc Middletown, CA 95222 USA
```
3. Save and close the file.

The new value becomes effective immediately and all subsequent tax calculations use the new address. You do not need to restart the CM.

5

Configuring Tax Calculation Options

You can configure global tax calculation settings in Oracle Communications Billing and Revenue Management.

Topics in this document:

- [Configuring Tax Calculation for Account Groups](#)
- [Validating Customer Addresses](#)
- [Reporting Zero Tax Amounts](#)
- [Reporting Taxes of Zero Percent](#)
- [Enabling and Disabling Taxation Globally](#)
- [Configuring Taxation in Pricing Center](#)
- [Configuring Taxation in loadpricelist](#)
- [Configuring Tax Calculation in ECE](#)

Configuring Tax Calculation for Account Groups

In an account group, parent accounts always pay taxes for child accounts with nonpaying bill units (**/billinfo** objects). However, there are two ways to calculate taxes. Taxes can be calculated for each individual nonpaying child bill unit and can be listed as separate items on the parent bill, or taxes can be consolidated into a single item for both the parent and nonpaying bill units.

You specify how BRM calculates taxes for account groups by entering one of the following values in the **cycle_tax_interval** entry in the Connection Manager (CM) configuration (**pin.conf**) file:

- **accounting:** (Default) BRM calculates taxes separately for the parent and each nonpaying bill unit and then lists the taxes as separate items on the parent bill.
- **billing:** BRM rolls activities for each nonpaying bill unit into the parent account and calculates taxes for the parent account only. The single tax item on the parent account includes taxes from both the parent and nonpaying bill units. This option improves overall performance of BRM tax computation.

Note:

Regardless of the option you select for the entry, Bill Now always rolls activities for each nonpaying bill unit (**/billinfo** object) into the parent bill unit and calculates taxes for the parent only. The single tax item for the parent includes taxes from both the parent and nonpaying bill units.

To specify how to calculate taxes for account groups:

1. Open the CM configuration file (*BRM_home/sys/cm/pin.conf*).
2. Edit the **cycle_tax_interval** entry:
 - To forward the tax from the child account to the parent account, enter **billing**. BRM calculates taxes for the parent account only, but the single tax item on the parent account includes taxes from both the parent and child accounts.
 - To calculate the taxes separately for the parent and child accounts, enter **accounting**. BRM lists the taxes as separate items on the parent bill.

For example:

```
fm_bill cycle_tax_interval billing
```

3. Save and close the file.
4. Stop and restart the CM.

Validating Customer Addresses

When taxes are calculated for customers in the United States, the tax rate is partially determined by the customer's ZIP code. If the customer's ZIP code is not correct and fails validation, or it is for a different state, the tax software calculates the taxes as **0**.

To avoid this problem, you can configure BRM to validate the customer's state and ZIP code when the customer creates an account.

Some third-party tax packages provide address validation. You can enable address validation by specifying which package to use for the **tax_valid** entry in the CM **pin.conf** file. The validation performed depends on the software you specify:

- Vertex Sales Tax Q Series validates city, state, and ZIP code.
- Vertex Communications Tax Q Series does not support address validation. If you configure BRM to use Communications Tax Q Series for validation, it always returns a positive result.

To specify a tax package to validate addresses:

1. Open the CM configuration file (*BRM_home/sys/cm/pin.conf*).
2. Set the value of the **tax_valid** entry to one of the values in [Table 5-1](#).

Table 5-1 tax_valid Values

Value	Description
0	BRM does not check the ZIP code. This is the default value.
3	BRM uses Vertex Sales Tax Q Series to validate the ZIP code. (The Vertex DM must be running.)
4	No effect. BRM uses Communications Tax Q Series to validate the ZIP code, but Communications Tax Q Series does not perform any validation. A positive result is always returned.

For example:

```
- fm_cust_pol tax_valid 3
```

3. Save and close the file.

4. Stop and restart the CM.

Reporting Zero Tax Amounts

By default, when a zero amount is returned from taxation, BRM does *not* report the tax amount and the TAX_JURISDICTIONS array is not created for the event. However, legal requirements in some jurisdictions demand that zero tax amounts be reported.

To configure BRM to report zero taxes:

1. Open the CM configuration file (*BRM_home/sys/cm/pin.conf*).
2. Uncomment the **fm_rate include_zero_tax** entry and change the value of the entry to **1**:

```
- fm_rate include_zero_tax 1
```

3. Save and close the file.
4. Stop and restart the CM.

If multiple taxes, including zero taxes, are reported from the same jurisdiction, they are aggregated by default. This may hide the effect of reporting zero taxes. To prevent zero taxes from being hidden in this way, you should configure BRM to itemize taxes by jurisdiction.

To configure BRM to itemize all taxes by jurisdiction:

1. Open the CM configuration file (*BRM_home/sys/cm/pin.conf*).
2. Change the value of the following entry to **itemize**:

```
- fm_rate tax_return_juris itemize
```

3. Save and close the file.
4. Stop and restart the CM.

Reporting Taxes of Zero Percent

When a charge's tax rate is zero percent, BRM reports a **0** tax percentage rate and sets the PIN_FLD_BAL_IMPACTS to **0** by default. To configure this feature, run the **pin_bus_params** utility to change the **exclude_percent_for_zero_tax** business parameter. For more information about this utility, see "pin_bus_params" in *BRM Developer's Guide*.

Note:

This business parameter applies to flat taxes and custom tax implementations only. It does not apply when Vertex Manager is used for tax calculation.

To configure whether BRM reports a zero tax percentage amount:

1. Go to *BRM_home/sys/data/config*.
2. Create an XML file from the */config/business_params* object:

```
pin_bus_params -r BusParamsBilling bus_params_billing.xml
```

3. In the file, set the **exclude_percent_for_zero_tax** parameter:

```
<exclude_percent_for_zero_tax>value</exclude_percent_for_zero_tax>
```

where *value* is:

- **enabled** to have BRM report a zero tax percentage amount.
 - **disabled** to have BRM not report zero tax percentage amounts.
4. Save this file as **bus_params_billing.xml**.
 5. Load the XML file into the BRM database:


```
pin_bus_params bus_params_billing.xml
```
 6. Stop and restart the CM.
 7. (Multischema systems only) Run the **pin_multidb** script with the **-R CONFIG** parameter. For more information, see *BRM System Administrator's Guide*.

Enabling and Disabling Taxation Globally

You can turn BRM taxation on and off globally by using the **taxation_switch** entry in the CM configuration file (*BRM_home/sys/cm/pin.conf*). You can choose to enable real-time tax calculation, deferred tax calculation, or both (the default setting). You can also disable tax calculation entirely.



Note:

Because deferred taxation procedures are triggered even if BRM is not configured to use taxation features, turning taxation off can improve performance. If BRM is configured for taxation, you can turn taxation off and on for testing.

To enable or disable taxation globally:

1. Open the Connection Manager (CM) configuration file (*BRM_home/sys/cm/pin.conf*).
2. Set the value of the **taxation_switch** entry:
 - **0**: Tax calculation is entirely disabled.
 - **1**: Only real-time tax calculation is enabled.
 - **2**: Only deferred (cycle-time) tax calculation is enabled.
 - **3**: (Default) Both real-time and deferred tax calculation are enabled.

For example, this specifies to enable real-time tax calculation only:

```
- fm_bill taxation_switch 1
```

3. Save and close the file.
4. Stop and restart the CM.

Configuring Taxation in Pricing Center

You can specify the following for each rate plan that you create in Pricing Center:

- The tax code to apply
- When to apply the tax code: during real-time rating or billing

To assign taxation to specific rate plans:

1. Open a real-time rate plan in Pricing Center.
2. In the Rate Plan Properties dialog box, specify when to calculate taxes and which tax code to apply as shown in [Figure 5-1](#).

Figure 5-1 Rate Plan Properties Dialog Box

The screenshot shows a dialog box titled 'Taxes'. It contains two fields: 'Tax when:' with a dropdown menu currently showing 'No Taxes', and 'Tax code:' with an empty text input field.

For more information on how to specify taxation in a rate plan, see Pricing Center Help.

Configuring Taxation in loadpricelist

To assign the tax calculation method for a specific rate plan using **loadpricelist**, create a price list XML file and set each rate plan's **tax_when** flag to one of the following:

- **NEVER**: Does not calculate and apply taxes.
- **NOW**: Calculates and applies taxes when an event occurs.
- **DEFER**: Defers tax calculation for an account until the end of its billing cycle.
- **DYNAMIC**: Defers tax calculation until the end of a billing cycle, but uses the tax rate from the time the event occurred. For example, if an account purchases a product on June 15 but its billing cycle ends on June 30, BRM would perform the tax calculation on June 30 but use the tax rate from June 15.

 **Note:**

The **DYNAMIC** flag is supported by BRM 12.0 Patch Set 5 and later releases.

For example, the following specifies to apply dynamic taxation to a rate plan:

```
<rate_plan tax_when="DYNAMIC">
...
</rate_plan>
```

After you create the price list XML file, you load it into the BRM database by using the **loadpricelist** utility. For more information, see "Using the XML Pricing Interface to Create a Price List" in *BRM Configuring Pipeline Rating and Discounting*.

Configuring Tax Calculation in ECE

ECE supports a fixed tax rate (a flat-rate taxation also known as GST or VAT). In ECE, you can apply a tax on both charges and alterations (discounts).

To configure taxation in ECE:

1. Access the ECE configuration MBeans:
 - a. Log on to the driver machine.
 - b. Start the ECE charging servers (if they are not started).
 - c. Connect to the ECE charging server node enabled for JMX management.

This is the charging server node set to **start CohMgt = true** in the *ECE_home/occeserver/config/eceTopology.conf* file.
 - d. Start a JMX editor that enables you to edit MBean attributes, such as JConsole.
 - e. In the editor's MBean hierarchy, find the ECE configuration MBeans.
2. Expand the **ECE Configuration** node.
3. Expand **charging.taxation**.
4. Expand **Operations**.
5. Click **addTaxDetails**.
6. Specify values for the following attributes:

 **Note:**

These attributes are mandatory. You must set all of them when configuring taxation.

- **taxCode:** Enter the tax code used by the charge offer or discount offer to which the tax applies.

The tax code is used by charge offers and discount offers to point to the tax rate that must be applied when a usage request is processed for the charge offer or discount offer.

Enter the same tax code entered in PDC or Pricing Center when the taxation section of the charge offer and discount offer was defined.
 - **taxRate:** Enter the tax rate to apply.

For example, entering **0.20** applies a 20% tax on the total usage impact.
 - **taxGllid:** Enter the General Ledger ID used for the tax impact.
7. Specify an additional **taxCode**, **taxRate**, and **taxGllid** value for each charge offer or discount offer to which a tax applies.
 8. Save your changes.

6

Configuring Vertex Manager

You can configure Vertex Manager to calculate taxes in your Oracle Communications Billing and Revenue Management (BRM) system.

Topics in this document:

- [About Implementing Vertex Tax Calculation](#)
- [Installing the Vertex Software](#)
- [Configuring the Sales Tax Q Series Shared Libraries](#)
- [Configuring the Communications Tax Q Series Shared Libraries](#)
- [Configuring the Vertex DM for Sales Tax Q Series](#)
- [Configuring the Vertex DM for Communications Tax Q Series](#)
- [Configuring the CM for Vertex Tax Calculation](#)
- [Verifying That Taxes Are Being Calculated](#)

About Implementing Vertex Tax Calculation

Vertex Manager connects BRM with the Vertex tax calculation database, which maintains current tax rates for a large variety of tax jurisdictions. By using Vertex, you can simplify how taxes are calculated and use the latest tax rates.

To calculate taxes by using Vertex, you install BRM Vertex Manager and then run the Vertex Data Manager.



Note:

BRM audits tax calculation data, but does not support the Vertex auditing features.

Installing the Vertex Software

You install BRM Vertex Manager when you install BRM.

Install the Vertex Sales Tax Q Series and Communications Tax Q Series software on the same machine as the Vertex Data Manager (DM). The Vertex software must be installed, configured, and tested as indicated in the Vertex documentation before any attempt is made to configure the Vertex DM.

See the Vertex documentation for installation instructions. You can install either package or both.



Note:

The Sales Tax Q Series and Communications Tax Q Series database can reside in the same tablespace as the BRM database, but it is recommended that you create a new tablespace for it.

Write down the values you entered for the following when you installed the Vertex Sales Tax Q Series.

- Datasource For Register Database
- Server Name For Register Database
- User Id For Register Database
- Password For Register Database

These values are used later when you configure the **pin.conf** entries.

Configuring the Sales Tax Q Series Shared Libraries

Make sure the following libraries are in the *BRM_home/lib* directory or in your **\$LD_LIBRARY_PATH**. These are shared libraries that come with the Sales Tax Q Series software.

Use files with extension **.so** for Linux and Solaris:

- **libvst**
- **libloc**
- **libqutil**

For more information about using Vertex toolkits, see the Vertex documentation.

Specifying Storage Manager Shared Library for Sales Tax Q Series

To configure the Vertex DM for Sales Tax Q Series, edit the *BRM_home/sysl* **dm_vertex/pin.conf** entries in [Table 6-1](#).

Table 6-1 pin.conf Entries for Vertex DM (Sales Tax Q Series)

Entry	Description
quantumdb_source	The schema where the STQ tables reside. Note: This parameter is required if the default tablespace for the user name and password parameters is not the same as the one that contains the STQ tables. For Indexed Sequential Access Method (ISAM) databases, this parameter specifies the ISAM data file directory.
quantumdb_server	The network identifier for the database on the server. For ISAM databases, this parameter should be commented out.
quantumdb_user	The user as a valid Oracle login name. For ISAM databases, this parameter should be commented out.

Table 6-1 (Cont.) pin.conf Entries for Vertex DM (Sales Tax Q Series)

Entry	Description
quantumdb_passwd	The user password.
quantum_sm_obj	The Storage Manager shared library that the DM uses to interact with a particular version of the Vertex STQ system. To use <ul style="list-style-type: none"> STQ 5.0, specify dm_vertex_stq500

To specify the appropriate Storage Manager shared library, set the value for **quantum_sm_obj** entry in the **pin.conf** file for Vertex DM to the particular version of Vertex STQ that you intend to use:

1. Open the Vertex DM configuration file (*BRM_home/sys/dm_vertex/pin.conf*).
2. Locate the **quantum_sm_obj** entry displayed in the following format in the file:

```
- dm_vertex quantum_sm_obj ./dm_vertex_stqVersion${LIBRARYEXTENSION}
```

where *Version* represents the STQ version.

For example, the following entry specifies that the Vertex DM needs to use STQ 4.0.6.

```
- dm_vertex quantum_sm_obj ./dm_vertex_stq406${LIBRARYEXTENSION}
```

3. Save and close the file.
4. Stop and restart the Vertex DM.

Configuring the Communications Tax Q Series Shared Libraries

Make sure the following libraries are in the *BRM_home/lib* directory or in your **\$LD_LIBRARY_PATH**. These are shared libraries that come with the Vertex Communications Tax Q Series software. Use files with extension **.so** for Linux and Solaris:

- **libadm**
- **libcch**
- **libcfg**
- **libcli**
- **libctq**
- **libctz**
- **libdbcper**
- **libgeo**
- **libhsh**
- **libobj**
- **libreg**
- **librpt**
- **librte**
- **libutl**
- **libxmlparse**

Specifying Storage Manager Shared Library for Communications Tax Q Series

Table 6-2 lists the **pin.conf** entries in the *BRM_home/sys/dm_vertex/pin.conf* file used to configure the Vertex DM for Communications Tax Q Series.

Table 6-2 pin.conf Entries for Vertex DM (Communications Tax Q Series)

Entry	Description
commtax_config_path	The location of Communications Tax Q Series configuration file (ctqcfg.xml). The default location is <i>CTQ_home/vertex/cfg</i> , where <i>CTQ_home</i> is the directory where you installed the Vertex software.
commtax_config_name	The Communications Tax Q Series configuration name. This name must match the configuration defined in the ctqcfg.xml file that is used with BRM. See the Vertex documentation for more information about defining configurations.
commtax_sm_obj	The Storage Manager shared library that the DM uses to interact with a particular version of the Vertex CTQ system. To use <ul style="list-style-type: none"> CTQ 2.02.16, specify dm_vertex_ctq20216 CTQ 3.00.02, specify dm_vertex_ctq300

To specify the appropriate Storage Manager shared library, set the value for **commtax_sm_obj** entry in the **pin.conf** file for Vertex DM to the particular version of Vertex CTQ that you intend to use:

1. Open the Vertex DM configuration file (*BRM_home/sys/dm_vertex/pin.conf*).
2. Locate the **commtax_sm_obj** entry displayed in the following format in the file:

```
dm_vertex commtax_sm_obj ./dm_vertex_ctqVersion${LIBRARYEXTENSION}
```

where *Version* represents the CTQ version.

For example, the following entry specifies that the Vertex DM needs to use CTQ 2.00.05.

```
dm_vertex commtax_sm_obj ./dm_vertex_ctq20005${LIBRARYEXTENSION}
```

3. Save and close the file.
4. Stop and restart the Vertex DM.

Configuring the Vertex DM for Sales Tax Q Series

To configure the Vertex DM for Sales Tax Q Series, you specify the required **pin.conf** entries in the *BRM_home/sys/dm_vertex/pin.conf* file:

1. Open the Vertex DM configuration file (*BRM_home/sys/dm_vertex/pin.conf*).
2. Change the following entries according to the instructions in the file:

- `dm_vertex quantumdb_source tablespace_for_Sales_Tax_Q_Series_tables`
- `dm_vertex quantumdb_server database_name`
- `dm_vertex quantumdb_user database_user`
- `dm_vertex quantumdb_passwd database_user_password`

where:

- The values for **quantumdb_source**, **quantumdb_server**, **quantumdb_user**, and **quantumdb_passwd** are the same values used when you installed the Sales Tax Q Series software. See "[Installing the Vertex Software](#)".
- **quantumdb_source** is required if the default tablespace for the user ID and password parameters is not the same tablespace that contains the Sales Tax Q Series tables.

 **Note:**

If the Vertex Sales Tax Q Series installation uses an ISAM database:

- **quantumdb_source** should point to the ISAM data file directory (as opposed to the Oracle data source).
- **quantumdb_server**, **quantumdb_user**, and **quantumdb_passwd** should be commented out.
- **dm_n_fe** and **dm_n_be** should be set to **1** to prevent multi-threading. The ISAM version of Vertex does not support multi-threading.

3. Save and close the file.
4. Stop and restart the Vertex DM.

Configuring the Vertex DM for Communications Tax Q Series

You set the required **pin.conf** entries in the `BRM_home/sys/dm_vertex/pin.conf` file to configure the Vertex DM for Communications Tax Q Series:

To configure the Vertex DM for Communications Tax Q Series:

1. Open the Vertex DM configuration file (`BRM_home/sys/dm_vertex/pin.conf`).
2. Specify the path to the Communications Tax Q Series configuration file in the **commtax_config_path** entry.

The following entry specifies the default path:

```
- dm_vertex commtax_config_path CTQ_home/vertex/cfg
```

3. Specify the Communications Tax Q Series configuration name in the **commtax_config_name** entry.

For example:

```
- dm_vertex commtax_config_name CTQ_Test
```

4. Save and close the file.
5. Stop and restart the Vertex DM.

Configuring the CM for Vertex Tax Calculation

Configure CM for Vertex calculation by completing these tasks:

- [Specifying Vertex DM Connection Entries in CM Configuration File](#)
- [Specifying whether to Validate ZIP Codes](#)
- [Itemizing or Summarizing Taxes for Each Jurisdiction Level](#)

Specifying Vertex DM Connection Entries in CM Configuration File

When you install BRM, you specify connection entries. You must change these entries if you change the Vertex database number, the host name, or the port number of the Vertex DM.

To specify the Vertex DM connection entry in the CM configuration file:

1. Open the CM configuration file (*BRM_home/sys/cm/pin.conf*).
2. Edit the **dm_pointer** entry.

```
- cm dm_pointer database ip hostname port
```

where:

- *database* identifies the Data Manager such as 0.0.8.1.
- *hostname* is the IP address or host name of the computer on which the DM is installed.
- *port* is the port number of the DM service.

Note:

The database number, host name, and port number must match the values in the Vertex DM configuration file (*BRM_home/sys/dm_vertex/pin.conf*).

3. Edit the **vertex_db** entry:

```
- fm_rate vertex_db database /_tax_db 0
```

where *database* is the database number specified in the Vertex DM configuration file.

4. Save and close the file.

You don't need to restart the CM to enable these entries.

Specifying whether to Validate ZIP Codes

If there is an error in an account's ZIP code, Sales Tax Q Series returns a tax amount of 0. If you set up Sales Tax Q Series to validate ZIP codes at account creation, you ensure that taxes are calculated correctly. This option specifies that Sales Tax Q Series checks the city, state and ZIP code when an account is created. If the ZIP code is not valid, account creation cannot be completed.

 **Note:**

- Communications Tax Q Series does not validate addresses and always returns a valid result.
- If you enable this option, the Vertex DM must be running when customers create accounts. If the connection to Vertex is offline, you can change this option to skip validating ZIP codes. This allows customers to create accounts.

To specify whether to validate ZIP codes:

1. Open the CM configuration file (*BRM_home/system/pin.conf*).
2. Change the **tax_valid** entry:
 - To enable ZIP code validation, enter **3**.
 - To disable ZIP code validation, enter **0**. This is the default setting.
3. Save and close the file.

You don't need to restart the CM to enable this entry.

Itemizing or Summarizing Taxes for Each Jurisdiction Level

You can opt to show the details of the tax types returned by Communications Tax Q Series for each tax jurisdiction level. These details can be mapped to a general ledger (G/L) ID and be shown on the invoice.

Some of the Vertex CTQ 2.00.05 and later jurisdiction codes are different from the internal BRM jurisdiction codes to which they are mapped. [Table 6-3](#) lists the Vertex-to-BRM mappings whose codes differ.

Table 6-3 Numbering Differences in Vertex-to-BRM Jurisdiction Code Mapping

Jurisdiction	Vertex Code	BRM Code
Other Municipality	6	12
County District	7	10
City District	9	11

To itemize the taxes for each jurisdiction:

1. Open the CM configuration file (*BRM_home/system/pin.conf*).
2. Add the following entry:


```
- fm_rate tax_return_juris itemize
```

To summarize the taxes, change the entry to **summarize** instead of **itemize**.

3. Save and close the file.
4. Stop and restart the CM.

Verifying That Taxes Are Being Calculated

After you have installed and configured the Vertex software, you can run the following test to verify that the Vertex software is calculating taxes.

1. Create a text file named **T502_40** with the following contents:

```

0 PIN_FLD_POID                POID [0] 0.0.0.1 /account 1 1
0 PIN_FLD_END_T               TSTAMP [0] (938224941) Fri Sep 24 19:02:21
1999
0 PIN_FLD_ACCOUNT_NO         STR [0] "ROOT"
0 PIN_FLD_CURRENCY           INT [0] 840
0 PIN_FLD_CURRENCY_NAME     STR [0] "USD"
0 PIN_FLD_TAXES              ARRAY [0] allocated 20, used 10
1   PIN_FLD_TAX_CODE         STR [0] "installVertex"
1   PIN_FLD_AMOUNT_TAXED    DECIMAL [0] 20
1   PIN_FLD_GL_ID           INT [0] 0
1   PIN_FLD_SHIP_TO         STR [0] "Cupertino;CA;95014;US;
[408572,2,1]"
1   PIN_FLD_SHIP_FROM       STR [0] "Denver; CO; 80205;US;
[303279,2,1]"
1   PIN_FLD_ORDER_ORIGIN    STR [0] ""
1   PIN_FLD_ORDER_ACCEPT    STR [0] ""
1   PIN_FLD_INTERNATIONAL_IND INT [0] 1
1   PIN_FLD_LOCATION_MODE   ENUM [0] 2
1   PIN_FLD_ELAPSED_TIME    TSTAMP [0] (249)
0 PIN_FLD_VAT_CERT           STR [0] ""
0 PIN_FLD_INCORPORATED_FLAG  ENUM [0] 0
0 PIN_FLD_RESIDENCE_FLAG     ENUM [0] 0
0 PIN_FLD_REGULATED_FLAG     ENUM [0]

```

2. Save the text file to the *BRM_home/setup/scripts* directory.
3. Start **dm_vertex**.
4. Open the *BRM_home/sys/cm.pinlog* file and verify that there are no errors.
5. Run **testnap** in the *BRM_home/setup/scripts* directory to verify that the taxes are being calculated:

```

testnap
r T502_40 1
xop 502 0 1

```

Note:

502 is the opcode reference number for PIN_FLD_RATE_TAX_CALC.

7

Tax Calculation Utilities

Learn about the tax calculation utilities that are included with Oracle Communications Billing and Revenue Management (BRM).

Topics in this document:

- [load_pin_ar_taxes](#)
- [load_tax_supplier](#)

To learn about tax calculation, see "[About Calculating Taxes](#)".

load_pin_ar_taxes

Use this utility to load configurable tax information into the **/config/ar_taxes** object in the BRM database. See "[Calculating Taxes for Accounts Receivable Actions](#)".

Location

BRM_home/bin

Syntax

```
load_pin_ar_taxes [-v] [-h] [-f input_file.xml]
```

Parameters

-v

Displays information about successful or failed processing as the utility runs.

-h

Displays the utility's syntax and parameters.

-f *input_file.xml*

Specifies the name and location of the file that defines the configurable tax information. For example: **C:\taxconf\adjust_taxes.xml**. If you do not specify the **-f** parameter and a path, **load_pin_ar_taxes** looks for a file named **pin_config_ar_taxes.xml** in the directory from which you started the utility.

load_tax_supplier

Use the **load_tax_supplier** utility to load one or more tax suppliers into the BRM database. See "[Creating Tax Suppliers](#)".

Location

BRM_home/bin

Syntax

```
load_tax_supplier [-d | -v | -t | -h] pin_tax_supplier.xml
```

Parameters

-d

Creates a log file for debugging purposes. Use this parameter for debugging when the utility appears to have run with no errors, but the data has not been loaded into the database.

-v

Displays information about successful or failed processing as the utility runs. To redirect the output to a log file, use the following command. Replace *filename.log* with the name of the log file:

```
load_tax_supplier pin_tax_supplier.xml -v > filename.log
```



Note:

If a file with the same name exists, it is overwritten.

-t

Checks the validity of the XML file, but doesn't create an object.

-h

Displays the syntax and parameters for this utility.

pin_tax_supplier

The name and location of the file that defines the tax suppliers. The default **pin_tax_supplier.xml** file is in *BRM_home/setup/scripts*.

If you do not run the utility from the directory in which the file is located, you must include the complete path to the file, for example:

```
load_pin_tax_supplier BRM_home/setup/scripts/pin_tax_supplier.xml
```

Results

If the **load_tax_supplier** utility doesn't notify you that it was successful, look in the **default.pinlog** file to find any errors. This file is either in the directory from which the utility was started or in a directory specified in the utility configuration file.