ONESOURCE INDIRECT TAX INTEGRATION FOR ORACLE 12

INSTALLATION AND CONFIGURATION GUIDE

VERSION 5.6.2.4

ORACLE
Validated Integration
Oracle E-Business Suite

August 2017

DOCUMENT HISTORY

Version Number	Version Date	Summary
V1	*October 1, 2024*	Updated for 5.6.2.4 release.
V2	*November 8, 2016*	Updated for 5.6.2.3 release.
V3	*August 15, 2017*	Updated for two new Profile options added by ONESOURCE Indirect Tax Integration for Oracle 12 Version 5.6.2.0 patch release. 1) eBTax: Sabrix AP Exclude VAT from Sabrix Tax Process 2) eBTax: Sabrix Compress Tax Lines
V4	*May 8, 2020*	Updated for 5.6.2.0 release.

TABLE OF CONTENTS

ABOUT THIS GUIDE

This guide tells you how to install the ONESOURCE Indirect Tax Integration for Oracle 12 to use with ONESOURCE Indirect Tax Determination. When the Integration is installed and configured, users of Oracle Purchasing, Oracle Payables, Oracle Order Management, and Oracle Receivables can receive fast and accurate sales, use, consumers use, and value-added tax (VAT) results.

This guide is intended for use by ONESOURCE Indirect Tax Professional Services, Oracle Applications Administrators, superusers, database administrators, and other IT professionals.

HOW THIS GUIDE IS ORGANIZED

This guide contains the following chapters:

- ***INTEGRATION OVERVIEW (page 3)*** shows the benefits of integrating ONESOURCE Indirect Tax Determination with Oracle Order Management, Oracle Receivables, Oracle Purchasing and Oracle Payables and provides an architectural and configurationoverview of the integration process.
- ***INSTALLATION (page 21)***, explains how to install the Integration.
- ***POST INSTALLATION: ORACLE CONFIGURATION (page 75)*** describes the changes you need to make in Oracle Applications after installing the Integration.
- ***POST INSTALLATION: TECHNICAL TOPICS (page 109)*** describes several IT tasks you must complete in order for ONESOURCE Indirect Tax Determination to integrate with Oracle release12.
- ***APPENDIX A: CONCURRENT PROCESSES (page 135)*** explains several concurrent programs created by the installation.
- ***APPENDIX B: NETWORK PROTOCOL OPTIONS (page 153)*** describes interface methods to ONESOURCE Indirect Tax Determination.
- ***APPENDIX C: ENTERING PARAMETERS FOR INSTALLATION SCRIPTS (page 157)*** describes the installation prompts for those who do not want to set their parameters in environment files.
- ***APPENDIX D: CONFIGURING HTTPS (OPTIONAL STEP) (page 161)*** explains how to set up this security protocol.
- ***APPENDIX E: ORACLE SOURCE MAPPED TO DETERMINATION XML DATA (page 167)*** shows the mapping between fields from Oracle Purchasing, Oracle Payables, and Oracle Receivables and ONESOURCE Indirect Tax Determination Input and Output XML.
- ***APPENDIX F: NON-STANDARD ORACLE APIS (page 227)*** lists various seeded database tables that receive insertions, updates, or triggers due to non-standard Oracle APIs.
- ***GLOSSARY (page 229)*** provides a selected list of definitions.

STYLE CONVENTIONS

This section describes the styles used in this document.

Bold text indicates commands, most User Interface elements (pages, windows, menu items, buttons, and so on), and values that can be selected from the User Interface.

Italic text indicates user input, file and directory names, user names, and web sites.

`Courier` text indicates command-line or text file input.

Indicates suggestions or additional, detailed information.

Indicates important text that should be carefully reviewed before proceeding.

INTEGRATION OVERVIEW

This chapter provides an overview of the Thomson Reuters Oracle E-Business Tax configuration, describes the benefits of tax calculation using the ONESOURCE Indirect Tax Determination, and gives you an installation checklist:

BENEFITS OF CALCULATING TAX WITH ONESOURCE INDIRECT TAX DETERMINATION

There are several benefits of using Determination to calculate tax for your Oracle Order Management, Oracle Receivables, Oracle Payables and Oracle Purchasing:

SEAMLESS INTEGRATION WITH ORACLE ORDER MANAGEMENT, ORACLE RECEIVABLES, ORACLE PURCHASING AND ORACLE PAYABLES

After integration, you can continue to use Oracle Order Management, Oracle Receivables, Oracle Purchasing and Oracle Payables as usual. Determination automatically obtains data elements necessary to perform tax calculations, and the results are automatically returned to Oracle.

ACCURATE TAX CALCULATION FOR GLOBAL TRANSACTION TAX MANAGEMENT

ONESOURCE Indirect Tax Determination:

- Integrates worldwide tax calculation.
- Enables global visibility and real time transactions.
- Gives you control of data and decision-making.
- Eliminates or minimizes IT involvement when tax jurisdictions, rates, and rules change.
- Provides a scalable, maintainable enterprise solution.

AUDITING AND REPORTING

Tax calculations processed using Determination are stored in audit tables. From this audit data, you can generate standard reports such as audit extract or US Tax Authority or you can run custom reports using other criteria.

See ONESOURCE Indirect Tax Determination *Online Help* for more information.

ACCRUALS

After you implement the Integration in Oracle Payables, you can accrue tax overpayments or underpayments to reduce penalties. Data is stored in the audit tables, and you can periodically run reports to view accruals for specific time periods.

ORACLE VALIDATED INTEGRATION

The ONESOURCE Indirect Tax Integration for Oracle 12 is an Oracle Validated Integration. Oracle Validated Integration is an Oracle Partner Network initiative where Oracle confirms that 3rd party software solutions adhere to integration standards and best practices, has been tested as functionally and technically sound, and operates and performs as documented. The scope of the Oracle Validated Integration program for 3rd party tax providers currently covers only the United States test scenarios.

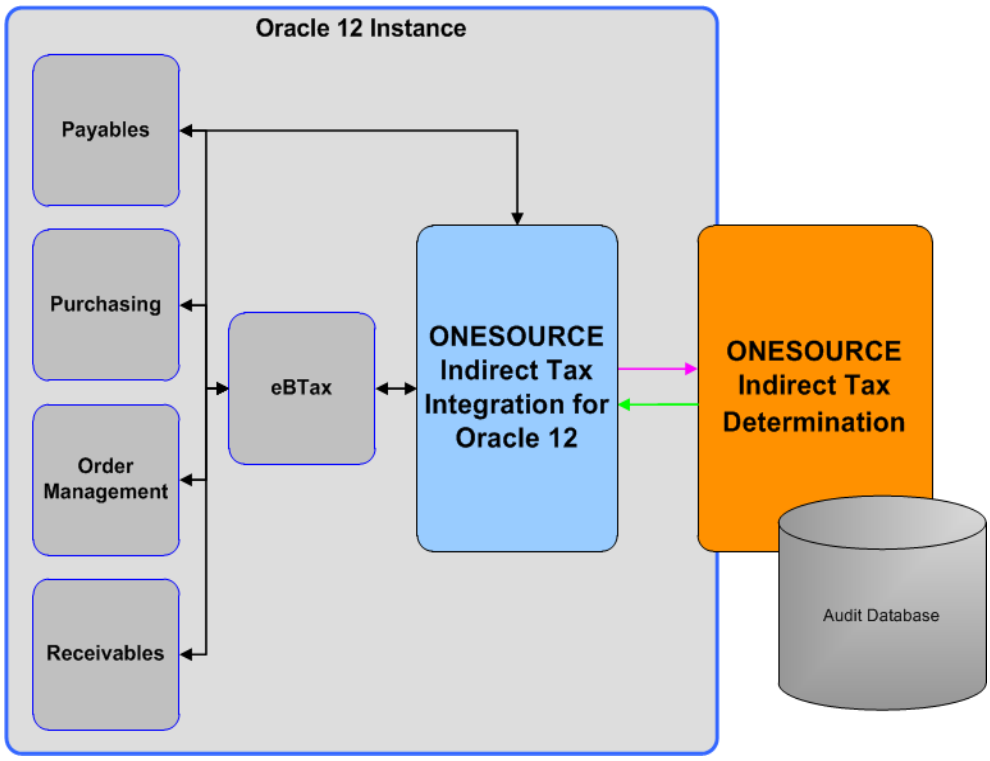
ARCHITECTURE

This section shows diagrams of:

- An overview of the ONESOURCE Indirect Tax Determination and Oracle applications.
- Elements of the process flows.

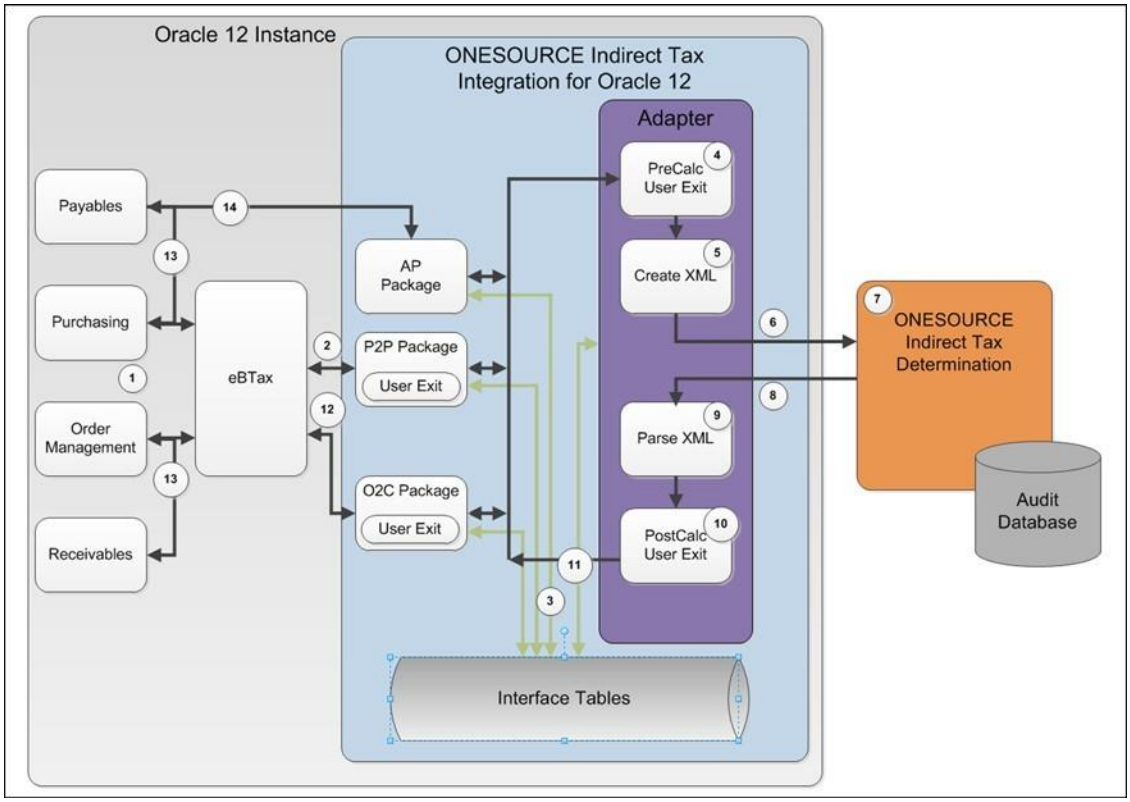
OVERVIEW OF ONESOURCE INDIRECT TAX DETERMINATION AND ORACLE APPLICATIONS

The following architecture diagram provides an overview how the ONESOURCE Indirect Tax Integration for Oracle 12 interacts with Oracle E-Business Suite Financials and the ONESOURCE Indirect Tax Determination.



PROCESS FLOWS

The following diagram shows the individual elements of the process flows. Each of the elements is described below:



Payables, Purchasing, Order Management, and Receivables

The numbers in the parentheses refer to the diagram above:

- (1) Business transactions relevant for tax call the Oracle E-Business Tax module using the Oracle-implemented hooks in the forms.
- (2) Oracle E-Business Tax routes the tax request based on the Regime setup to the ONESOURCE Indirect Tax Integration for Oracle 12.
- (3) The Integration manages data provided by the E-Business Tax module for Purchasing, Payables, Order Management, and Receivables transactions.
- (4) Using table objects to stage the tax data allows for processing of large invoices.
- (5) Through the use of a predefined user-exit for pre-processing data, customer-specific business rules and mappings can be implemented.
- (6) The Integration makes use of the Oracle XMLDB function to map the transaction data into XML format.
- (7) The Oracle HTTP POST method is called to send the XML data to ONESOURCE Indirect Tax Determination.

- (8) ONESOURCE Indirect Tax Determination calculates tax, and where applicable, writes the tax liabilities to the audit database.
- (9) Tax Results are returned via the HTTP POST method which writes the results into the Interface tables.
- (10) A post-processing user-exit allows customers to implement customer-specific rules.
- (11) The tax results are passed back to the Integration.
- (12) The Integration maps the data to Oracle E-Business Tax.
- (13) Oracle E-Business Tax returns the final tax results to the calling form, and where applicable, writes to the Oracle Financials database.
- (14) For U.S. only Payables transactions an additional process can be called to determine applicable tax accrual on an invoice based on configuration options and tolerance settings. The accrual amounts are then returned to Oracle AP and stored in the database tables.

REGIME TO RATE FLOW CONFIGURATION OVERVIEW

This section includes the following information:

- An overview of ONESOURCE Indirect Tax eBTax configuration
- Setting up tax flows with one-step configuration
- Creating Source Flows
- Integration behavior
- Disabling automatic tax flow creation
- Using default accounting when creating tax flows
- Updating existing tax rate codes GL accounts
- Creating Tax Recovery Rates
- Migrating existing tax flows
- Using reports to help guide the process

OVERVIEW OF ONESOURCE INDIRECT TAX EBTAX CONFIGURATION

The Oracle E-Business Tax Partner interface requires rigid configuration of taxes, jurisdictions, statuses, and rates. All accounting for partner tax providers is derived at the Tax Rate Code level. For each tax flow, a distinct tax, jurisdiction, status and rate must be configured. In this release, Thomson Reuters has expanded the eBTax footprint to include Tax to Rate Flows for each distinct authority.

The Oracle eBTax setup for Thomson Reuters includes one regime per country. Each regime is associated with one tax, one jurisdiction, one status, and one tax rate code for each authority. The Thomson Reuters supplied Tax to Rate flow includes all authorities per business workflow (Order to Cash and/or Procure to Pay). However, the list of authorities can be modified to fit your business needs prior to loading into Oracle. Unidentified authorities returned to Oracle during tax determination can automatically be created as new Tax Flows. This process will use the general ledger accounts from an existing tax flow defined for the ERP Tax Code or a Default Tax Flow defined for that operating unit. To ensure a unique Tax Flow for each tax line, each UUID (Unique Authority Identifier) is mapped to a unique Tax Flow.

There may be multiple instances of the same authority in the OUTPUT XML for the same transaction line (i.e. an authority with a tiered rate). Because a distinct Tax Flow for each Tax Line is required, multiple instances of the same authority in the Tax to Rate Flow are necessary. When a tiered rate is encountered, a sequence is assigned to the mapping (in the Sabrix Authorities Mapping table) and the Tax to Rate Flows are created and assigned accordingly.

Tax authorities files (supplied with content updates by Thomson Reuters) are used to match the Tax Flows to an authority. To ensure that your implementation has the latest tax authority information, you can load the monthly content updates into Oracle. Alternatively, you can use the automatic tax flow creation process (described below) and not load the tax authority file. Manage flow creation using the method which works best for your organization.

ONE-STEP CONFIGURATION

Use the Thomson Reuters One-Step Configuration to set up Tax to Rate Flows for Oracle eBTax. This process uses comma separated variable length files (or CSV) prepared from a template supplied by Thomson Reuters. From these files, the One-Step Configuration builds Regimes,

Taxes, Jurisdictions, Status, and Rates. It also subscribes Operating Units to the Regimes and assigns general ledger accounts to the Tax Rates.

To allow for periodic changes to authorities and ERP Tax Codes, you can download the latest template files of all the authorities from the ONESOURCE Indirect Tax Customer Center. Alternatively, you can use the automatic tax flow creation process (described below) and not load the tax authority file.

An optional step has been added to the One-Step Configuration to load the authorities. Two new files, *INTL_sabrix_stage_authorities.dat* and *US_sabrix_stage_authorities.dat*, have been added. The process loads the new files into the Sabrix schema and joins the resulting table to the SABRIX_STAGE_CONFIG table to build the Tax Flows. If these files are not loaded, the authority flows will automatically be created at transaction time.

A new persistent table, SABRIX_AUTHORITY_MAPPING, is used to look up the Tax Flow by AUTHORITY_UUID in the tax line returned by the Determination in the OUTDATA TAX block. One-Step Configuration merges the data from the SABRIX_STAGE_CONFIG table and the SABRIX_STAGE_AUTHORITY table to populate the SABRIX_AUTHORITY_MAPPING table.

An authority file can be constructed from your local ONESOURCE Determination database to process Custom Authorities with the One-Step Configuration. If custom ERP Tax Codes are used with custom authorities, then the custom ERP Tax Codes must be included in the *Sabrix_stage_config.dat* file when running the One-Step Configuration. For more information on the *Sabrix_stage_config.dat* file see ***Edit INTL_sabrix_stage_config.dat and US_sabrix_stage_config.dat (page 29)***.

Directional taxes, such as Value Added Taxes, create a pair of Tax to Rate Flows, one for each tax direction (input and output). These flows contain the direction as part of the name. This allows Tax to Rate Flows with different tax directions to have different general ledger accounts.

The naming convention for Tax to Rate Flows is ERP Tax Code, Business Workflow, Direction, and Sequence. The sequence is a database-generated incremental number that is independent of Regime, Business Workflow, and ERP Tax Code. An example Tax Code is 'USWA_O2C_1254' or 'CAGST_P2P_I_2398'. The description of the Tax Code is the authority name, such as 'USWA - SEATTLE, CITY SALES/USE TAX.'

SOURCE FLOWS

The description for each element in the Tax to Rate Flows defined during the One-Step Configuration is 'Source Flow for P2P Tax flow for XXXX', where XXXX is the ERP Tax Code. The integration uses the Source Flow for the ERP Tax Code to automatically define the authority level flows, described in ***INTEGRATION BEHAVIOR***.

Also, for Payables transactions, the *Sabrix Tax Process* sums the taxes per ERP Tax Code and uses the Source Flow to derive the general ledger accounts used on the expense and accrual entries. The Tax Classification field on the payables transaction is then populated with the Source Flow Tax Code.

INTEGRATION BEHAVIOR

Tax Flow mappings are determined by the UUID of the TAX Block.

If the UUID is not mapped to an existing Tax to Rate Flow and creation of new Tax Flows is not disabled, the ONESOURCE Integration will attempt to create a new Tax to Rate Flow using the accounting from a source Tax to Rate Flow as follows:

1. Look for a source Tax to Rate Flow with a match on ERP Tax Code, Operating Unit and Tax Direction. The Tax to Rate flow with the lowest mapping sequence will be used. This is normally the Source Flow for the ERP Tax Code set up during configuration.
2. Look for a source Tax to Rate Flow with a match on ERP Tax Code and Operating Unit. The Tax to Rate flow with the lowest mapping sequence will be used.
3. Look for a default (DFLT) Tax to Rate Flow with a match on Operating Unit and Tax Direction.
4. Look for a default (DFLT) Tax to Rate Flow with a match on Operating Unit.

If the ERP Tax Code group is undefined in Oracle and the default information cannot be determined, an Invalid Tax Code exception is generated.

DISABLING AUTOMATIC TAX FLOW CREATION

Customers who load authorities using the *INTL_sabrix_stage_authorities.dat* and *US_sabrix_stage_authorities.dat* files may not want to have automatic tax flows created including the Source tax flows. In order to allow customers the option of automatically creating Tax Flows or not automatically creating Tax Flows a profile option **eBTax: Sabrix Disable Tax Flow Creation** has been added. Setting the profile option to "*Yes"* will disable Automatic Tax Flow Creation from existing tax flow including Source tax flows associated to an ERP Tax Code. Transactions will error with no tax code found message. Setting the profile option to *"No"* will enable creating automatic tax flows.

In order to suppress Source Flow creation, the Profile Option eBTax: Sabrix Disable Tax Flow Creation must be set to 'Yes' prior to running the sabrix_insert_config.sh script.

Customers who have disabled the automatic tax flow creation can manually create new tax flows by running a concurrent program called *Sabrix Add Authority*. The error messaging for O2C and P2P transaction processing will display the necessary information to create new tax flows and authority mapping.

DEFAULT ACCOUNTING WHEN CREATING TAX FLOWS

The Integration can create a Tax to Rate Flow automatically with valid accounting. It does so by creating default accounting for each Regime and Operating Unit. Your tax personnel can be notified using a scheduled concurrent process of all new Tax Flows created with the default accounting (see the *New Tax Flows Report* below). The Default Accounting Flow is intended to specify a clearing account for an Operating Unit.

The Default Accounting Flow adheres to the naming convention of DFLT (in place of the ERP Tax Code), workflow, and unique sequence. The resulting Tax Flow is created as DFLT_WFx_ LX. For example, the Default Accounting Flow for O2C workflow of the Vision Operations Operating Unit would be 'DFLT_O2C_123456' and the description would be 'Source O2C Tax Flow for Vision Operations operating unit.'

When the Integration encounters an ERP Tax Code that is not configured into a Tax to Rate Flow, it looks for a Default Accounting Flow for the Operating Unit and Regime. If a Default Accounting Flow is configured, the Integration creates a new Tax Flow automatically and assigns the general

ledger account associated to the Default Accounting Flow to the new Authority Tax to Rate Flow. If a Default Accounting Flow is not configured, the transaction fails. Once the proper flow has been created, the transaction can be reprocessed.

If the Tax to Rate Flow is defined, but accounting is not configured for the operating unit on the transaction, the Integration will assign the new operating unit and define the general ledger accounts from the DFLT Tax to Rate Flow to the existing Tax to Rate Flow.

The One-Step Configuration or the supplied Tax Flow creation concurrent programs can be used to create a Default Accounting Flow.

UPDATES TO EXISTING TAX RATE CODES GL ACCOUNT

The Integration has a programmatic process to mass update the GL tax accounts on existing Tax Rate Codes. Prior to the Integration version 5.6.0.0 release, the *sabrix_stage_config.dat* file could only be used to create new tax rate codes. A new bash shell script *sabrix_update_config.sh* uses a new procedure *sabrix_ebtax.update_accounts _from_staging* to load data from the *sabrix_stage_config.dat* file into the *SABRIX_STAGE_CONFIG* table.

Any authority specific GL accounts are overwritten for the Tax to Rate flows with the same ERP Tax Codes in the sabrix_stage_config.dat file based on whether you are using Basic or Extended Tax Code functionality: Basic ERP Tax Code i.e. DEVAT, ITVAT, USWA or USTX or Extended ERP Tax Code i.e. DEVAT.CU.CU.I.

The behavior is identical to the concurrent program Sabrix Add Account to Tax Rate Code to update Tax Rate Codes GL accounts.

TAX RECOVERY RATES

Tax Recovery Rates are required for any non-U.S. country where recoverable taxes are applicable for P2P transactions. Starting with Integration version 5.6.0.0 release, P2P Tax Recovery Rates are also created using the One-Step Configuration or the supplied Tax Flow creation concurrent programs.

When creating the Tax Recovery Rates, the existing Tax Rate Code (e.g. GBVAT_P2P_I_21) will have the suffix '_REC_RATE" added, resulting in a Tax Recovery Rate such as GBVAT_P2P_I_21_REC_RATE.

For existing Tax to Rate Flows from Integration versions prior to 5.6.0.0, a concurrent program *Sabrix Setup Recovery Rates* has been added to update the appropriate flags to allow tax recovery, create the tax recovery rate and associate the tax recovery rate at the Tax level and to the standard percentage Tax Rate Code. The concurrent program is meant only to create recovery rates for any and all existing non-U.S. standard percentage Tax Rate Codes. For details on how to run the concurrent program and to verify the Tax Recovery Rates have been created, see ***SABRIX SETUP RECOVERY RATES (page 151)***.

MIGRATION OF EXISTING TAX FLOWS

A migration path for existing Tax to Rate Flows from Integration versions 5.2.0.0 or later has been added to the One-Step Configuration. This path recognizes existing Tax to Rate Flows and builds new flows from the authority file.

If upgrading from Integration version 5.3.0.0 to version 5.4.0.0, set the installation parameter USE_EXISTING_CONFIG to 'Y' instead of creating a *sabrix_stage_config.dat* file. This causes the *sabrix_insert_config.sh* script to build the *SABRIX_STAGE_CONFIG* table from your existing Tax to Rate Flows. The script maps the existing entries to the new Authority table and creates the new Tax to Rate Flows with the new naming convention. The existing Tax to Rate Flows remain unchanged but are not used by the Integration.

The installation parameter USE_EXISTING_CONFIG is used only during the upgrade from Integration version 5.3.0.0 to version 5.4.0.0. This parameter has subsequently been deprecated in newer Integration releases.

REPORTS

The *Sabrix New Tax Flows Report* lists all the Tax to Rate Flows created within a date range. The source of the default information to create the Tax to Rate Flow is provided in the report.

The *GL Tax Account Discrepancies For Sabrix Tax Rate Codes* report lists Tax Flows within an ERP Tax Code group that have different general ledger accounts. Thomson Reuters recommends that you use the same general ledger accounts within the ERP Tax Code group. An ERP Tax Code group relates to all the Tax Flows associated with the same ERP Tax Code. All the city and county authorities within a state have the same ERP Tax Code. For example, the ERP Tax Code for Washington is USWA. Directional taxes defined with each direction posting to a separate general ledger account (i.e. Input to one account and Output to a different account) are also displayed on this report.

TRIGGERING ONESOURCE INDIRECT TAX DETERMINATION TAX CALCULATIONS

After Determination is fully integrated with Oracle Purchasing, Oracle Payables, Oracle Order Management and Oracle Receivables, you can automatically generate correct tax results when performing the following actions:

- Creating or modifying a purchase requisition.
- Creating or modifying a purchase order.
- Creating or modifying a sales order.
- Importing an invoice using AutoInvoice.
- Creating batches for invoice processing.
- Using recurring invoices.
- Creating or updating an invoice.
- Creating a debit or credit memo.

PREREQUISITES

This guide assumes you have a successful installation of the ONESOURCE Indirect Tax Determination and an instance of Oracle Applications Release 12 to connect to using the Integration.

Please see the following document which details the Support Lifecycle of the Integration version on the ONESOURCE Indirect Tax Customer Center:

- *ONESOURCE Indirect Tax Product Support Lifecycle Guide*: https://www.onesourceidtsupport.com/cgi-bin/sabrix.cfg/php/enduser/std_adp.php?p_faqid=679

If you are not using software components supported by ONESOURCE Indirect Tax, you may encounter errors.

To confirm that you are using supported software components for the supported Integration versions, see the Integration Version, ERP Version, Patches Required and Supported Database tables below.

Integration for Oracle 12 Order to Cash (O2C) and Procure to Pay (P2P) 5.6.x

OLM/CRM, iStore, Service Contracts, iProcurement, and Project Billing not supported. Oracle Localizations are not supported.

Integration Version	ERP Version	Patches Required	Supported Databases
5.6.x Oracle	Financials 12.1.3	10095488 10296081 12360358 14122718 14273383 14277162 14466506 9411521 9787808 18041417 19256028 21112600 21696441	Oracle RDBMS 11g: 11.2.x Single and RAC databases are supported.
5.6.x Oracle	Financials 12.2.x	17305947 18010252 18041417 19256028 21270528 21696441	Oracle RDBMS 11g: 11.2.x Oracle RDBMS 12c: 12.1.x Oracle RDBMS 19c: 19.24.0.0.0 Single node, two node, RAC and Edition-Based Redefinition databases are supported.

Note:

1. Replace TLS 1.0 with version TLS 1.2 for Oracle Hosted customers. this is mandatory for connecting to our cloud/hosted systems from Oracle.
2. If patches are absolute, we can apply superseded packages.

Integration for Oracle 12 Order to Cash (O2C) and Procure to Pay (P2P) 5.5.0.2

NOTE OLM/CRM, iStore, Service Contracts, iProcurement, and Project Billing not supported. Oracle Localizations are not supported.

Integration Version	ERP Version	Patches Required	Supported Databases
5.5.0.2	Oracle Financials 12.1.3	10095488 10296081 12360358 13563481 14122718 14273383 14277162 14466506 9411521	Oracle RDBMS 11g: 11.2.x Single and RAC databases are supported.
5.5.0.2	Oracle Financials 12.2.x	17305947 18010252	Oracle RDBMS 11g: 11.2.x Oracle RDBMS 12c: 12.1.x Single, RAC, and Edition-Based Redefinition databases are supported.

NOTE Apply patches in the order listed.

Integration for Oracle 12 Order to Cash (O2C) and Procure to Pay (P2P) 5.5.0.1

NOTE OLM/CRM, iStore, Service Contracts, iProcurement, and Project Billing not supported. Oracle Localizations are not supported.

Integration Version	ERP Version	Patches Required	Supported Databases
5.5.0.1	Oracle Financials 12.1.3	10095488 10296081 12360358 13563481 14122718 14273383 14277162 14466506 9411521	Oracle RDBMS 11g: 11.2.x Single and RAC databases are supported.
5.5.0.1	Oracle Financials 12.2.2	17305947	Oracle RDBMS 11g: 11.2.x Oracle RDBMS 12c: 12.1.x Single, RAC, and Edition-Based Redefinition databases are supported.
5.5.0.1	Oracle Financials 12.2.3	18010252	Oracle RDBMS 11g: 11.2.x Oracle RDBMS 12c: 12.1.x Single, RAC, and Edition-Based Redefinition databases are supported.

NOTE Apply patches in the order listed.

INSTALLATION QUICK REFERENCE

This guide describes three installation types:

- Combined Order to Cash (O2C) and Procure to Pay (P2P)
- Order to Cash (O2C) Only
- Procure to Pay (P2P) Only

Use the following checklist as you proceed through the installation chapters.

Topic and Page number	R = Required		
	O2C+P2P	O2C	P2P
DOWNLOADING THE ONESOURCE INDIRECT TAX INTEGRATION FOR ORACLE 12 (page 22)	R	R	R
DETERMINING PARAMETER VALUES FOR YOUR SITE (page 23)	R	R	R
CREATING THE XXIDT SCHEMA (page 35)	R	R	R
CREATE THE API OWNER ID (page 37)	R	R	R
DEFINING PAYABLES DESCRIPTIVE FLEXFIELDS (page 39)	R	N/A	R
PROCESS FLOW FOR INSTALL STEPS (page 48)	R	R	R
CONFIGURING PARAMETERS IN ENVIRONMENT FILES (page 49)	R	R	R
EXECUTING THE SABRIX INTEGRATION INSTALLATION SCRIPT (page 54)	R	R	R
REGISTERING THE ONESOURCE INDIRECT TAX INTEGRATION FOR ORACLE 12 (page 56)	R	R	R
DEFINING THE SABRIX APPLICATION DATA GROUP (page 59)	R	R	R
SETTING UP REGIME TO RATE FLOW (page 60)	R	R	R
DESCRIPTIVE FLEXFIELDS: ASSOCIATING VALUE SETS (page 76)	R	R	R
CONFIGURING PROFILE OPTIONS (page 79)	R	R	R
SETTING TAX EVENTS ON ORDER MANAGEMENT TRANSACTION TYPES (page 101)	R	R	N/A
DEFINING SABRIX AP HOLDS (page 102)	R	N/A	R
BYPASS TCA GEOGRAPHY HIERARCHY VALIDATIONS (page 104)	OPTIONAL	OPTIONAL	OPTIONAL
STORING CUSTOM .SQL SCRIPTS (page 109)	R	R	R
VERIFYING THE INSTALLATION (page 111)	R	R	R
SYNCHRONIZING ZONE NAMES (page 107)	OPTIONAL		
CUSTOMIZING THE INTEGRATION (page 118)	OPTIONAL		

Topic and Page number	R = Required		
	O2C+P2P	O2C	P2P
PROGRAMMING USER EXITS (page 121)	OPTIONAL		
PERFORMANCE TUNING (page 130)	OPTIONAL	N/A	OPT

INSTALLATION

This chapter describes the installation of the ONESOURCE Indirect Tax Integration for Oracle 12 software in the system running Oracle Applications.

There are three types of new installations:

- Combined Order to Cash (O2C) and Procure to Pay (P2P)
- Order to Cash (O2C) Only
- Procure to Pay (P2P) Only

At the beginning of each section, look for a table that indicates whether that particular step is required for your type of installation. The letter “R” appears in the right-hand columns if it is required.

This chapter covers the following topics:

REQUIRED SKILLS AND ACCESS

Before proceeding with the installation, ensure that you and your colleagues have the following:

Skills and Access	Description
Technical	• Using SQL*Plus • UNIX System Administration • Using Oracle 12 Account Payables • Using Oracle 12 Purchasing • Using Oracle 12 Order Management • Using Oracle 12 Receivables • Configuring Profile Options • Defining Descriptive Flexfields
Oracle Responsibilities	• Application Developer • System Administrator • Payables • Purchasing • Order Management • Receivables • Tax Managers
DBA and System Administration Access	• Ability to modify and re-compile Oracle Applications • System Administrator for log file location access • Database Administrator for log file location access

DOWNLOADING THE ONESOURCE INDIRECT TAX INTEGRATION FOR ORACLE 12

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle DBA	Obtain the Customer Center Username and Password from Technical Support	R	R	R

Two files located on the ONESOURCE Indirect Tax Customer Center contain data used to setup Oracle E-Business Tax. Download both files:

- *Integration for Oracle 12* contains the installation files
- *Integration for Oracle 12 Setup File* contains the configuration files

1. Navigate to the ONESOURCE Indirect Tax Customer Center at (*https://customercenter.sabrix.com/sabrixcc*).
2. Log on using the **User Name** and **Password** provided by ONESOURCE Indirect Tax Technical Support.
3. In the **Available Products** list, verify that the Integration for Oracle 12 **Current Installed Version** is set.

4. Click **Download** for the *Integration for Oracle 12*.
5. Save the file *ONESOURCEIDTIntegrationOracle12_5.6.2.4.zip* and unzip it on aclient machine that has access to the Oracle Applications database from the OracleClient.
6. In the **Available Products** list, verify that the *Integration for Oracle 12 Setup File* **Current Installed Version** is set.
7. Click **Download** for the *Integration for Oracle 12 Setup File*.
8. Save the file *IntegrationOracle12SetupFile.zip* and unzip it on a client machine, in the same directory as in Step 5.

Your client machine must have a local BASH shell to execute the installation scripts. If you are using Microsoft Windows, you could use a command line interface such as CYGWIN.

DETERMINING PARAMETER VALUES FOR YOUR SITE

The scripts and procedures used in this guide reference the following information about your implementation. You should determine these values *before* running the installation:

- ***PRIMARY INSTALLATION SCRIPT PARAMETERS (this page)***
- ***REGIME TO RATE FLOW PARAMETERS (page 27)***

PRIMARY INSTALLATION SCRIPT PARAMETERS

The following table contains a list of parameters you will enter into the environment files *sabrix_install.env* and *sabrix_p2p.env*. You will use these values when you reach the section ***CONFIGURING PARAMETERS IN ENVIRONMENT FILES (page 49)***. Use the table below to record the values at your site.

Item	Description	R = Required			Your Value
		O2C + P2P	O2C	P2P	
Configure eBTax	Determines if the installation script will configure core eBTax.	R	R	R	
Retain Data in Sabrix schema staging tables	For Sabrix Integration upgrade installations only -- Determines if the installation script will retain existing data in Sabrix schema staging tables for sabrix_invoice, sabrix_invoice_out, sabrix_invoice_tax, sabrix_license, sabrix_license_out, sabrix_line, sabrix_line_tax, sabrix_line_out, sabrix_message, and sabrix_registration tables.	R	R	R	

Item	Description	R = Required			Your Value
		O2C + P2P	O2C	P2P	
Use Existing Configuration	Use existing tax flows to create new tax flows in an Integration upgrade scenario. Note: This installation parameter is used only during the upgrade from Integration version 5.3.x.x to version 5.4.x.x. This parameter has subsequently been deprecated in newer Integration releases.	R	R	R	
Packages Only	Install only packages and views (leaving schema objects, value sets, and profile options intact). Used for testing purposes. Default is *N*.	R	R	R	
Sabrix Owner Name	New schema name. The default is *XXIDT.*	R	R	R	
Sabrix Owner Password	New schema password. The default is *XXIDT.*	R	R	R	
APPS Schema Name	Oracle APPS Schema name if other than APPS.	R	R	R	
APPS Schema Password	Oracle APPS Schema password if other than APPS.	R	R	R	
Oracle Database Name	The name of the database running Oracle Applications. Example: *production or test*	R	R	R	
Sabrix Tablespace Name	The name of the tablespace for the new Sabrix Schema. Example: *XXIDT_DATA*	R	R	R	
Sabrix Index Tablespace Name	The name of the index for the new Sabrix Schema. Example: XXIDT_*IDX*	R	R	R	

Item	Description	R = Required			Your Value
		O2C + P2P	O2C	P2P	
Tax Status Attribute	For Payables implementations: The descriptive flexfield attribute used when creating the Tax Status flag segment in Payables invoice distributions. Example: *ATTRIBUTE1* See *Defining Payables Invoice Distribution Flexfield* for more information.	R		R	
Sabrix Tax Rate/Amount Attribute	For Payables implementations: The descriptive flexfield attribute used to store the tax rate and amount returned from Sabrix in Payables invoice distributions. Example: ATTRIBUTE3 See *Defining Payables Invoice Distribution Flexfield* for more information.	R		R	
Sabrix Line Link	For Payables implementations: The descriptive flexfield segment used to link the Sabrix tax line to the item line. Example: *ATTRIBUTE4* See Defining Payables Invoice Distribution Flexfield for more information.	R		R	
Sabrix Application ShortName	Sabrix Application ShortName used when registering Sabrix as an application in Oracle. Example: *XXIDT*	R	R	R	

Item	Description	R = Required			Your Value
		O2C + P2P	O2C	P2P	
Sabrix Application Top	The path to the Sabrix application files when registering Sabrix as an application in Oracle. Example: *XXIDT_TOP*	R	R	R	
Effective Start Date for EB Tax Transactions	The effective start of eBTax Transactions used when installing/configuring Sabrix Tax Regime within various Oracle Tax tables. Date format is (DD-MON-YYYY)	R	R	R	

REGIME TO RATE FLOW PARAMETERS

The Oracle eBTax setup for Thomson Reuters includes one regime per country. Each regime is associated with Tax to Rate flows containing one tax, one jurisdiction, one status, and one tax rate code for each authority. The general ledger accounts (O2C Recoverable/Liability Account, P2P Recoverable/Liability Account, and P2P Expense Account) are associated with the ERP Tax Code and defined at the eBTax Tax Rate Code level. The operating units are then subscribed to Sabrix regime(s).

To set up the Regime to Rate flow, you must gather various data about your transactions. The information gathered will be inserted into one data file. The data file is loaded into your installation through a shell script described later in this guide. The section below describes the type of data to gather and how to insert the information into the data file:

- Regimes
- Tax Codes for Tax Flows
- Operating Units
- Gather Values for Regimes, Tax Codes for Tax Flows, and Operating Units

Creating the data file is an optional step. The Regimes, Tax to Rate flows, Tax Rate Codes and Subscribe to Service provider can be entered individually in concurrent processes described later in this guide. Thomson Reuters recommends that you create the data file if you are implementing multiple Regimes and multiple Tax Rate Codes. The data file also must be completed if the general ledger accounts vary by Tax Rate Code.

Regimes

The first type of data you need to gather is the regimes in which your company transacts business. As defined in eBTax, a regime is equivalent to a country. The countries will translate to regimes during the installation.

Although you should set up the regimes for which each operating unit transacts business, future regimes can be added and subscribed to using the concurrent programs. See details in the ***CREATING REGIME TO RATE FLOW USING SABRIX CONCURRENT PROCESSES (page 136)***. You could also set up all regimes for all operating units to limit your risk of missing regime(s).

Tax Codes for Tax Flows

The second type of data you need to gather for the regime to rate flow setup is the list of ERP Tax Codes to be used in your implementation. The ERP Tax Code is a field on the authorities in the ONESOURCE Indirect Tax Determination engine. The files within the Integration for *Oracle 12 Setup File.zip* file contain a listing of the Determination supported ERP Tax Codes. These files can be modified to fit your implementation.

For Custom Authorities defined in your local ONESOURCE Determination database, you can also include the custom ERP Tax Codes used with custom authorities, similar to the data provided in the *Oracle 12 Setup File.zip* file.

Operating Units

The last type of data you need to prepare for the regime to rate flow is a list of ERP Tax Codes each operating unit uses, along with the associated General Ledger account numbers (O2C Recoverable/Liability Account, the P2P Recoverable/Liability Account, and the P2P Expense Account).

There are three steps for preparing Regime to Rate Flow information:

- ***Gather Regime, Tax Code and Operating Unit Values (page 28)***
- ***Edit INTL_sabrix_stage_config.dat and US_sabrix_stage_config.dat (page 29)***
- ***Rename the INTL_sabrix_stage_authorities.dat and US_sabrix_stage_authorities.dat (page 34)***

Gather Regime, Tax Code and Operating Unit Values

Collect information about your regimes so that you can enter it into each of the data files below:

- **Regimes**: Identify the countries in which your implementation will do business. Compile a list of every country where each operating unit transacts business (for example, the customer Ship To countries in an O2C implementation).
- **Country Code**: Review the associated country code for the respective ERP Tax Code. Country Code is based on ISO two character codes.
- **Currency**: Currency for each country identified as a Regime.
- **Exchange Rate Type**: The exchange rate type for your implementation. It is the CONVERSION_TYPE field from the GL_DAILY_CONVERSION_TYPES table. The file is set to *Corporate* for all rows. The exchange rate type can be updated for your implementation. The exchange rate type defined on the regime will default to defined taxes.
- **ERP Tax Code**: Identify the tax codes for your implementation based on the countries/regimes created above in the regime setups. This code can be found on the **Authorities Edit** page for each authority in the Determination. Alternatively all authorities can be loaded.
- **Tax Direction**: Specifies input/output values for VAT recovery identification. The Tax Direction would be null for all United States ERP Tax Codes and other non-directional taxes.
  - Specify the tax direction if the general ledger (GL) accounting is different for Input and Output (I or O). If the tax direction is left null in the .dat file, then both (I or O) tax directions will post to the same general ledger (GL) accounts. If entering two flows in the data file, one for Input and one for Output, the setup process creates two tax rate codes. If appropriate use different general ledger (GL) account for both (I or O) tax directions, see **P2P Recoverable/Liability Account** and **Tax Expense Account** bullets below. Operating Unit ID: Determine the operating unit id authorized to use the Sabrix Regimes and Tax Rate Flows.
- **O2C Recoverable/Liability Account**: A general ledger (GL) account that records tax recoverable amounts or relieves tax liability amounts.
- **P2P Recoverable/Liability Account**: A general ledger (GL) account that records tax recoverable amounts or relieves tax liability amounts.
  - For U.S. P2P tax rate codes enter a Sales and Use Tax Liability GL account.
  - For non-U.S. P2P Tax to Rate flows for VAT or other non-U.S. ERP tax rate codes with Input (I) Tax Direction enter a Recoverable Tax GL account. The same Recoverable Tax GL account will also be on the associated Input (I) Tax Recovery Rate.

- For non-U.S. P2P Tax to Rate flows for VAT or other non-U.S. ERP tax rate codes with Output (O) Tax Direction (usually used for VAT Reverse Charge/Self Assess) enter a Liability Tax GL account. The same Liability Tax GL account will also be on the associated Output (O) Tax Recovery Rate.
- For further details on VAT P2P processing, see the *ONESOURCE Indirect Tax Integration for Procure to Pay User's Guide, Value Added Tax (VAT) Transactions section.*

- **Tax Expense Account**: A general ledger (GL) account for Payables tax account that records tax amounts from invoice distributions; or a Receivables tax account that records taxes collected from customers.
  - For non-U.S. P2P Tax to Rate flows with Output (O) Tax Direction (usually used for VAT Reverse Charge/Self Assess) you may want to enter the same Liability Tax GL account as used on the P2P Tax to Rate flows with Output (O) Tax Direction for the Tax Expense Account. This will ensure recording the Output (O) Tax Direction taxes to the appropriate GL account.
  - If the **Tax Expense Account** is left null on the Tax Rate Code, the default Oracle functionality is to derive the GL Tax Expense account from the item distribution lines.
  - For further details on VAT P2P processing, see the *ONESOURCE Indirect Tax Integration for Procure to Pay User's Guide, Value Added Tax (VAT) Transactions section.*

The general ledger accounts **must** be valid code combinations in Oracle General Ledger prior to loading the spreadsheet into Oracle eBTax. The loading process will not automatically create new account combinations.

Edit INTL_sabrix_stage_config.dat and US_sabrix_stage_config.dat

One-Step Configuration uses the filename *sabrix_stage_config.dat* file to create the Regimes to Rate Flows. The two supplied config.dat files must be renamed to *sabrix_stage_config.dat* after editing. The supplied files can be merged into one file for installation, or the files can be processed separately by the One-Step Configuration.

For Custom Authorities defined in your local ONESOURCE Determination database, you can also include the custom ERP Tax Codes associated to the custom authorities, similar to the data provided in the *Oracle 12 Setup File.zip* file.

The configuration files can be described as the ERP Tax Code file. The information in this file is used to construct regimes and subscribe operating units to regimes. You can edit the sample configuration files for your regime values by doing the following:

1. The Integration for Oracle 12 Setup File.*zip* file (downloaded from the ONESOURCE Indirect Tax Customer Center at *https://customercenter.sabrix.com/sabrixcc*) contains four files:
   - *INTL_sabrix_stage_authorities.dat* - contains a list of all international authorities from ONESOURCE Indirect Tax Determination. (No edits required). It is **optional** to use this file.
   - *INTL_sabrix_stage_config.dat* - sample file to be used to record your regime to rate flow information for international authorities.

- *US_sabrix_stage_authorities.dat* - contains a list of all United States authorities from ONESOURCE Indirect Tax Determination. (No edits required). It is **optional** to use this file.
- *US_sabrix_stage_config.dat* - sample file to be used to record your regime to rateflow information for United States authorities.

If you are using the ONESOURCE Indirect Tax extended tax code functionality, the expected extended tax rate code values can be entered into the *US_sabrix_stage_config.dat* file and/or *INTL_sabrix_stage_config.dat* file for easy configuration in Oracle. See the *Sabrix Integration for Oracle 12 Procure to Pay or Order to Cash User's Guide,* Setting eBTax: Sabrix Tax Accounting Level for Extended Tax Code Processing section for more information on extended tax code functionality.

If the *INTL_sabrix_stage_authorities.dat* and the *US_sabrix_stage_authorities.dat* files are used (it is an optional step), they merely need to be renamed, they do not need to be edited. The supplied files can be merged into one file for intallation, or the files can be processed separately by the One-Step Configuration.For more information, see ***Rename the INTL_sabrix_stage_authorities.dat and US_sabrix_stage_authorities.dat (page 34)***.

If using either the *US_sabrix_stage_config.dat and the INTL_sabrix_stage_config.dat* file to create *sabrix_stage_config.dat* and want to suppress automatic Source Flow creation, the Profile Option eBTax: Sabrix Disable Tax Flow Creation must be set to 'Yes' prior to running the sabrix_insert_config.sh script..

2. Edit the *US_sabrix_stage_config.dat* and the *INTL_sabrix_stage_config.dat* files with a spreadsheet program such as Microsoft Excel and save as a *.csv* file. Each row in the spreadsheet represents a unique combination of Country Code, Country Name, Currency Code, Exchange Rate Type, ERP Tax Code, Tax Direction, Operating Unit ID, and general ledger account numbers for Tax Recoverable/Liability and TaxExpense.
    - **Country Code**: The file is pre-populated with a country code for each ERP Tax Code the Determination supports.
    - **Country Name (for Regimes**): The file is pre-populated with all of the country codes the Determination supports. All Regimes can be loaded or the file can be edited by removing the regimes you do not require for your implementation. If an edited file is loaded in the initial installation, additional regimes can be added in the future. See the section ***CREATING REGIME TO RATE FLOW USING SABRIX CONCURRENT PROCESSES (page 136)*** for more information.

Each operating unit and regime combination needs to be identified separately.

    - **Currency Code**: The file is pre-populated with a currency for each country that Determination supports. Review the currency values for your implementation.
    - **Exchange Rate Type**: The file has a column for exchange rate type, set to *Corporate* for all rows. It is the CONVERSION_TYPE field from the GL_DAILY_CONVERSION_TYPES table. The exchange rate type can be updated for your implementation. The exchange rate type defined on the regime will default to defined taxes.

NOTE The initial value for Exchange Rate Type is *Corporate* in *US_sabrix_stage_config.dat* and *INTL_sabrix_stage_config.date.* If you are not using the *Corporate* Exchange Rate Type, enter the value for your implementation.

- **ERP Tax Code**: The file contains an initial long list of ERP Tax Codes and their related country codes that Determination supports. The file can be edited by deleting unwanted tax codes or adding new rows (in the same format) for custom authorities or Determination extended tax codes needed for your implementation.

NOTE The ERP Tax Codes must relate to the regimes defined above in the Country Name for Regimes.

- **Tax Direction**: Specifies input/output values for VAT recovery identification. Direction would be null for all United States ERP Tax Codes and other non-directional taxes. Specify the tax direction if the general ledger (GL) accounting is different for Input and Output (I or O). If the tax direction is left null in the .dat file, then both (I or O) tax directions will post to the same general ledger (GL) accounts. If entering two flows in the data file, one for Input and one for Output, the setup process creates two tax rate codes. If appropriate use different general ledger (GL) account for both (I or O) tax directions, see **P2P Recoverable/Liability Account** and **Tax Expense Account** bullets below. The Tax
- **Operating Unit ID**: Determines the operating unit id authorized to use the Sabrix Regimes and Tax Rate Flows.
- **Tax Recovery Liability O2C**: A general ledger (GL) account that records tax recoverable amounts or relieves tax liability amounts.
- **Tax Recovery Liability P2P**: A general ledger (GL)account that records tax recoverable amounts or relieves tax liability amounts.
- For U.S. P2P tax rate codes enter a Sales and Use Tax Liability GL account.
- For non-U.S. P2P Tax to Rate flows for VAT or other non-U.S. ERP tax rate codes with Input (I) Tax Direction enter a Recoverable Tax GL account. The same Recoverable Tax GL account will also be on the associated Input (I) Tax Recovery Rate.
- For non-U.S. P2P Tax to Rate flows for VAT or other non-U.S. ERP tax rate codes with Output (O) Tax Direction (usually used for VAT Reverse charge/Self Assess) enter a Liability Tax GL account. The same Liability Tax GL account will also be on the associated Output (O) Tax Recovery Rate.
- For further details on VAT P2P processing, see the *ONESOURCE Indirect Tax Integration for Procure to Pay User's Guide, Value Added Tax (VAT) Transactions* section.
- **Tax Expense Account**: A general ledger (GL) account for Payables tax account that records tax amounts from invoice distributions; or a Receivables tax account that records taxes collected from customers.
- For non-U.S. P2P Tax to Rate flows with Output (O) Tax Direction (usually used for VAT Reverse charge/ Self Assess) you may want to enter the same Liability Tax GL account as used on the P2P Tax to Rate flows with Output (O) Tax Direction for the Tax Expense Account. This will ensure recording the Output (O) Tax Direction taxes to the appropriate GL account.
- If the **Tax Expense Account** is left null on the Tax Rate Code, the default Oracle functionality is to derive the GL Tax Expense account from the item distribution lines.
- For further details on VAT P2P processing, see the *ONESOURCE Indirect Tax Integration for Procure to Pay User's Guide, Value Added Tax (VAT) Transactions* section.

The general ledger accounts **must** be valid code combinations in Oracle General Ledger prior to loading the spreadsheet into Oracle eBTax. The loading process will not automatically create new account combinations.

3. To allow the integration to automatically create tax flows for unexpected ERP Tax Codes returned from Determination, add a row to the file for the ERP Tax Code of 'DFLT'. Specify default accounting by workflow and by operating unit. This allows for the flow to be created and the related transaction to complete normally.
4. Save the file with an extension of *.csv* in the same directory where you unzipped the Integration for Oracle 12 files.
5. Navigate to that directory and rename the file to *sabrix_stage_config.dat* so that it has an extension of *.dat*.

One-Step Configuration uses the filename *sabrix_stage_config.dat* to create the Regimes to Rate Flows. The supplied INTL and US files must be renamed to *sabrix_stage_config.dat* after editing. The two files can be merged into one file for the installation. Or the files can be be processed by the One-Step Configuration separately.

The following table contains the details of the US_*sabrix_stage_config* and the *INTL_sabrix_stage_config* data files:

DETAILS OF US_SABRIX_STAGE_CONFIG AND INTL_SABRIX_STAGE_CONFIG DATA FILES		
Data File Name	*US_sabrix_stage_config.dat* or *INTL_sabrix_stage_config.dat*	
Delimiter/Separator	Comma (Records with commas should be enclosed in double quotes)	
Format (All fields must be on same line)	"CountryCode","CountryName","CurrencyCode","ExchangeRateType", "ERPTaxCode","Tax Direction", "OperatingUnitID","TaxRecoveryLiabilityO2C","TaxRecoveryLiabilityP2P","TaxExpense"	
Data File Column Name	**Data Type, Size and Description**	**Example Data (Comma delimiter based on Format above after each data element)**
Country Code	Varchar2(2) - ISO Standard Country codes	CA
Country Name	Varchar2(150)	CANADA
Currency Code	Varchar2(3) - ISO Standard Currency Code	CAD
Exchange Rate Type	Varchar2(150)	Corporate
ERP Tax Code	Varchar2(50)	GAGST
Tax Direction	Varchar2(1)	O
Operating Unit ID	Up to 10 digit number	7272
TaxRecoveryLiabilityO2C	Varchar2(150)	01-000-2310-0000-000
TaxRecoveryLiabilityP2P	Varchar2(150)	01-000-2520-0000-000
Tax Expense (Optional)	Varchar2(150)	01-410-7710-0000-000 (Optional to enter Tax Expense GL Account)

The *sabrix_stage_config.dat* data file can have a header row listing the column names. During the installation of sabrix regime to rate flow (*sabrix_insert_config.sh* script) the header row in *sabrix_stage_config.dat* file will be ignored.

For an O2C-only installation, the P2P accounts may be omitted. For a P2P-only installation, a place holder is required for the O2C account. This can be accomplished by leaving the account value blank for O2C Recoverable/Liability Account (for example, leave two consecutive commas such as *CA,CANADA,CAD,Corporate, CAGST,O,7272,,01-000-2520-0000-000,01-410-7710-0000-000*).

When the Tax Expense GL account, used for AP transactions, is null, the Integration will derive the Tax Expense GL account from the AP invoice item lines GL account.

When the profile option 'eBTax: Sabrix AP Tax Expense Account Source' is set to 'Tax', the GL Expense Account is required on the Tax Rate Codes for the *Sabrix Tax Process* program in Payables.

Examples of Alternative Approaches for *sabrix_stage_config.dat*

Each row in the spreadsheet represents a unique combination of Country Code, Country Name, Currency Code, Exchange Rate Type, ERP Tax Code, Tax Direction, Operating Unit ID, and general ledger account numbers for Tax Recoverable/Liability and Tax Expense. The spreadsheet can be created three different ways (using the same format) to load the sabrix_stage_config table:

- **Create separate spreadsheets for each operating unit**: For example, one spreadsheet for all the tax codes, country codes, and general ledger account combinations for Operating Unit 1 and a separate spreadsheet for the combinations for Operating Unit 2. There is only one *.dat* file (named *sabrix_stage_config.dat)* used by the script. Therefore, the first spreadsheet for the first operating unit needs to be loaded into the staging table and into Oracle before loading the second spreadsheet into the staging table; otherwise, the second spreadsheet will overwrite the first spreadsheet when you save it as the same filename (*sabrix_stage_config.dat)*.
- **Create one spreadsheet with all the combinations**: For example, one spreadsheet is created with all the tax codes, country codes, and general ledger account combinations for all operating units. In the scenario where both Operating Unit 1 (org id *1235*) and Operating Unit 2 (org id *9987*) will be subscribing to the US regime and tax code USWA then there will be two rows in the spreadsheet, one for each operating unit ID as follows:

US,UNITED STATES, USD, Corporate, USWA,,1235, 01-000-2310-0000-000, 01-000-2520-0000-000,01-410-7710-0000-000

US,UNITED STATES, USD, Corporate, USWA,,9987, 01-000-2310-0000-000, 01-000-2520-0000-000,01-410-7710-0000-000

- **Create one spreadsheet with your Custom Authorities combinations**: For example, one spreadsheet is created with all the Custom tax codes, country codes, and general ledger account combinations for all operating units.

For Custom Authorities defined in your local ONESOURCE Determination database, you can also include the custom ERP Tax Codes associated to the custom authorities, similar to the data provided in the *Oracle 12 Setup File.zip* file.

See the table on the previous page for example data and formatting of *sabrix_stage_config.dat*.

Rename the INTL_sabrix_stage_authorities.dat and US_sabrix_stage_authorities.dat

It is optional to load the two supplied authorities.dat files. If these files are not loaded and a *DFLT* flow is defined, the authority flows will automatically be created at transaction time. See ***DEFAULT ACCOUNTING WHEN CREATING TAX FLOWS (page 10)*** for more information on the automatic creation of Tax to Rate Flows.

The One-Step Configuration uses the filename *sabrix_stage_authorities.dat* file to create the authority mappings. The two supplied authorities.dat files (*INTL_sabrix_stage_authorities.dat* and *US_sabrix_stage_authorities.dat*) must be renamed to *sabrix_stage_authorities.dat*.The supplied files can be merged into one file for the installation, or the files can be processed by the One- Step Configuration separately.

CREATING THE XXIDT SCHEMA

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle DBA	None			

The script *sabrix_create_schema.sql* creates Data and Index Tablespaces for XXIDT objects and creates the schema user that owns XXIDT objects. It also provides default values but allows you to override these values.

Do not execute the script *sabrix_create_schema.sql* script if you are using Oracle Real Application Clusters (RAC). If this is a RAC environment, the DBA must manually create the Sabrix Schema.

Before running the script, use the following table to note any values specific to your implementation:

Prompt	Default Value	Your Value
Log File Name	*Sabrix_create_schema_ddmmyyhhmm.log*	
Sabrix Adapter Schema Owner Name	*XXIDT*	
Sabrix Adapter Owner Password	*XXIDT*	
Data Tablespace Name	*XXIDT_DATA*	
Data File Size	*200M*	
Data File Name	(No default) Example*: d:\oracle\visdata\xxidt_data.dbf*	
Index Tablespace Name	*XXIDT_IDX*	
Index File Size	*50M*	
Index File Name	(No default) Example*: d:\oracle\visdata\xxidt_idx.dbf*	
Temp Tablespace Name	*TEMP*	

1. Log into SQL*Plus as the *SYSTEM* user.
2. Run the following command to create the Sabrix schema.

```
SQL> @sabrix_create_schema.sql
```

The following is an example of the prompts and defaults in this script:

```
SQL>  @sabrix_create_schema.sql

Enter log file name [sabrix_create_schema1108151135]:
+-------------------------------------------------------------------------------+
+   Install script : [sabrix_create_schema.sql]
+   Run date       : [11-AUG-2015 11:35:55]
+
+ Please enter the following information to create
+ 1. Data and Index Tablespaces for Sabrix objects
+    If tablespaces already exist, this script will not re-create them.
+ 2. Schema user that owns Sabrix objects
+    If schema already exists, this script will not re-create it.
+
+ Important!!! .. Please log in as SYSTEM/SYS/User with full DBA privileges.
+
+-------------------------------------------------------------------------------+
+ Parameters:
+-------------------------------------------------------------------------------+
Sabrix adapter schema owner    [XXIDT] :
Sabrix adapter schema password [XXIDT] :
Data Tablespace Name           [XXIDT_DATA] :
Data File Size                 [200M] :
Data File Name                 [] :
Index Tablespace Name          [XXIDT_IDX] :
Index File Size                [50M] :
Index File Name                [] :
Temp Tablespace Name           [TEMP] :
```

CREATE THE API OWNER ID

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle Superuser	Confirmation that Oracle Legal Entity Configurator completed successfully (Refer to *R12 Oracle Financials Implementation Guide*).	R	R	R

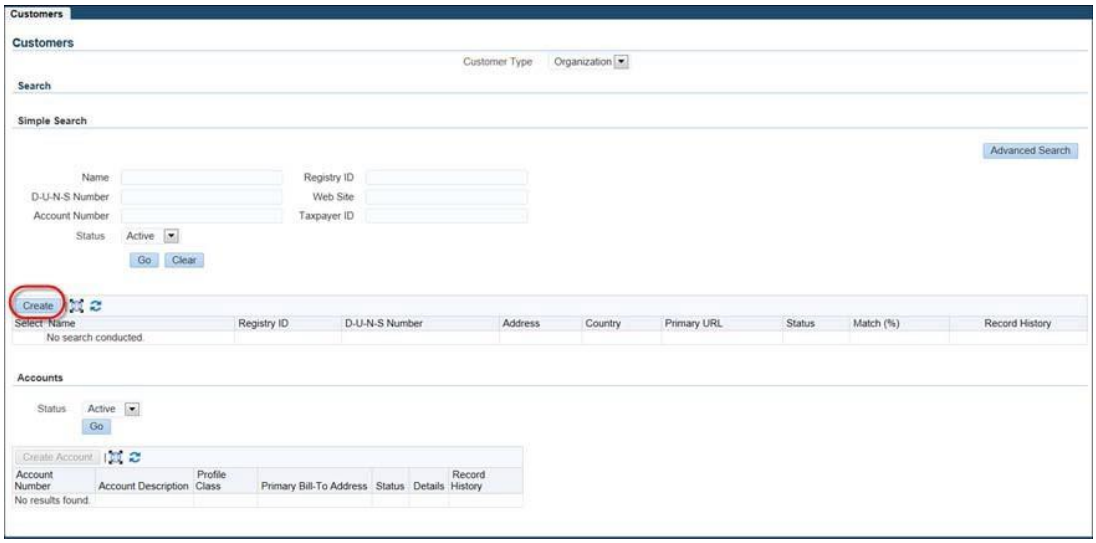
The API Owner ID defines which Tax Partner tax interface to use. This ID for the Integration is created by entering *Sabrix, Inc.* as a party in the Oracle Trading Community Architecture. Once the party is created, a **party_tax_profile_id** is created and this ID is equivalent to the API Owner ID. The Install Integration script will automatically use the **party_tax_profile_id** created for *Sabrix, Inc.* when configuring eBTax for the Integration.

Use a responsibility with customer setup (Receivables, Order Management, Trading Community Manager, etc.). Navigate to the party or customer setup. The following example outlines creating a customer (which automatically creates a party) in Oracle Receivables.

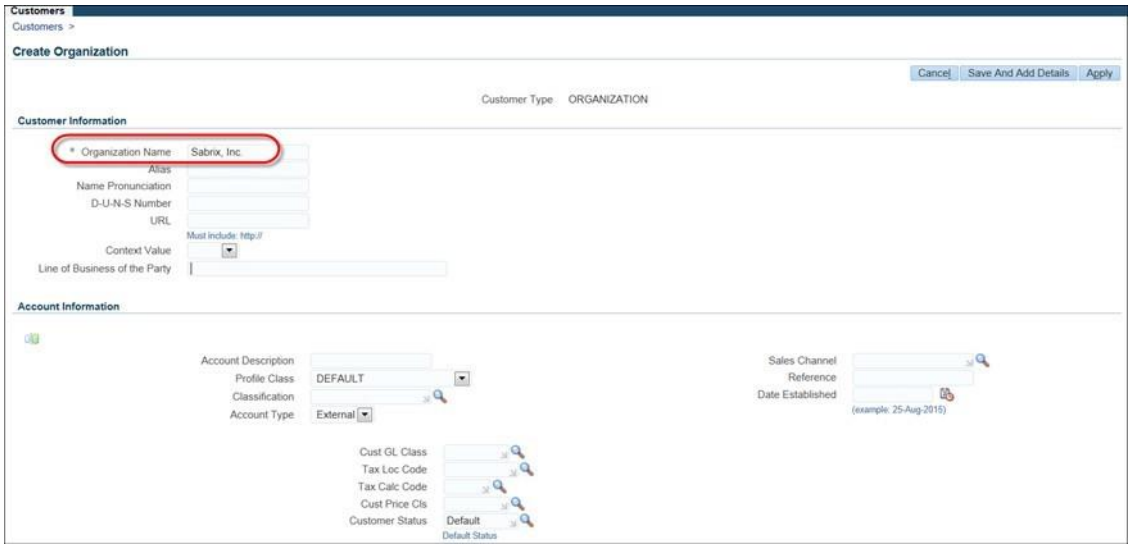
1. Select the **Receivables Manager** Responsibility.
2. Navigate to **Customers > Create/Maintain Customers**.
3. Search for Name of *Sabrix%.* If it does not exist then go the next step.

If upgrading from a prior Oracle version like 11i or prior ONESOURCE Integration version, search to confirm if Sabrix, Inc. already exists as a tax partner. If Sabrix, Inc. is already defined, then from Step 4 Create to Step 8 is not needed.

4. Click **Create**.

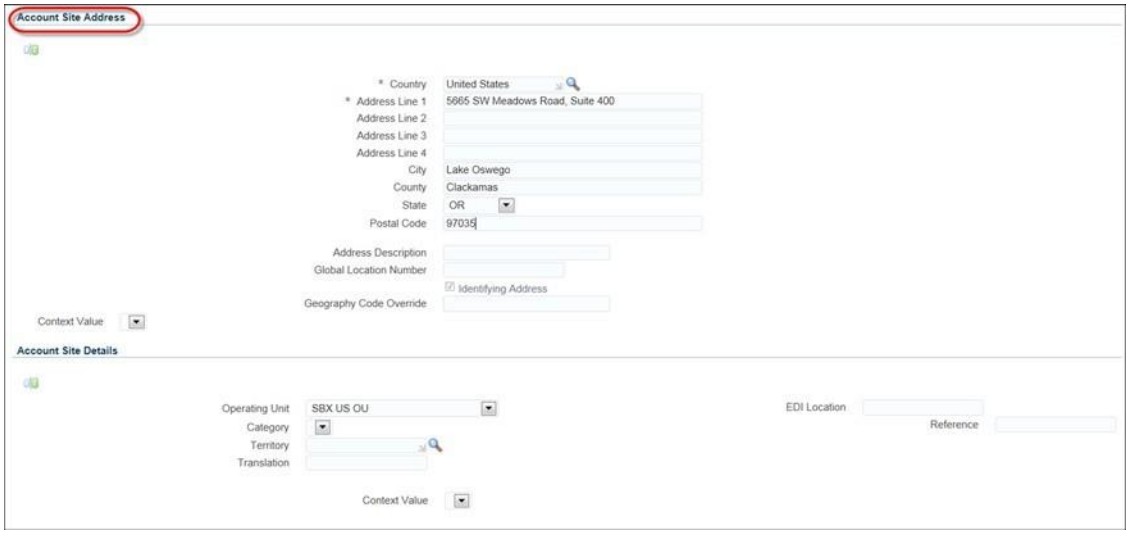


5. Enter *Sabrix, Inc.* as the **Organization Name**. No other values are required for the Customer Information section.



Be sure to enter the exact value of *Sabrix, Inc.* when you complete this step. If you use another value, the installation will fail.

6. Enter the **Address** according to your business rules and Oracle TCA setup.



NOTE The operating unit assignment is based on the operating units your responsibility has access to. For *Sabrix, Inc.* party creation, an address is not needed, but Oracle requires an address in order to save the customer record (not required on a party creation).

7. Click **Apply** on the **Create Organization** form.
8. Click **Apply** on the **Customer** form.
9. A confirmation message appears, indicating that the *Sabrix, Inc.* customer is now created and the changes have been saved.

DEFINING PAYABLES DESCRIPTIVE FLEXFIELDS

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle Superuser	None			

You must complete this step before you execute the script *sabrix_install_integration.sh* because it relies on the attribute numbers.

The following sections tell you how to define your descriptive flexfields.

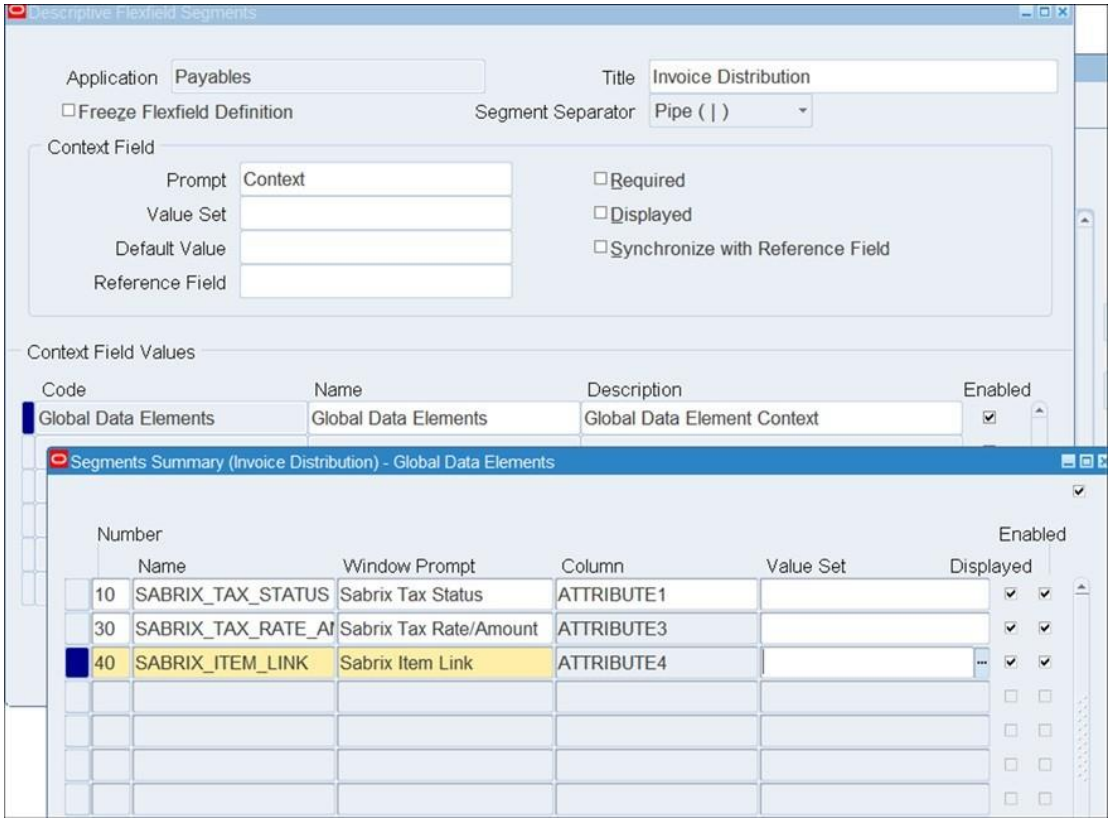
- ***DEFINING PAYABLES INVOICE DISTRIBUTION FLEXFIELD (this page)***
- ***DEFINING DESCRIPTIVE FLEXFIELD ATTRIBUTE TO ALLOW PROFILESETTINGS BY COUNTRY (page 43)***
- ***DEFINING DESCRIPTIVE FLEXFIELD ATTRIBUTE TO ALLOW PROFILE SETTINGS BY STATE (page 45)***

DEFINING PAYABLES INVOICE DISTRIBUTION FLEXFIELD

To define descriptive flexfields required for the Integration:

1. Select the **Payables Manager** responsibility (or another that gives you the ability to define descriptive flexfields).
2. Navigate to **Setup > Flexfields > Descriptive > Segments**.
3. Query **Payables** in the **Application** field and **Invoice Distribution** in the **Title** field.
4. Uncheck **Freeze Flexfield Definition**.
5. In the **Context Field** region, clear the **Required** and **Displayed** check boxes.

6. Click the **Segments** in the **Global Data Elements** region and create the segments shown in the table below:



Segment Name	Window Prompt	Column	Required	Display Size	Description Size	Concatenated Description Size
SABRIX_TAX_STATUS	Sabrix Tax Status	Attribute X	N DFFs are not operating unit specific. • Defined DFFs are viewable in all orgs, even if ONESOURCE Inidrect Tax Integration is not installed. • The decision to make theSabrixTax Status flagrequired is company specific.	2	50	25
SABRIX_TAX_RATE_AMOUNT	Sabrix Tax Rate/Amount	Attribute X	N	150	50	80
SABRIX_ITEM_LINK	Sabrix Item Link	Attribute X	N	10	50	25

The attribute number is specific to your installation. You will need this attribute number as part of the sabrix_install_integration.sh script. Also, the installation script creates the value sets; and you associate these with the descriptive flexfields, along with freeze and compile flexfield definitions when you complete the steps in *POST INSTALLATION: ORACLE CONFIGURATION (page 75)*.

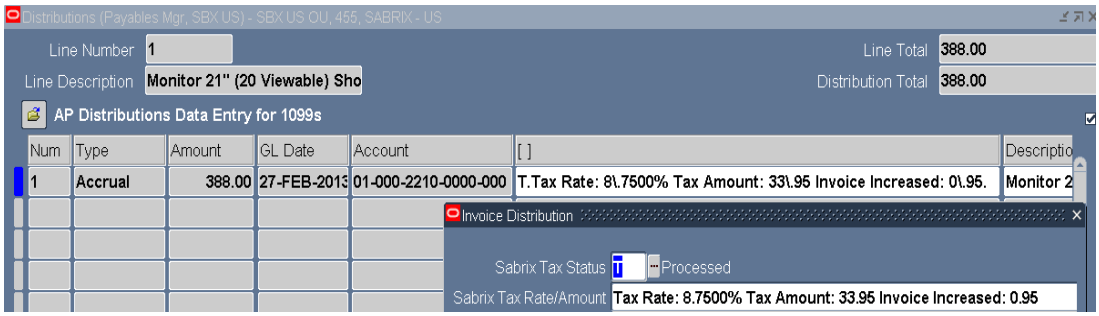
7. **Required** and **Display Size** are set when opening the segment.

Click **Open** for each segment and set the required *Y* or *N* flags; otherwise, all attributes default to *Required (Y)* during the initial setup.

8. Save your work.

NOTE

Before you define the descriptive flexfields, note that the descriptive flexfield Segment Separator defaults by Oracle to a period symbol (.). This causes the **Invoice Distributions** form to add a backslash symbol (\ ) before the decimal in the number: for example, 9.50% becomes 9\.50% (see the following screen shot). The separator can be changed to another symbol such as a pipe (|) so the number is properly displayed in the **Invoice Distributions** form. Set this separator value according to the needs of your implementation.

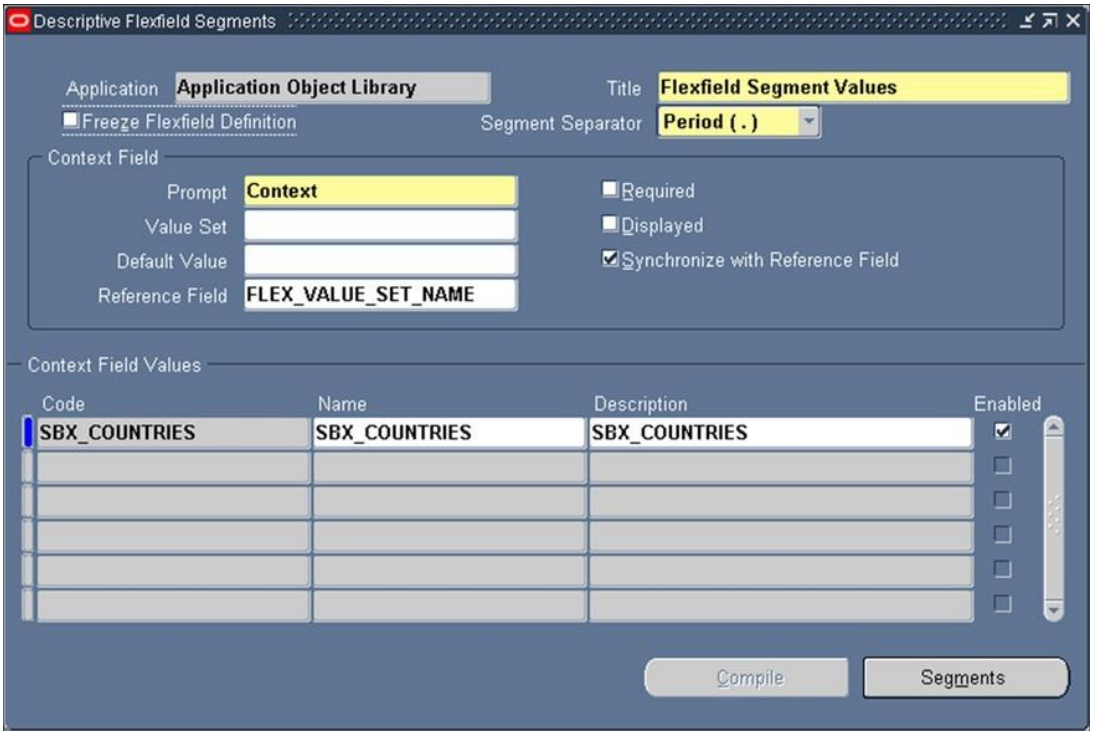


DEFINING DESCRIPTIVE FLEXFIELD ATTRIBUTE TO ALLOW PROFILE SETTINGS BY COUNTRY

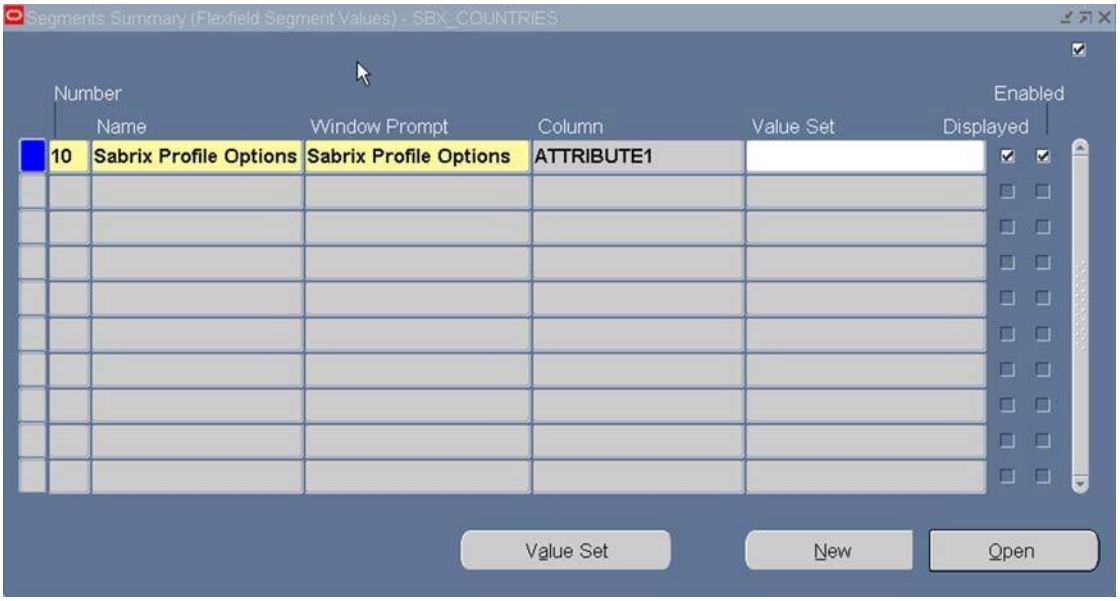
Depending on your business processes, you may require different values in profile options for each country in which you do business. This optional configuration allows the Sabrix profile options to be set at the Country level as attributes on a value set named SBX_COUNTRIES.

To define the optional descriptive flexfield for country level profile options:

1. Select the **Payables Manager** responsibility (or another that gives you the ability to define descriptive flexfields).
2. Navigate to **Setup > Flexfield > Descriptive > Segments.**
3. Query **Application Object Library** in the **Application** field and **Flexfield Segment Values** in the **Title** field.
4. Confirm the checkbox **Freeze Flexfield Definition** is unchecked.
5. Verify that **Context** is in the **Context Field** prompt.
6. Enter *FLEX_VALUE_SET_NAME* in the **Reference Field** in the context field section.
7. In the **Context Field** section, clear the **Required** and **Displayed** check boxes.
8. Check the **Synchronize with Reference Field** box.
9. In the **Context Field Values** section, add a new record**.**
10. Enter *SBX_COUNTRIES* in both the **Code** and **Description** fields.



11. Click **Segments** for *SBX_COUNTRIES* and enter the following, using the table below as a guide:



Segment Name	Window Prompt	Column	Required	Display Size
Sabrix Profile Options	Sabrix Profile Options	Attribute *X*	N	50

NOTE The attribute number is specific to your installation.

12. Save your work.

NOTE **Required** and **Display Size** column values are set when opening the segment.

Click **Open** for each segment to set the required *Y* or *N* flags; otherwise, all attributes default to Required (*Y*) during the initial setup.

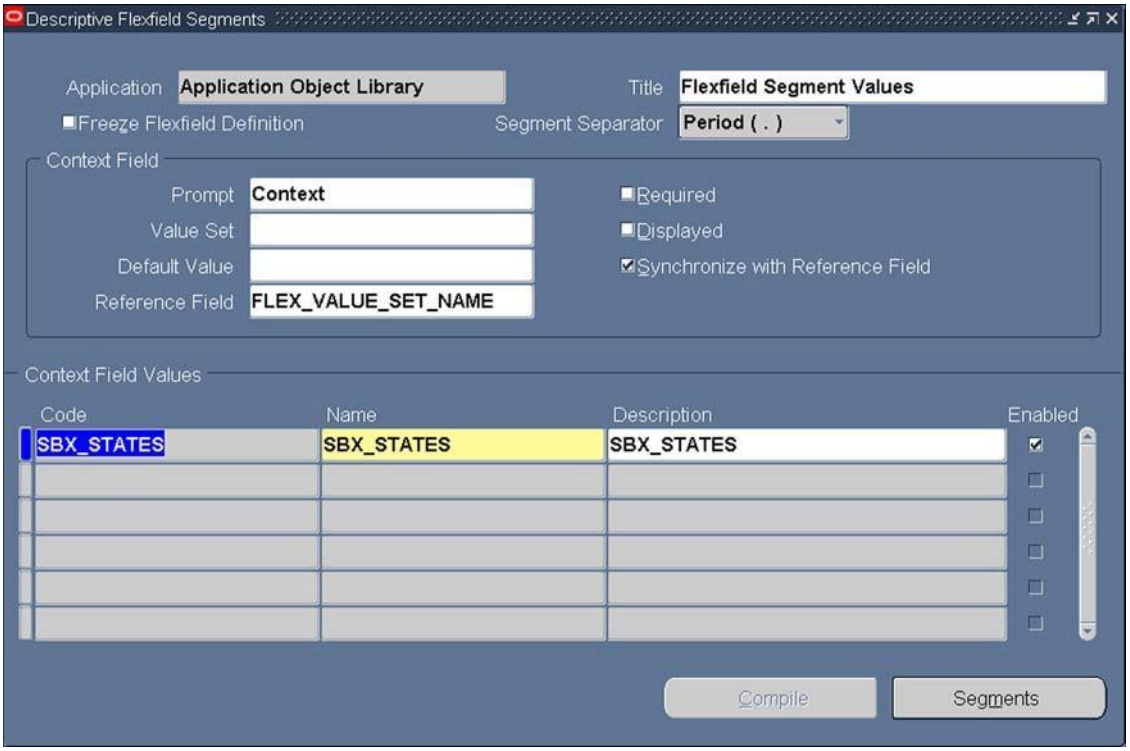
DEFINING DESCRIPTIVE FLEXFIELD ATTRIBUTE TO ALLOW PROFILE SETTINGS BY STATE

Depending on your business processes, you may require different values for the *AP Tax Calc Result Handling* profile option per state. This optional configuration allows the Sabrix profile option *AP Tax Calc Result Handling* to be set at the state level as attributes on the value set named SBX_STATES. See the *ONESOURCE Indirect Tax Integration for Oracle 12 Procure to Pay User Guide* for more information.

The *AP Tax Calc Result Handling* profile option is the only profile option set at the state level.

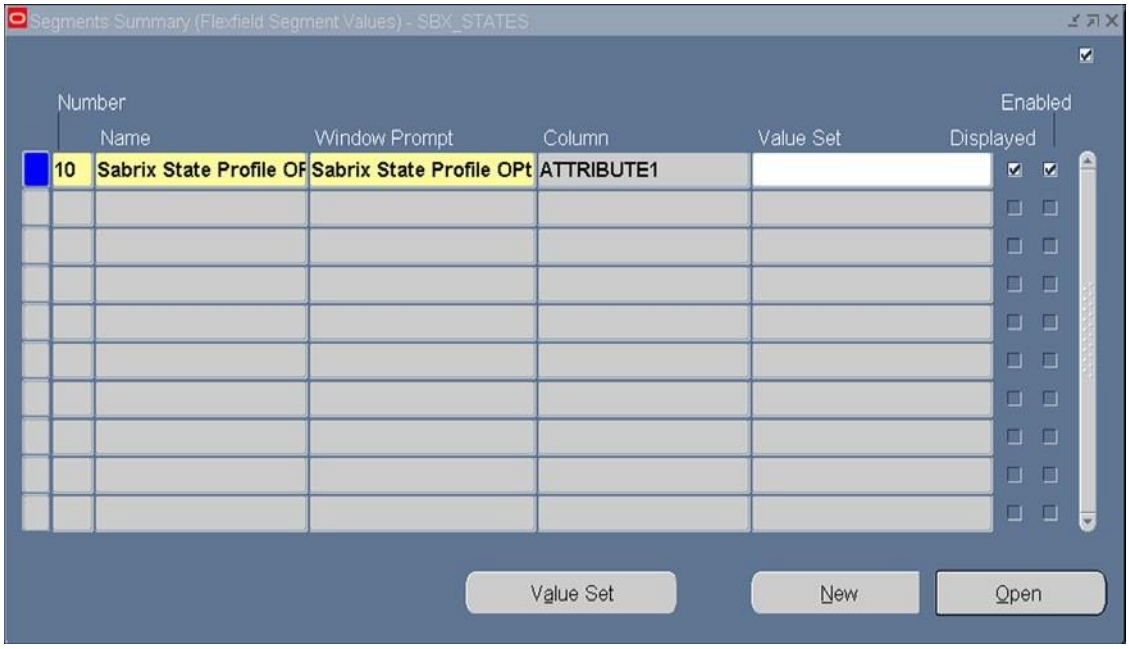
To define the optional descriptive flexfield for state level profile options:

1. Select the **Payables Manager** responsibility (or another that gives you the ability to define descriptive flexfields).
2. Navigate to **Setup > Flexfield > Descriptive > Segments.**
3. Query **Application Object Library** in the **Application** field and **Flexfield Segment Values** in the **Title** field.
4. Confirm the checkbox **Freeze Flexfield Definition** is unchecked.
5. Verify that **Context** is in the **Context Field** prompt.
6. Enter *FLEX_VALUE_SET_NAME* in the **Reference Field** in the context field section.
7. In the **Context Field** section, clear the **Required** and **Displayed** check boxes.
8. Check the **Synchronize with Reference Field** box.
9. In the **Context Field Values** section, add a new record**.**
10. Enter *SBX_STATES* in both the **Code** and **Description** fields



11. Click **Segments** for *SBX_STATES* and enter the following, using the table below as a guide:

Segment Name	Window Prompt	Column	Required	Display Size
Sabrix State Profile Options	Sabrix State Profile Options	Attribute *X*	N	50



NOTE The attribute number is specific to your installation.

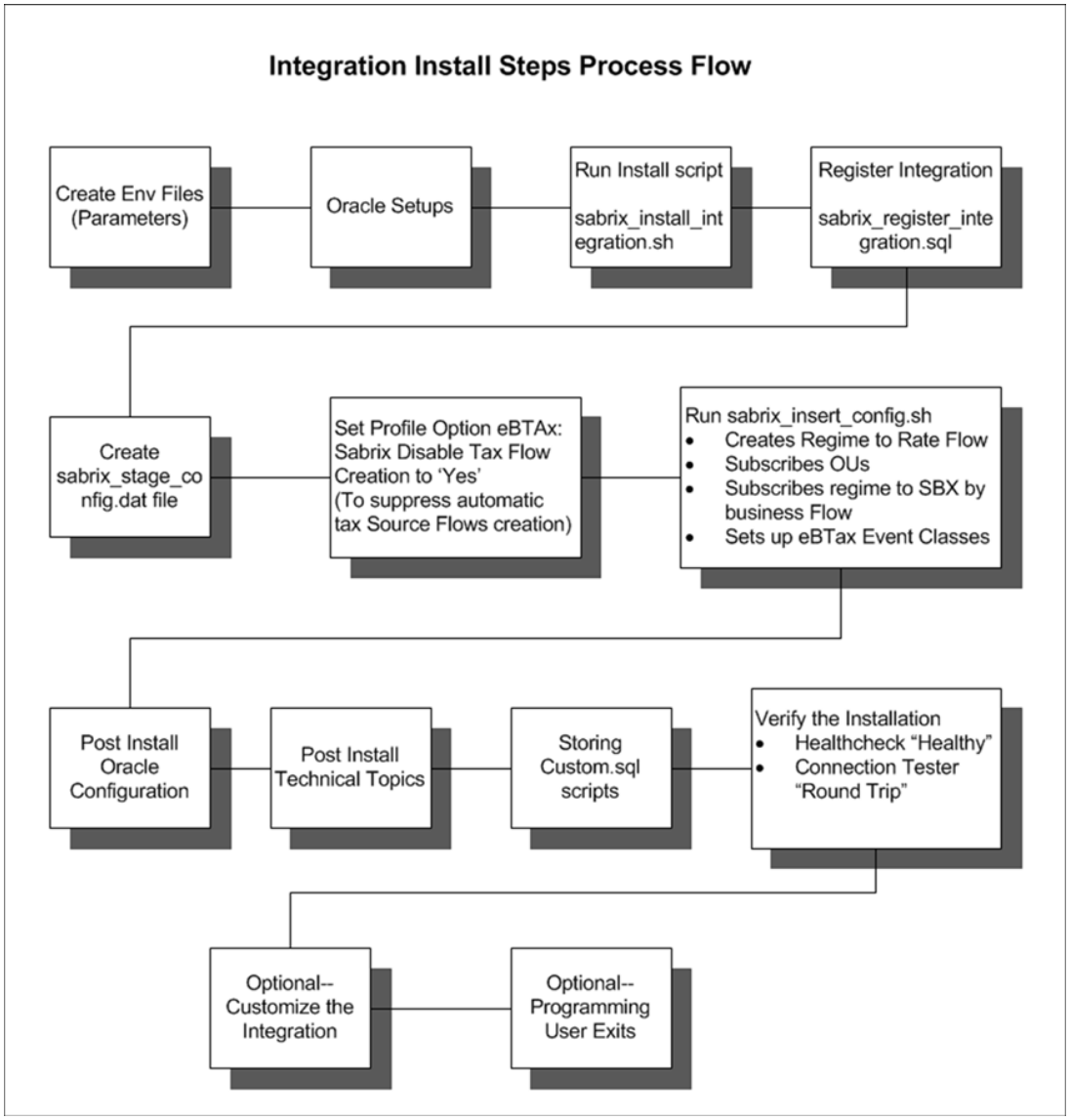
12. Save your work.

NOTE **Required** and **Display Size** column values are set when opening the segment.

Click **Open** for each segment to set the required *Y* or *N* flags; otherwise, all attributes default to Required (*Y*) during the initial setup.

PROCESS FLOW FOR INSTALL STEPS

The below process flow is a visual of the various steps required to install the ONESOURCE Indirect Tax Integration for Oracle 12 and is not an inclusive list. There are numerous steps which need to be completed before and after the actual installation of ONESOURCE Indirect Tax Integration for Oracle 12. For further details, review all of the various chapters in this guide.



CONFIGURING PARAMETERS IN ENVIRONMENT FILES

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle DBA	Successful creation of the XXIDT schema.	R	R	R
	Successful creation of API Owner ID.	R	R	R
	Confirmation that Oracle Legal Entity Configurator completed successfully (Refer to *R12 Oracle Financials Implementation Guide*).	R	R	R

The primary installation script *sabrix_installation_integration.sh* requires certain parameters. Thomson Reuters recommends that you set the parameters in reusable environment files so you can store the information for other installations or for automating installations. With little effort, you can modify key parameters in the files and use them for your various environments.

If you prefer to enter the parameters at run-time instead of using environment files, see *APPENDIX C: ENTERING PARAMETERS FOR INSTALLATION SCRIPTS (page 157)* for details.

In the sections that follow, there are instructions for creating new environment files from the following two template (*.tmp)* files:

- *sabrix_install.tmp*
- *sabrix_p2p.tmp*

Once you finish the steps below, you will have two environment files with the extension *.env*:

- *sabrix_install.env*
- *sabrix_p2p.env* (Required only for combined O2C/P2P installations or P2P only)

You must edit the *.env* files before you run the *sabrix_install_integration.sh* script. This ensures you have identified the correct installation information.

EDITING SABRIX_INSTALL.ENV

To edit the file:

1. Create *sabrix_install.env* by opening the template file *sabrix_install.tmp* in a text editor (The file *sabrix_install.tmp* is in the directory where you unzipped *ONESOURCEIDTIntegrationOracle12_5.6.2.4.zip* in the Import Files sub-directory.)

When using Windows NOTEPAD to edit either parameter or data files, NOTEPAD will append carriage return characters to every line in the file. On some Linux and Unix systems this may cause problems with the shell scripts and SQL*Loader scripts used in the installation of the Sabrix Integration. If you are using ftp to transfer files directly to a Linux or Unix server you should convert the files *sabrix_install.env*, *sabrix_p2p.env and sabrix_stage_config.dat* by using the dos2unix utility.

2. Save the file with the extension*.env*, creating *sabrix_install.env*. This file should be in the same directory as the script *sabrix_install_integration.sh*.
3. Continue in the text editor, adding the values from the table below.
4. When you are finished editing the file, save and close it, making sure the file is named *sabrix_install.env.*

Prompt	Description	Example	R = Required			Your Value
			O2C+P2P	O2C	P2P	
SETUP_EBTAX	Configure eBTax?	*Y*	R	R	R	
INSTALL_O2C	Installing Sabrix Integration for Oracle 12 Order to Cash (O2C)?	*Y or N*	R	R	R Enter *N*	

Prompt	Description	Example	R = Required			Your Value
			O2C+P2P	O2C	P2P	
INSTALL_P2P	Installing Sabrix Integration for Oracle 12 Procure to Pay (P2P)?	*Y or N*	R	R Enter *N*	R	
RETAIN_DATA	For Sabrix Integration upgrade installations only -- Determines if the installation script will retain existing data in Sabrix schema staging tables for sabrix_invoice, sabrix_invoice_out, sabrix_invoice_tax, sabrix_license, sabrix_license_out, sabrix_line, sabrix_line_tax, sabrix_line_out, sabrix_message, and sabrix_registration tables.	*Y or N*	R	R	R	
USE EXISTING CONFIG	Use existing tax flows to build new tax flows? Note: This installation parameter is used only during the upgrade from Integration version 5.3.x.x to version 5.4.x.x. This parameter has subsequently been deprecated in newer Integration releases.	*Y or N*	R	R	R	

Prompt	Description	Example	R = Required			Your Value
			O2C+P2P	O2C	P2P	
PACKAGES ONLY	Install only packages and views? (leaving schema objects, value sets, and profile options intact). Default is *N*	*Y or N*	R	R	R	
USER	XXIDTSchema User	*XXIDT*	R	R	R	
PWD	XXIDT Schema Password	*XXIDT*	R	R	R	
APPSUSER	Application (APPS) Schema User	*APPS*	R	R	R	
APPSPWD	Application Schema Password	*XXXX*	R	R	R	
DBNAME	Database name/alias	*oracle12dev*	R	R	R	
SABRIX_DATA	XXIDT Data Table Space	*XXIDT_DATA*	R	R	R	
SABRIX_IDX	XXIDT Index Table Space	*XXIDT_IDX*	R	R	R	
SBX_APPL_ SHORT_NAME	XXIDT Application ShortName	*XXIDT*	R	R	R	
SBX_EBTAX_STA RT_DATE	Effective start date for EBTax configuration.	*DD-MMM-YYYY*	R	R	R	

Be sure to supply a valid value for each parameter. Omissions may cause your installation to fail.

EDITING SABRIX_P2P.ENV

Edit this file if you are performing a combined Order to Cash and Procure to Pay installation or if you are only installing Procure to Pay. To edit the file:

1. Create *sabrix_p2p.env* by opening the template file *sabrix_p2p.tmp* in a text editor (The file *sabrix_p2p.tmp* is in the directory where you unzipped *ONESOURCEIDTIntegrationOracle12_5.6.2.4.zip* in the Import Files sub-directory.)
2. Save the file with the extension .env, creating *sabrix_p2p.env*. This file should be in the same directory as the script *sabrix_install_integration.sh*.

When using Windows NOTEPAD to edit either parameter or data files, NOTEPAD will append carriage return characters to every line in the file. On some Linux and Unix systems this may cause problems with the shell scripts and SQL*Loader scripts used in the installation of the Sabrix Integration. If you are using ftp to transfer files directly to a Linux or Unix server you should convert the files *sabrix_install.env*, *sabrix_p2p.env and sabrix_stage_config.dat* by using the dos2unix utility.

3. Continue in the text editor, adding the values from the table below.
4. When you are finished editing the file, save and close it, making sure the file is named *sabrix_p2p.env.*

Prompt	Description	Example	R = Required			Your Value
			O2C+P2P	O2C	P2P	
AP_TAX_FLAG_DFF	AP Sabrix Tax Status Flag Descriptive Flexfield (DFF)	*ATTRIBUTE1*	R		R	
AP_TAX_AMOUNT_DFF	AP Sabrix Tax Rate/Amount Descriptive Flexfield (DFF)	*ATTRIBUTE3*	R		R	
AP_TAX_LINK_DFF	AP Sabrix Item Line Link Descriptive Flexfield (DFF)	*ATTRIBUTE4*	R		R	

The attribute numbers above are specific to your installation and were created in the Defining Payables Descriptive Flexfields step.

Be sure to supply a valid value for each parameter. Omissions may cause your installation to fail.

EXECUTING THE SABRIX INTEGRATION INSTALLATION SCRIPT

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle DBA	• Successful creation of the Sabrix schema. • Created API Owner ID. • The two environment files have been edited or you have reviewed the command line parameters.	R	R	R
Technical	• Successful Installation of the ONESOURCE Indirect Tax Determination			

The primary installation script is *sabrix_install_integration.sh.* To execute it, complete the following:

1. Ensure that you edited the environment files (see Installation Parameters):
   - *sabrix_install.env* (Required)
   - *sabrix_p2p.env* (Required only for combined O2C/P2P installations or P2P-only)
2. Invoke a Bash shell on your client machine.
3. Run the following script from the directory in which you unzipped the *ONESOURCEIDTIntegrationOracle12_5.6.2.4.zip* files.

```
$ ./sabrix_install_integration.sh
```

This installation script is designed to run in a BASH shell.

4. Review the log file and check for errors. The log file is located in the directory you ran the install script. Verify the core EBTax setup for Sabrix Integration completed. Specifically look for ORA errors and scan for other errors.

The example below is the beginning section of the output after the *sabrix_install_integration.sh* script has started:

```
+-------------------------------------------------------------------------------+
+   Install script          : [sabrix_install_integration.sh]
+   RUN DATE                : [19-Aug-2015 15:46:14]
+   LOG FILE NAME           : [sabrix_install_integration1908151546.log]
+-------------------------------------------------------------------------------+
+ PARAMETERS:
+ Sabrix   SCHEMA NAME            : [XXIDT]
+ APPS     SCHEMA NAME            : [APPS]
+ Database                        : [pdxsasin103]
+ Setup EBTax                     : [Y]
+ EBTax Start Date                : [01-JAN-2014]
+ Install O2C                     : [N]
+ Install P2P                     : [Y]
+ Retain Staging Data             : [N]
+ Install Packages and Views Only : [N]
+ Sabrix Data Tablespace          : [XXIDT_DATA]
+ Sabrix Index Tablespace         : [XXIDT_IDX]
+ Sabrix Application Short Name   : [XXIDT]
+-------------------------------------------------------------------------------+
  AP Tax Status Flag DFF = ATTRIBUTE1
  AP Tax Amount DFF = ATTRIBUTE3
  AP Item Link DFF = ATTRIBUTE4
+-------------------------------------------------------------------------------+
+ Checking username AND passwords
+ Connecting TO SCHEMA XXIDT

SQL*Plus: Release 11.2.0.1.0 Production on Wed Aug 19 15:46:17 2015
```

The *sabrix_install_integration.sh* will submit the Oracle concurrent program, 'Compile Non-Compiled Flexfields.' This program is run to compile flexfields that may become uncompiled during the installation. It is submitted by the first user found with a responsibility defined with Request Group Name: 'Oracle E-Business Tax Request'. The installation *sabrix_install_integration.sh* log file will indicate success or failure in submission. If the submission fails,the message will read: 'Unable to find a suitable user to submit Compile Non-Compiled Flexfields process. If you have difficulty submitting concurrent requests from the Schedule Request form, please manually submit the Compile Non-Compiled Flexfields program.' In that case, the concurrent program can be run manually from the Oracle Application.

REGISTERING THE ONESOURCE INDIRECT TAX INTEGRATION FOR ORACLE 12

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle DBA logged on as APPS	Successful completion of installation script *sabrix_install_integration.sh*	R	R	R

1. In SQL*Plus, log on to the Oracle Applications database as *APPS*.
2. Run the following script from the directory in which you unzipped the Sabrix Integration installation files:

```
SQL> @sabrix_register_integration.sql
```

If you are installing O2C and P2P, run the script twice with a unique integration name for O2C and P2P.

3. Enter the following data:

Script Prompts	Description	O2C + P2P	O2C	P2P
Application Module Name	*O2C or P2P*	R Enter *O2C* in first execution and *P2P* in second	R Enter *O2C*	R Enter *P2P*
Integration Name	Example: *IntegrationO2C or IntegrationP2P*	R Enter *IntegrationO2C* in first execution and *IntegrationP2P* in second	R Enter *IntegrationO2C*	R Enter *IntegrationP2P*
Sabrix Tax Calc URL	The full URL of the Sabrix Solution tax calculation engine. Example: *http://neptune:7777/sabrix/xmlinvoice*	R	R	R
Application Module Name	Short Name for Concurrent Program "Sabrix AP Tax Thread" *SBXAPPROC* (default)	R Accept default	R Accept default	R Accept default

Script Prompts	Description	O2C + P2P	O2C	P2P
Application Short Name	Application Short Name for Concurrent Program "Sabrix AP Tax Thread" *XXIDT* (default) Use the same value as SBX_SHORT_NAME in *sabrix_install.env*. See *Editing sabrix_install.env*.	R	R	R
Debug Level	Set to one of the following: • *0* to log everything. • *5* to log the indata and outdata XML both to and from Sabrix. • 7 to generate timing statements for diagnosingperformance issues. • *10* to log errors only (Default).	R	N/A **Warning**: This parameter in the Sabrix_Connection_Options table is no longer used for O2C. It will be deprecated in a future release. The value for the profile option **eBTax: Sabrix Debug Level** replaces this parameter functionality for O2C.	R (Only for AP, not PO) **Warning**: This parameter in the Sabrix_Connection_Options table is no longer used for PO. It will be deprecated in a future release. The value for the profile option **eBTax: Sabrix Debug Level** replaces this parameter functionality for PO.
URL Protocol	Defines whether to use the XMLINVOICE Interface (HTTP) or the Web Services Interface (SOAP)	R	R	R
Proxy URL	The Proxy Address, if applicable [http://]host[:port][/]	R Only if applicable	R Only if applicable	R Only if applicable

The following is an example of the script's prompts and defaults for O2C.

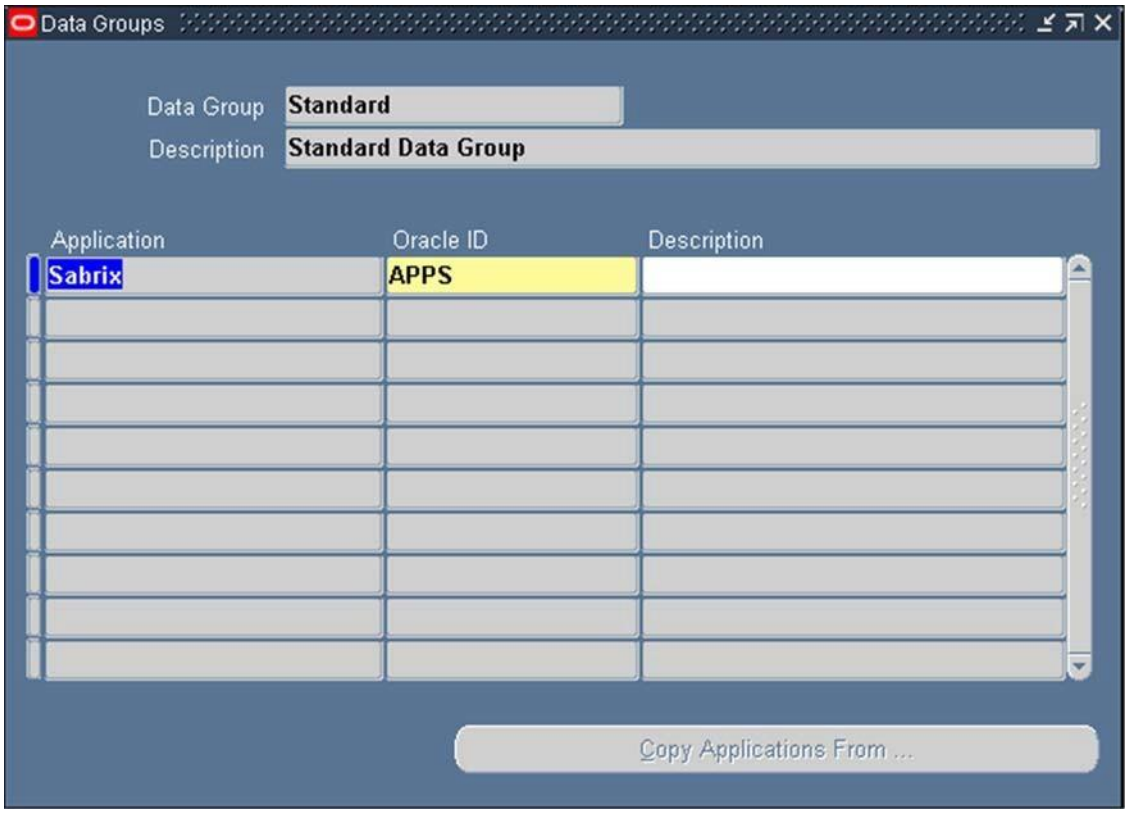
```
+   Script to Register Sabrix Integration Server Options
+   Install script : [sabrix_register_integration.sql]
+   Run date       : [19-AUG-2015 09:40:19]
+ All Parameter values are case sensitive
+ Parameters:
+------------------------------------------------------------------------------+


Enter Application Module Name (O2C/P2P)
    Application Module Name [P2P]  : O2C
Enter Integration Name (e.g. IntegrationP2P)
     Integration Name       [IntegrationO2C]    : IntegrationO2C
    Sabrix Tax Calc URL       []  : http://pdxsasin082.corp.ositax.com:7003/sabrix/services/taxcalculationservice
/2011-09-01/taxcalculationservice
Enter Short Name for Concurrent Program "Sabrix AP Tax Thread"
    Application Module Name [SBXAPPROC] :
'Enter Application Short Name for Concurrent Program "Sabrix AP Tax Thread"'
    Application Short  Name [XXIDT]  : XXIDT
'Enter Debug Level (0 - All Messages, 5 - XML, 10 - No messages) : '
 Debug Level [10] : 5
'Enter URL Protocol (HTTP or SOAP) : '
 URL Protocol [HTTP] : SOAP
'Enter Proxy address if applicable [http://]host[:port][/]'
 Proxy URL :
+------------------------------------------------------------------------------+
```

NOTE Once you register the integration, you can configure HTTPS. See *APPENDIX D: CONFIGURING HTTPS (OPTIONAL STEP) (page 161)*.

DEFINING THE SABRIX APPLICATION DATA GROUP

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle Superuser	Successful completion of *sabrix_install_integration.sh*	R	R	R

1. Select the **System Administrator**responsibility,
2. Navigate to **Security > ORACLE > Data Group**.
3. Query the **Data Group** *Standard* (or your company's custom data group).
4. Add a new record and select *Sabrix* in the **Application** field.
5. Select **APPS** as the **Oracle ID**.



6. Save and close the form.

SETTING UP REGIME TO RATE FLOW

Who	Prerequisites		O2C + P2P	O2C	P2P
Oracle Superuser	Country Code	The Country Code is the ISO country code and must be part of the Oracle TCA Geography Hierarchy. (Country Code must be set up in hz_geographies as a geography_type = *COUNTRY*)	R	R	R
	API Owner ID	This is the party_tax_profile_id for the party (Sabrix, Inc.). You should have completed this step as a prerequisite for executing the Integration installation script *sabrix_install_integration.sh*.			
	Application Data Group	See the preceding section *Defining the Sabrix Application Data Group*.			

To set up the regime to rate flow, complete these three primary tasks, as described in the following sections:

1. Download the *INTL_sabrix_stage_authorities.dat* (for international implementations) and the *US_sabrix_stage_authorities.dat* (for United States implementations) included in the *IntegrationOracle12SetupFile.zip* from the ONESOURCE Indirect Tax Customer Center at *https://customercenter.sabrix.com/sabrixcc*.
2. Create the *sabrix_stage_config.dat* file.
3. Run the *sabrix_insert_config.sh* script.

DOWNLOAD SABRIX_STAGE_AUTHORITIES FILES (*OPTIONAL*)

The first step, an optional step, for creating a functioning Regime to Rate Flow is to download two files: *INTL_sabrix_stage_authorities.dat* and the *US_sabrix_stage_authorities.dat.* Rename the files to *sabrix_stage_authorities.dat.* For details, see ***REGIME TO RATE FLOW PARAMETERS (page 27)***.

This step is optional. The authorities files contain a listing of all authorities from Determination. The authorities can be loaded into the Sabrix schema and join the resulting table to the SABRIX_STAGE_CONFIG table to build the Tax to Rate Flows. If these files are not loaded and a *DFLT* flow is defined, the authority flows will automatically be created at transaction time. See ***DEFAULT ACCOUNTING WHEN CREATING TAX FLOWS (page 10)*** for more information on the automatic flow creation process.

CREATE SABRIX_STAGE_CONFIG.DAT FILES

The next step for creating a functioning Regime to Rate Flow is to create the *sabrix_stage_config.dat* file. For details, see ***REGIME TO RATE FLOW PARAMETERS (page 27)***.

The Oracle E-Business Tax Partner interface requires a rigid configuration of taxes, jurisdictions, status and rates. All accounting for third party tax providers is derived at the rate level. For each tax flow, a distinct tax, jurisdiction, status and rate must be configured.

The Regime to Rate Flow information must be populated in the SABRIX_STAGE_CONFIG staging table before the interface can create the Regime to Rate Flows. Load your data file information about regimes, operating units and general ledger accounts in the SABRIX_STAGE_CONFIG staging table. Enter Country Code, Country Name, Currency Code, Exchange Rate Type, ERP Tax Code, Tax Direction, Operating Unit ID, and general ledger account numbers for Tax Recoverable/Liability and Tax Expense. The shell script routine helps you load your Regime to Rate Flow data into the staging table.

If the data file contains a row for the ERP Tax Code *DFLT*, defining default accounting by workflow and by operating unit, a *DFLT* Tax to Rate Flow will be created. If an unexpected ERP Tax Code is returned by Determination, the *DFLT* Tax to Rate Flow will be used to automatically create a new Tax to Rate Flow for the new ERP Tax Code at the transaction time. Thus, the transaction is allowed to complete normally. A report, *Tax Flows created by ONESOURCE Indirect Tax for Oracle 12,* displays the newly created Tax Flows by date range. See ***DEFAULT ACCOUNTING WHEN CREATING TAX FLOWS (page 10)*** for more information on the *DFLT* Tax to Rate Flows.

If using either the *US_sabrix_stage_config.dat and the INTL_sabrix_stage_config.dat* file to create *sabrix_stage_config.dat* and want to suppress automatic Source Flow creation, the Profile Option eBTAx: Sabrix Disable Tax Flow Creation must be set to 'Yes' prior to running the sabrix_insert_config.sh script.

RUN THE SABRIX_INSERT_CONFIG.SH

Execute the Shell Script: sabrix_insert_config.sh.

To create regimes using the shell script, execute *sabrix_insert_config.sh*:

1. Confirm that your Regime to Rate Flow information is correctly entered in the data file *sabrix_stage_config.dat* ***(page 8)***.
2. If your data file is complete, invoke a Bash shell on your client machine.
3. Execute the following shell script from the directory in which you unzipped the Integration.

```
$ ./sabrix_insert_config.sh
```

This script loads the data file information into the SABRIX_STAGE_CONFIG table and then executes the interface *SABRIX_EBTAX.setup_everything* to create the regime to rate flows in Oracle eBTax. The script deletes records from the SABRIX_STAGE_CONFIG staging table after successfully loading them into Oracle.

Be sure to review the *sabrix_stage_config.log*, *sabrix_stage_config.dsc* (discarded data file for header row in the *sabrix_stage_config.dat* file) and the *sabrix_stage_config.bad* files for errors with the *sabrix_insert_config.sh* script.

If you perform the optional step of loading the *sabrix_stage_authorities.dat,* be sure to review the *sabrix_stage_authorities.log*, *sabrix_stage_authorities.dsc* (discarded data file for header row in the *sabrix_stage_authorities.dat* file) and the *sabrix_stage_authorities.bad* files for errors with the *sabrix_insert_config.sh* script.

The sabrix_insert_config log file lists the number of tax codes that were inserted for a given workflow. For example:

-Inserted 82 tax codes for O2C

The script does not prompt you if it can get the parameters from your environment files, as described in ***CONFIGURING PARAMETERS IN ENVIRONMENT FILES (page 49)***. If you are prompted, it means one of the following:

- You chose to use the .env files for the primary installation script *sabrix_install_integration.sh* but did not complete all of the required parameters. If this is the case, you must add those missing values, see ***CONFIGURING PARAMETERS IN ENVIRONMENT FILES (page 49)***.
- You chose *not* to use the ENV files for the primary installation script *sabrix_install_integration.sh*; instead, you provided the parameters at execution. If you chose this path, answer the prompts listed below.

Prompt	Details
Install O2C	Create an O2C Tax Flow? *Yes* or *No*
Install P2P	Create a P2P Tax Flow? *Yes* or *No*
Use Existing Config	Use existing tax flows to build new tax flows? *Yes* or *No* Note: This installation parameter is used only during the upgrade from Integration version 5.3.x.x to version 5.4.x.x. This parameter has subsequently been deprecated in newer Integration releases.
Start Date	Tax Effective Start Date (DD-MMM-YYYY)

The example below is the beginning section of the output after the *sabrix_insert_config.sh* script has started:

```
+-------------------------------------------------------------------------------+
+   Install script          : [sabrix_insert_config.sh]
+   RUN DATE                : [29-Sep-2011 09:55:12]
+   LOG FILE NAME           : [sabrix_insert_config2909110955.log]
+-------------------------------------------------------------------------------+
+ PARAMETERS:
+ APPS     SCHEMA NAME                              : [APPS]
+ Database                                          : [pdxsasin008]
+ Start Date for eBTax                              : [01-JAN-2011]
+ Install O2C                                       : [Y]
+ Install P2P                                       : [Y]
+ Use Existing Configuration?                       : [N]
+-------------------------------------------------------------------------------+
+-------------------------------------------------------------------------------+
+ Checking username AND passwords
+ Connecting TO SCHEMA APPS
USER is "APPS"
+-------------------------------------------------------------------------------+
+ Connected TO schema successfully .. PARAMETERS OK
+-------------------------------------------------------------------------------+
+-------------------------------------------------------------------------------+
```

For new installations, the installation script sets the APPLIED_AMT_HANDLING_FLAG column to 'P' (prorate). For installations prior to ONESOURCE Indirect Tax Integration for Oracle12 Version 5.5.0.0, this flag is not updated by the installation script. To update this column post-install, use one of the following methods included in the ONESOURCEIDTIntegrationOracle12_5.6.2.4.zip file:

- sabrix_update_applied_amt_hdlg.sql script -single update for applicable rows in the zx_taxes_b table, updating the applied_amount_handling_flag to 'P'.
- sabrix_update_applied_amt_hdlg_api.sql script - utilizes the zx_taxes_pkg.update_row api and updates the applied_amount_handling_flag to 'P'.

Thomson Reuters recommends NOT changing the flag 'Allow Tax Applicability' from checked to unchecked in the Configuration Owner Tax Options in Oracle eBTax. This is checked in the Thomson Reuters Integration for Oracle 12 configuration process. Oracle has stated this field must be checked for Third Party Tax Providers. Customers creating new Configuration Owner Tax Options with the 'Allow Tax Applicability' flag unchecked are not supported by Thomson Reuters.

Results of Regime Creation

The following Core and Workflow objects are affected after you execute the interface *SABRIX_EBTAX.setup_everthing:*

Object Name	Core Installation	Description of Changes
ZX_REGIMES_B	TAX_REGIME_CODE	SBX_REGIME_ + Country Code
	HAS_SUB_REGIME_FLAG	N
	COUNTRY_OR_GROUP_CODE	COUNTRY
	COUNTRY_CODE	US
	EFFECTIVE_FROM	Tax Start Date
	TAX_STATUS_RULE_FLAG	N
	DEF_PLACE_OF_SUPPLY_TYPE_CODE	SHIP_FROM
	APPLICABILITY_RULE_FLAG	N
	PLACE_OF_SUPPLY_RULE_FLAG	N
	TAX_CALC_RULE_FLAG	N
	TAXABLE_BASIS_THRSHLD_FLAG	N
	TAX_RATE_THRSHLD_FLAG	N
	TAX_AMT_THRSHLD_FLAG	N
	TAX_RATE_RULE_FLAG	N
	TAXABLE_BASIS_RULE_FLAG	N
	DEF_INCLUSIVE_TAX_FLAG	N
	HAS_OTHER_JURISDICTIONS_FLAG	N
	ALLOW_ROUNDING_OVERRIDE_FLAG	N
	ALLOW_EXEMPTIONS_FLAG	Y
	ALLOW_EXCEPTIONS_FLAG	Y
	ALLOW_RECOVERABILITY_FLAG	Y
	AUTO_PRVN_FLAG	N
	HAS_TAX_DET_DATE_RULE_FLAG	N
	HAS_EXCH_RATE_DATE_RULE_FLAG	N
	HAS_TAX_POINT_DATE_RULE_FLAG	N
	USE_LEGAL_MSG_FLAG	N
	REGN_NUM_SAME_AS_LE_FLAG	N

Object Name	Core Installation	Description of Changes
ZX_SUBSCRIPTION_DETAILS	SUBSCRIPTION_OPTION_ID	0
	FIRST_PTY_ORG_ID	-99 (indicates global content owner)
	PARENT_FIRST_PTY_ORG_ID	-99 (indicates global content owner)
	VIEW_OPTIONS_CODE	NONE
	TAX_REGIME_CODE	SBX_REGIME_ + Country Code
	EFFECTIVE_FROM	Tax Start Date
	RECORD_TYPE_CODE	USER_DEFINED
ZX_API_OWNER_STATUSES	STATUS_CODE is set to GENERATED to force tpi_plugin generation. Process will create a row if it does not exist for the Sabrix Party (API_OWNER_ID).	

Workflow Installation		
ZX_API_CODE_COMBINATIONS	Two Rows Created (One for each context)	
	Context	**Context_Flex_Structure_ID**
	Service Partner	53261
	User Extension	53262
	Field Name	**Value**
	ENABLED_FLAG	Y
	SEGMENT_ATTRIBUTE1	SBX_REGIME_ + Country Code
	SEGMENT_ATTRIBUTE2	Workflow (O2C or P2P)
	SUMMARY_FLAG	N

<table>
<tr><th colspan="3">Workflow Installation</th></tr>
<tr><td rowspan="8">ZX_API_REGISTRATIONS
This table is dependent on ZX_API_CODE_COMBINATIONS with the context_ccid populated from the code_combination_id</td><td colspan="2">7 Rows Created for each Context ID</td></tr>
<tr><td>Context</td><td>Procedures</td></tr>
<tr><td rowspan="5">Service Partner</td><td>CALCULATE</td></tr>
<tr><td>SYNCHRONIZE</td></tr>
<tr><td>UPDATE_DOCUMENT</td></tr>
<tr><td>COMMIT_FOR_TAX</td></tr>
<tr><td>IMPORT_EXEMPTIONS</td></tr>
<tr><td rowspan="2">User Extension</td><td>DERIVE_HDR_ATTRS</td></tr>
<tr><td>DERIVE_LINE_ATTRS</td></tr>
<tr><td>ZX_EVNT_CLS_MAPPINGS</td><td colspan="2">For P2P only, where application_id = 200:
MANUAL_LINES_FOR_PARTNER_FLAG = ‘Y’
MAN_TAX_ONLY_LIN_FOR_PTNR_FLAG = ‘Y’</td></tr>
<tr><td rowspan="10">ZX_PRODUCT_OPTIONS</td><td colspan="2">For O2C only, to allow entry and override of customer exemptions in AR.</td></tr>
<tr><td>PRODUCT_OPTIONS_ID</td><td>ZX_PRODUCT_OPTIONS_S.nextval</td></tr>
<tr><td>ORG_ID</td><td>Subscribing Operating Unit</td></tr>
<tr><td>APPLICATION_ID</td><td>222 (AR)</td></tr>
<tr><td>TAX_METHOD_CODE</td><td>‘EBTAX’</td></tr>
<tr><td>TAX_USE_CUSTOMER_EXEMPT_FLAG</td><td>‘Y’</td></tr>
<tr><td>TAX_USE_CLASSIFICATION_FLAG</td><td>‘Y’</td></tr>
<tr><td>OBJECT_VERSION_NUMBER</td><td>‘1’</td></tr>
<tr><td>RECORD_TYPE_CODE</td><td>User defined</td></tr>
</table>

Results of Tax Flow Creation

The following Core and Workflow objects are affected after you execute the interface *SABRIX_EBTAX.setup_everything*:

Object Name	Description of Changes	
ZX_TAXES_B	Naming Convention: ERP Tax Code_WF_Direction_sequence	
	RECOVERY_RATE_OVERRIDE_FLAG	'N'
	PRIMARY_RECOVERY_TYPE_CODE	'STANDARD'
	PRIMARY_REC_RATE_DET_RULE_FL	'N'
	SEC_REC_RATE_DET_RULE_FLAG	'N'
	OFFSET_TAX_FLAG	'N'
	RECORD_TYPE_CODE	'USER_DEFINED'
	ALLOW_ROUNDING_OVERRIDE_FLAG	'Y'
	TAX	ERP Tax Code + WF + Direction + sequence
	EFFECTIVE_FROM	Tax Start Date
	TAX_REGIME_CODE	SBX_REGIME + Country
	ALLOW_MANUAL_ENTRY_FLAG	'Y'
	ALLOW_TAX_OVERRIDE_FLAG	'Y'
	EXCHANGE_RATE_TYPE	From Parent Regime
	TAX_CURRENCY_CODE	From Parent Regime
	TAX_PRECISION	2
	MINIMUM_ACCOUNTABLE_UNIT	0.01
	ROUNDING_RULE_CODE	'NEAREST'
	TAX_STATUS_RULE_FLAG	'N'
	TAX_RATE_RULE_FLAG	'N'
	DEF_PLACE_OF_SUPPLY_TYPE_COD	'SHIP_TO_BILL_TO'
	PLACE_OF_SUPPLY_RULE_FLAG	'N'
	APPLICABILITY_RULE_FLAG	'N'
	TAX_CALC_RULE_FLAG	'N'
	TAXABLE_BASIS_RULE_FLAG	'N'
	DEF_INCLUSIVE_TAX_FLAG	'N'
	HAS_OTHER_JURISDICTIONS_FLAG	'N'
	ALLOW_EXEMPTIONS_FLAG	'Y'
	ALLOW_EXCEPTIONS_FLAG	'Y'
	ALLOW_RECOVERABILITY_FLAG	'N' for U.S. Taxes 'Y' for Non-U.S. Taxes
	DEF_TAX_CALC_FORMULA	'STANDARD_TC'

Object Name	Description of Changes	
ZX_TAXES_B	TAX_INCLUSIVE_OVERRIDE_FLAG	'Y'
	DEF_TAXABLE_BASIS_FORMULA	'STANDARD_TB'
	DEF_REGISTR_PARTY_TYPE_CODE	'SHIP_TO_BILL_TO'
	REGISTRATION_TYPE_RULE_FLAG	'N'
	REPORTING_ONLY_FLAG	'N'
	LIVE_FOR_PROCESSING_FLAG	'Y'
	PRIMARY_RECOVERY_TYPE_CODE	'STANDARD'
	TAX_FULL_NAME	ERP Tax Code + WF + Direction + sequence
	ZONE_GEOGRAPHY_TYPE	'COUNTRY'
	DEF_REC_SETTLEMENT_OPTION_CO	'IMMEDIATE'
	REGN_NUM_SAME_AS_LE_FLAG	'N'
	CONTENT_OWNER_ID	-99 (indicates global content owner)
	APPLIED_AMT_HANDLING_FLAG	'P'
	PARENT_GEOGRAPHY_TYPE	'COUNTRY'
	PARENT_GEOGRAPHY_ID	Geography ID of Country
	ALLOW_MASS_CREATE_FLAG	'Y'
	SOURCE_TAX_FLAG	'Y'
	DEF_PRIMARY_REC_RATE_CODE	For non-U.S. Taxes, Tax rate code of recovery rate
	SPECIAL_INCLUSIVE_TAX_FLAG	'N'
	ALLOW_DUP_REGN_NUM_FLAG	'N'
	LIVE_FOR_APPLICABILITY_FLAG	'Y'
	APPLICABLE_BY_DEFAULT_FLAG	'Y'

Object Name	Description of Changes	
ZX_JURISDICTIONS_B	Naming Convention: ERP Tax Code_WF_Direction_sequence	
	'TAX_JURISDICTION_CODE	ERP Tax Code + WF + Direction + sequence
	EFFECTIVE_FROM	Tax Start Date
	TAX_REGIME_CODE	SBX_REGIME + Country
	ZONE_GEOGRAPHY_ID	Geography ID of Country
	TAX	ERP Tax Code + WF + Direction + sequence
	DEFAULT_JURISDICTION_FLAG	'Y'
	DEFAULT_FLG_EFFECTIVE_FROM	Tax Start Date
	TAX_JURISDICTION_NAME	ERP Tax Code + WF + Direction + sequence
	RECORD_TYPE_CODE	'USER_DEFINED'
	PRECEDENCE_LEVEL	300
	ALLOW_TAX_REGISTRATIONS_FLAG	'Y'
ZX_STATUS_B	Naming Convention: ERP Tax Code_WF_Direction_sequence	
	TAX_STATUS_CODE	ERP Tax Code + WF + Direction + sequence
	EFFECTIVE_FROM	Tax Start Date
	TAX	ERP Tax Code + WF + Direction + sequence
	TAX_REGIME_CODE	SBX_REGIME + Country
	RULE_BASED_RATE_FLAG	'N'
	ALLOW_RATE_OVERRIDE_FLAG	'Y'
	ALLOW_EXEMPTIONS_FLAG	'Y'
	ALLOW_EXCEPTIONS_FLAG	'Y'
	DEFAULT_STATUS_FLAG	'Y'
	DEFAULT_FLG_EFFECTIVE_FROM	Tax Start Date
	DEF_REC_SETTLEMENT_OPTION_CO	'IMMEDIATE'
	RECORD_TYPE_CODE	'USER_DEFINED'
	TAX_STATUS_NAME	ERP Tax Code + WF + Direction + sequence
	CONTENT_OWNER_ID	-99 (indicates global content owner)
	TAX_RATE_CODE	ERP Tax Code + WF + Direction + sequence

Object Name	Description of Changes	
ZX_RATES_B	CONTENT_OWNER_ID	-99 (indicates global content owner)
	EFFECTIVE_FROM	Tax Start Date
	TAX_REGIME_CODE	SBX_REGIME + Country
	TAX	ERP Tax Code + WF + Direction + sequence
	TAX_STATUS_CODE	ERP Tax Code + WF + Direction + sequence
	RATE_TYPE_CODE	'PERCENTAGE' for Tax Rate Codes 'RECOVERY' for Tax Recovery Rate Codes
	PERCENTAGE_RATE	1 for Tax Rate Codes 100 for Tax Recovery Rate Codes
	RECOVERY_TYPE_CODE	'STANDARD'
	ACTIVE_FLAG	'Y'
	DEFAULT_RATE_FLAG	'Y'
	DEFAULT_FLG_EFFECTIVE_FROM	Tax Start Date
	DEFAULT_REC_TYPE_CODE	'STANDARD'
	DEFAULT_REC_RATE_CODE	For non-U.S. 'PERCENTAGE' RATE_TYPE_CODE, Tax rate code of recovery rate
	DEF_REC_SETTLEMENT_OPTION_CODE	'IMMEDIATE'
	RECORD_TYPE_CODE	'USER_DEFINED'
	TAX_RATE_NAME	ERP Tax Code + WF + Direction + sequence
	ALLOW_ADHOC_TAX_RATE_FLAG	'Y'
	ADJ_FOR_ADHOC_AMT_CODE	'TAXABLE_BASIS'
	INCLUSIVE_TAX_FLAG	'N'
	TAX_INCLUSIVE_OVERRIDE_FLAG	'Y'
	OBJECT_VERSION_NUMBER	1
	ALLOW_EXEMPTIONS_FLAG	'Y'
	ALLOW_EXCEPTIONS_FLAG	'Y'

Results of Sabrix Subscribe Operating Unit

The following Core and Workflow objects are affected after you execute the interface *SABRIX_EBTAX.setup_everything:*

Core Installation	
Object Name	**Description of Change**
ZX_PARTY_TAX_PROFILE	If the operating unit does not have a Party Tax Profile, one will be created dynamically.
ZX_REGIMES_USAGES	A row will be inserted into ZX_REGIME_USAGES if one does not already exist for the regime and operating unit.
ZX_SUBSCRIPTION_OPTIONS	One row for the REGIME_USAGE_ID, with SUBSCRIPTION_OPTION_CODE = 'GCO_ONLY' is inserted.
ZX_SUBSCRIPTION_DETAILS	One row is created in ZX_SUBSCRIPTION_DETAILS, with VIEW_OPTIONS_CODE set to "NONE".

Workflow Installation	
ZX_EVNT_CLS_OPTIONS	
For P2P/AP	Event Class Codes •'STANDARD INVOICES' •'PREPAYMENT INVOICES •'EXPENSE REPORTS' Entity Code = 'AP_INVOICES'
For P2P/Purchasing	• Event Class Code = 'PO_PA' Entity Code = 'PURCHASE_ORDER' • Event Class Code = 'REQUISITION' Entity Code = 'REQUISITION' • Event Class Code ='RELEASE' Entity Code = 'RELEASE'
For O2C (Shared by AR and OM)	Event_class_codes •'INVOICE' •'CREDIT_MEMO' •'DEBIT_MEMO' Entity_Code = 'TRANSACTIONS'

Results of Sabrix Add Account to Tax Rate Code

The following Workflow objects are affected after you execute the interface *SABRIX_EBTAX.setup_everything:*

Workflow Installation	
ZX_ACCOUNTS	
O2C	CCID for O2C Recoverable/Liability
P2P	CCID for P2P Recoverable/Liability CCID for Tax Expense
ZX_SRVC_SUBSCRIPTIONS	One row for each Workflow

POST INSTALLATION: ORACLE CONFIGURATION

This chapter describes Oracle 12 configurations you must make after installing the ONESOURCE Indirect Tax Integration.

Follow the procedures in this order:

- ***DESCRIPTIVE FLEXFIELDS: ASSOCIATING VALUE SETS (page 76)***
- ***CONFIGURING PROFILE OPTIONS (page 79)***
- ***SETTING TAX EVENTS ON ORDER MANAGEMENT TRANSACTION TYPES (page 101)***
- ***DEFINING SABRIX AP HOLDS (page 102)***
- ***BYPASS TCA GEOGRAPHY HIERARCHY VALIDATIONS (page 104)***
- ***SYNCHRONIZING ZONE NAMES (page 107)***

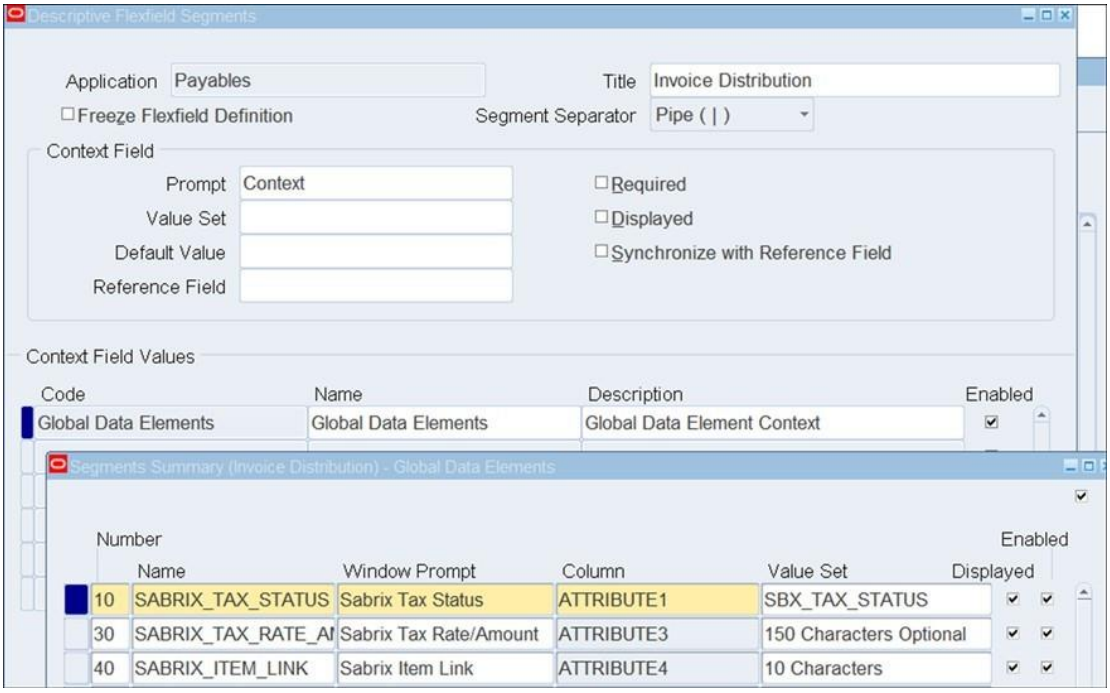
To successfully configure profile options and descriptive flexfields, you must have both the System Administrator and Application Developer responsibilities.

DESCRIPTIVE FLEXFIELDS: ASSOCIATING VALUE SETS

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle Superuser	Successful completion of Integration installation	R		R

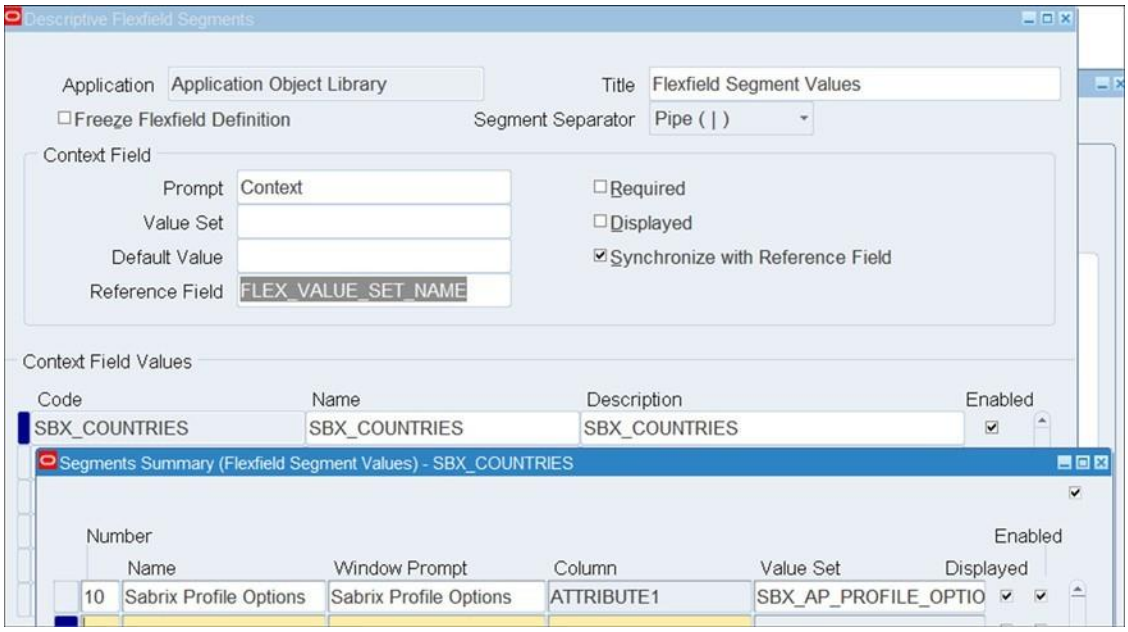
During the installation process, you defined descriptive flexfields in Oracle and assigned attribute columns. Subsequently, the *sabrix_install_integration.sh* script automatically created value sets and value set values to be used for the attributes listed in *sabrix_p2p.env*. Now, you must associate several of these value sets with the descriptive flexfields. To do so:

1. Select the **Payables Manager** responsibility (or another that gives you the ability to define descriptive flexfields).
2. Navigate to **Setup > Flexfield > Descriptive > Segments**.
3. Query the **Payables** in the **Application** field and **Invoice Distribution** in the **Title** field.
4. Confirm that the **Freeze Flexfield Definition** checkbox is unchecked, click **Segments**, and add the **Value Sets** following the table below.



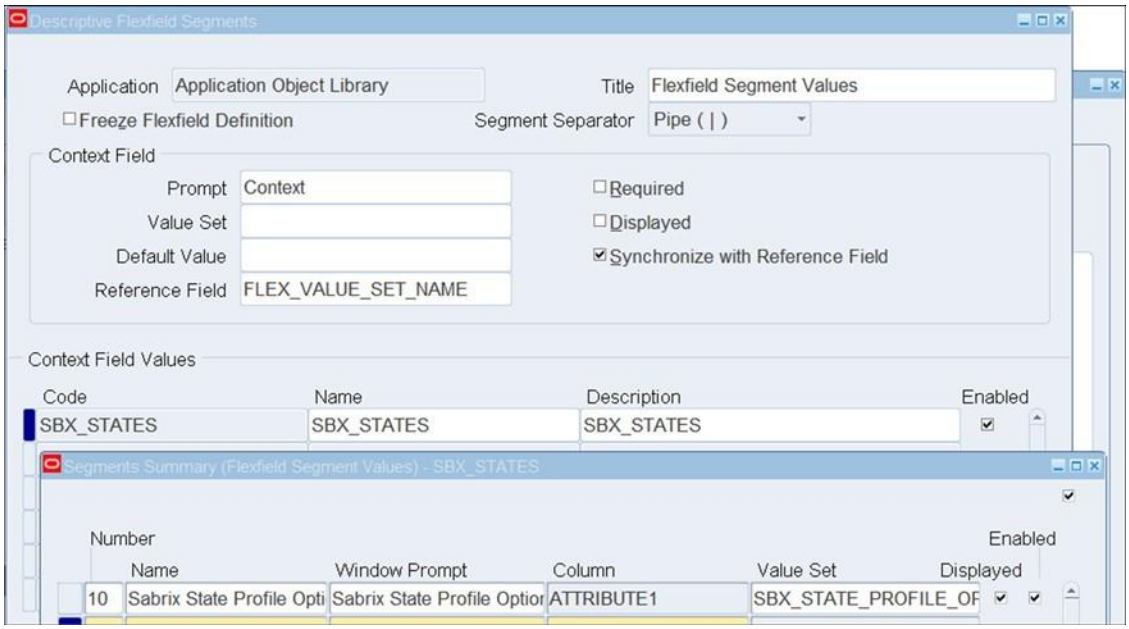
Segment Name	Value Set
SABRIX_TAX_STATUS	SBX_TAX_STATUS
SABRIX_TAX_RATE_AMOUNT	150 Characters Optional
SABRIX_ITEM_LINK	10 Characters

5. Save your work.
6. Freeze and compile the flexfield (Click through error messages stating that this is supported only in Oracle Forms).
7. Query **Application Object Library** in the **Application** field and **Flexfield Segment Values** in the **Title** field.
8. Unfreeze the flexfield definition and place your cursor in the **Context Field Values for SBX_COUNTRIES** and click **Segments**.
9. Select ***SBX_AP_PROFILE_OPTIONS*** value set for **Sabrix Profile Options**.



10. Save your work.
11. Close the Segments Summary window.

12. On the Descriptive Flexfield Segments window, place your cursor in the **Context Field Values for SBX_STATES** and click **Segments**.



13. Select ***SBX_STATE_PROFILE_OPTIONS*** value set for **Sabrix State Profile Options**.

14. Freeze and compile the flexfield (Click through error messages stating that this is supported only in Oracle Forms).

CONFIGURING PROFILE OPTIONS

You can configure profile options for four levels of organization:

- Site level
- Application level
- Organization level
- Country level
- State level

CONFIGURING PROFILE OPTIONS AT THE SITE LEVEL

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle Superuser	Successful completion of Integration installation	X	X	X

NOTE An *X* in the profile options tables indicates that the value is used for functionality within the corresponding business flow.

Profile options affect the way your application looks and behaves and may be changed to suit your preferences. The following profile options were created during the installation of the ONESOURCE Indirect Tax Integration and are configured in Oracle.

To set profile options:

1. Select the **System Administrator** responsibility.
2. Navigate to **Profile > System**.

3. Enter *%SABRIX%* in the **Profile** field and click **Find**.
4. Using the table below as a guide, enter and save the appropriate information. You can record your own choices in the **Your Value** column below:

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: AP Sabrix Audit Determination Tax Amount	• *Yes*: Saves the Tax Control Amount in the **vendor_tax** and Determination calculated tax in the **tax_amount** audit fields. • *No*: Saves the Determination calculated tax in both **vendor_tax** and **tax_amount** audit fields. • *NULL* value behaves as a *Yes* value. **Note**: This profile option only applies to non-U.S. transactions.	X		X	
eBTax: AP Sabrix Use Trx Date to Lookup Registrations	• *Yes*: The supplier registration is looked up using the AP Transaction Date. • *No*: The supplier registration is looked up using the sysdate. • *NULL* value behaves as a *Yes* value.	X		X	
eBTax: Sabrix AP Approve Invoices	Set to one of the following for AP: • *Yes*: Allows an invoice that is short-paid to retain its approved status. • *No*: Sends the transaction back through the AP approval workflow (if enabled at your company). • *NULL* value behaves as a *Yes* value.	X		X	
eBTax: Sabrix AP Audit Online Tax Calc	Set to one of the following for AP: • *Yes*: AP Transactions are sent to Determination Audit. • *No*: AP Transactions are not sent to Determination Audit. • *NULL* value behaves as a *Yes* value. **Note**: This profile option does not apply to U.S. AP invoices.	X		X	

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix AP Batch by Lines	Set to one of the following for AP: • *Yes:* Use the number of **invoice lines** to determine the number of workers to submit when running the Sabrix Tax Process concurrent process. • *No:* Use the number of **invoices** to determine how many workers to submit when running the Sabrix Tax Process concurrent process.	X		X	
eBTax: Sabrix AP Create Transactions	Set to one of the following for AP: • *Yes*: Create data in the SABRIX_TRANSACTIONS_AP table (data used for Sabrix Accrual Report). Also used to troubleshoot the outcome of the state level profile option processing. • *No*: Do not create data in the sabrix_transaction_ap table. • *NULL* value defaults to *No*. **Note**: If this profile option is not set to *Yes*, there will be no data for the Sabrix Accrual Report in the SABRIX_TRANSACTIONS_AP table.	X		X	
eBTax: Sabrix AP Exclude VAT from Sabrix Tax Process	To bypass tax calculation from the *ONESOURCE Integration* concurrent tax process (Sabrix Tax Process) for non-U.S. AP invoices processed prior to the *ONESOURCE Integration* 5.6.x.x online tax calculation. • *Yes*: Excludes non-U.S. AP invoices processed prior to 5.6.x.x from Sabrix Tax Process. • *No*: Does not exclude non-U.S. AP invoices processed prior to 5.6.x.x from Sabrix Tax Process. • *NULL* value defaults to *No*.	X		X	

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix AP Hold?	• Set to *Yes* to apply an AP Hold Reason for Sabrix. • Set to No to not apply a Hold on US invoices for manual processing of US Out of Tolerance Hold invoices.	X		X	
eBTax: Sabrix AP Maximum Workers	Allows you to set a maximum number of concurrent processes that are submitted from the **Sabrix Tax Process** for AP. If the limit is reached, any transactions that are not picked up will be processed in the next run of the **Sabrix Tax Process**. A null value defaults to a limit of 20.	X		X	

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix AP Overcharge Shortpay Action	To handle Vendor Tax Overcharges outside the tolerance levels specified by the two tolerance profile options below, you can set this option to: • *ACCRUE - Prorate By Item:* Enables accrue shortpay functionality for incorrect vendor tax charges, prorated by itemline amounts. The invoice amount will not change, but invoice lines and distribution line entries are created as Sabrix Tax Accrual and Sabrix Tax Expense for the vendor overcharged tax amount. • *ACCRUE - Prorate By Tax:* Enables accrue shortpay functionality for incorrect vendor tax charges, prorated by individual tax percentages. The invoice amount will not change, but invoice lines and distribution line entries are created as Sabrix Tax Accrual and Sabrix Tax Expense for the vendor overcharged tax amount. • *OFF:* The invoice amount will not change and **NO** invoice lines and distribution line entries are created as Sabrix Tax Accrual and Sabrix Tax Expense for the vendor overcharged tax amount. **Use this option to eliminate negative tax accruals.** • *PARTIAL*: This will reduce the invoice total by the difference between the calculated and invoiced vendor tax amount, and Sabrix generated partial short pay entry lines are created for the vendor overcharged tax amount. • *TOTAL*: This will reverse the entire vendor tax amount, reduce the invoice amount by the vendor tax amount, and make entries to self-assess the tax amount. • *Null* value default is treated as *ACCRUE - Prorate By Item* setting.	X		X	

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix AP Skip Fully Paid U.S. Invoices	Set to one of the following for AP: • *Yes*: Skip fully paid U.S. AP invoices from Sabrix tax calculation. • *No*: Do not skip fully paid U.S. AP invoices from Sabrix tax calculation. • *NULL* value defaults to *NO*.	X		X	
eBTax: Sabrix AP Tax Expense Account Source	**This is a required setting.** Set to one of the following for AP: • *Tax* to use the GL account that is set up on the tax code for expense accounting. • *Item* if the GL account should be taken from the line item.	X		X	
eBTax: Sabrix AP Tax Grouping Rule	Defines how Project Accounting will be used when the Integration groups tax distributions for AP. • *DEFAULT* groups by the tax distribution account. • *PROJECT* groups by 6 default project fields. • *CUSTOM* groups by up to 10 user-definable project fields. For more information, see *Programming User Exits*.	X		X	
eBTax: Sabrix AP Tolerance Amount	A tolerance amount to determine at which point Sabrix should accrue tax. If tax is within the amount entered here, no accruals will be returned to the AP invoice. **Note**: A null or 0 value is acceptable if you do not wish to use tolerances.	X		X	

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix AP Tolerance Percent	A tolerance percentage to determine at which point Sabrix should accrue tax. If tax is within the percentage entered here, no accruals will be returned to the AP invoice. Example: enter *10* if you wish to use 10% as your threshold. **Note**: A null or 0 value is acceptable if you do not wish to use tolerances.	X		X	

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix AP Undercharge Shortpay Action	To handle Vendor Tax Undercharges outside the tolerance levels specified by the two tolerance profile options below, you can set this option to: • *ACCRUE - Prorate By Item:* Enables accrue shortpay functionality for incorrect vendor tax charges, prorated by itemline amounts. The invoice amount will not change, but invoice lines and distribution line entries are created as Sabrix Tax Accrual and Sabrix Tax Expense for the vendor undercharged tax amount. • *ACCRUE - Prorate By Tax:* Enables accrue shortpay functionality for incorrect vendor tax charges, prorated by individual tax percentages. The invoice amount will not change, but invoice lines and distribution line entries are created as Sabrix Tax Accrual and Sabrix Tax Expense for the vendor undercharged tax amount. • *OFF:* The invoice amount will not change and **NO** invoice lines and distribution line entries are created as Sabrix Tax Accrual and Sabrix Tax Expense for the vendor undercharged tax amount. **Use this option to eliminate negative tax accruals.** • *PARTIAL*: This will reduce the invoice total by the difference between the calculated and invoiced vendor tax amount, and Sabrix generated partial short pay entry lines are created for the vendor undercharged tax amount. • *TOTAL*: This will reverse the entire vendor tax amount, reduce the invoice amount by the vendor tax amount, and make entries to self-assess the tax amount. • *Null* value default is treated as *ACCRUE - Prorate By Item* setting.	X		X	

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix AP Write Distributions to Invoice?	Set to *Yes* or *No* to prevent Integration for Oracle R12 from creating AP Distribution Lines. • *YES*: Writes lines to the AP_DISTRIBUTION table. • *NO*: Does not write lines to the AP_DISTRIBUTION table. • *NULL* value behaves as a *Yes* value. **Note**: For more information on tax types, see *ONESOURCE Indirect Tax Determination Online Help*.	X		X	
eBTax: Sabrix AR Audit On Complete	Set to one of the following for AR: • *Yes*: AR Transactions are sent to Determination Audit only when the ARtransaction has been completed and has status of 'Complete'. • *No*: AR Transactions are sent to Determination Audit during the Integration tax call (usually when transaction is saved) . • *NULL* value defaults to *No*.	X	X		
eBTax: Sabrix Allow order to book with Severity 2 error	• *No*: The sales order will be prevented from booking until the severity 2 issue is resolved. • *Yes*: The sales order can be booked and fulfilled in spite of a severity 2 error. The error will need to be resolved before invoicing can occur but it will not hold up the order processing cycle. **Note**: The sales order must be Booked a second time in order for the order to be Booked with a severity 2 error, such as Unknown Sabrix Company.	X	X		

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix Allow Sabrix Table Truncate	• *Yes*: The 'Sabrix Log Maintenance' concurrent program will process the 'PURGE' action, and purge the records in the SABRIX_LOG table according to the specified parameters. • *No*: The 'Sabrix Log Maintenance' concurrent program will not process the 'PURGE' action. • *NULL*: value defaults to *No*.	X	X	X	
eBTax: Sabrix Calculate OM Document Tax	When the tax event is set to booking, set to one of the following for OM performance: • *Yes*: OM Transactions sent to the Determination for a tax call is 'Document at a Time' instead of line by line. Only valid if on Oracle OM 12.1.3. • *No*: Each sales order line will make a tax call via eBTax to the Determination. • *NULL*: value defaults to *No*. **Note**: If on Oracle 12.1.3 environment and have OM patched to the equivalent level as 12.2.x. set to No. For more information, see "Setting EBTAX: Sabrix Calculate Order Management Document Tax" in the *ONESOURCE Indirect Tax Integration for Oracle 12 Order to Cash User's Guide*.	X	X		
eBTax: Sabrix Calculate Tax on Purchase Orders	Set to one of the following for AP: • *Yes*: Allows Purchase Orders to be taxed by Sabrix. • *No*: Does NOT allow Purchase Orders to be taxed by Sabrix. • *NULL:* value defaults to *Yes*	X		X	

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix Company	**This is a required setting.** A value that maps **exactly** to an External Company ID set in the Determination for the desired company. This option can be set multiple times (by operating unit responsibility) depending on the complexity of your configuration. You must choose one of the options below: • The External Company ID in the Determination must be identical to the Oracle accounting flexfieldbalancing segment. • To use the accrual roll-up by company, the value entered in this field would be either BALANCE_SEGMENT or SEGMENT*_VALUE. The company value in Oracle AP>Setup>Options>Financial Options (T) Accounting Liability account field is used as the company value to send to Determination during the Payables tax call. See "Multiple Company Segments for a Single AP Invoice" in the *ONESOURCE Indirect Tax Integration for Oracle 12 Procure to Pay User's Guide*. For the other Integration your company is using, this field needs to be set at the Application level. This would include Order Management, Receivables, Purchasing and Payables.	X	X	X	

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix Compress Tax Lines	To summarize tax results instead of displaying detailed tax and distribution lines for U.S. transactions. • *Yes*: Tax results for each item line are compressed/summed into a single tax and distribution line. • *No*: Tax results for each item line will have multiple tax lines and distributions if multiple tax blocks are returned by *ONESOURCE Determination*. • *NULL:* value defaults to *No*	X	X	X	
eBTax: Sabrix Customer Number Source	Set to one of the following: • *Party:* Oracle Party Number and Oracle Party Name to be passed as the **Customer Number** and **Customer Name** fields, respectively, when setting up Determination Customer Exemption Certificates based on Oracle R12 functionality. • *Customer*: Oracle Customer Account Number and Oracle Account Description to be passed as the **Customer Number** and **Customer Name** fields, respectively, when setting up Determination Customer Exemption Certificates for Oracle R11i backward compatibility. • *Null*: Value defaults to Oracle Party Number and Party Name.	X	X		

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix Debug Level	Set to one of the following for SABRIX_LOG table: • *ALL* to write all debug messages. • *XML* to write only XML messages. **Note**: In order to generate the Sabrix XML Report or Sabrix Log Report, the profile option must be set to either ALL or XML prior to executing a transaction. • *OFF* to not write messages. • *USER* to write all debug messages. • W*ARNING* to write non-severe warning messages. • *TIMING* to write procedure entry and exit messages for AP transactions in the SABRIX_LOG table.	X	X	X	
eBTax: Sabrix Disable Tax Flow Creation	Set to one of the following for automatic creation of Tax to Rate flows: • *Yes*: Disables Automatic Tax Flow Creation from existing tax flow including Source tax flows associated to an ERP Tax Code. Transactions will error with no tax code found message. • *No*: Enables creating automatic tax flows. • *NULL:* value defaults to *No* **NOTE:** In order to suppress Source Flow creation the Profile Option eBTax: Sabrix Disable Tax Flow Creation must be set to 'Yes' prior to running the *sabrix_insert_config.sh* script.	X	X	X	

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix Enable Persistent Connection	Set to one of the following: • *Yes*: Enables persistent connection. • *No*: Does not enable persistent connection • *NULL:* value defaults to *No* **Note**: Set to No or leave null value when using *ONESOURCE Integration* for *Oracle* connected to *ONESOURCE Determination* in a Cloud environment.	X	X	X	
eBTax: Sabrix Enable Tax Calculation ByPass	Set to one of the following to enable bypassing tax calculation: • *Yes*: Allows bypassing tax calculation for online tax calculations. No tax will be calculated on the transactions. • *No*: Does not allow bypassing tax calculation for online tax calculations. Tax will be calculated on the transactions. • *NULL:* value defaults to *No* **Note 1:** Yes setting has to be used in conjunction with Bypass Calc User Exit to not have tax calculated. For more information, see ***BYPASSING TAX CALCULATIONS (page 128)*** **Note 2:** Yes setting takes precedence over existing profiles (i.e. eBTax: Sabrix Calculate Tax on Purchase Orders and eBTax: Sabrix O2C Tax Calculation)	X	X	X	

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix Hosted Unique Identifier	• Any non-null value. • Used to pre-pend to the External_Company_ID in the Sabrix_Invoice table, making the External_Company_ID unique across all Determination companies in a cloud environment. • The value will be reflected in the INDATA XML. • There is no default.	Only for Determination Cloud Environment	Only for Determination Cloud Environment	Only for Determination Cloud Environment	
eBTax: Sabrix O2C Tax Calculation	• *AR ONLY*: Tax is not calculated for OM transactions. • *AR_OM*: Tax is calculated for both OM and AR transactions. • *OFF*: Tax is *NOT* calculated for either OM or AR transactions. • *Null*: default value; which is AR_OM.	X	X		
eBTax: Sabrix Set HTTP Header SOAP Action	• *Yes*: The SOAP Action is set in the HTTP header. • *No*: The SOAP Action is not set in the HTTP header. • *Null*: Defaults to *No*.	X	X	X	

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix System Print Structure	This profile option is used in conjunction with **eBTax: Sabrix Debug Level**. • To print data structures set the value to *ON*. • *OFF* and any other value prevents the structures from printing to the sabrix_log table. **Note**: For O2C and P2P transactions the input transaction view values can be used for debugging. Set to *OFF*' in a production environment in conjunction with the profile option **eBTax: Sabrix Debug Level** set to *OFF*.	X	X	X	
eBTax: Sabrix Use Bill-To for Null Ship-To	Set to one of the following for AR: • *No:* Generates an error if no Ship-To is given. • *Yes*: Uses the Bill-To address as the Ship-To address. • *Allow:* Allows the transaction to go to Determination with no Ship-To. It will be filled in via precalc code or a Determination TransEditor. • *NULL* value is treated as *NO*.	X	X		

The following profile options are specific to Sabrix Extended Tax Codes.

Profile Option	Set To	O2C + P2P	O2C	P2P	Your Value
eBTax: Sabrix TCE: Delimiter	Enter the delimiter to use when concatenating the tax code with other elements. The options are: • *. (period):* Use a period to delimit tax code extension segments • null (no delimiter): Do not use any delimiter to separate tax code extension segments. • _ (underscore): Use an underscore to delimit tax code extension segments.	X	X	X	
eBTax: Sabrix TCE: Exempt	Enter the character(s) to denote that the tax code is for an exempt transaction.	X	X	X	
eBTax: Sabrix TCE: Tax Direction	Enter one of the following: • *N* disables the Tax Direction extension for tax codes and tax accounting. • *Y* enables the Tax Direction extension for tax codes and tax accounting.	X	X	X	
eBTax: Sabrix TCE: Tax Rate	Enter one of the following: • *N* disables the Tax Rate extension for tax codes and tax accounting. • *Y* enables the Tax Rate extension for tax codes and tax accounting.	X	X	X	
eBTax: Sabrix TCE: Tax Type	Enter one of the following: • *N* disables the Tax Type extension for tax codes and tax accounting. • *Y* enables the Tax Type extension for tax codes and tax accounting.	X	X	X	
eBTax: Sabrix Tax Accounting Level	**This is a required setting.** Select one of the following: • *BASIC* enables Sabrix basic tax accounting preferences. • *EXTENDED* enables Sabrix extended tax accounting preferences.	X	X	X	

CONFIGURING PROFILE OPTIONS AT THE ORGANIZATION LEVEL

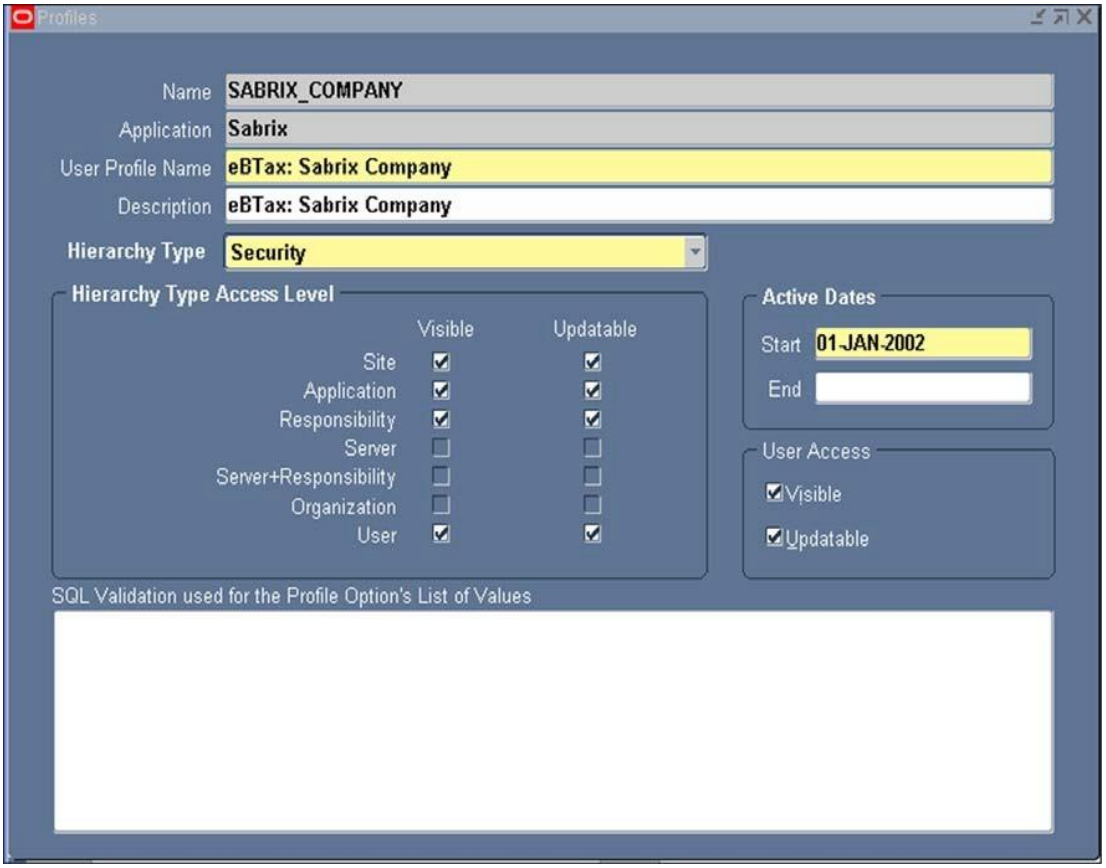
Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle Superuser	Successful completion of Integration installation	X	X	X

NOTE An *X* in the profile options tables indicates that the value is used for functionality within the corresponding business flow.

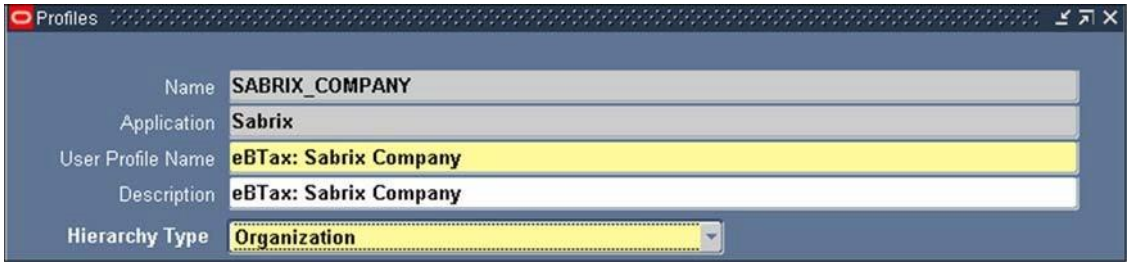
You can set profile options at various levels depending on how you want the applications to respond to various users. You can group and administer profile options based on the hierarchy type to which they are attached.

To set up **Sabrix_Company** profile at the **Org** level:

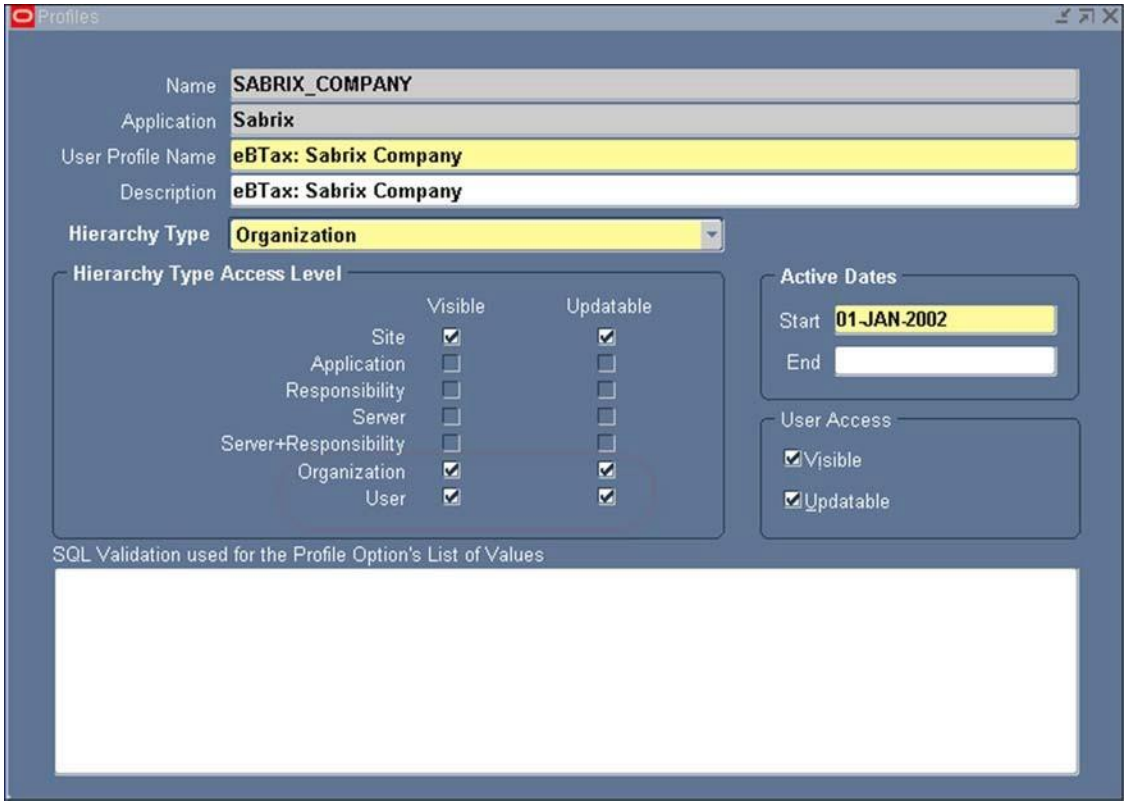
1. Select the **Application Developer** responsibility.
2. Navigate to **Profile**.
3. Query **Name** *SABRIX_COMPANY*.



4. In the **Hierarchy Type** drop-down menu, choose **Organization**.



5. In the **Hierarchy Type Access Level** section, select the **Visible** and **Updatable** checkboxes for **Organization** and **User**.



6. Save changes and close the form.

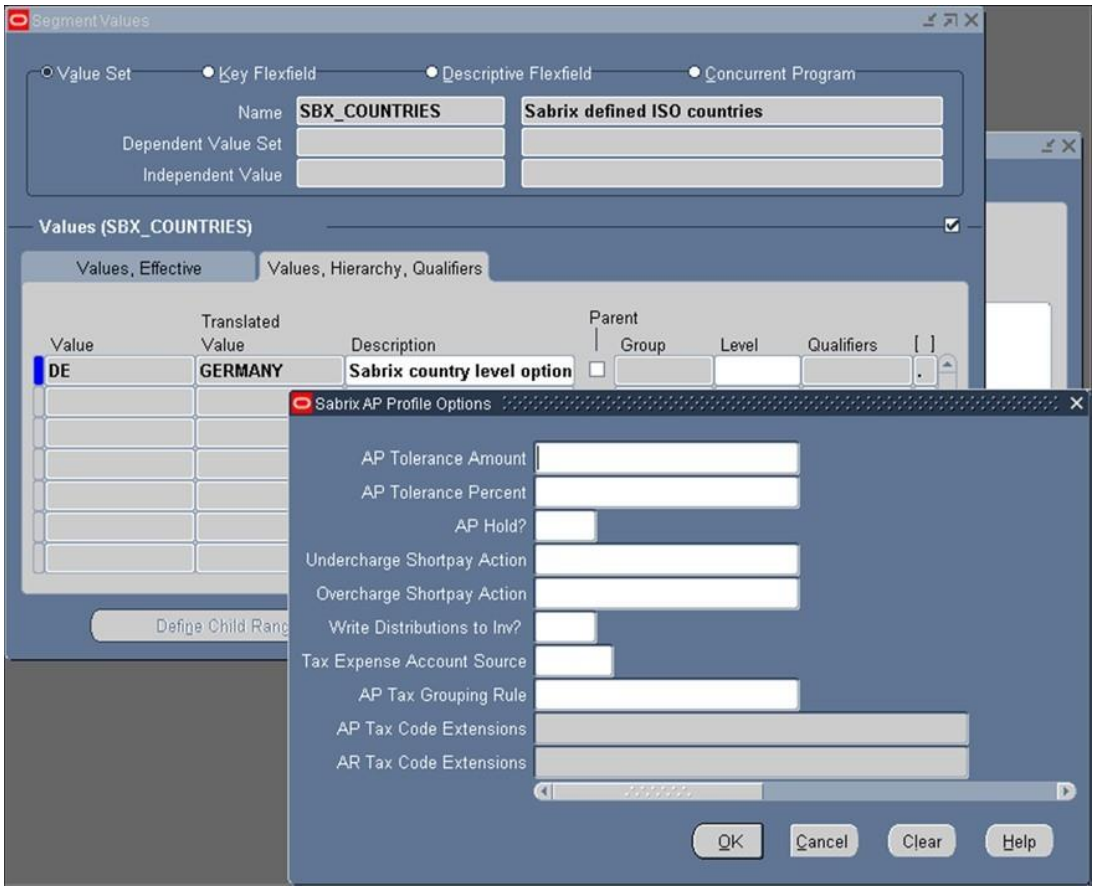
CONFIGURING PROFILE OPTIONS AT THE COUNTRY LEVEL

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle Superuser	Successful completion of Integration installation	X		X

NOTE An *X* in the profile options tables indicates that the value is used for functionality within the corresponding business flow.

You can override site level AP profile options with those set at the country level:

1. Select the **Payables Manager** responsibility.
2. Navigate to **Setup > Flexfields > Validation > Values.**
3. Find the value set *SBX_COUNTRIES*.Query the country to which you want to set the profile option.
4. Enter and save the profile option values for each profile.



CONFIGURING PROFILE OPTIONS AT THE STATE LEVEL

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle Superuser	Successful completion of Integration installation	X		X

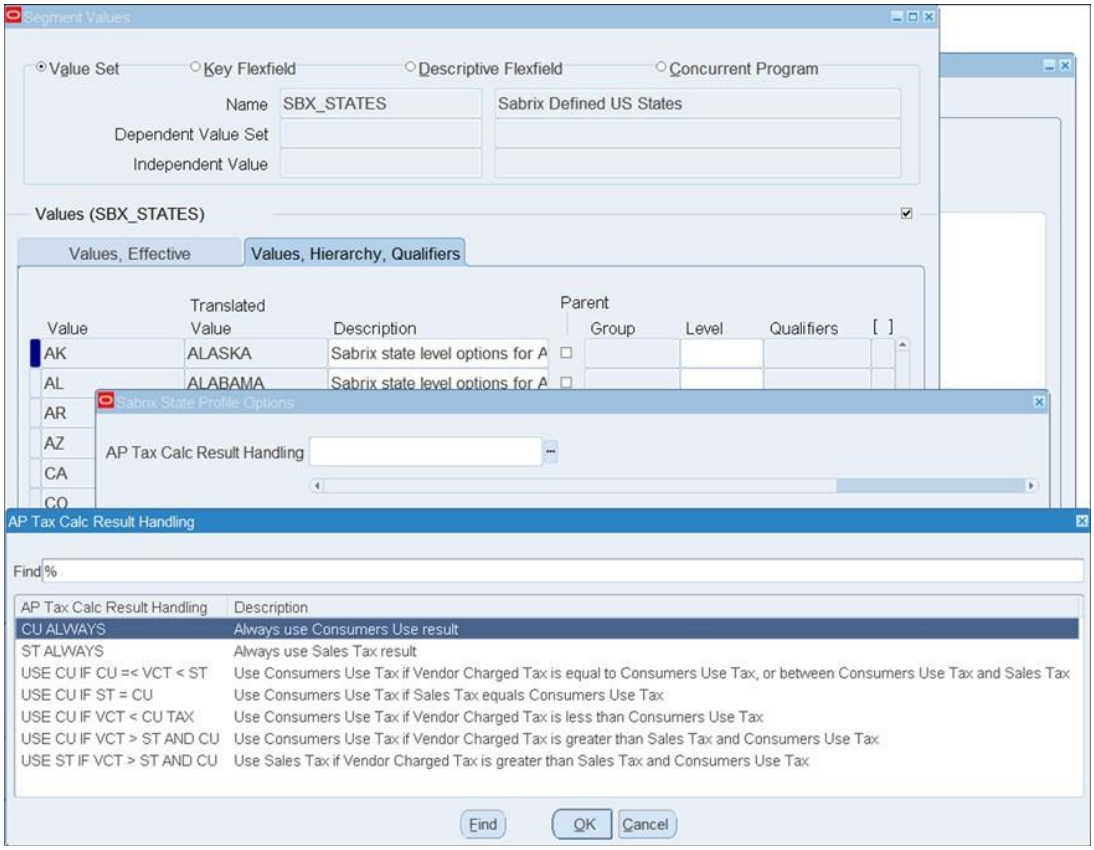
An *X* in the profile options tables indicates that the value is used for functionality within the corresponding business flow.

Currently, only the *AP Tax Calc Result Handling* profile option is available to set at the state level. See the *ONESOURCE Indirect Tax Integration for Oracle 12 Procure to Pay User Guide* for more information.

To set the AP profile option at the state level:

1. Select the **Payables Manager** responsibility.
2. Navigate to **Setup > Flexfields > Validation > Values.**
3. Find the value set *SBX_STATES*.
4. Query the state to which you want to set the profile option.

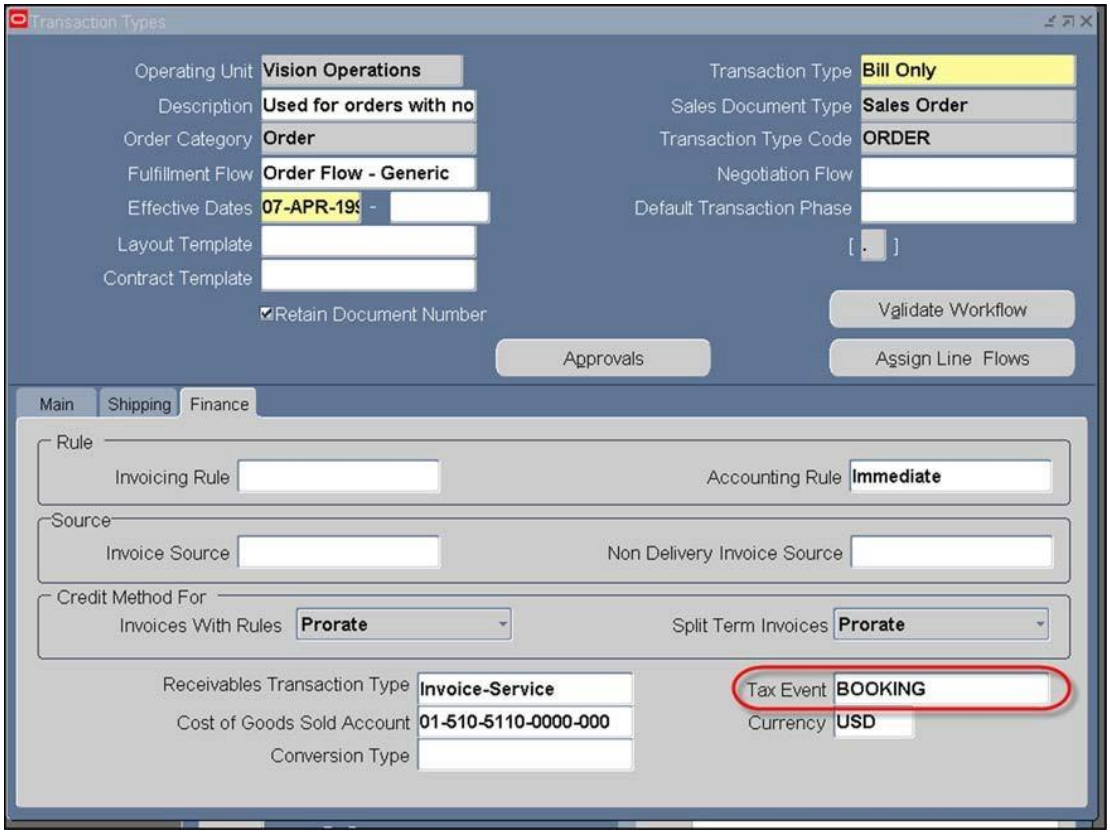
Enter and save the profile option values for each profile.



SETTING TAX EVENTS ON ORDER MANAGEMENT TRANSACTION TYPES

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle Superuser	Successful completion of Integration installation	R	R	

1. Select the **Order Management Superuser** responsibility.
2. Navigate to **Setup > Transaction Types > Define.**
3. Query each **Order Type** (Transaction Type Code = ORDER).
4. Click the **Finance** tab.
5. Set the **Tax Event** field to one of the following:
   - **Entering**: If you require tax after each item line is entered.
   - **Booking**: If you require a tax estimate on your order before fulfillment (*Recommended*).
   - **Invoicing**: If you only want tax called during invoicing.



DEFINING SABRIX AP HOLDS

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle Superuser	Successful completion of Integration installation	R	R	R

If the profile option **eBTax: Sabrix AP Hold?** is set to *NO* (indicating that holds are not to be placed on invoices), you may skip this task. This profile option drives the Sabrix Tax holds:

- System Tax holds to be placed on transactions when there is a Severity 2 error for the tax results.
- System Tax holds to be placed on transactions when the GL account is invalid or cannot be overridden.
- System Tax holds when there is an attempt to shortpay a paid invoice.
- An optional manual processing hold for US invoices that are out of tolerance.

Thomson Reuters recommends disabling all other tax holds.

Create the following invoice holds in Oracle Payables.

1. In **Payables Manager**, navigate to the **Invoice Hold and Release Names** window. Because companies have varying degrees of user responsibilities, this could be found using the navigation path in Payables as **Setup > Invoice > Hold and Release Names**.
2. Complete the **Invoice Hold and Release Names** form using the following attributes (the value in the **Name** column must be exactly as outlined below):

Name	Description (80 characters or less)	Type	Accounting Allowed	Manual Release Allowed
Tax Process Error Hold	Sabrix encountered an error processing tax; Resolve and resubmit.	Invoice Hold Reason	No (leave checkbox null)	No (leave checkbox null)
Tax Account Hold	Sabrix encountered an Invalid Tax Code/GL Account error. Resolve and resubmit.	Invoice Hold Reason	No (leave checkbox null)	No (leave checkbox null)
Tax Determination Hold	Sabrix Determination tax call encountered errors. Resolve and resubmit.	Invoice Hold Reason	No (leave checkbox null)	Yes (leave checkbox checked)
Tax Process Release	The Sabrix Tax Process released the hold for reprocessing.	Invoice Release Reason	No (leave checkbox null)	No (leave checkbox null)
Tax Release Determination	Sabrix manually resolved and released Determination hold.	Invoice Release Reason	No (leave checkbox null)	No (leave checkbox null)
Tax Shortpay Hold	The amount calculated by Sabrix to shortpay exceeds the AMT remaining to be paid	Invoice Hold Reason	No (leave checkbox null)	No (leave checkbox null)
Tax Shortpay Release	The Sabrix Shortpay Hold has been resolved manually.	Invoice Release Reason	No (leave checkbox null)	No (leave checkbox null)
Tax Tolerance Hold	Sabrix Tolerance threshold exceeded; Resolve and resubmit.	Invoice Hold Reason	No (leave checkbox null)	No (leave checkbox null)
The following hold is optional. Configure it if you want to manually review every US transaction that has been identified as requiring a shortpay or accrual for the difference. Please see the *ONESOURCE Indirect Tax Integration for Oracle 12 Purchases User's Guide* for more information.				
US Out of Tolerance	Sabrix US out of tolerance hold. Manually correct or review,release and resubmit	Invoice Hold Reason	No (leave checkbox null)	Yes

3. Save the form.

Set up the **US Out of Tolerance Hold** only if you want to review every invoice before an accrual or shortpay (depending on your profile settings) occurs. To resubmit, you must remove the hold and set the status of line 1 back to *R* (Ready to process) or *A* for (Accept). See the *ONESOURCE Indirect Tax Integration for Oracle 12 Procure to Pay User's Guide* for more detail about how to use this hold.

BYPASS TCA GEOGRAPHY HIERARCHY VALIDATIONS

ONESOURCE Indirect Tax Integration no longer requires having the Oracle Trading Community Architecture (TCA) Geography Name Referencing (GNR) program run to validate new address combinations for transaction tax calls.

However, by default using standard Oracle SYNCHRONIZE API functionality still requires the TCA GNR program to be executed for Tax Partners.

ONESOURCE Indirect Tax Integration is using the standard Oracle SYNCHRONIZE API to populate ONESOURCE Determination with the AR Autoinvoice number and not the transaction id number during the COMPLETE-INVOICE event.

For customers who want to bypass the TCA GNR program, plus want to use the ONESOURCE Indirect Tax Integration feature to populate ONESOURCE Determination with the AR Autoinvoice number, four steps are required:

- Apply Oracle patch 21696441 to bypass TCA GNR hierarchy validations.
- Create manually an Oracle eBTax **Tax Reporting Type Code** named *"VALIDATE GNR FOR PARTNERS".*
- Associate the **Tax Reporting Type Code** to the appropriate **Party Tax Profile** for the Operating Unit (OU) which was used for the eBTax Regime subscription.
- Set the profile option "**eBTax: Sabrix AR Audit on Complete**" to *"Yes".*

Once all of the above steps have been completed, you will not have to run the GNR for the specified OU.

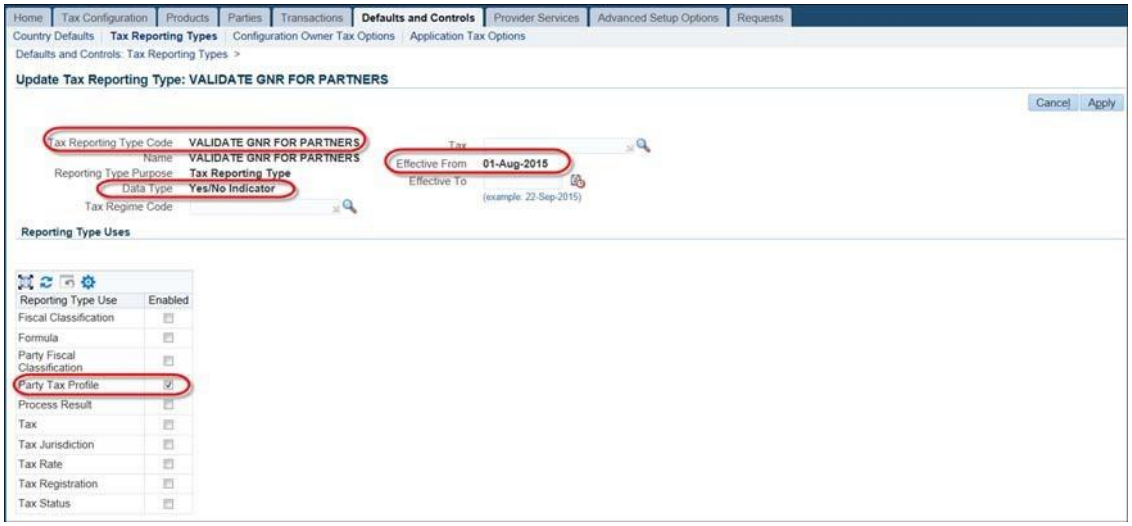
The same setup has to be done for all the Party Tax Profile OUs using ONESOURCE Indirect Tax Integration even though the GNR may not be defined.

CREATE TAX REPORTING TYPE CODE

1. Select the **Tax Managers** responsibility.
2. Navigate to **Defaults and Controls > Tax Reporting Types.**
3. Click the **Create** button.
4. Enter into the **Tax Reporting Type Code** field *"VALIDATE GNR FOR PARTNERS".*

The Tax Reporting Type Code must match exactly (without the quotes) "*VALIDATE GNR FOR PARTNERS".*

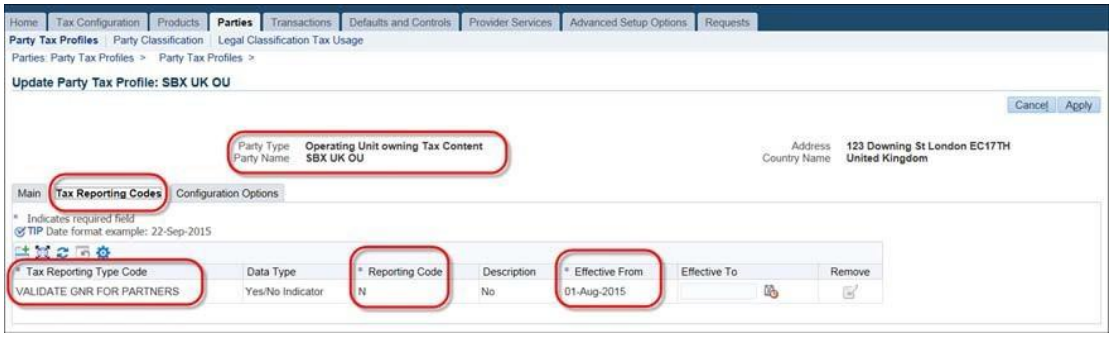
5. Enter into the **Date/Type** field *"Yes/No Indicator".*
6. Enter the **Effective From** date (Date should be on or after the Effective From date used on other eBTax configurations like Regimes).
7. Check the box for **Reporting Type Uses** for **Party Tax Profile.**
8. Click Apply to save the record.



ASSOCIATE PARTY TAX PROFILE TO TAX REPORTING TYPE

Determine which Operating Unit (OU) will not have GNR program run.

1. Select the **Tax Managers** responsibility.
2. Navigate to **Parties > Party Tax Profiles.**
3. Query Party Tax Profile of OU/LE and Party Name (used for regime subscription).
4. Click on the **Update Tax Profile** icon.
5. Go to the **Tax Reporting Codes** tab
6. Click on **Add Another Row** icon.
7. Choose from list of value the **Tax Reporting Type Code** of *"VALIDATE GNR FOR PARTNERS".*
8. The **Date/Type** field *"Yes/No Indicator"* defaults from the selected **Tax Reporting Type Code** of *"VALIDATE GNR FOR PARTNERS".*
9. Enter **Reporting Code** as "N".
10. Enter the **Effective From** date (Date should be on or after the Effective From date used on other eBTax configurations like Regimes and Tax Reporting Type Codes).
11. Click Apply to save the record.


Home
Tax Configuration
Products
Parties
Transactions
Defaults and Controls
Provider Services
Advanced Setup Options
Requests
Party Tax Profiles
Party Classification
Legal Classification Tax Usage
Parties: Party Tax Profiles >
Party Tax Profiles >
Update Party Tax Profile: SBX UK OU
Cancel
Apply
Party Type
Operating Unit owning Tax Content
Party Name
SBX UK OU
Address
123 Downing St London EC17TH
Country Name
United Kingdom
Main
Tax Reporting Codes
Configuration Options
* Indicates required field
TIP Date format example: 22-Sep-2015
* Tax Reporting Type Code
Data Type
* Reporting Code
Description
* Effective From
Effective To
Remove
VALIDATE GNR FOR PARTNERS
Yes/No Indicator
N
No
01-Aug-2015

SYNCHRONIZING ZONE NAMES

Who	Prerequisite	O2C + P2P	O2C	P2P
Superuser	• Successful Integration installation • All configurations completed	R	R	R

Determination and Oracle E-Business Suite can use slightly different spellings for several zone names. If these names are not synchronized, Determination will not correctly process an incoming transaction containing the unknown zone. You can synchronize zone names in one of the following ways:

- Create a Zone Alias to map the Oracle zone name to the equivalent Determination zone name.
- Create a TransEditor in Determination to change address data from Oracle.
- Update the zone name within Oracle TCA to match the expected Determination name.

For example, you could have locations with an address in the Nunavut Territories of Canada. The location record contains the full name of “Nunavut Territories” yet Determination contains the zone name “Nunavut.” When a transaction using one of these addresses is processed using the Integration, Determination does not recognize the address from Oracle. An error for the transaction is generated.

You can choose among the following options to handle this:

- Create a Zone Alias in Determination to map “Nunavut Territories” to “Nunavut.”
- Create a TransEditor in Determination to replace “Nunavut Territories” with “Nunavut.”
- Update the customer master address in Oracle for each customer in “Nunavut Territories” to “Nunavut.”

Contact Customer Support for more information.

POST INSTALLATION: TECHNICAL TOPICS

This chapter contains information about the following *required* procedures:

- ***STORING CUSTOM .SQL SCRIPTS***
- ***VERIFYING THE INSTALLATION (page 111)***

These sections describe *optional* procedures:

- ***CUSTOMIZING THE INTEGRATION (page 118)***
- ***PROGRAMMING USER EXITS (page 121)***
- ***PERFORMANCE TUNING (page 130)***

STORING CUSTOM .SQL SCRIPTS

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle DBA	• Successful Integration installation • All configurations completed	R	R	R

A directory hierarchy (known as the *XXIDT* top or *$XXIDT_TOP*) needs to be built to store the *SBXAPACCR.sql* (Sabrix AP Accrual Report), *SBXTRXREG.sql* (Sabrix Transaction Register for AR), *SBXLOGRPT.sql* (Sabrix Log Report), *SBXNTFRPT.sql* (Sabrix New Tax Flows Report), and *SBXADRRPT.sql* (GL Tax Account Discrepancies For Sabrix Tax Rate Codes) reports.

This directory should be built under the *APPL_TOP* or $*APPL_TOP* directory and should look like *$APPL_TOP/xxidt/12.0.0/sql*.

After the directory structure has been created, the custom applications environment file must be modified to include the *XXIDT_TOP* path. The custom applications environment file is usually of the form *customCONTEXT_NAME.env* or *customSID_hostname.env* (for example, *customVIS_xxidtdev.env*). If the file doesn't exist, a new file will need to be created, owned by the applications operating system user.

For example:

If *$APPL_TOP = /oracle/VIS/apps/apps_st/appl, Oracle SID = VIS* and *hostname = xxidtdev*, then the file */oracle/VIS/apps/apps_st/appl/customVIS_xxidt.env* should be modified/created with the following lines:

```
XXIDT_TOP=/oracle/VIS/apps/apps_st/appl/xxidt/12.0.0
export XXIDT_TOP
```

Once the structure has been completed, save *SBXAPACCR.sql*, *SBXTRXREG.sql, SBXLOGRPT.sql, SBXNTFRPT.sql, and SBXADRRPT.sql* in the *$XXIDT_TOP/sql* directory.

A sub folder in the installation zip file contains the sql files used for concurrent processing, such as those listed above. These files must be uploaded to the Application Tier (i.e. the system that hosts the Oracle Application Home).

To upload the report files (SBX****.sql) to the Oracle Application server, open a cygwin window (using a bash shell):

- **cd^/cygdrive/c/c** (return)
- **cd^Install/Sabrix**... (return)
- **cd^/Reports/** (return) *cd to the directory you extracted the installation files

Transfer the files using sftp (secured file transfer protocol).

1. **sftp *yourapplicationserveruser@yoursystemname*** *(i.e.*applvis@pdxsasin085)
2. Enter application server password
3. Within the sftp program, cd to the $XXIDT_TOP/sql directory (i.e. cd^/opt/oracle/VIS/apps/apps_st/appl/xxidt/12.0.0/sql/)
4. **put^*** (this will transfer all the files at once)
5. **Exit** the sftp program

For the cygwin (bash shell) line entries above for **cd ^** and **put^,** you do not need to type the caret **(^)** character**.** The caret **(^)** character indicates a blank space.

VERIFYING THE INSTALLATION

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle DBA logged in as *APPS*.	• Successful Sabrix Integration installation • All configurations completed	R	R	R

You will use two procedures to verify the installation:

- Run the HealthCheck shell script
- Run the Connection Tester

HEALTHCHECK SCRIPT

This script verifies that all required objects to run the ONESOURCE Indirect Tax Integration are installed properly. If objects are missing, they will be clearly identified as *Missing*. If objects have become obsolete, they will be tagged as *Removed*. The *sabrix_healthcheck.sh* needs to be run separately for each operating unit identified in the *sabrix_stage_config.dat* file. See the ***REGIME TO RATE FLOW PARAMETERS (page 27)*** for details.

Complete the following to execute the shell script:

1. Ensure that the two prerequisites are completed.
2. Invoke a Bash shell on your client machine.

This installation script is designed to run in a Bash shell.

3. Run the following script from the directory in which you unzipped the ONESOURCEIndirect Tax Integration for Oracle 12 files.

```
$ ./sabrix_healthcheck.sh
```

4. When prompted, enter the Operating Unit ID to Check. You can find the relevant Operating Unit IDs in the *sabrix_stage_config.dat* file, see ***REGIME TO RATE FLOW PARAMETERS (page 27)*** for details.

sabrix_healthcheck Prompt Name	Example Value
Enter the Operating Unit ID to Check	XXXX

This script prompts you for the Operating Unit ID to Check only. All other parameters are from your sabrix_install.env environment file, if the file exists. See ***EDITING SABRIX_INSTALL.ENV (page 50)*** for details.

5. The example below is the beginning section of the output after the *sabrix_healthcheck.sh* script has started:

```
$ ./sabrix_healthcheck.sh
 +-------------------------------------------------------------------------------+
 +   Install script          : [sabrix_healthcheck.sh]
 +   RUN DATE                : [08-Nov-2010 13:26:31]
 +   LOG FILE NAME           : [sabrix_healthcheck0811101326.log]
 +-------------------------------------------------------------------------------+
 +-------------------------------------------------------------------------------+
  Enter the Operating Unit ID to check
7857
 +-------------------------------------------------------------------------------+
 + PARAMETERS:
 + Sabrix   SCHEMA NAME                                  : [SABRIX]
 + APPS     SCHEMA NAME                                  : [APPS]
 + Database                                              : [loein13]
 + Check O2C                                            : [Y]
 + Check P2P                                            : [Y]
 + Operating Unit                                       : [7857]
 +-------------------------------------------------------------------------------+
 +-------------------------------------------------------------------------------+
 + Checking username AND passwords
 + Connecting TO SCHEMA APPS
USER is "APPS"
 +-------------------------------------------------------------------------------+
 + Connected TO schema successfully .. PARAMETERS OK
 +-------------------------------------------------------------------------------+
 +-------------------------------------------------------------------------------+
 + Running Script sabrix_healthcheck.sql...
```

6. If you are prompted for any other parameters than Operating Unit ID, then you may have chosen **not** to use the *sabrix_install.env* file for the primary installation script *sabrix_install_integration.sh*; instead, you provided the parameters at execution. If you chose this path, proceed to answer the prompts listed below:

Healthcheck Prompt Name	Example Value for running both O2C and P2P
Sabrix Adapter schema name	*SABRIX*
Oracle Applications schema name	*APPS*
Check O2C? [Y]	*Y*
Check P2P? [Y]	*Y*
Org ID	XXXX

If you want to run a business flow individually set the parameter values of 'Check O2C' or 'Check P2P' as appropriate.

7. Review the *sabrix_healthcheck.sh* log. Confirm the *sabrix_healthcheck.sh* log displays the message “Sabrix Oracle 12 Healthcheck is complete. Installation is healthy.” Review and correct all “Missing” items until the *sabrix_healthcheck.sh* log returns this healthy message at the end of the file.
8. Repeat steps 2-4 for each operating unit identified in the *sabrix_stage_config.dat* file, by entering the Org ID parameter when running the healthcheck script.

The following is an example of a healthcheck that is missing items:

```
...All Oracle12 seeded regime-to-rate view records found.
--------------------------------------------------------------------------------
--------- GL Tax Accounts attached to Tax Rate Codes ----------
Checking GL Tax Accounts have been attached to Tax Rate Codes...
--------------------------------------------------------------------------------
O2C Workflow         # Tax Rate Codes    # GL Tax Liability      # GL Tax Expense    # GL Tax Liability      # GL Tax Expense

Tax Regime                      Found        Accounts Found        Accounts Found      Accounts Missing      Accounts Missing

SBX_REGIME_JP                       1                     0                     0                     1                     1

P2P Workflow         # Tax Rate Codes    # GL Tax Liability      # GL Tax Expense    # GL Tax Liability      # GL Tax Expense

Tax Regime                      Found        Accounts Found        Accounts Found      Accounts Missing      Accounts Missing

SBX_REGIME_JP                       1                     0                     0                     1                     1

...2 Regimes missing accounts for Operating Unit[7860].
--------------------------------------------------------------------------------
------------------------------------------------------
------------------ Missing Components ------------------
------------------------------------------------------
---------- Note: results may be truncated ------------
Sabrix Oracle12 Healthcheck error.  The following expected objects are missing:

...1 Regimes missing accounts for Operating Unit[7860]; Workflow[O2C].
...1 Regimes missing accounts for Operating Unit[7860]; Workflow[P2P].


+------------------------------------------------------------------------------+
+   Script : [healthcheck.sql]
+   Complete ..
+------------------------------------------------------------------------------+
```

The following is an example of a healthy healthcheck:

```
Row for ZX.ZX_TAXES_TL                                          : found
Row for ZX.ZX_STATUS_B                                          : found
Row for ZX.ZX_STATUS_TL                                         : found
Row for ZX.ZX_JURISDICTIONS_B                                   : found
Row for ZX.ZX_JURISDICTIONS_TL                                  : found
Row for ZX.ZX_RATES_B                                           : found
Row for ZX.ZX_RATES_TL                                          : found
Row for APPLSYS.FND_LOOKUP_VALUES                               : found
Row for ZX.ZX_ID_TCC_MAPPING_ALL                                : found
...All Sabrix Oracle12 default regime-to-rate flow records found.
--------------------------------------------------------------------------------
--------- GL Tax Accounts attached to Tax Rate Codes ----------
Checking GL Tax Accounts have been attached to Tax Rate Codes...
--------------------------------------------------------------------------------
...All Tax Rate Codes for Operating Unit have Tax accounts.
--------------------------------------------------------------------------------
------------------------------------------------------
-  Sabrix Oracle12 Healthcheck complete.     -
-                 Installation is healthy    -
------------------------------------------------------

+------------------------------------------------------------------------------+
+   Script : [healthcheck.sql]
+   Complete ..
+------------------------------------------------------------------------------+
```

RUNNING THE CONNECTION TESTER

The Connection Tester validates communication from Oracle to Determination and back again.

The Determination must be initially configured before running the Connection Tester. For more information see ***REGISTERING THE ONESOURCE INDIRECT TAX INTEGRATION FOR ORACLE 12 (page 56)*** in this guide.

While logged into SQL*Plus as *APPS* user, run the following commands:

```
SQL> set serveroutput on size 100000
SQL> exec Sabrix_util.contest('Integration Name','External_Company_Id')
```

Connection Tester Parameters	Values	O2C + P2P	O2C	P2P
Integration Name	The exact value you entered when running the *sabrix_register_integration.sql* script. For more information, see *Executing the Sabrix Integration Installation Scripts*.	R Enter IntegrationO2C during 1st execution and IntegrationP2P in 2nd.	R Enter *IntegrationO2C*	R Enter *IntegrationP2P*
External_Company_Id	The value you set on the Oracle Profile Option **eBTax: Sabrix Company.** See *Configuring Profile Options at the Site Level*.	R	R	R

If you are doing a combined installation of P2P (Procure to Pay) and O2C (Order to Cash), run the script for both O2C and P2P.

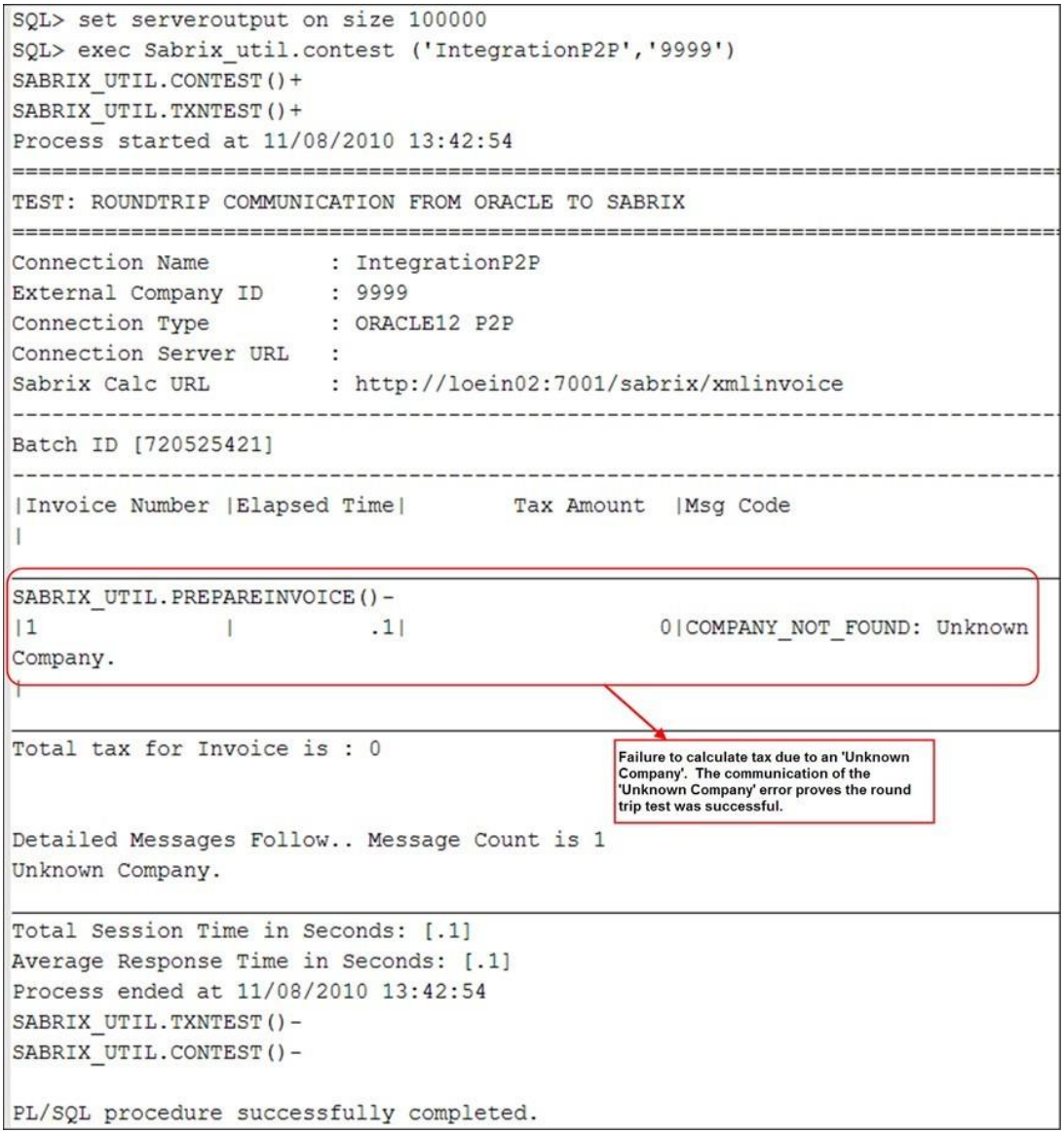
Two examples of Connection Tester output for O2C are shown below. In this example, the fact that tax results were returned by Determination indicates a successful round trip from Oracle to Determination.

```
SQL> set serveroutput on size 100000
SQL> exec Sabrix_util.contest ('IntegrationO2C','01')
SABRIX_UTIL.CONTEST()+
SABRIX_UTIL.TXNTEST()+
Process started at 11/08/2010 13:38:58
================================================================================
TEST: ROUNDTRIP COMMUNICATION FROM ORACLE TO SABRIX
================================================================================
Connection Name         : IntegrationO2C
External Company ID     : 01
Connection Type         : ORACLE12 O2C
Connection Server URL   :
Sabrix Calc URL         : http://loein02:7001/sabrix/xmlinvoice
--------------------------------------------------------------------------------
Batch ID [720501937]
--------------------------------------------------------------------------------
|Invoice Number |Elapsed Time|        Tax Amount  |Msg Code
|_______________________________________________________________________________
SABRIX_UTIL.PREPAREINVOICE()-
|1              |        4.06|                950|
|_______________________________________________________________________________
Total tax for Invoice is : 950

Line Taxes Follow.. Line Tax Count is 3
.       TAX_LINE_ID             : [1]
.       COMMENT                 : [ORACLE[USWA]]
.       TAXABLE_BASIS           : [10000]
.       TAX_AMOUNT              : [650]
.       TAX_RATE                : [.065]
.       ZONE_LEVEL              : [State]
.       ZONE_NAME               : [WASHINGTON]
.       TAX_LINE_ID             : [2]
.       COMMENT                 : [ORACLE[USWA]]
.       TAXABLE_BASIS           : [10000]
.       TAX_AMOUNT              : [0]
.       TAX_RATE                : [0]
.       ZONE_LEVEL              : [District]
.       ZONE_NAME               : [KIRKLAND]
.       TAX_LINE_ID             : [3]
.       COMMENT                 : [ORACLE[USWA]]
.       TAXABLE_BASIS           : [10000]
.       TAX_AMOUNT              : [300]
.       TAX_RATE                : [.03]
.       ZONE_LEVEL              : [City]
.       ZONE_NAME               : [KIRKLAND]

Detailed Messages Follow.. Message Count is 1
--------------------------------------------------------------------------------
Total Session Time in Seconds: [4.11]
Average Response Time in Seconds: [4.11]
Process ended at 11/08/2010 13:39:02
SABRIX_UTIL.TXNTEST()-
SABRIX_UTIL.CONTEST()-

PL/SQL procedure successfully completed.
```

Tax results confirm the round trip test was successful.

This example shows an error message generated by the ONESOURCE Indirect Tax Determination. While a tax result was not returned, the Connection Tester again validated round-trip communication between Oracle and Determination.

The error indicates a configuration error (a company name mismatch).



CUSTOMIZING THE INTEGRATION

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle Superuser	Completed Post Installation Oracle Configurations	OPTIONAL	OPTIONAL	OPTIONAL
Determination Superuser	Successful Integration installation	OPTIONAL	OPTIONAL	OPTIONAL

The standard, out-of-the-box installation of the Integration allows you to pass basic transaction data to Determination, which calculates the tax and returns the correct result to Oracle Order Management, Oracle Receivables, Oracle Purchasing and Oracle Payables.

You can customize both the ONESOURCE Indirect Tax Integration for Oracle 12 and Oracle release 12 so you can more effectively pass additional data to Determination to trigger specialized calculations or enable additional auditing or reporting capabilities.

These customizations can be straightforward or complex, and require you to thoroughly analyze your business processes and requirements. You can contact Professional Services to assist you when you are ready to implement them.

The following sections provide an overview of some of these special customizations:

- ***PASSING CUSTOM ATTRIBUTES TO ONESOURCE INDIRECT TAX DETERMINATION (page 118)***
- ***MANAGING PRODUCTS AND PRODUCT CATEGORIES (page 119)***
- ***IDENTIFYING TRANSACTION SOURCE IN THE AUDIT TABLES (page 119)***
- ***PROCESSING SERVICES TRANSACTIONS (page 120)***

PASSING CUSTOM ATTRIBUTES TO ONESOURCE INDIRECT TAX DETERMINATION

Determination's Input XML specification allows the source system to specify up to 50 invoice-level and up to 50 line-level custom attributes. These attributes are primarily passed into the system for two reasons:

- To enable reporting on an aspect of the enterprise not covered by the standard data elements (such as project accounting data, and data by corporate division, by store, etc.).
- To enable triggering of custom processing using Determination's TransEditor functionality. Each TransEditor contains a user-defined condition (matching a specified Element/Value pattern found in the Input XML) that can trigger one or more actions.

The Integration architecture contains two user exits (one on input, one on output) that enable you to specify the data you want to pass to and return from custom attributes. See ***PROGRAMMING USER EXITS (page 121)***.

After you configure custom attributes, you may also need to configure TransEditors in Determination. See ONESOURCE Indirect Tax Determination *Online Help* for more information.

Attribute elements 41-50 are reserved for ONESOURCE Indirect Tax Integration use. You can configure these elements to trigger TransEditors or to perform reporting but they should not be modified by pre- or post-calculation user exits. See *APPENDIX E: ORACLE SOURCE MAPPED TO DETERMINATION XML DATA (page 167)* for more information.

MANAGING PRODUCTS AND PRODUCT CATEGORIES

Only products and product categories that need special tax treatment need to be maintained in Determination's product exceptions list.

For tax calculations outside the United States, Determination provides a complete list of product categories and exceptions. For US tax calculations, these products and exceptions must be entered manually. However, because only exceptions need to be entered, the vast majority of products do not need to be maintained in the system.

Determination also provides product mapping, in which source system products (such as those maintained in your Oracle implementation) can be mapped to Sabrix product exceptions.

Because product mappings are always implementation-specific, your company's tax department needs to configure these mappings in Determination before passing product codes to Determination as part of a transaction.

See ONESOURCE Indirect Tax Determination *Online Help* for more information about product mapping.

IDENTIFYING TRANSACTION SOURCE IN THE AUDIT TABLES

Determination creates a unique key to identify transactions in the Audit database. The components of the key are the audit fields HOST_SYSTEM, CALLING_SYSTEM_NUMBER and UNIQUE_INVOICE_NUMBER. These are populated by the Integration as follows:

Audit Field	Description
HOST_SYSTEM	NAME value of the V$DATABASE view.
CALLING_SYSTEM_NUMBER	application_id.
UNIQUE_INVOICE_NUMBER for AP/PO	Payables: transaction id from Oracle (invoice_id), company ID is appended to the Oracle invoice_id when the **Tax: Sabrix Company** profile option is set to either BALANCE_SEGMENT or SEGMENTx_VALUE Purchasing: po_headers_id from po_headers_all table.
UNIQUE_INVOICE_NUMBER for AR/OM	Receivables: customer_Trx_id from the ra_customer_Trx_All table. Order Management: Header_id from OE_Order _Headers table.

PROCESSING SERVICES TRANSACTIONS

Determination can calculate tax for both goods and services. However, the determination of whether a transaction falls under one of these categories is implementation-specific. Consequently, the ONESOURCE Indirect Tax Integration for Oracle 12 architecture makes the assumption that all transactions passed to Determination for calculation are goods transactions.

To indicate that a transaction is a services transaction:

- Identify which transaction elements indicate that the transaction should be taxed as a service. For example, you might determine that there are three distinct fields within Oracle that must be satisfied to indicate that a transaction is a service.
- Make use of the user exit and Determination custom attributes to pass the fields that indicate that the transaction is a service to Determination.
- Create a TransEditor to check for the custom attributes in the incoming XML, and to set the TRANSACTION_TYPE to the desired services type if the attributes and matching values are found.

See ONESOURCE Indirect Tax Determination *Online Help* for Custom Attributes and TransEditor information and the *Programmer's Topics* in ONESOURCE Indirect Tax Determination *Online Help* for XML Element information, including <TRANSACTION_TYPE>.

PROGRAMMING USER EXITS

Who	Prerequisite	O2C + P2P	O2C	P2P
• Oracle DBA • Database Developer	• Successful Integration installation • Completed post installation Oracle Configurations	OPTIONAL	OPTIONAL	OPTIONAL

The Integration architecture allows for a variety of user exits that enable you to specify which data you want to pass to and return from a transaction.

- PreCalc (After Input mapping but before calling Determination)
  - The PreCalc (input) exit might be used to look up product codes or alternate Determination companies to use when assessing tax.
- PostCalc (After calling Determination but before Output mapping)
  - The PostCalc (output) exit might be used to modify output results or to map Determination results to Oracle.
- ByPassCalc (Bypass eBTax CALCULATE API)
  - The ByPassCalc exit might be used to bypass tax determination for aspecific Transaction Type or a particular Operating Unit.
  - The ByPassCalc exit may also be used to modify initialization variables such as Company ID or Host System.

The following User Exit topics are discussed below:

- ***PASSING CUSTOM ATTRIBUTES (page 122)***
- ***DRIVING PROJECT ACCOUNTING USING CUSTOM ATTRIBUTES (page 126)***
- ***USING ORACLE CCIDS IN PLACE OF THE TAX CODE DEFAULTS FOR AP (page 127)***
- ***HANDLING CUSTOM TAX CODES (page 127)***
- ***ENABLING LOGGING TO THE INTEGRATION-OWNED LOG FILE (page 128)***
- ***BYPASSING TAX CALCULATIONS (page 128)***

PASSING CUSTOM ATTRIBUTES

Determination's Input XML specification allows the source system to specify up to 50 invoice-level and up to 50 line-level custom attributes. These attributes are primarily passed into the system for the following reasons:

- To enable reporting on an aspect of the enterprise not covered by the standard data elements (such as by corporate division, by store, etc.).
- To enable triggering of custom processing using Determination's TransEditor functionality. Each TransEditor contains a user-defined condition (matching a specified Element/Value pattern found in the Input XML) that can trigger one or more actions.
- To override the values of standard elements populated by Oracle and the Integration.

Certain Invoice and Line Attributes are reserved for the Integration. The following tables detail those attributes reserved for integration:

- Oracle Payables ***(page 123)***
- Oracle Purchasing ***(page 124)***
- Oracle Receivables ***(page 125)***

Once you configure custom attributes, you may also need to configure TransEditors in Determination. See ONESOURCE Indirect Tax Determination *Online Help* for more information.

Custom Attributes Used by Oracle Payables

Invoice Level Attributes	
User_Element_Attribute41	ap_invoices_all.invoice_id
User_Element_Attribute42	ap_invoices_all.vendor_site_id
User_Element_Attribute43	hr_locations_all.location_id
User_Element_Attribute44	View name: AP_VIEW
User_Element_Attribute45	ap_invoices_all.org_id
User_Element_Attribute46	Reserved
User_Element_Attribute47	Reserved
User_Element_Attribute48	Reserved
User_Element_Attribute49	application_id (always 200)
User_Element_Attribute50	Reserved
Line Level Attributes	
User_Element_Attribute31	ap_invoice_distributions_all.project_id
User_Element_Attribute32	ap_invoice_distributions_all.task_id
User_Element_Attribute33	ap_invoice_distributions_all.expenditure_item_date
User_Element_Attribute34	ap_invoice_distributions_all.expenditure_organization_id
User_Element_Attribute35	ap_invoice_distributions_all.pa_quantity
User_Element_Attribute36	ap_invoice_distributions_all.expenditure_type
User_Element_Attribute41	ap_invoice_distributions_all.line_id
User_Element_Attribute42	ap_invoices_all.vendor_site_id
User_Element_Attribute43	hr_locations_all.location_id
User_Element_Attribute44	tax_amount
User_Element_Attribute45	for matched invoices, po_line_locations_all.line_location_id
User_Element_Attribute46	ap_invoice_distributions_all.distribution_id
User_Element_Attribute47	ap_invoice_distributions_all.line_type_lookup_code
User_Element_Attribute48	Reserved
User_Element_Attribute49	Reserved
User_Element_Attribute50	when audit is deleted, 'INVOICE IN TOLERANCE'

Custom Attributes used by Oracle Purchasing

Invoice Level Attributes	
User_Element_Attribute41	transaction_id
User_Element_Attribute42	bill_to_party_number
User_Element_Attribute43	related_doc_trx_id
User_Element_Attribute44	event_type_code
User_Element_Attribute45	Reserved
User_Element_Attribute46	Reserved
User_Element_Attribute47	Reserved
User_Element_Attribute48	Reserved
User_Element_Attribute49	application_id
User_Element_Attribute50	event_class_mapping_id
Line Level Attributes	
User_Element_Attribute41	transaction_line_id
User_Element_Attribute42	product_id
User_Element_Attribute43	applied_from_line_id
User_Element_Attribute44	null
User_Element_Attribute45	ship_to_party_name
User_Element_Attribute46	ship_to_party_number
User_Element_Attribute47	entity_code
User_Element_Attribute48	transaction_line_type
User_Element_Attribute49	tax_event_class_code
User_Element_Attribute50	tax_event_type_code

Custom Attributes used by Oracle Receivables

The following table lists the attributes used when implementing the Integration:

Invoice Level Attributes	
User_Element_Attribute41	Header ID
User_Element_Attribute42	Bill To Customer ID
User_Element_Attribute43	Previous Transaction Header ID
User_Element_Attribute44	Tax Info Rec - attribute 10
User_Element_Attribute45	Reserved
User_Element_Attribute46	Reserved
User_Element_Attribute47	Reserved
User_Element_Attribute48	Reserved
User_Element_Attribute49	Application ID
User_Element_Attribute50	Deleted by Oracle Receivables Flag
Line Level Attributes	
User_Element_Attribute41	Line ID
User_Element_Attribute42	Inventory Item ID
User_Element_Attribute43	Previous Line ID
User_Element_Attribute44	Location ID
User_Element_Attribute45	Ship To Customer Name
User_Element_Attribute46	Ship To Customer Number
User_Element_Attribute47	Tax Line Number
User_Element_Attribute48	If Freight Line is from OM, where the List_Line_Type_Code = Freight_Charge, then the Charge_type_code is populated.
User_Element_Attribute49	VAT Tax ID
User_Element_Attribute50	Tax Code

After you configure your exits, you may also need to configure TransEditors in Determination. See ONESOURCE Indirect Tax Determination *Online Help* for more information.

DRIVING PROJECT ACCOUNTING USING CUSTOM ATTRIBUTES

Oracle standard functions enable the association of AP tax distributions with Oracle Projects. To enable this association, the Integration can use custom attributes to pass product and project data with the invoice. This enables Oracle Projects to accurately associate expenses to projects.

You can choose whether and how to implement Oracle Projects accounting. Depending on the choice you make when setting the profile option **eBTax: Sabrix AP Tax Grouping Rule**, the Integration groups tax distributions for posting to Oracle by:

- Account (no project accounting - **DEFAULT** setting).
- Project (based on the values found in line attributes 31-36 above: **PROJECT** setting).
- A user-defined custom grouping (based on the values found in line attributes 31-40: **CUSTOM** setting).

Setting this profile option does not affect tax calculations, only how tax distributions are posted back to Oracle. For examples, see the *ONESOURCE Indirect Tax Integration for Oracle 12 Procure to Pay User's Guide*.

Other than setting the profile option, you do not need additional configurations to implement either account- or project-based grouping. If you want to implement a custom grouping for tax distributions, you need to populate line attributes 31-40 with the desired Oracle field data using either the pre- or post-calculation user exit.

USING ORACLE CCIDS IN PLACE OF THE TAX CODE DEFAULTS FOR AP

By default, the AP accounting for tax distributions in Oracle is driven by the related Tax Code returned by Determination. This Tax Code is generated from the Authority/Rule combination in Determination that was used to generate the tax result. In some instances, more data is needed to provide for correct and complete accounting in Oracle. For example, a company may have multiple stores in the same tax jurisdiction and a requirement to track the tax being reported for each store.

You can populate the Oracle Code Combination ID (CCID) for a transaction during the post-calculation user exit. To do so, evaluate transaction data, and then populate two elements in the SABRIX_LINE_TAX table with the desired CCIDs:

- ERP_ACCOUNT_ID: used to specify the CCID for the tax expense/non-recoverable.
- ERP_OFFSET_ACCOUNT_ID: used to specify the CCID for the tax liability/recoverable.

When the Integration returns the results structure to Oracle, it evaluates the elements above first. If they are populated, tax distributions are posted to the accounts specified therein. If they are not populated, tax distribution posting is driven by the Determination-returned Tax Code.

HANDLING CUSTOM TAX CODES

If you are you using custom Tax Codes (i.e. tax codes other than those returned by the ERP_TAX_CODE element from Determination), you will need to write post-calc user exits in order to translate the Determination tax code into the appropriate tax code for your implementation.

The post-calc needs to be written in the *SABRIX_ADAPTER_TABLE_POSTCALC* stored procedure. This user exit is executed after the results are returned from Determination and before the results are mapped back into Oracle Financials. In this procedure, you will need to directly update the ERP_TAX_CODE in the results staging table, SABRIX_LINE_TAX.

ENABLING LOGGING TO THE INTEGRATION-OWNED LOG FILE

The following procedure enables developers to log messages to the SABRIX_LOG table in the pre-calc and post-calc user exits. The procedure specification is as follows:

```
Sabrix_log_pkg.put_line
    ( p_procedure      IN VARCHAR2
    , p_location       IN VARCHAR2
    , p_message        IN VARCHAR2
    , p_output_type    IN VARCHAR2 DEFAULT 'LOG'
    , p_severity       IN VARCHAR2 DEFAULT '0'
    , p_invoice_number IN VARCHAR2 DEFAULT g_invoice_number
    , p_request_id     IN NUMBER   DEFAULT g_request_id
    )
```

Usage:

```
  SABRIX_LOG_PKG.put_line
    ( p_procedure      => 'SABRIX_ADAPTER_TABLE_PRECALC'
    , p_location       => 'Startup'
    , p_message        => 'Setting IS_AUDIT to ''N'''
    , p_output_type    => 'LOG'
    , p_severity       => sabrix_log_pkg.k_info
    )
```

You should create log messages whenever values are modified in the pre- and post-calcs.

BYPASSING TAX CALCULATIONS

The BypassCalc User Exit allows for the return 1 of 2 return codes; NORMAL or BYPASS. NORMAL will result in the continuation of the tax calculation while BYPASS will return control back to EBTax without executing the tax calculation.

If the BypassCalc User Exit returns BYPASS, it is expected that it populates the EBTax Error_Status and Error Msg. If a successful status is to be returned to EBTax, the User Exit will need to set the ERROR_STATUS to 'S' and leave the ERROR_DEBUG_MSG_TABLE empty. If an error condition is returned to EBTax, the User Exit will need to set the ERROR_STATUS to 'E' and put an appropriate error message in the ERROR_DEBUG_MSG_TABLE.

BypassCalc User Exit is used in conjunction with Yes setting on Profile Option eBTax: Sabrix Enable Tax Calculation ByPass.

Setting Global Variables for Bypassing Tax Calculations

The following global variables can be updated in the BypassCalc User Exit:

Variable Name	Format
g_use_bt_null_st	varchar2(10)
g_invoice_freight_as_line	varchar2(1)
g_allow_order_book	varchar2(1)
g_customer_number_source	varchar2(20)
g_sabrix_calc_o2c_tax	varchar2(255)
g_om_document_tax	varchar2(1)
g_sabrix_audit_on_complete	varchar2(1)
g_Sabrix_Company	varchar2(100) := null
g_o2c_quote_appl	number
g_instance_name	varchar2(100)
g_host_system	varchar2(200)
g_us_invoice	BOOLEAN
g_sabrix_tolerance	number
g_sabrix_tolerance_pct	number

PERFORMANCE TUNING

Who	Prerequisite	O2C + P2P	O2C	P2P
Oracle DBA	• Successful Integration installation • Completed post installation Oracle Configurations	R	N/A	R

The Integration Procure to Pay uses a view to aggregate information about an AP invoice during processing. This view is optimized for performance when using the Integration. However, large numbers of existing Payables invoices may degrade performance. If performance is degraded, or if you expect to process large volumes of invoices, you should create the following indexes in the Payables (AP) schema to sustain the desired performance levels.

The following sections explain the four indexes for the Integration:

- ***INDEX ON SABRIX TAX STATUS IN AP_INVOICE_DISTRIBUTIONS_ALL (this page)***
- ***FUNCTION BASED INDEX IN AP_INVOICE_DISTRIBUTIONS_ALL (page 131)***
- ***INDEX ON CANCELLED_DATE IN AP_INVOICES_ALL (page 131)***
- ***INDEX ON PARENT_REVERSAL_ID IN AP_INVOICE_DISTRIBUTIONS_ALL (page 132)***

INDEX ON SABRIX TAX STATUS IN AP_INVOICE_DISTRIBUTIONS_ALL

Create an index on the **Sabrix Tax Status** attribute column in the AP_INVOICE_DISTRIBUTIONS_ALL table. This column is a descriptive flexfield (DFF) defined at ONESOURCE Indirect Tax Integration for Oracle 12 installation time to store the processing status of AP invoices. Review your DFF setups on the Invoice Distribution DFF definition in Oracle Payables for the correct column name.

Below is a sample script that can be used to create this index. The DBA **must** modify the values shown here to reflect the installation's individual processing and storage requirements.

```
CREATE                  INDEX                  AP.AP_INV_DIST_TAXSTATUS
ONAP.AP_INVOICE_DISTRIBUTIONS_ALL(&ENTER_TAX_STATUS_DFF_ATTRIBUTE_HERE)
PCTFREE 10 INITRANS 2 MAXTRANS 255
 STORAGE(
 INITIAL 16 K
 NEXT 16 K
 MINEXTENTS 1
 MAXEXTENTS 50
 PCTINCREASE 50
 FREELISTS 1
 FREELIST GROUPS 1
 BUFFER_POOL DEFAULT
 )
TABLESPACE USER_DATA
```

FUNCTION BASED INDEX IN AP_INVOICE_DISTRIBUTIONS_ALL

Performance of the Sabrix view SABRIX_AP_INVOICE_V can be improved by creating a function-based index. See the following index descriptions for further information.

Review *HANDLING SKEWED TAX STATUS INFORMATION IN TABLE AP_INVOICE_DISTRIBUTIONS_ALL (page 132)* for additional information.

1. Create a function-based index against the newly-enabled predicate as shown here.

```
CREATE INDEX ap.sabrix_tax_status2
ON AP_INVOICE_DISTRIBUTIONS_ALL
(
DECODE(&TAX_STATUS_DFF_ATTRIBUTE, 'R','Y','E','Y','A','Y',NULL) ASC
)
PCTFREE 10 INITRANS 2 MAXTRANS 255
STORAGE(
INITIAL 16 K
NEXT 16 K
MINEXTENTS 1
MAXEXTENTS 50
PCTINCREASE 50
FREELISTS 1
FREELIST GROUPS 1
BUFFER_POOL DEFAULT
)
TABLESPACE <AP Index tablespace>
/
```

2. Analyze the index.

```
analyze index ap.sabrix_tax_status2 compute statistics:
```

INDEX ON CANCELLED_DATE IN AP_INVOICES_ALL

Create an index on the cancelled_date column in the AP_INVOICES_ALL table.

Below is a sample script you can use to create this index. The DBA **must** modify the values shown here to reflect the installation's individual processing and storage requirements.

```
CREATE  INDEX  AP.AP_INVOICES_TAXCANCEL
ON  AP.AP_INVOICES_ALL(CANCELLED_DATE)
PCTFREE  10  INITRANS  2  MAXTRANS 255
 STORAGE(
 INITIAL 16 K
NEXT 16 K
 MINEXTENTS 1
 MAXEXTENTS 50
 PCTINCREASE 50
       FREELISTS 1
 FREELIST GROUPS 1
 BUFFER_POOL DEFAULT
 )
TABLESPACE USER_DATA
```

INDEX ON PARENT_REVERSAL_ID IN AP_INVOICE_DISTRIBUTIONS_ALL

Create a non-unique index on the parent_reversal_id column in the AP_INVOICE_DISTRIBUTIONS_ALL table.

Below is a sample script that can be used to create this index:

```
CREATE INDEX AP_INVOICE_DISTRIBUTIONS_SBX2
ON AP_INVOICE_DISTRIBUTIONS_ALL (PARENT_REVERSAL_ID)
PCTFREE 10 INITRANS 2 MAXTRANS 255
   STORAGE(
       INITIAL 16 K
      NEXT 16 K
       MINEXTENTS 1
       MAXEXTENTS 50
       PCTINCREASE 50
       FREELISTS 1
       FREELIST GROUPS 1
       BUFFER_POOL DEFAULT
       )

TABLESPACE <AP Index tablespace>
```

HANDLING SKEWED TAX STATUS INFORMATION IN TABLE AP_INVOICE_DISTRIBUTIONS_ALL

In an environment including uniformly distributed data and up-to-date statistics, the cost-based approach normally makes accurate estimations of the cost of executing a particular statement. However, when the data is not uniformly distributed, the optimizer cannot accurately estimate the selectivity of a query.

Over time, the column defined as the Sabrix Tax Status flag can become skewed because of the relatively low number of daily invoices requiring tax calculations compared with the total number of processed transactions. This in turn impacts the optimizer, and can result in missed index selection and poor performance. There are several ways to avoid this issue, some of which are described below. Review these carefully so you can understand the optimal solution within a given environment.

Function Based Index. See *FUNCTION BASED INDEX IN AP_INVOICE_DISTRIBUTIONS_ALL (page 131)*.

Histogram. Oracle allows you to store histograms describing the data distribution of a particular column when that column does not have a uniform data distribution. Histograms are stored in the dictionary and computed by using the ANALYZE command on a particular column. Therefore, there is a maintenance and space cost for using histograms. You should only compute histograms for the Sabrix Tax Status flag once the data has become skewed over time.

The DBMS_STATS builtin can be used to maintain histograms and can easily be included within the normally scheduled stats collection tasks. An example script is below. Substitute &TAX_STATUS_DFF_ATTRIBUTE for the specific attribute being used.

```
BEGIN
 DBMS_STATS.gather_table_stats
 (ownname    => 'AP',
 tabname    =>  'ap_invoice_distributions_all',
 method_opt => 'FOR COLUMNS (&TAX_STATUS_DFF_ATTRIBUTE)',
 DEGREE     => '2',
 CASCADE    => TRUE
 );
END;
/
```

Session variable OPTIMIZER_INDEX_COST_ADJ. When rows making up the table AP_INVOICE_DISTRIBUTIONS_ALL contain highly skewed values for the Sabrix Tax Status flag, the optimizer may begin choosing sub-optimal full table scans over index access paths, even with high volume tables where single column indexes exist. One way to tune optimizer behavior is by utilizing OPTIMIZER_INDEX_COST_ADJ. This session parameter lets you adjust plan access path selection to be more or less index-friendly.

Decreasing the default value of 100 can force indexes to look more appealing to the optimizer in a CBO environment. This can dramatically improve index hit rates on previously long running queries. Great care must be taken when adjusting this setting however, as index selection is not always preferable to full table scans. Results from modifying this or other session level parameters can vary from query to query, making a holistic testing approach critically important.

Contact Technical Support for specific concerns about the ONESOURCE Indirect Tax Integration for Oracle 12 performance.

ACCESS CONTROL LIST

When the Oracle application does not return tax and receives the following Error: &GENERIC_TEXT (GENERIC_TEXT=), the Oracle application is not communicating with ONESOURCE Indirect Tax Determination.

In order to resolve the error, drop the Access Control List (ACL) for OracleEBX.xml and re-create the ACL. Below is a script from Oracle Support to drop and re-create the ACL:

```
BEGIN
DBMS_NETWORK_ACL_ADMIN.drop_acl
(acl => 'OracleEBS.xml');
commit;
END;

BEGIN
DBMS_NETWORK_ACL_ADMIN.CREATE_ACL('OracleEBS.xml','APPS access for UTL_HTTP',
'APPS', TRUE, 'connect');
DBMS_NETWORK_ACL_ADMIN.ASSIGN_ACL('OracleEBS.xml','*');
Commit;
END;

BEGIN
DBMS_NETWORK_ACL_ADMIN.add_privilege (
acl => 'OracleEBS.xml',
principal => 'APPS',
is_grant => TRUE,
privilege => 'resolve',
position => NULL,
start_date => NULL,
end_date => NULL);
COMMIT;
END;
```

APPENDIX A: CONCURRENTPROCESSES

This appendix provides an overview of the following concurrent processes.

Regime to Rate Flow processes:

- ***SABRIX SETUP REGIME (page 136)***
- ***SABRIX SETUP TAX (page 137)***
- ***SABRIX SUBSCRIBE OPERATING UNIT (page 138)***
- ***SABRIX ADD ACCOUNT TO TAX RATE CODE (page 139)***

Other concurrent processes:

- ***SABRIX LOG REPORT (page 141)***
- ***SABRIX XML REPORT (page 142)***
- ***SABRIX TAX PROCESS AND SABRIX AP TAX THREAD (page 143)***
- ***SABRIX LOG TABLE MAINTENANCE (page 144)***
- ***SABRIX TRANSACTION MANAGER FOR AP AND AR (page 146)***
- ***SABRIX NEW TAX FLOWS REPORT (page 147)***
- ***GL ACCOUNT DISCREPANCIES FOR SABRIX TAX RATE CODES REPORT (page 149)***
- ***SABRIX SETUP RECOVERY RATES (page 151)***

CREATING REGIME TO RATE FLOW USING SABRIX CONCURRENT PROCESSES

The following section describes how to create only one regime to rate flow for a fresh installation, or if you are adding a new, individual regime to rate flow for an existing complete installation.

If you want to create multiple regimes, multiple tax flows, subscribe the regime to rate flow to multiple Operating Units, or add different general ledger tax accounts to multiple Tax Rate Codes, see ***REGIME TO RATE FLOW PARAMETERS (page 27)*** and ***SETTING UP REGIME TO RATE FLOW (page 60)***

The Sabrix Application Data Group must be configured before you can execute these concurrent process. See *DEFINING THE SABRIX APPLICATION DATA GROUP (page 59)*.

CONCURRENT PROCESSES

This section describes the following:

- Sabrix Setup Regime
- Sabrix Setup Tax
- Sabrix Subscribe Operating Unit
- Sabrix Add Account to Tax Rate Code

SABRIX SETUP REGIME

This is the first step for creating a functioning Regime to Rate flow. This creates a single Regime and is typically used to add a Regime after a setup of multiple regimes. See ***SETTING UP REGIME TO RATE FLOW (page 60)***.

To create a single regime, execute the concurrent process **Sabrix Setup Regime**:

1. Select the **Tax Managers** responsibility.
2. Navigate to **Requests > Other > Requests > Run** and select a single request.
3. In the name field, select the concurrent process **Sabrix Setup Regime**.
4. Enter the parameters from the following table:

Prompt	Description
Country	Country code for Regime.
Currency	Tax currency code.
Exchange Rate Type	The source of an exchange rate used to perform foreign currency conversion; for example, *User Defined*, *Spot*, or *Corporate rate*. This is the **CONVERSION_TYPE** field from the GL_DAILY_CONVERSION_TYPES table.
Install O2C	Create an O2C Tax Flow? *Yes* or *No.*

Prompt	Description
Install P2P	Create a P2P Tax Flow? *Yes* or *No.*
Start Date	Tax Effective Start Date (DD-MMM-YYYY)

5. Click **Submit**.
6. Review the concurrent request log file to confirm successful completion.

It is recommended that you end date the non-critical non-Sabrix regimes and their associated flow records. These records could negatively affect performance. The records should be end dated in the following tables:

- ZX_REGIMES_B
- ZX_TAXES_B
- ZX_STATUS_B
- ZX_JURISDICTIONS_B
- ZX_RATES_B

SABRIX SETUP TAX

This is the second step for creating a functioning Regime to Rate flow.This creates a single tax flow and is typically used to add a tax flow to a completed installation. See ***SETTING UP REGIME TO RATE FLOW (page 60)***.

Configure a Single Tax Flow

To create a single tax flow, execute the concurrent process **Sabrix Setup Tax***:*

1. Select the **Tax Managers** responsibility.
2. Navigate to **Requests > Other > Requests > Run** and select a single request.
3. In the name field, select the concurrent process **Sabrix Setup Tax**.
4. Enter the parameters from the following table:

Prompt	Description
Tax Regime	Tax Regime Name. Choose from value set.
ERP Rate Code	Sabrix ERP Tax Code. Choose from value set.
Tax Direction	Specifies input/outp values for VAT recovery identification. Choose from value set.
Install O2C	Create an O2C Tax Flow? *Yes* or *No*
Install P2P	Create a P2P Tax Flow? *Yes* or *No*
Start Date	Tax Effective Start Date (*DD-MMM-YYYY*) Must be on or after Regime Start Date

5. Click **Submit**.
6. Review the concurrent request log file to confirm successful completion.

SABRIX SUBSCRIBE OPERATING UNIT

This is the third step for creating a functioning Regime to Rate flow. This subscribes a single Operating Unit to a single Regime. See ***SETTING UP REGIME TO RATE FLOW (page 60)***.

To subscribe an operating unit to a regime in Oracle e-Business Tax, run the *Sabrix Subscribe Operating Unit* concurrent process. The concurrent process subscribes the defined operating unit to the defined regime.

You must execute the concurrent process once for each combination of operating unit and regime.

To run the **Sabrix Subscribe Operating Unit** concurrent process:

1. Select the **Tax Managers** responsibility.
2. Navigate to **Requests > Other > Requests > Run** and select a single request.
3. In the name field, select the concurrent process **Sabrix Subscribe Operating Unit**.
4. Enter the parameters from the following table:

Prompt	Description
Tax Regime	Tax Regime Name the operating unit will be subscribed. Choose from value set.
Operating Unit	Operating unit to subscribe to Tax Regime selected above. Choose from value set.
Subscribe for O2C	Subscribe an O2C Tax Flow? *Yes* or *No.*
Subscribe for P2P	Subscribe to a P2P Tax Flow? *Yes* or *No.*
Start Date	Tax Effective Start Date (DD-MMM-YYYY): Date must be on or after Regime Start Date.

5. Click **Submit**.

You must execute the concurrent process once for each combination of operating unit and regime.

6. Review the concurrent request log file to confirm successful completion.

SABRIX ADD ACCOUNT TO TAX RATE CODE

This is the fourth and final step for creating a functioning Regime to Rate flow. This assigns all the Tax Rate Codes for the selected ERP Tax Code to a specific operating unit's general ledger tax accounts. This process can be run to change existing general ledger tax accounts for all the Tax Rate Codes for the ERP Tax Code specified.

You must execute the concurrent process once for each combination of regime, operating unit regime and tax rate code.

To run the **Sabrix Add Account to Tax Rate Code** concurrent process:

1. Select the **Tax Managers** responsibility.
2. Navigate to **Requests > Other > Requests > Run** and select a single request.
3. In the name field, select the concurrent process **Sabrix Add Account to Tax Rate Code**.
4. Enter the parameters from the following table:

Prompt	Description
Tax Regime	Tax Regime Name the Tax Rate Code belongs. Choose from value set.
Operating Unit	Operating Unit for the the general ledger accounts. Choose from value set.
ERP Tax Rate Code	Choose from value set.
Workflow	*Procure to Pay* or *Order to Cash.* Choose from value set.
Tax Direction	Specifies input/ouput values for VAT recovery identification. Choose from value set.
Recoverable/Liability Acct	Enter a valid General Ledger account combination for the Operating Unit. Required for O2C and P2P Tax Rate Codes For P2P Tax Rate Codes, see *Gather Regime, Tax Code and Operating Unit Values (page 28)* for *P2P Recoverable/Liability Account.*
Tax Expense Acct	Enter a valid General Ledger account combination for the Operating Unit Not Required for O2C Tax Rate Codes. For P2P Tax Rate Codes, see *Gather Regime, Tax Code and Operating Unit Values (page 28)* for *P2P Tax Expense Account.*

5. Click **Submit**.
6. Review the concurrent request log file to confirm successful completion.

Thomson Reuters recommends that you maintain the same general ledger accounts for all the Tax to Rate Flows created for the ERP Tax Code.

To manually enter general ledger accounts for a new Tax Rate Code or manually change the general ledger accounts for an existing Tax Rate Code, see the "Tax Rate Codes and General Tax Accounts" chapter in the *ONESOURCE Indirect Tax Integration for Oracle 12 Order to Cash User Guide* or *ONESOURCE Indirect Tax Integration for Oracle 12 Procure to Pay User Guide*.

OTHER SABRIX CONCURRENT PROCESSES

This section describes the following processes:

- Sabrix Log Report
- Sabrix XML Report
- Sabrix Tax Process and Sabrix AP Tax Thread
- Sabrix Log Table Maintenance
- Sabrix Transaction Manager for AP and AR
- Sabrix New Tax Flows Report
- GL Accounts Discrepancies for Sabrix Tax Rate Codes Report

SABRIX LOG REPORT

The **Sabrix Log Report** gathers information from the SABRIX_LOG table. User can run this report to review data about specific transactions from the SABRIX_LOG table. For messages that are not errors, the **eBTax: Sabrix Debug Level** profile option must be set to a value other than *OFF* prior to executing the transaction.

To run the **Sabrix Log Report**:

1. Select the **Tax Managers** responsibility.
2. Navigate to **Requests > Other > Requests > Run** and select a single request.
3. In the name field, select the concurrent process **Sabrix Log Report**.
4. Enter the following optional parameters:

Prompt	Description
Log Date Start	Choose from list of values.
Log Date End	Choose from list of values.
Workflow	Choose from list of values.
Batch ID	Choose from list of values.
Document Number	Choose Transaction number from list of values.
Concurrent Request ID	Choose from list of values.
Severity	Choose from list of values.

5. Click **Submit**.
6. Review the concurrent request log file to confirm successful completion.

SABRIX XML REPORT

The **Sabrix XML Report** displays XML InData and XML OutData from the SABRIX_LOG table. The **eBTax: Sabrix Debug Level** profile option must be set to a value *ALL* or *XML* prior to executing the transaction.

To run the **Sabrix XML Report**:

1. Select the **Tax Managers** responsibility.
2. Navigate to **Requests > Other > Requests > Run** and select a single request.
3. In the name field, select the concurrent process **Sabrix XML Report**.
4. Enter optional parameters from the following table:

Prompt	Description
XML Type	Choose from list of values.
Workflow	Choose from list of values.
Document Number	Choose Transaction number from list of values.
Concurrent Request ID	Choose from list of values.
Sabrix Batch ID	Choose from list of values.
Start Time	Choose from list of values.
End Time	Choose from list of values.

5. Click **Submit**.
6. Review the concurrent request log file to confirm successful completion

SABRIX TAX PROCESS AND SABRIX AP TAX THREAD

The Integration for payables transactions runs as a concurrent process within Oracle Payables. This concurrent process is usually set up as a scheduled job, or it can also be run manually.

You can run the Integration concurrent processes in one of two configurations:

- Single-threaded, by using the **Sabrix AP Tax Thread**.
- Multi-threaded, by using the **Sabrix Tax Process** to spawn multiple Sabrix AP Tax Threads and manage processing workload.

Thomson Reuters recommends using the multi-threaded configuration for improved data delivery and automatic load balancing.

For details about **Sabrix Tax Process**, see the section “Sabrix Tax Process for AP” in the *ONESOURCE Indirect Tax Integration for Oracle 12 Procure to Pay User’s Guide*.

SABRIX LOG TABLE MAINTENANCE

The Integration is fully integrated with Oracle's error handling capabilities. The *Sabrix Log Table Maintenance* process can be used to clean and purge the SABRIX_LOG table. This process is not operating unit-specific because it is used for Sabrix tables, which are not operating unit-specific. A significant amount of detail for every transaction is included in the SABRIX_LOG table, including:

- Input value from Oracle
- Output value from Determination
- Process dialog and interim values
- Transaction source
- Profile option values
- XML Input and Output to and from Determination

All transaction logging data is stored in the SABRIX_LOG table, including the appropriate workflow (O2C or P2P). The information can be cleaned or purged using the **Sabrix Log Table Maintenance** concurrent program.

The log messages for all transaction data are written to an Oracle database table, SABRIX_LOG table, located in the Sabrix schema. The Integration can maintain the logs resulting from Determination processes. The Integration writes logs to tables in the Sabrix schema.

For details on logging when processing Oracle Payables transactions, see “Configuring Tax Process Logging and XML Settings” in the ONESOURCE Indirect Tax Integration for Oracle 12 Procure to Pay User Guide.

When processing Receivables, Order Management, and Purchasing transactions, log data is controlled by the profile option **eBTax: Sabrix Debug Level** settings of *ALL, OFF, USER, XML,* and *WARNING*, in conjunction with the **eBTax: Sabrix System Print Structure** setting**.** See the ***CONFIGURING PROFILE OPTIONS (page 79)*** section, for more information.

Timing statistics for AP transactions are stored in the SABRIX_LOG table if the profile option **eBTax: Sabrix Debug Level** is set to *Timing.* This setting will write procedure entry and exit messages for AP transactions in the SABRIX_LOG table.

For details on logging when processing Oracle Payables transactions, see the section “Configuring Sabrix Tax Process Logging and XML Settings Configuration Options” in the *ONESOURCE Indirect Tax Integration for Oracle 12 Procure to Pay User’s Guide*.

Maintenance of these logs are managed by the **Sabrix Log Table Maintenance** concurrent program. Maintenance actions depend upon the setting of the program parameter values listed below:

- CLEAN - data for the parameters (retention days, start and end dates, etc) selected willbe deleted from the log table.
- PURGE - the log table is emptied, either entirely or all transactions based on the workflow parameter. This action requires that the profile option **eBTax: Sabrix Allow Sabrix Table Truncate** be set to *Yes* in conjunction*.* If the profile option is set to *No* and the user attempts to run the **Sabrix Log Maintenance** for PURGE, the log file will show that the purge is not allowed.

NOTE The selected action outlined above will be performed on ALL data in the SABRIX_LOG table, including Receivables, Order Management, Payables, and Purchasing transactions.

NOTE Once the data is deleted or purged from the SABRIX_LOG table, the XML data is also deleted or purged. Therefore, reports (Sabrix XML Report) will contain no data for the filtered criteria.

The process parameters include the Action (Clean or Purge), Workflow (O2C or P2P), Retention Days (the number of days to retain the transaction data - used for CLEAN action only), the start and end dates. (date parameters are for CLEAN action only).

If an error occurs during processing of the transaction, you are notified. Full rollback capabilities ensure that Determination and Oracle Applications remain in agreement. Even with these safeguards in place, with the level of control granted to administrators and superusers, it is possible to manually circumvent these controls and introduce inconsistencies. For this purpose, the Integration includes reconciliation reports for matching Oracle distributions to audit transactions.

The Sabrix Log Report concurrent process enables users to review the data in the SABRIX_LOG table without requiring a DBA or Developer to gather the data. See the ***SABRIX LOG REPORT (page 141)*** for more information.

See the “Reporting” section in the *ONESOURCE Indirect Tax Integration for Oracle 12 Procure to Pay User’s Guide or ONESOURCE Indirect Tax Integration for Oracle 12 Order to Cash User’s Guide* for additional details.

SABRIX TRANSACTION MANAGER FOR AP AND AR

The Integration provides a simple method to maintain the transaction tables resulting from Determination processes. The Integration writes transactions to tables located within the Sabrix schema. This process is not Operating Unit-specific since it is used for Sabrix tables that are not Operating Unit-specific. The **Sabrix Transaction Manager** concurrent program manages these tables, and depending upon the setting of parameters listed in the table below, can:

- Delete transactions older than a specified date.
- Delete transactions for a specified date range.
- Purge all records from the transaction table.

See the 'Reporting' section in the *ONESOURCE Indirect Tax Integration for Oracle 12 Procure to Pay User's Guide or ONESOURCE Indirect Tax Integration for Oracle 12 Order to Cash User's Guide* for additional details.

SABRIX NEW TAX FLOWS REPORT

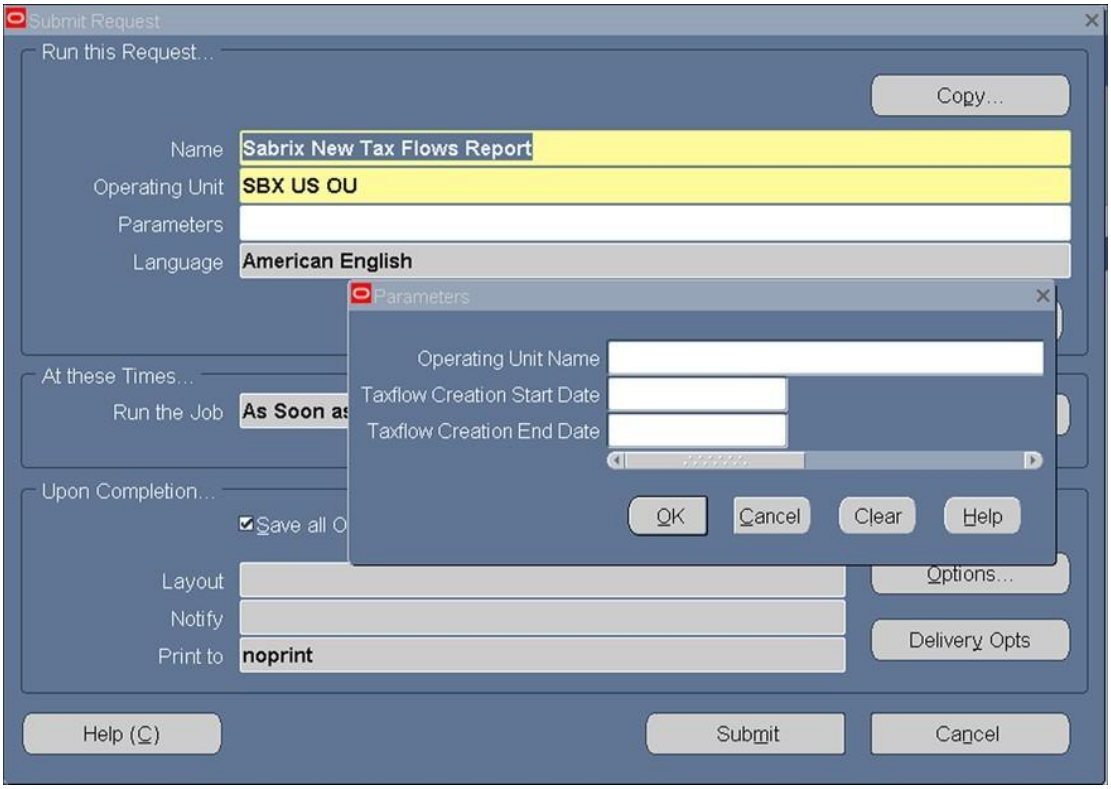
The **Sabrix New Tax Flows Report** lists all the tax flows created within a date range. The source for the default information to create the Tax Flow is provided in the report. This report can be scheduled to run daily to notify the tax personnel of new tax flows that were created.

The report contains the following information for the newly created tax flow:

- Operating Unit
- Regime
- Tax Name
- Tax Rate Code
- Liability Account
- Expense Account
- Source Tax Flow
- Date Added

To run the **Sabrix New Tax Flows Report**:

1. Select the **Tax Managers** responsibility.
2. Navigate to **Requests > Other > Requests > Run** and select a single request.
3. In the name field, select the concurrent process **Sabrix New Tax Flows Report.**
4. Enter information on the **Parameters** page and click **OK**.
5. Click **Submit**.
6. Navigate to the **Concurrent Manager** to view the report.



GL ACCOUNT DISCREPANCIES FOR SABRIX TAX RATE CODES REPORT

The *GL Tax Account Discrepancies For Sabrix Tax Rate Codes* report lists Tax Flows within an ERP Tax Code group that have different general ledger accounts. Thomson Reuters recommends that you use the same general ledger accounts within the ERP Tax Code group. An ERP Tax Code group relates to all the Tax Flows associated with the same ERP Tax Code. For example, all the city and county authorities within a state will have the same ERP Tax Code, like Washington State (USWA).

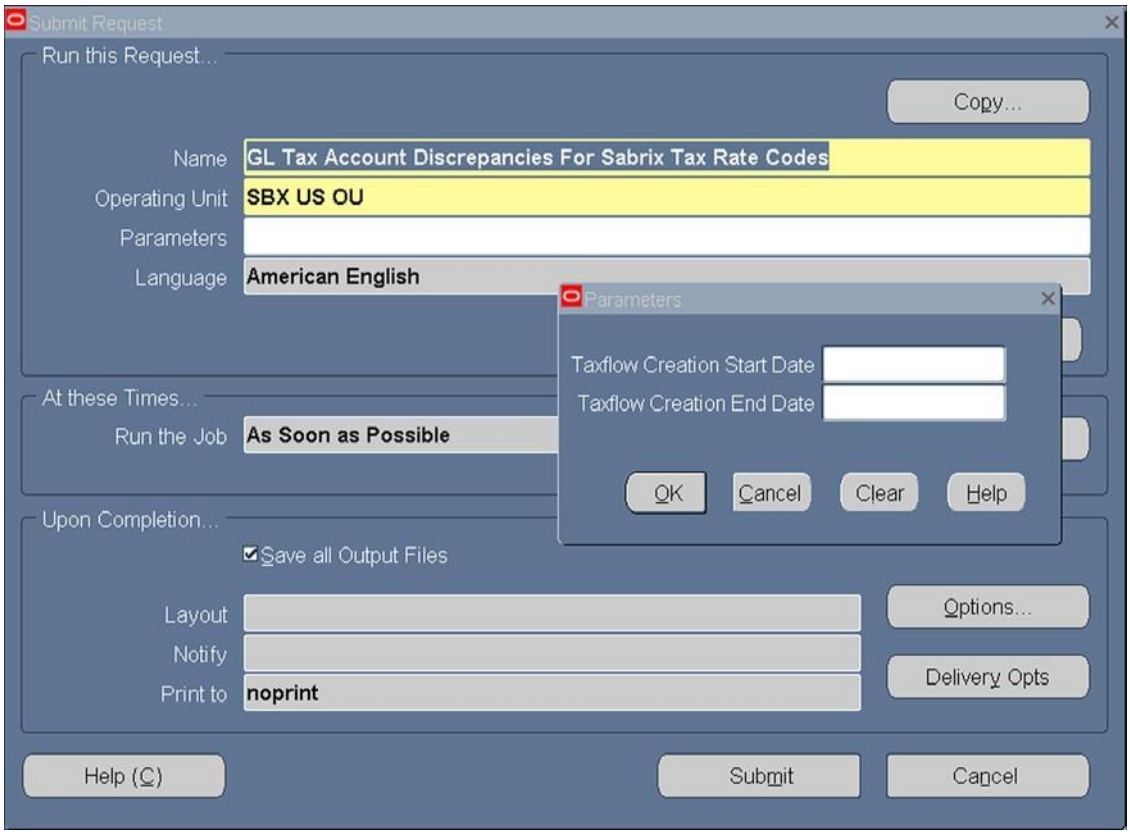
The report contains the following information for each discrepancy found:

- Operating Unit Name
- Regime
- Source ERP Tax Code
- Source ERP Tax Code Liability Account
- Source ERP Tax Code Expense Account
- Tax Rate Code
- Tax Rate Name
- Tax Rate Code Liability Account
- Tax Rate Code Expense Account
- Date Added

If there are no GL Accounting discrepancies, then the concurrent request output will not have any data and wil be blank.

To run the **GL Account Discrepancies For Sabrix Tax Rate Codes** Report:

1. Select the **Tax Managers** responsibility.
2. Navigate to **Requests > Other > Requests > Run** and select a single request.
3. In the name field, select the concurrent process **GL Account Discrepancies For Sabrix Tax Rate Codes.**
4. Enter information on the **Parameters** page and click **OK**.



5. Click **Submit**.
6. Navigate to the **Concurrent Manager** to view the report.

SABRIX SETUP RECOVERY RATES

For existing Tax to Rate Flows from Integration versions prior to 5.6.0.0, a concurrent program *Sabrix Setup Recovery Rates* has been added to update the appropriate flags to allow tax recovery, create the tax recovery rate and associate the tax recovery rate at the Tax level and to the standard percentage Tax Rate Code. The concurrent program is meant only to create recovery rates for any and all existing non-U.S. standard percentage Tax Rate Codes.

To run the Sabrix Setup Recovery Rates concurrent process:

1. Select the **Tax Managers** responsibility.
2. Navigate to **Requests > Other > Requests > Run** and select a single request.
3. In the name field, select the concurrent process **Sabrix Setup Recovery Rates.**

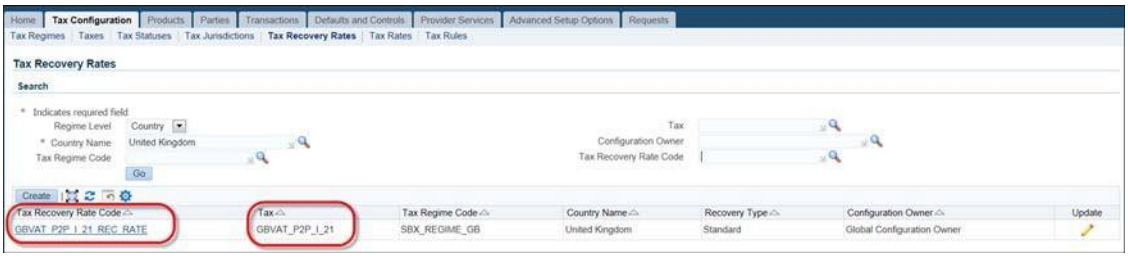
There are no parameters or a report to **View Output** on the **Sabrix Setup Recovery Rates** concurrent process. However, the concurrent **process View Log** will be available.

4. Click **Submit**.
5. Review the concurrent request log file to confirm successful completion.

Verify Tax Recovery Rates

To verify that Tax Recovery Rates have been created for all existing non-U.S. standard percentage Tax Rate Codes.

1. Select the **Tax Managers** responsibility.
2. Navigate to **Tax Configuration > Tax Recovery Rates**.
3. Search by entering the Country Name (required field)
4. Click on **Go** to view the *Tax Recovery Rate Code*



5. Click on the Update icon to view the additional details for the Tax Recovery Rate Code

.

APPENDIX B: NETWORK PROTOCOL OPTIONS

Web Service Interface

ONESOURCE Indirect Tax Integration for Oracle 12 has an option to select either the XMLINVOICE interface or the Web Services interface to communicate with Determination. Determination is the same using either interface. The INDATA/OUTDATA saved to the sabrix_log reflects the format of the interface with the Web Services XML format reflecting the SOAP envelope.

Tax Calc URL

The addition of the Web Services interface has increased the maximum size of the Tax Calc URL. The Tax Calc URL can now be as big as 4000 bytes.

The Tax Calc URL for Web Services (or SOAP) can be determined by

- Navigating to the services page of the Tax Determination application ; i.e. http://yourtaxcalcURL/sabrix/services.
- Scrolling down to the TaxCalculationService section and open (i.e. click on) the WSDL URL.
- Looking at the bottom of the WSDL for the address of the TaxCalculationService in the <soap:address location> element: (i.e. http://yourtaxcalcURL/sabrix/services/taxcalculationservice/2011-09-01/taxcalculationservice). Unlike the XMLINVOICE interface, this URL cannot be addressed directly by a browser.

New Parameters

The Web Service interface introduced new parameters to the sabrix_register_integration.sql script. These parameters are independent of each other.

Parameter	Description
Enter URL Protocol (HTTP or SOAP):	HTTP is the XMLINVOICE interface and SOAP is the Web Services Interface. The default is HTTP.
Enter Proxy address if applicable (http://hosst[:port][/]	The Proxy URL; i.e. http://www-proxy.my-company.com:80

Security

If security is implemented (i.e. Determination requires a username/password) the username and password will have to be set via a pre-calc user exit (SABRIX_ADAPTER_TABLE_PRECALC). Once the values are set, the sabrix_adapter.send_xml procedure will automatically create a SOAP header with the appropriate security values set.

Proxy

The sabrix_register_integration.sql script contains a Connection_Options parameter named PROXY. The Integration will check for a value in this table and will set the proxy value via utl_http.set_proxy, but only if the proxy is not already set (i.e. it will not overwrite the proxy if it has already been set at the server level).

Transfer-Encoding/chunked

The http request header transfer-encoding/chunked must be supported by the proxy server. Oracle HTTP Server is an example of an HTTP Proxy that supports transfer-encoding/chunked http request headers. If the request fails with a 500 error and the proxy is set, an exception will be thrown and a Sev 2 error message, “HTTP Error contacting URL:[tax_calc_url]:Your proxy[proxy_server_addr] may not allow transfer-encoded/chunked requests.” will be logged and raised to the calling API.

Content-Length

Many proxy servers do not support chunked transer-encoding. To allow the routing of the Determination request, there is a branch in the sabrix_adapter package to aggregate the entire document and replace the transfer-encoding, chunked HTTP header with the content-length (length of document) HTTP header.

The sabrix_adapter package contains a settable boolean variable, g_use_proxy_workaround. The default value is FALSE. If set to TRUE, the adapter will branch and prepare the Determination request with a content-length header.

Requests routed through this branch are limited to 32K bytes in size (which is less than 20 lines). If the request is greater than 32K, an exception will be thrown and a sev 2 message will be logged and raised to the calling API. The error message will be “The invoice is too large to process via your proxy [proxy server address].”

Setting the Proxy

Once the proxy is set for the database session, it cannot be overridden.

There are multiple methods to set the proxy for the database.

1. The database host server can set the HTTP_PROXY environment variable and the Oracle RDBMS will set the proxy to this value at startup. This value will be used for all database sessions. If the proxy is set in this manner, it will take precedence over all other methods.
2. The proxy can be set via a pre-calc by executing the url_http.set_proxy API.
3. The sabrix_register_integration.sql script has been enhanced to accept the proxy server HTTP address and will populate a row in the SABRIX_CONNECTION_OPTIONS with an OPTION_NAME = 'PROXY'. If this option is populated, the Integration will use the utl_http.set_proxy to set the HTTP proxy address for the database session. If a proxy is already configured via the HTTP_PROXY environment variable of the database server or another method, this value will have no effect.

APPENDIX C: ENTERING PARAMETERS FOR INSTALLATION SCRIPTS

If you choose to set the installation parameters at run-time instead of using the environment files, complete the tables below with notes about your environment. Then proceed to sections for executing the shell script *sabrix_installation_integration.sh* (***(page 54)***)

Ensure that you have no environment files (*.env*) in the directory where you unzipped ONESOURCE Indirect Tax Integration for Oracle 12.

PROMPTS FOR SABRIX_INSTALLATION_INTEGRATION.SH

The following table contains the prompts displayed by the installation script. Once you have recorded your values in the table below, see ***EXECUTING THE SABRIX INTEGRATION INSTALLATION SCRIPT (page 54)***:

Prompt	Description	Example	O2C + P2P	O2C	P2P	Your Value
Install Integration for Oracle Order to Cash?	Installing Integration Order to Cash (O2C)?	Y or N	R	R	R Enter *N*	
Install Integration for Oracle Procure to Pay?	Installing Integration Procure to Pay (P2P)?	Y or N	R	R Enter *N*	R	
Install packages and views only?	Install only packages and views? (leaving schema objects, value sets, and profile options intact). If set to 'Y', the user will not be prompted to retain data.	Y or N	R	R	R	

Prompt	Description	Example	O2C + P2P	O2C	P2P	Your Value
RETAIN_DATA	For ONESOURCE Indirect Tax Integration upgrade installations only -- Determines if the installation script will retain existing data in Sabrix schema staging tables for sabrix_invoice, sabrix_invoice_out, sabrix_invoice_tax, sabrix_license, sabrix_license_out, sabrix_line, sabrix_line_tax, sabrix_line_out, sabrix_message, and sabrix_registration tables.	*Y or N*	R	R	R	
Setup eBTax?	Enter *Y* to have the EB Tax Setup completed by the installation. Default is *Y*. **Note**: Only set this value to *N* if you are upgrading or updating.	*Y* or *N*	R	R	R	
The following prompt appears only if Setup eBTax = Y						
Enter the eBTax Start Date [01-JAN- 2008]	Effective Start date for Sabrix Rate Flow. Default is *01-JAN-2008*.	*01-JAN-2010*	R	R	R	
Enter the name of the Sabrix Schema	Sabrix Schema User	*XXIDT*	R	R	R	
Enter the password of the Sabrix Schema	Sabrix Schema Password	*XXIDT*	R	R	R	
Enter the name of the Apps Schema	Application (APPS) Schema User	*APPS*	R	R	R	
Enter the password of Apps Schema	Application Schema Password	*XXXX*	R	R	R	
Enter the name of the ERP database	Database name/alias	*oracle12dev*	R	R	R	

Prompt	Description	Example	O2C + P2P	O2C	P2P	Your Value
Enter the Sabrix Schema Data Tablespace	Sabrix Data Table Space	*XXIDT_DATA*	R	R	R	
Enter the Sabrix Schema Index Tablespace	Sabrix Index Table Space	*XXIDT_IDX*	R	R	R	
Enter the Sabrix Application Short Name	Sabrix Application ShortName	*XXIDT*	R	R	R	
Enter the Effective Start Date for eBTax Transactions	Effective Start Date for E-Business Tax Transactions	*01-JAN-2010*	R	R	R	
Additional prompts if you are installing Procure to Pay						
Enter the AP Tax Status Flag DFF	AP Sabrix Tax Status Flag Descriptive Flexfield (DFF)	*ATTRIBUTE1*	R		R	
Enter the AP Tax Amount DFF	AP Sabrix Tax Rate/Amount Descriptive Flexfield (DFF)	*ATTRIBUTE3*	R		R	
Enter the AP Item Link DFF	AP Sabrix Item Line Link Descriptive Flexfield (DFF)	*ATTRIBUTE4*	R		R	

APPENDIX D: CONFIGURING HTTPS (OPTIONAL STEP)

To use the ONESOURCE Indirect Tax Integration for Oracle 12 over a secure HTTP (HTTPS) connection, you must create an Oracle Wallet to store the appropriate security certificates.

This section describes how a DBA can use the *sabrix_register_wallet.sql* script to create and populate the Oracle Wallet in the database, and then register the Integration for use over https.

For additional details on setting up an Oracle Wallet, refer to the *Oracle® Database Advanced Security Administrator's Guide 10g Release 1 (10.1)*, Chapter 8: Using Oracle Wallet Manager.

GATHERING SECURITY CERTIFICATES

To use https in your integration, gather the security certificates for your ONESOURCE Indirect Tax Determination *xmlinvoice* interface.

You need every certificate in the certification path. The certificates must be in the *Base-64 encoded X.509* format. You can export the certificates using any web browser. See your browser's help section on exporting certificates for specific instructions.

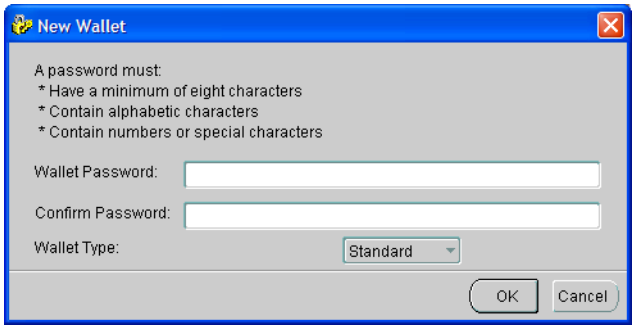
After successfully exporting the certificates, you can import them into an Oracle Wallet, as described in the next section.

CREATING THE ORACLE WALLET

You must create the Oracle Wallet on the database server.

1. Run the **Oracle Wallet Manager**.
   - On Linux / UNIX systems an X-Term session must be used; you will then run **owm**. The wallet directory may need to be set up by the system administrator.
   - On Windows systems, open the **Oracle Wallet Manager**.

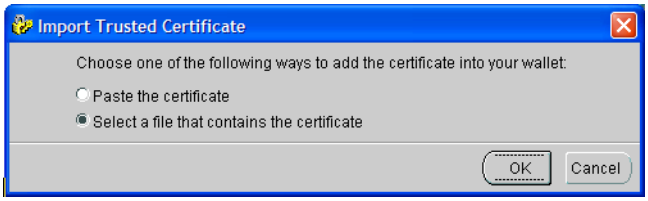
2. Choose **Wallet > New** to create a new wallet.



3. Specify a password for the wallet and click **OK**.

4. Select **No** when prompted to create a certificate request.

5. Choose **Operations > Import Trusted Certificate**.



6. Choose **Select a file that contains the certificate**. Click **OK**.

7. Select one of the saved certificates. Click **Open**.
8. Repeat steps 5-7 for the other certificate.

ADDING ORACLE WALLET DATA TO TABLE SABRIX_CONNECTION_OPTIONS

Who	Prerequisite
Oracle DBA, logged in as *APPS*.	Successful Integration installation.

To add to the required Oracle Wallet information to the SABRIX_CONNECTION_OPTIONS table:

1. In SQL*Plus, log into the Oracle Applications database as *APPS*.
2. Run the following script from the directory in which you unzipped the installation files:

   ```
   SQL> @sabrix_register_wallet.sql
   ```

3. Enter the following data:

Prompts	Example Value
Application Module Name	O2C or P2P
Oracle Wallet Directory	/home/oravis/wallets
Oracle Wallet Password	welcome1

The following is an example of the script's prompts and defaults:

```
SQL> @sabrix_register_wallet.sql

------------------------- Register the Sabrix Connection -------------------------
.                        (sabrix_register_wallet.sql)

LOGFILE                                  RUNDATE
---------------------------------------- ------------------------------
sabrix_register_wallet2310121114.log 23-OCT-2012 11:14:55
Log file is [sabrix_register_wallet2310121114.log]
+   Script to Register Sabrix Connection Server Options
+   Install script : [sabrix_register_wallet.sql]
+   Run date       : [23-OCT-2012 11:14:55]
+ Connect to SQL*Plus as Oracle user SABRIX
+ All Parameter values are case sensitive
+ Parameters:
+-------------------------------------------------------------------------------+


Enter Application Module Name (O2C/P2P)
    Application Module Name [P2P]  : O2C
'Enter Oracle Wallet Directory (e.g. [/etc/oracle/wallet]): '
 Oracle Wallet Directory [] :
```

You must configure a WALLET_DIRECTORY and WALLET_PASSWORD for each CONNECTION_ID installed, for example, *CONNECTION_ID 1* for O2C and *CONNECTION_ID 2* for P2P. The CONNECTION_ID is dependent upon the order in which the Integration installations were completed.

In the following examples from UNIX/Linux-based Oracle environments, the O2C (Sales) Integration was installed first, so its CONNECTION_ID is *1*. The P2P (Purchases) Integration was installed second so its CONNECTION_ID is *2*.

The following shows the **SABRIX_CONNECTION_OPTIONS** table for **CONNECTION_OPTION_ID**s *9* and *10* when both O2C and P2P are installed.

CONNECTION_ OPTION_ID	CONNECTION_ ID	OPTION_NAME	OPTION_VALUE
1	1	DEBUG_LEVEL	5
2	1	LOG_FILE	
3	1	BATCH_SIZE	100
4	2	DEBUG_LEVEL	10
5	2	LOG_FILE	
6	2	BATCH_SIZE	100
7	2	APPL_SHORT_NAME	SBX
8	2	TAX_PROCESS_WORKER	SBXAPPROC
9	1	WALLET_DIRECTORY	Example: *file:/home/oravis/wallets* **Note**: Directory name must be the same directory as when creating the Oracle Wallet Directory
10	1	WALLET_PASSWORD	Example: *welcome1* **Note**: Use the same WALLET_PASSWORD as when creating the Oracle Wallet
9	2	WALLET_DIRECTORY	Example: *file:/home/oravis/wallets* **Note**: Directory name must be the same directory as when creating the Oracle Wallet Directory
10	2	WALLET_PASSWORD	Example: *welcome1* **Note**: Use the same WALLET_PASSWORD as when creating the Oracle Wallet

APPENDIX E: ORACLE SOURCEMAPPED TO DETERMINATION XML DATA

This chapter shows the mapping from Oracle Purchases and Oracle Receivables fields to ONESOURCE Indirect Tax Determination Input and Output XML. Determination XML is organized hierarchically. A single XML data file may contain multiple batches, invoices, and lines.

- Batches contain invoices.
- Invoices contain lines.

An element value set at the batch level is passed down to the invoice and line levels, unless it is overridden at a lower level.

In the P2P mapping table there are references to view aliases. In the AP column there are references to 'trx'. This refers to the sabrix_ap_item_details_v view. In the PO and AP Online Call column there are references to i_calc_txn. This refers to the sabrix_p2p_calc_txn_input_v view..

This chapter has the following P2P and O2C mappings:

P2P MAPPING

P2P: INVOICE-LEVEL INPUT XML MAPPED TO ORACLE

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
BATCH_ID	Sequence-generated ID	Request ID followed by '.' and sequence-generated ID
INVOICE_ID	*1*	*1*
CREATION_DATE	SYSDATE	SYSDATE
MERCHANT_ID	UNMAPPED	UNMAPPED
MERCHANT_NAME	UNMAPPED	UNMAPPED
CALLING_SYSTEM_NUMBER	*201*	*200*
HOST_SYSTEM	Database instance name	Database instance name
EXTERNAL_COMPANY_ID	profile option **Sabrix company**	Both calls: profile option **Sabrix company** If profile option = *BALANCE_SEGMENT*, then balance segment value from ap_invoice_distributions_all ccid. If profile option = *SEGMENT*_VALUE*, then from ap_invoice_distributions_all ccid segment *.
MERCHANT_ROLE	*B*	First call: If US invoice total tax = 0, then *B*, else *S* First call: If Non-US invoice, then *B* Second call: *B*
STATISTICS_NAME	UNMAPPED	UNMAPPED
STATISTICS_VALUE	UNMAPPED	UNMAPPED
XML_GROUP_NAME	UNMAPPED	UNMAPPED
XML_GROUP_OWNER	UNMAPPED	UNMAPPED
USERNAME	UNMAPPED	UNMAPPED
PASSWORD	UNMAPPED	UNMAPPED
ALLOCATION_GROUP_NAME	UNMAPPED	UNMAPPED
ALLOCATION_GROUP_OWNER	UNMAPPED	UNMAPPED
ALLOCATION_NAME	UNMAPPED	UNMAPPED
AUTO_CREATE_CERTIFICATES	UNMAPPED	*N*
AUTO_CREATE_CUSTOMERS	UNMAPPED	*N*
BASIS_PERCENT	UNMAPPED	UNMAPPED
VENDOR_NAME	substrb(po_vendors.vendor_name,1,100) Note: Vendor Name on requisitions are at the line level.	Both calls: ap_suppliers.vendor_name
VENDOR_NUMBER	substrb(po_vendors.vendor_number,1,20) Note: Vendor Number on requisitions are at the line level	Both calls: ap_suppliers.segment1
PRODUCT_MAPPING_GROUP_OWNER	UNMAPPED	UNMAPPED
PRODUCT_MAPPING_GROUP_NAME	UNMAPPED	UNMAPPED
BT_COUNTRY	UNMAPPED	UNMAPPED
BT_PROVINCE	UNMAPPED	UNMAPPED
BT_STATE	UNMAPPED	UNMAPPED
BT_COUNTY	UNMAPPED	UNMAPPED
BT_CITY	UNMAPPED	UNMAPPED

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
BT_DISTRICT	UNMAPPED	UNMAPPED
BT_POSTCODE	UNMAPPED	UNMAPPED
BT_GEOCODE	UNMAPPED	UNMAPPED
BT_IS_BONDED	UNMAPPED	UNMAPPED
BT_LOCATION_TAX_CATEGORY	UNMAPPED	UNMAPPED
BT_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
BP_COUNTRY	UNMAPPED	UNMAPPED
BP_PROVINCE	UNMAPPED	UNMAPPED
BP_STATE	UNMAPPED	UNMAPPED
BP_COUNTY	UNMAPPED	UNMAPPED
BP_CITY	UNMAPPED	UNMAPPED
BP_DISTRICT	UNMAPPED	UNMAPPED
BP_POSTCODE	UNMAPPED	UNMAPPED
BP_GEOCODE	UNMAPPED	UNMAPPED
BP_IS_BONDED	UNMAPPED	UNMAPPED
BP_LOCATION_TAX_CATEGORY	UNMAPPED	UNMAPPED
BP_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
CALCULATION_DIRECTION	UNMAPPED	First call: *F* Second call: If invoice total tax = 0, then *F*, else *R*
COUNTRY_OF_ORIGIN	UNMAPPED	UNMAPPED
CURRENCY_CODE	Zx_lines_det_factors.trx_currency_code	Both calls: ap_invoices_all.currency_code
CUSTOMER_NAME	Zx_ptnr_location_info_gt.bill_to_party_name	UNMAPPED
CUSTOMER_NUMBER	Both calls: profile option **Sabrix company** If profile option = BALANCE_SEGMENT, then balance segment value from ap_invoice_distributions_all ccid If profile option = SEGMENT*_VALUE, then from ap_invoice_distributions_all ccid segment *	Both calls: profile option **Sabrix company** If profile option = BALANCE_SEGMENT, then balance segment value from ap_invoice_distributions_all ccid If profile option = SEGMENT*_VALUE, then from ap_invoice_distributions_all ccid segment *
DEPT_OF_CONSIGN	UNMAPPED	UNMAPPED
END_USER_NAME	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_COUNTRY	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_PROVINCE	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_STATE	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_COUNTY	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_CITY	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_DISTRICT	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_POSTCODE	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_GEOCODE	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_COUNTRY_FUNC	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_PROVINCE_FUNC	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_STATE_FUNC	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_COUNTY_FUNC	UNMAPPED	UNMAPPED

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
EXEMPT_AMOUNT_CITY_FUNC	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_DISTRICT_FUNC	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_POSTCODE_FUNC	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_GEOCODE_FUNC	UNMAPPED	UNMAPPED
EXEMPT_CERTIFICATE_COUNTRY	UNMAPPED	UNMAPPED
EXEMPT_CERTIFICATE_PROVINCE	UNMAPPED	UNMAPPED
EXEMPT_CERTIFICATE_STATE	UNMAPPED	UNMAPPED
EXEMPT_CERTIFICATE_COUNTY	UNMAPPED	UNMAPPED
EXEMPT_CERTIFICATE_CITY	UNMAPPED	UNMAPPED
EXEMPT_CERTIFICATE_DISTRICT	UNMAPPED	UNMAPPED
EXEMPT_CERTIFICATE_POSTCODE	UNMAPPED	UNMAPPED
EXEMPT_CERTIFICATE_GEOCODE	UNMAPPED	UNMAPPED
EXEMPT_REASON_COUNTRY	UNMAPPED	UNMAPPED
EXEMPT_REASON_PROVINCE	UNMAPPED	UNMAPPED
EXEMPT_REASON_STATE	UNMAPPED	UNMAPPED
EXEMPT_REASON_COUNTY	UNMAPPED	UNMAPPED
EXEMPT_REASON_CITY	UNMAPPED	UNMAPPED
EXEMPT_REASON_DISTRICT	UNMAPPED	UNMAPPED
EXEMPT_REASON_POSTCODE	UNMAPPED	UNMAPPED
EXEMPT_REASON_GEOCODE	UNMAPPED	UNMAPPED
FILTER_GROUP_NAME	UNMAPPED	UNMAPPED
FILTER_GROUP_OWNER	UNMAPPED	UNMAPPED
FISCAL_DATE	UNMAPPED	ap_invoices_all.gl_date
FREIGHT_ON_BOARD	substr(zx_lines_det_factors. fob_point,1,1)	If PO matched, substr(po_headers.fob_ lookup_code,1,3) if not null Else substr(ap_supplier_sites.fob_lookup_ code, 1, 3)
GROSS_AMOUNT	sum(zx_lines_det_factors.line_amt)	sum(ap_invoice_distribution.amount)
FUNCTIONAL_CURRENCY_CODE	UNMAPPED	UNMAPPED
DELIVERY_TERMS	zx_lines_det_factors.fob_point	If PO matched, substr(po_headers.fob_lookup_code,1,3) if not null Else substr(ap_supplier_sites.fob_lookup_code, 1, 3)
INPUT_RECOVERY_TYPE	UNMAPPED	UNMAPPED
INVOICE_DATE	SYSDATE	SYSDATE
INVOICE_NUMBER	Zx_lines_det_factors.trx_number	Both calls: ap_invoices_all.invoice_num
IS_BUSINESS_SUPPLY	UNMAPPED	*N*
IS_CREDIT	UNMAPPED	If ap_invoices_all.invoice_type_lookup_ code=CREDIT then *Y*, else *N*
IS_EXEMPT_ALL	UNMAPPED	UNMAPPED
IS_EXEMPT_COUNTRY	UNMAPPED	UNMAPPED
IS_EXEMPT_PROVINCE	UNMAPPED	UNMAPPED
IS_EXEMPT_STATE	UNMAPPED	UNMAPPED
IS_EXEMPT_COUNTY	UNMAPPED	UNMAPPED
IS_EXEMPT_CITY	UNMAPPED	UNMAPPED
IS_EXEMPT_DISTRICT	UNMAPPED	UNMAPPED

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
IS_EXEMPT_POSTCODE	UNMAPPED	UNMAPPED
IS_EXEMPT_GEOCODE	UNMAPPED	UNMAPPED
IS_NO_TAX_ALL	UNMAPPED	UNMAPPED
IS_NO_TAX_COUNTRY	UNMAPPED	UNMAPPED
IS_NO_TAX_PROVINCE	UNMAPPED	UNMAPPED
IS_NO_TAX_STATE	UNMAPPED	UNMAPPED
IS_NO_TAX_COUNTY	UNMAPPED	UNMAPPED
IS_NO_TAX_CITY	UNMAPPED	UNMAPPED
IS_NO_TAX_DISTRICT	UNMAPPED	UNMAPPED
IS_NO_TAX_POSTCODE	UNMAPPED	UNMAPPED
IS_NO_TAX_GEOCODE	UNMAPPED	UNMAPPED
IS_REPORTED	UNMAPPED	UNMAPPED
IS_REVERSED	UNMAPPED	UNMAPPED
IS_ROUNDING	UNMAPPED	UNMAPPED
LOCATION_SHIP_FROM	UNMAPPED	UNMAPPED
LOCATION_BILL_TO	UNMAPPED	UNMAPPED
LOCATION_SUPPLY	UNMAPPED	UNMAPPED
LOCATION_MIDDLEMAN	UNMAPPED	UNMAPPED
LOCATION_ORDER_ORIGIN	UNMAPPED	UNMAPPED
LOCATION_ORDER_ACCEPTANCE	UNMAPPED	UNMAPPED
LOCATION_SHIP_TO	UNMAPPED	UNMAPPED
LOCATION_SET	UNMAPPED	UNMAPPED
MM_COUNTRY	UNMAPPED	UNMAPPED
MM_PROVINCE	UNMAPPED	UNMAPPED
MM_STATE	UNMAPPED	UNMAPPED
MM_COUNTY	UNMAPPED	UNMAPPED
MM_CITY	UNMAPPED	UNMAPPED
MM_DISTRICT	UNMAPPED	UNMAPPED
MM_POSTCODE	UNMAPPED	UNMAPPED
MM_GEOCODE	UNMAPPED	UNMAPPED
MM_IS_BONDED	UNMAPPED	UNMAPPED
MM_LOCATION_TAX_CATEGORY	UNMAPPED	UNMAPPED
MM_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
MODE_OF_TRANSPORT	UNMAPPED	UNMAPPED
MOVEMENT_DATE	Zx_lines_det_factors.trx_line_date	If po matched, nvl(max(rcv_shipment_headers.shipped_date) ,sysdate) If receipt matched, rcv_shipment_headers.shipped_date Else *NULL*
MOVEMENT_TYPE	UNMAPPED	UNMAPPED
NATURE_OF_TRANSACTION_CODE	UNMAPPED	UNMAPPED
OA_COUNTRY	UNMAPPED	UNMAPPED
OA_PROVINCE	UNMAPPED	UNMAPPED
OA_STATE	UNMAPPED	UNMAPPED
OA_COUNTY	UNMAPPED	UNMAPPED

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
OA_CITY	UNMAPPED	UNMAPPED
OA_DISTRICT	UNMAPPED	UNMAPPED
OA_POSTCODE	UNMAPPED	UNMAPPED
OA_GEOCODE	UNMAPPED	UNMAPPED
OA_IS_BONDED	UNMAPPED	UNMAPPED
OA_LOCATION_TAX_CATEGORY	UNMAPPED	UNMAPPED
OA_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
OO_COUNTRY	UNMAPPED	UNMAPPED
OO_PROVINCE	UNMAPPED	UNMAPPED
OO_STATE	UNMAPPED	UNMAPPED
OO_COUNTY	UNMAPPED	UNMAPPED
OO_CITY	UNMAPPED	UNMAPPED
OO_DISTRICT	UNMAPPED	UNMAPPED
OO_POSTCODE	UNMAPPED	UNMAPPED
OO_GEOCODE	UNMAPPED	UNMAPPED
OO_IS_BONDED	UNMAPPED	UNMAPPED
OO_LOCATION_TAX_CATEGORY	UNMAPPED	UNMAPPED
OO_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
ORIGINAL_INVOICE_NUMBER	UNMAPPED	UNMAPPED
ORIGINAL_DOCUMENT_ID	UNMAPPED	UNMAPPED
ORIGINAL_DOCUMENT_ITEM	UNMAPPED	UNMAPPED
ORIGINAL_DOCUMENT_TYPE	UNMAPPED	UNMAPPED
ORIGINAL_INVOICE_DATE	UNMAPPED	UNMAPPED
ORIGINAL_MOVEMENT_DATE	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_COUNTRY	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_PROVINCE	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_STATE	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_COUNTY	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_CITY	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_DISTRICT	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_POSTCODE	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_GEOCODE	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_COUNTRY_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_PROVINCE_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_STATE_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_COUNTY_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_CITY_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_DISTRICT_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_POSTCODE_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_GEOCODE_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_RATE_COUNTRY	UNMAPPED	UNMAPPED
OVERRIDE_RATE_PROVINCE	UNMAPPED	UNMAPPED
OVERRIDE_RATE_STATE	UNMAPPED	UNMAPPED
OVERRIDE_RATE_COUNTY	UNMAPPED	UNMAPPED

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
OVERRIDE_RATE_CITY	UNMAPPED	UNMAPPED
OVERRIDE_RATE_DISTRICT	UNMAPPED	UNMAPPED
OVERRIDE_RATE_POSTCODE	UNMAPPED	UNMAPPED
OVERRIDE_RATE_GEOCODE	UNMAPPED	UNMAPPED
POINT_OF_TITLE_TRANSFER	UNMAPPED	UNMAPPED
PORT_OF_ENTRY	UNMAPPED	UNMAPPED
PORT_OF_LOADING	UNMAPPED	UNMAPPED
REGIME	UNMAPPED	UNMAPPED
SP_COUNTRY	UNMAPPED	UNMAPPED
SP_PROVINCE	UNMAPPED	UNMAPPED
SP_STATE	UNMAPPED	UNMAPPED
SP_COUNTY	UNMAPPED	UNMAPPED
SP_CITY	UNMAPPED	UNMAPPED
SP_DISTRICT	UNMAPPED	UNMAPPED
SP_POSTCODE	UNMAPPED	UNMAPPED
SP_GEOCODE	UNMAPPED	UNMAPPED
SP_IS_BONDED	UNMAPPED	UNMAPPED
SP_LOCATION_TAX_CATEGORY	UNMAPPED	UNMAPPED
SP_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
SF_COUNTRY	zx_ptnr_location_info_gt.ship_ from_geography values	Both calls: ap_supplier_sites_all.country
SF_PROVINCE	zx_ptnr_location_info_gt.ship_ from_geography values	Both calls: ap_supplier_sites_all.province
SF_STATE	zx_ptnr_location_info_gt.ship_ from_geography values	Both calls: ap_supplier_sites_all.state
SF_COUNTY	zx_ptnr_location_info_gt.ship_ from_geography values	Both calls: ap_supplier_sites_all.county
SF_CITY	zx_ptnr_location_info_gt.ship_ from_geography values	Both calls: ap_supplier_sites_all.town_or_city
SF_DISTRICT	zx_ptnr_location_info_gt.ship_ from_geography values	UNMAPPED
SF_POSTCODE	zx_ptnr_location_info_gt.ship_ from_geography values	Both calls: ap_supplier_sites_all.zip (1st five chars for USA)
SF_GEOCODE	zx_ptnr_location_info_gt.ship_from_g eography values	Both calls: ap_supplier_sites_all.zip (chars 7-10 for USA)
SF_IS_BONDED	UNMAPPED	UNMAPPED
SF_LOCATION_TAX_CATEGORY	UNMAPPED	UNMAPPED
SF_COMPANY_BRANCH_ID	UNMAPPED	Both calls: ap_supplier_sites_all.vat_registration_num
ST_COUNTRY	zx_ptnr_location_info_gt.ship_ to_geography values	Both calls: Load first line when shipto_source = HEADER, else NULL. hr_locations_all.country or value from ADHOC
ST_PROVINCE	zx_ptnr_location_info_gt.ship_ to_geography values	Both calls: Load first line when shipto_source = HEADER, else NULL. hr_locations_all.country or value from ADHOC
ST_STATE	zx_ptnr_location_info_gt.ship_ to_geography values	Both calls: Load first line when shipto_source = HEADER, else NULL. hr_locations_all.country or value from ADHOC

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
ST_COUNTY	zx_ptnr_location_info_gt.ship_to_geography values	Both calls: Load first line when shipto_source = HEADER, else NULL. hr_locations_all.country or value from ADHOC
ST_CITY	zx_ptnr_location_info_gt.ship_to_geography values	Both calls: Load first line when shipto_source = HEADER, else NULL. hr_locations_all.country or value from ADHOC
ST_DISTRICT	zx_ptnr_location_info_gt.ship_to_geography values	UNMAPPED
ST_POSTCODE	zx_ptnr_location_info_gt.ship_to_geography values	Both calls: Load first line when shipto_source = HEADER, else NULL. hr_locations_all.country or value from ADHOC
ST_GEOCODE	zx_ptnr_location_info_gt.ship_to_geography values	Both calls: Load first line when shipto_source = HEADER, else NULL. hr_locations_all.country or value from ADHOC
ST_IS_BONDED	UNMAPPED	UNMAPPED
ST_LOCATION_TAX_CATEGORY	UNMAPPED	UNMAPPED
ST_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
STATISTICAL_PROCEDURE	UNMAPPED	UNMAPPED
SU_COUNTRY	UNMAPPED	UNMAPPED
SU_PROVINCE	UNMAPPED	UNMAPPED
SU_STATE	UNMAPPED	UNMAPPED
SU_COUNTY	UNMAPPED	UNMAPPED
SU_CITY	UNMAPPED	UNMAPPED
SU_DISTRICT	UNMAPPED	UNMAPPED
SU_POSTCODE	UNMAPPED	UNMAPPED
SU_GEOCODE	UNMAPPED	UNMAPPED
SU_IS_BONDED	UNMAPPED	UNMAPPED
SU_LOCATION_TAX_CATEGORY	UNMAPPED	UNMAPPED
SU_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_COUNTRY	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_PROVINCE	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_STATE	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_COUNTY	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_CITY	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_DISTRICT	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_POSTCODE	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_GEOCODE	UNMAPPED	UNMAPPED
SUPPLY_TYPE	UNMAPPED	UNMAPPED
TAX_AMOUNT	UNMAPPED	UNMAPPED
TAX_CODE	UNMAPPED	UNMAPPED
TAX_TYPE_ALL	UNMAPPED	UNMAPPED
TAX_TYPE_COUNTRY	UNMAPPED	UNMAPPED
TAX_TYPE_PROVINCE	UNMAPPED	UNMAPPED
TAX_TYPE_STATE	UNMAPPED	UNMAPPED
TAX_TYPE_COUNTY	UNMAPPED	UNMAPPED
TAX_TYPE_CITY	UNMAPPED	UNMAPPED
TAX_TYPE_DISTRICT	UNMAPPED	UNMAPPED

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
TAX_TYPE_POSTCODE	UNMAPPED	UNMAPPED
TAX_TYPE_GEOCODE	UNMAPPED	UNMAPPED
TITLE_TRANSFER_LOCATION	UNMAPPED	UNMAPPED
TRANSACTION_TYPE	GS Constant	GS Constant
IS_AUDIT_UPDATE	If zx_lines_det_factors.event_class_code = *INVOICE_ADJUSTMENT*' then *Y*, otherwise NULL	UNMAPPED
BUYER_EST_BILL_TO	UNMAPPED	UNMAPPED
BUYER_EST_BUYER_PRIMARY	UNMAPPED	UNMAPPED
BUYER_EST_MIDDLEMAN	UNMAPPED	UNMAPPED
BUYER_EST_ORDER_ORIGIN	UNMAPPED	UNMAPPED
BUYER_EST_ORDER_ACCEPTANCE	UNMAPPED	UNMAPPED
BUYER_EST_SELLER_PRIMARY	UNMAPPED	UNMAPPED
BUYER_EST_SHIP_FROM	UNMAPPED	UNMAPPED
BUYER_EST_SHIP_TO	UNMAPPED	UNMAPPED
BUYER_EST_SUPPLY	UNMAPPED	UNMAPPED
MIDDLEMAN_EST_BILL_TO	UNMAPPED	UNMAPPED
MIDDLEMAN_EST_BUYER_PRIMARY	UNMAPPED	UNMAPPED
MIDDLEMAN_EST_MIDDLEMAN	UNMAPPED	UNMAPPED
MIDDLEMAN_EST_ORDER_ORIGIN	UNMAPPED	UNMAPPED
MIDDLEMAN_EST_ORDER_ACCEPTANCE	UNMAPPED	UNMAPPED
MIDDLEMAN_EST_SELLER_PRIMARY	UNMAPPED	UNMAPPED
MIDDLEMAN_EST_SHIP_FROM	UNMAPPED	UNMAPPED
MIDDLEMAN_EST_SHIP_TO	UNMAPPED	UNMAPPED
MIDDLEMAN_EST_SUPPLY	UNMAPPED	UNMAPPED
SELLER_EST_BILL_TO	UNMAPPED	UNMAPPED
SELLER_EST_BUYER_PRIMARY	UNMAPPED	UNMAPPED
SELLER_EST_MIDDLEMAN	UNMAPPED	UNMAPPED
SELLER_EST_ORDER_ORIGIN	UNMAPPED	UNMAPPED
SELLER_EST_ORDER_ACCEPTANCE	UNMAPPED	UNMAPPED
SELLER_EST_SELLER_PRIMARY	UNMAPPED	UNMAPPED
SELLER_EST_SHIP_FROM	UNMAPPED	UNMAPPED
SELLER_EST_SHIP_TO	UNMAPPED	UNMAPPED
SELLER_EST_SUPPLY	UNMAPPED	UNMAPPED
IS_SIMPLIFICATION	UNMAPPED	UNMAPPED
UNIQUE_INVOICE_NUMBER	Zx_lines_det_factors.transaction_id	If profile option Sabrix Company is set to BALANCE_SEGMENT or SEGMENTx_VALUE, then ap_invoices_all.invoice_id || '.' || derived external_company_id, else ap_invoices_all.invoice_id
VENDOR_TAX	sum(zx_lines.tax_amt)	Sum of ap_invoice_distributions_all.amount for invoice where line_type_lookup_code = *TAX* prior to processing by Determination
VENDOR_TAX_FUNC	UNMAPPED	UNMAPPED
US_PRODUCT_XREF_GROUP_OWNER	UNMAPPED	UNMAPPED

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
US_PRODUCT_XREF_GROUP_NAME	UNMAPPED	UNMAPPED
INTL_PRODUCT_XREF_GROUP_OWNER	UNMAPPED	UNMAPPED
INTL_PRODUCT_XREF_GROUP_NAME	UNMAPPED	UNMAPPED
US_REGISTRATION_GROUP_OWNER	UNMAPPED	UNMAPPED
US_REGISTRATION_GROUP_NAME	UNMAPPED	UNMAPPED
INTL_REGISTRATION_GROUP_OWNER	UNMAPPED	UNMAPPED
INTL_REGISTRATION_GROUP_NAME	UNMAPPED	UNMAPPED
VAT_GROUP_REGISTRATION	UNMAPPED	UNMAPPED
USER_ELEMENT_ATTRIBUTE1 – 40	UNMAPPED	UNMAPPED
USER_ELEMENT.ATTRIBUTE41	Zx_lines_det_factors.transaction_id	Both calls: ap_invoices_all.invoice_id
USER_ELEMENT.ATTRIBUTE42	i_calc_txn.ship_from_loc_id	Both calls: trx.sf_address_id
USER_ELEMENT.ATTRIBUTE43	i_calc_txn_ship_to_loc_id	Both calls: trx.st_address_id, if ship_to_shource = HEADER, else NULL
USER_ELEMENT.ATTRIBUTE44	Zx_lines_det_factors.event_ class_code	Both calls: AP_VIEW
USER_ELEMENT.ATTRIBUTE45	i_calc_trn.internal_organization_id	Both calls: ap_invoices_all.org_id
USER_ELEMENT.ATTRIBUTE46	Reserved	Reserved
USER_ELEMENT.ATTRIBUTE47	Reserved	Reserved
USER_ELEMENT.ATTRIBUTE48	Reserved	Reserved
USER_ELEMENT.ATTRIBUTE49	Zx_lines_det_factors. application_id	Both calls: 200
USER_ELEMENT.ATTRIBUTE50	Zx_lines_det_factors. event_class_mapping_id	First call: INVOICE IN TOLERANCE Second call: Returned only if True
COMPANY_NAME	UNMAPPED	UNMAPPED
HOST_REQUEST_ID	UNMAPPED	UNMAPPED
HOST_REQUEST_LOG_ENTRY_ID	UNMAPPED	UNMAPPED
AUDIT_MESSAGE_THRESHOLD	UNMAPPED	UNMAPPED
CUSTOMER_GROUP_NAME	UNMAPPED	UNMAPPED
CUSTOMER_GROUP_OWNER	UNMAPPED	UNMAPPED
DOCUMENT_TYPE	UNMAPPED	UNMAPPED
END_USE	UNMAPPED	UNMAPPED
AUTHORITY_TYPE	UNMAPPED	UNMAPPED
FULLY_INCLUSIVE	UNMAPPED	UNMAPPED
IS_AUDITING_MESSAGES	UNMAPPED	UNMAPPED
TAX_DETERMINATION_DATE	UNMAPPED	UNMAPPED
TAX_EXCHANGE_RATE_DATE	UNMAPPED	UNMAPPED
TAX_POINT_DATE	UNMAPPED	UNMAPPED

AP: LINE-LEVEL INPUT XML MAPPED TO ORACLE

Determination Line-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
BATCH_ID	Sequence-generated ID	Sequence-generated ID
INVOICE_ID	1	1
LINE_ID	Input cursor row number	Sequence-generated ID
CREATION_DATE	Sysdate	Sysdate
ACCOUNTING_CODE	UNMAPPED	UNMAPPED
ALLOCATION_GROUP_NAME	UNMAPPED	UNMAPPED
ALLOCATION_GROUP_OWNER	UNMAPPED	UNMAPPED
ALLOCATION_NAME	UNMAPPED	UNMAPPED
IS_ALLOCATABLE	UNMAPPED	UNMAPPED
BASIS_PERCENT	UNMAPPED	UNMAPPED
BT_COUNTRY	UNMAPPED	UNMAPPED
BT_PROVINCE	UNMAPPED	UNMAPPED
BT_STATE	UNMAPPED	UNMAPPED
BT_COUNTY	UNMAPPED	UNMAPPED
BT_CITY	UNMAPPED	UNMAPPED
BT_DISTRICT	UNMAPPED	UNMAPPED
BT_POSTCODE	UNMAPPED	UNMAPPED
BT_GEOCODE	UNMAPPED	UNMAPPED
BT_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
BP_COUNTRY	UNMAPPED	UNMAPPED
BP_PROVINCE	UNMAPPED	UNMAPPED
BP_STATE	UNMAPPED	UNMAPPED
BP_COUNTY	UNMAPPED	UNMAPPED
BP_CITY	UNMAPPED	UNMAPPED
BP_DISTRICT	UNMAPPED	UNMAPPED
BP_POSTCODE	UNMAPPED	UNMAPPED
BP_GEOCODE	UNMAPPED	UNMAPPED
BP_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
COMMODITY_CODE	UNMAPPED	UNMAPPED
COUNTRY_OF_ORIGIN	UNMAPPED	UNMAPPED
CUSTOMER_NAME	UNMAPPED	UNMAPPED
CUSTOMER_NUMBER	UNMAPPED	UNMAPPED
DEPT_OF_CONSIGN	UNMAPPED	UNMAPPED
DESCRIPTION	Zx_lines_det_factors.product_ description	Both calls: ap_invoice_distributions_all.description
DISCOUNT_AMOUNT	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_COUNTRY	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_PROVINCE	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_STATE	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_COUNTY	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_CITY	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_DISTRICT	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_POSTCODE	UNMAPPED	UNMAPPED

Determination Line-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
EXEMPT_AMOUNT_GEOCODE	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_COUNTRY_FUNC	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_PROVINCE_FUNC	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_STATE_FUNC	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_COUNTY_FUNC	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_CITY_FUNC	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_DISTRICT_FUNC	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_POSTCODE_FUNC	UNMAPPED	UNMAPPED
EXEMPT_AMOUNT_GEOCODE_FUNC	UNMAPPED	UNMAPPED
EXEMPT_CERTIFICATE_COUNTRY	Zx_lines_det_factors.exempt_ certificate_number	UNMAPPED
EXEMPT_CERTIFICATE_PROVINCE	Zx_lines_det_factors.exempt_ certificate_number	UNMAPPED
EXEMPT_CERTIFICATE_STATE	Zx_lines_det_factors.exempt_ certificate_number	UNMAPPED
EXEMPT_CERTIFICATE_COUNTY	Zx_lines_det_factors.exempt_ certificate_number	UNMAPPED
EXEMPT_CERTIFICATE_CITY	Zx_lines_det_factors.exempt_ certificate_number	UNMAPPED
EXEMPT_CERTIFICATE_DISTRICT	Zx_lines_det_factors.exempt_ certificate_number	UNMAPPED
EXEMPT_CERTIFICATE_POSTCODE	Zx_lines_det_factors.exempt_ certificate_number	UNMAPPED
EXEMPT_CERTIFICATE_GEOCODE	Zx_lines_det_factors.exempt_ certificate_number	UNMAPPED
EXEMPT_REASON_COUNTRY	Substr(Zx_lines_det_factors. exempt_reason, 1, 20)	UNMAPPED
EXEMPT_REASON_PROVINCE	Substr(Zx_lines_det_factors. exempt_reason, 1, 20)	UNMAPPED
EXEMPT_REASON_STATE	Substr(Zx_lines_det_factors. exempt_reason, 1, 20)	UNMAPPED
EXEMPT_REASON_COUNTY	Substr(Zx_lines_det_factors. exempt_reason, 1, 20)	UNMAPPED
EXEMPT_REASON_CITY	Substr(Zx_lines_det_factors. exempt_reason, 1, 20)	UNMAPPED
EXEMPT_REASON_DISTRICT	Substr(Zx_lines_det_factors. exempt_reason, 1, 20)	UNMAPPED
EXEMPT_REASON_POSTCODE	Substr(Zx_lines_det_factors. exempt_reason, 1, 20)	UNMAPPED
EXEMPT_REASON_GEOCODE	Substr(Zx_lines_det_factors. exempt_reason, 1, 20)	UNMAPPED
DELIVERY_TERMS	Zx_lines_det_factors.fob_point	If PO matched, substr(po_headers.fob_lookup_code,1,3) if not null Else substr(ap_supplier_sites.fob_lookup_code, 1, 3)
FREIGHT_ON_BOARD	substr(Zx_lines_det_factors.fob_point,1,1)	If PO matched, substr(po_headers.fob_lookup_code,1,3) if not null Else substr(ap_supplier_sites.fob_lookup_code, 1, 3)
GROSS_AMOUNT	Zx_lines_det_factors.line_amt	Both calls: ap_invoice_distributions_all.amount
GROSS_AMOUNT_FUNC	UNMAPPED	UNMAPPED
INPUT_RECOVERY_TYPE	UNMAPPED	UNMAPPED

Determination Line-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
INPUT_RECOVERY_AMOUNT	UNMAPPED	UNMAPPED
INPUT_RECOVERY_AMOUNT_FUNC	UNMAPPED	UNMAPPED
INPUT_RECOVERY_PERCENT	UNMAPPED	UNMAPPED
IS_BUSINESS_SUPPLY	UNMAPPED	UNMAPPED
IS_EXEMPT_ALL	UNMAPPED	Both calls: if ap_tax_codes_all.name = *EXEMPT* then *Y*, else NULL
IS_EXEMPT_COUNTRY	UNMAPPED	UNMAPPED
IS_EXEMPT_PROVINCE	UNMAPPED	UNMAPPED
IS_EXEMPT_STATE	UNMAPPED	UNMAPPED
IS_EXEMPT_COUNTY	UNMAPPED	UNMAPPED
IS_EXEMPT_CITY	UNMAPPED	UNMAPPED
IS_EXEMPT_DISTRICT	UNMAPPED	UNMAPPED
IS_EXEMPT_POSTCODE	UNMAPPED	UNMAPPED
IS_EXEMPT_GEOCODE	UNMAPPED	UNMAPPED
IS_MANUFACTURING	UNMAPPED	UNMAPPED
IS_NO_TAX_ALL	UNMAPPED	UNMAPPED
IS_NO_TAX_COUNTRY	UNMAPPED	UNMAPPED
IS_NO_TAX_PROVINCE	UNMAPPED	UNMAPPED
IS_NO_TAX_STATE	UNMAPPED	UNMAPPED
IS_NO_TAX_COUNTY	UNMAPPED	UNMAPPED
IS_NO_TAX_CITY	UNMAPPED	UNMAPPED
IS_NO_TAX_DISTRICT	UNMAPPED	UNMAPPED
IS_NO_TAX_POSTCODE	UNMAPPED	UNMAPPED
IS_NO_TAX_GEOCODE	UNMAPPED	UNMAPPED
ITEM_VALUE	UNMAPPED	UNMAPPED
ITEM_VALUE_FUNC	UNMAPPED	UNMAPPED
LINE_NUMBER	Input cursor row number	Both calls: ap_invoice_distributions_all.line_number
LOCATION_SHIP_FROM	UNMAPPED	UNMAPPED
LOCATION_BILL_TO	UNMAPPED	UNMAPPED
LOCATION_SUPPLY	UNMAPPED	UNMAPPED
LOCATION_MIDDLEMAN	UNMAPPED	UNMAPPED
LOCATION_ORDER_ORIGIN	UNMAPPED	UNMAPPED
LOCATION_ORDER_ACCEPTANCE	UNMAPPED	UNMAPPED
LOCATION_SHIP_TO	UNMAPPED	UNMAPPED
LOCATION_SET	UNMAPPED	UNMAPPED
MASS	UNMAPPED	UNMAPPED
MM_COUNTRY	UNMAPPED	UNMAPPED
MM_PROVINCE	UNMAPPED	UNMAPPED
MM_STATE	UNMAPPED	UNMAPPED
MM_COUNTY	UNMAPPED	UNMAPPED
MM_CITY	UNMAPPED	UNMAPPED
MM_DISTRICT	UNMAPPED	UNMAPPED
MM_POSTCODE	UNMAPPED	UNMAPPED
MM_GEOCODE	UNMAPPED	UNMAPPED

Determination Line-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
MM_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
MODE_OF_TRANSPORT	UNMAPPED	UNMAPPED
MOVEMENT_DATE	i_calc_txn.transaction_line_date	If po matched, nvl(max(rcv_shipment_headers.shipped_ date) ,sysdate) If receipt matched, rcv_shipment_headers.shipped_date Else NULL
MOVEMENT_TYPE	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_COUNTRY	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_PROVINCE	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_STATE	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_COUNTY	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_CITY	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_DISTRICT	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_POSTCODE	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_GEOCODE	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_COUNTRY_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_PROVINCE_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_STATE_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_COUNTY_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_CITY_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_DISTRICT_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_POSTCODE_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_AMOUNT_GEOCODE_FUNC	UNMAPPED	UNMAPPED
OVERRIDE_RATE_COUNTRY	UNMAPPED	UNMAPPED
OVERRIDE_RATE_PROVINCE	UNMAPPED	UNMAPPED
OVERRIDE_RATE_STATE	UNMAPPED	UNMAPPED
OVERRIDE_RATE_COUNTY	UNMAPPED	UNMAPPED
OVERRIDE_RATE_CITY	UNMAPPED	UNMAPPED
OVERRIDE_RATE_DISTRICT	UNMAPPED	UNMAPPED
OVERRIDE_RATE_POSTCODE	UNMAPPED	UNMAPPED
OVERRIDE_RATE_GEOCODE	UNMAPPED	UNMAPPED
OA_COUNTRY	UNMAPPED	UNMAPPED
OA_PROVINCE	UNMAPPED	UNMAPPED
OA_STATE	UNMAPPED	UNMAPPED
OA_COUNTY	UNMAPPED	UNMAPPED
OA_CITY	UNMAPPED	UNMAPPED
OA_DISTRICT	UNMAPPED	UNMAPPED
OA_POSTCODE	UNMAPPED	UNMAPPED
OA_GEOCODE	UNMAPPED	UNMAPPED
OA_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
OO_COUNTRY	UNMAPPED	UNMAPPED
OO_PROVINCE	UNMAPPED	UNMAPPED
OO_STATE	UNMAPPED	UNMAPPED
OO_COUNTY	UNMAPPED	UNMAPPED

Determination Line-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
OO_CITY	UNMAPPED	UNMAPPED
OO_DISTRICT	UNMAPPED	UNMAPPED
OO_POSTCODE	UNMAPPED	UNMAPPED
OO_GEOCODE	UNMAPPED	UNMAPPED
OO_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
PART_NUMBER	zx_lines_det_factors.product_code	First call: mtl_system_items.segment1 Second call: mtl_system_items.segment1 (for matched invoices only)
POINT_OF_TITLE_TRANSFER	UNMAPPED	UNMAPPED
PORT_OF_ENTRY	UNMAPPED	UNMAPPED
PORT_OF_LOADING	UNMAPPED	UNMAPPED
PRODUCT_CODE	zx_lines_det_factors.product_code	First call: mtl_system_items.segment1 Second call: mtl_system_items.segment1 (for matched invoices only)
QUANTITY	zx_lines_det_factors.trx_line_quantity	Both calls: ap_invoice_distributions_all.quantity_invoiced
REGIME	UNMAPPED	UNMAPPED
SP_COUNTRY	UNMAPPED	UNMAPPED
SP_PROVINCE	UNMAPPED	UNMAPPED
SP_STATE	UNMAPPED	UNMAPPED
SP_COUNTY	UNMAPPED	UNMAPPED
SP_CITY	UNMAPPED	UNMAPPED
SP_DISTRICT	UNMAPPED	UNMAPPED
SP_POSTCODE	UNMAPPED	UNMAPPED
SP_GEOCODE	UNMAPPED	UNMAPPED
SP_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
SF_COUNTRY	zx_ptnr_location_info_gt.ship_from_geography values	Both calls: ap_supplier_sites_all.country
SF_PROVINCE	zx_ptnr_location_info_gt.ship_from_geography values	Both calls: ap_supplier_sites_all.province
SF_STATE	zx_ptnr_location_info_gt.ship_from_geography values	Both calls: ap_supplier_sites_all.state
SF_COUNTY	zx_ptnr_location_info_gt.ship_from_geography values	Both calls: ap_supplier_sites_all.county
SF_CITY	zx_ptnr_location_info_gt.ship_from_geography values	Both calls: ap_supplier_sites_all.city
SF_DISTRICT	UNMAPPED	UNMAPPED
SF_POSTCODE	zx_ptnr_location_info_gt.ship_from_geography values	Both calls: ap_supplier_sites_all.zip (1st five chars for USA)
SF_GEOCODE	zx_ptnr_location_info_gt.ship_from_geography values	Both calls: ap_supplier_sites_all.zip (chars 7-10 for USA)
SF_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
ST_COUNTRY	zx_ptnr_location_info_gt.ship_to_geography values	Both calls: hr_locations_all.country or value from ADHOC
ST_PROVINCE	zx_ptnr_location_info_gt.ship_to_geography values	Both calls: hr_locations_all.country or value from ADHOC
ST_STATE	zx_ptnr_location_info_gt.ship_to_geography values	Both calls: hr_locations_all.country or value from ADHOC

Determination Line-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
ST_COUNTY	zx_ptnr_location_info_gt.ship_to_geography values	Both calls: hr_locations_all.region_2 (US only)
ST_CITY	zx_ptnr_location_info_gt.ship_to_geography values	Both calls: hr_locations_all.town_or_city
ST_DISTRICT	UNMAPPED	UNMAPPED
ST_POSTCODE	zx_ptnr_location_info_gt.ship_to_geography values	Both calls: hr_locations_all.postal_code (1st five chars for USA)
ST_GEOCODE	zx_ptnr_location_info_gt.ship_to_geography values	Both calls: hr_locations_all.postal_code (chars 7-10 for USA)
ST_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
SUPPLEMENTARY_UNIT	UNMAPPED	UNMAPPED
SU_COUNTRY	UNMAPPED	UNMAPPED
SU_PROVINCE	UNMAPPED	UNMAPPED
SU_STATE	UNMAPPED	UNMAPPED
SU_COUNTY	UNMAPPED	UNMAPPED
SU_CITY	UNMAPPED	UNMAPPED
SU_DISTRICT	UNMAPPED	UNMAPPED
SU_POSTCODE	UNMAPPED	UNMAPPED
SU_GEOCODE	UNMAPPED	UNMAPPED
SU_COMPANY_BRANCH_ID	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_COUNTRY	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_PROVINCE	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_STATE	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_COUNTY	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_CITY	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_DISTRICT	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_POSTCODE	UNMAPPED	UNMAPPED
SUPPLY_EXEMPT_PERCENT_GEOCODE	UNMAPPED	UNMAPPED
SUPPLY_TYPE	UNMAPPED	UNMAPPED
TAX_AMOUNT	NULL	First call: NULL Second call: if invoice total tax = 0, then NULL else amount to accrue For VAT, handle with recoverability
TAX_AMOUNT_FUNC	UNMAPPED	UNMAPPED
TAX_CODE	Zx_lines_det_factors.input_tax_classification_code	UNMAPPED
TAX_TYPE_ALL	UNMAPPED	UNMAPPED
TAX_TYPE_COUNTRY	UNMAPPED	UNMAPPED
TAX_TYPE_PROVINCE	UNMAPPED	UNMAPPED
TAX_TYPE_STATE	UNMAPPED	UNMAPPED
TAX_TYPE_COUNTY	UNMAPPED	UNMAPPED
TAX_TYPE_CITY	UNMAPPED	UNMAPPED
TAX_TYPE_DISTRICT	UNMAPPED	UNMAPPED
TAX_TYPE_POSTCODE	UNMAPPED	UNMAPPED
TAX_TYPE_GEOCODE	UNMAPPED	UNMAPPED
TRANSACTION_TYPE	UNMAPPED	UNMAPPED
BUYER_EST_BILL_TO	UNMAPPED	UNMAPPED

Determination Line-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
BUYER_EST_BUYER_PRIMARY	UNMAPPED	UNMAPPED
BUYER_EST_MIDDLEMAN	UNMAPPED	UNMAPPED
BUYER_EST_ORDER_ORIGIN	UNMAPPED	UNMAPPED
BUYER_EST_ORDER_ACCEPTANCE	UNMAPPED	UNMAPPED
BUYER_EST_SELLER_PRIMARY	UNMAPPED	UNMAPPED
BUYER_EST_SHIP_FROM	UNMAPPED	UNMAPPED
BUYER_EST_SHIP_TO	UNMAPPED	UNMAPPED
BUYER_EST_SUPPLY	UNMAPPED	UNMAPPED
MIDDLEMAN_EST_BILL_TO	UNMAPPED	UNMAPPED
MIDDLEMAN _EST_BUYER_PRIMARY	UNMAPPED	UNMAPPED
MIDDLEMAN _EST_MIDDLEMAN	UNMAPPED	UNMAPPED
MIDDLEMAN _EST_ORDER_ORIGIN	UNMAPPED	UNMAPPED
MIDDLEMAN_EST_ORDER_ ACCEPTANCE	UNMAPPED	UNMAPPED
MIDDLEMAN _EST_SELLER_PRIMARY	UNMAPPED	UNMAPPED
MIDDLEMAN _EST_SHIP_FROM	UNMAPPED	UNMAPPED
MIDDLEMAN _EST_SHIP_TO	UNMAPPED	UNMAPPED
MIDDLEMAN _EST_SUPPLY	UNMAPPED	UNMAPPED
SELLER_EST_BILL_TO	UNMAPPED	UNMAPPED
SELLER_EST_BUYER_PRIMARY	UNMAPPED	UNMAPPED
SELLER_EST_MIDDLEMAN	UNMAPPED	UNMAPPED
SELLER_EST_ORDER_ORIGIN	UNMAPPED	UNMAPPED
SELLER_EST_ORDER_ACCEPTANCE	UNMAPPED	UNMAPPED
SELLER_EST_SELLER_PRIMARY	UNMAPPED	UNMAPPED
SELLER_EST_SHIP_FROM	UNMAPPED	UNMAPPED
SELLER_EST_SHIP_TO	UNMAPPED	UNMAPPED
SELLER_EST_SUPPLY	UNMAPPED	UNMAPPED
IS_SIMPLIFICATION	UNMAPPED	UNMAPPED
GROSS_PLUS_TAX	UNMAPPED	UNMAPPED
GROSS_PLUS_TAX_FUNC	UNMAPPED	UNMAPPED
MIDDLEMAN_MARKUP_AMOUNT	UNMAPPED	UNMAPPED
MIDDLEMAN_MARKUP_AMOUNT_ FUNC	UNMAPPED	UNMAPPED
MIDDLEMAN_MARKUP_RATE	UNMAPPED	UNMAPPED
UNIQUE_LINE_NUMBER	Zx_lines_det_factors.trx_line_id	UNMAPPED
UNIT_OF_MEASURE	Zx_lines_det_factors.uom_code	If PO or receipt matched, Po_lines.unit_meas_lookup_code, else null
VAT_GROUP_REGISTRATION	UNMAPPED	UNMAPPED
VENDOR_NAME	UNMAPPED for PO and AP Online Call Requisitions: substr(po_vendors.vendor_name,1,100)	UNMAPPED
VENDOR_NUMBER	UNMAPPED for PO and AP Online Call Requisitions: substr(po_vendors.vendor_number,1,20)	UNMAPPED
VENDOR_TAX	UNMAPPED	UNMAPPED
VENDOR_TAX_FUNC	UNMAPPED	UNMAPPED
USER_ELEMENT_ATTRIBUTE1 – 30	UNMAPPED	UNMAPPED

Determination Line-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO and AP Online Call**	**AP First and Second Calls to Determination**
USER_ELEMENT.ATTRIBUTE31	ap_invoice_distributions.project_id	ap_invoice_distributions.project_id
USER_ELEMENT.ATTRIBUTE32	ap_invoice_distributions.task_id	ap_invoice_distributions.task_id
USER_ELEMENT.ATTRIBUTE33	ap_invoice_distributions.expenditure_item_date	ap_invoice_distributions.expenditure_item_date
USER_ELEMENT.ATTRIBUTE34	ap_invoice_distributions.expenditure_organization_id	ap_invoice_distributions.expenditure_organization_id
USER_ELEMENT.ATTRIBUTE35	ap_invoice_distributions.pa_quantity	ap_invoice_distributions.pa_quantity
USER_ELEMENT.ATTRIBUTE36	ap_invoice_distributions.expenditure_type	ap_invoice_distributions.expenditure_type
USER_ELEMENT.ATTRIBUTE41	Zx_lines_det_factors.trx_line_id	Both calls: ap_invoice_distributions_all.invoice_distribution_id
USER_ELEMENT.ATTRIBUTE42	Zx_lines_det_factors.sf_location_id	Both calls: txn.sf_address_id
USER_ELEMENT.ATTRIBUTE43	i_calc_txn.ship_to_loc_id	Both calls: trx.st_address_id
USER_ELEMENT.ATTRIBUTE44	i_calc_txn.ship_from_loc_id	Both calls: ap_invoices_all.tax_amount
USER_ELEMENT.ATTRIBUTE45	Substr(zx_ptnr_location_info_gt.ship_to_party_name, 1, 200)	Both calls: po_line_locations_all.line_location_id
USER_ELEMENT.ATTRIBUTE46	zx_ptnr_location_info_gt.ship_to_party_number	Both calls: ap_invoice_distributions_all.distribution_id
USER_ELEMENT.ATTRIBUTE47	i_calc_txn.entity_code	Both calls: ap_invoice_distributions_all. Line_type_lookup_code
USER_ELEMENT.ATTRIBUTE48	Zx_lines_det_factors.trx_line_type	Reserved
USER_ELEMENT.ATTRIBUTE49	UNMAPPED	Reserved
USER_ELEMENT.ATTRIBUTE50	Zx_lines_det_factors.input_tax_classification_code	Reserved
CALCULATED_UNIT_AMOUNT	UNMAPPED	UNMAPPED
CUSTOMER_GROUP_NAME	UNMAPPED	UNMAPPED
CUSTOMER_GROUP_OWNER	UNMAPPED	UNMAPPED
END_USE	UNMAPPED	UNMAPPED
INVOICE_DATE	UNMAPPED	UNMAPPED
IS_CREDIT	UNMAPPED	UNMAPPED
TAX_DETERMINATION_DATE	UNMAPPED	UNMAPPED
TAX_EXCHANGE_RATE_DATE	UNMAPPED	UNMAPPED
TAX_POINT_DATE	UNMAPPED	UNMAPPED

AP: REGISTRATION-LEVEL INPUT XML MAPPED TO ORACLE

Determination Registration-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO**	**AP First and Second Calls to Determination**
LINE_ID	-1	Sequence-generated ID
MERCHANT_ROLE	*S*	*S*
REGISTRATION_NUMBER	Zx_registrations.registration_number	Zx_registrations.registration_number

AP:LICENSE_IN INVOICE-LEVEL INPUT XML MAPPED TO ORACLE

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field	**PO**	**AP First and Second Calls to Determination**
LICENSE_NUMBER	UNMAPPED	UNMAPPED
LICENSE_TYPE	UNMAPPED	UNMAPPED
LICENSE_NUMBER_TYPE	UNMAPPED	UNMAPPED

AP: DETERMINATION OUTPUT MAPPED TO ORACLE TABLE AP_INVOICE_LINES_ALL

Determination Output XML Mapped to Oracle Table ap_invoice_lines_all		
Column Name	**Value**	**Comment**
INVOICE_ID	Invoice_id	
LINE_NUMBER	New_invoice_line_number	Max line number + 1 for the invoice.
LINE_TYPE_LOOKUP_CODE	*TAX*	
TAX_REGIME_CODE	*'SBX_REGIME_ '+Taxable Country*	
TAX_JURISDICTION_CODE	zx_jurisdictions_b.tax_jurisdiction_code	
TAX_STATUS_CODE	zx_status_b.tax_status_code	
TAX	zx_taxes_b.tax	
TAX_RATE_CODE	zx_rates_b.tax_rate_code	
ACCOUNTING_DATE	Accounting_date	If the invoice has been posted, this is the invoice gl_date otherwise the accounting date of the associated line item. If accounting date not open, searches for next available open period. If none,error.
SET_OF_BOOKS_ID	value from item: set_of_books_id	
AMOUNT	Value established by Determination	Based upon tax calculations.
WFAPPROVAL_STATUS	*NOT REQUIRED*	
CREATED_BY	G_sabrix_user	User ID for user SABRIX from FND_USER.
CREATION_DATE	sysdate	

Determination Output XML Mapped to Oracle Table ap_invoice_lines_all		
Column Name	**Value**	**Comment**
LAST_UPDATED_BY	G_sabrix_user	User ID for user SABRIX from FND_USER.
LAST_UPDATE_DATE	Sysdate	
DESCRIPTION	L_description	Dependent upon action.
LINE_SOURCE	*SABRIX*	
ORG_ID	value from item: org_id	
GENERATE_DISTS	N	
MATCH_TYPE	*NOT_MATCHED*	
DEFAULT_DIST_CCID	Value established by Determination	Based upon configurations.
PERIOD_NAME	value from item: period_name	
DEFERRED_ACCTG_FLAG	*N*	
ORIGINAL_AMOUNT	Value established by Determination	Based upon tax calculations.
ASSETS_TRACKING_FLAG	value from item: assets_tracking_flag	
PROGRAM_APPLICATION_ID	Fnd_global.prog_appl_id	
PROGRAM_ID	Fnd_global.conc_program_id	
PROGRAM_UPDATE_DATE	Sysdate	
REQUEST_ID	Fnd_global.conc_request_id	
TAX_ALREADY_CALCULATED_FLAG	*Y*	

AP: DETERMINATION OUTPUT MAPPED TO ORACLE TABLE AP_INVOICE_DISTRIBUTIONS_ALL

Determination Output XML Mapped to Oracle Table ap_invoice_distributions_all		
Column Name	**Value**	**Comment**
ACCOUNTING_DATE	accounting_date	If the invoice has been posted, this is the invoice gl_date otherwise the accounting date of the associated line item. If accounting date not open, searches for next available open period. If none, error.
ACCRUAL_POSTED_FLAG	*N*	
ASSETS_ADDITION_FLAG	*N*	*P* for accruals
ASSETS_TRACKING_FLAG	value from item: assets_tracking_flag	*N* for accruals
CASH_POSTED_FLAG	*N*	
DISTRIBUTION_LINE_NUMBER	*1*	
INVOICE_LINE_NUMBER	New_invoice_line_number	Max line number + 1 for the invoice.
DIST_CODE_COMBINATION_ID	<Determination determined value>	Based upon configurations.
INVOICE_ID	value from item: invoice_id	
LAST_UPDATED_BY	G_sabrix_user	User ID for user SABRIX from FND_USER.
LAST_UPDATE_DATE	sysdate	Current date/time.
LINE_TYPE_LOOKUP_CODE	L_line_type	If recoverable, *REC_TAX*, else *NONREC_TAX*.
PERIOD_NAME	value from item: period_name	
SET_OF_BOOKS_ID	value from item: set_of_books_id	
ACCTS_PAY_CODE_COMBINATION_ID	value from item: accts_pay_code_combination_id	
AMOUNT	Value established by Determination	Based upon tax calculations.
BATCH_ID	value from item: batch_id	
CREATED_BY	g_sabrix_user	User ID for user SABRIX from FND_USER.
CREATION_DATE	sysdate	Current date/time.
PROGRAM_APPLICATION_ID	Fnd_global.prog_appl_id	
PROGRAM_ID	Fnd_global.conc_program_id	
PROGRAM_UPDATE_DATE	Sysdate	
DESCRIPTION	Value established by Determination	Based upon the type of distribution: Shortpay, Expense or Accrual, Recoverable.
LAST_UPDATE_LOGIN	*-1*	
MATCH_STATUS_FLAG	*A*	
POSTED_FLAG	*N*	
QUANTITY_INVOICED	NULL	
ORG_ID	value from item: org_id	
REVERSAL_FLAG	value from item: reversal_flag	
PROJECT_ID	value from item: project_id	
TASK_ID	value from item: task_id	
EXPENDITURE_ITEM_DATE	value from item: expenditure_item_date	
EXPENDITURE_ORGANIZATION_ID	value from item: expenditure_organization_id	
PA_QUANTITY	value from item: pa_quantity	

Determination Output XML Mapped to Oracle Table ap_invoice_distributions_all		
Column Name	**Value**	**Comment**
EXPENDITURE_TYPE	value from item: expenditure_type	
PROJECT_ACCOUNTING_CONTEXT	Value from item: Project_accounting_context	
PA_ADDITION_FLAG	*N*	
ATTRIBUTE_CATEGORY	value from item: attribute_category	
TAX_CODE_ID	zx_rates_b.tax_rate_id	
INVOICE_DISTRIBUTION_ID	ap.ap_invoice_distributions_s. nextval	
REC_NREC_RATE	input_recovery_percent	
RECOVERY_RATE_NAME	tax percentage rate name plus '_REC_RATE'	
TAX_RECOVERABLE_FLAG	Depends on value of tax_recovery_rate	If tax_recovery_rate = 0 then NULL, else *Y*
ATTRIBUTEn	*Sabrix Tax Accrual* or *Sabrix Tax Expense*	The specific attribute is dependent upon the value entered during installation of the Integration. The value is dependent upon the specific type of distribution being created.
TYPE_1099	value from item: type_1099	
INCOME_TAX_REGION	value from item: income_tax_region	

O2C MAPPING

O2C: INVOICE-LEVEL INPUT XML MAPPED TO ORACLE

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field SABRIX_INVOICE Columns	**Oracle Source**	**Comments**
BATCH_ID	Autogenerated from sequence.	
INVOICE_ID	Always *1*.	
CREATION_DATE	System generated.	
MERCHANT_ID	UNMAPPED	
MERCHANT_NAME	UNMAPPED	
CALLING_SYSTEM_NUMBER	*222* (application_id of AR), *660* (application_id of OM).	
HOST_SYSTEM	instance_name from v$instance.	
EXTERNAL_COMPANY_ID	Profile value of SABRIX_COMPANY.	
MERCHANT_ROLE	*S*	
STATISTICS_NAME	UNMAPPED	
STATISTICS_VALUE	UNMAPPED	
XML_GROUP_NAME	UNMAPPED	
XML_GROUP_OWNER	UNMAPPED	
USERNAME	UNMAPPED	
PASSWORD	UNMAPPED	
ALLOCATION_GROUP_NAME	UNMAPPED	
ALLOCATION_GROUP_OWNER	UNMAPPED	
ALLOCATION_NAME	UNMAPPED	
AUTO_CREATE_CERTIFICATES	N	
AUTO_CREATE_CUSTOMERS	UNMAPPED	
BASIS_PERCENT	UNMAPPED	
VENDOR_NAME	UNMAPPED	
VENDOR_NUMBER	UNMAPPED	
PRODUCT_MAPPING_GROUP_OWNER	UNMAPPED	
PRODUCT_MAPPING_GROUP_NAME	UNMAPPED	
BT_COUNTRY	zx_ptnr_location_info_gt.billto_ geography_value#	Determination expects E-Business Tax to correctly populate these fields. When the fields are not populated correctly (or at all), the Integration falls back on the HZ_LOCATION and HR_LOCATION tables. The HZ_LOCATION table has precedence over the HR_LOCATION table.
BT_PROVINCE	zx_ptnr_location_info_gt.billto_ geography_value#	
BT_STATE	zx_ptnr_location_info_gt.billto_ geography_value#	
BT_COUNTY	zx_ptnr_location_info_gt.billto_ geography_value#	
BT_CITY	zx_ptnr_location_info_gt.billto_ geography_value#	
BT_DISTRICT	zx_ptnr_location_info_gt.billto_ geography_value#	

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field SABRIX_INVOICE Columns	**Oracle Source**	**Comments**
BT_POSTCODE	zx_ptnr_location_info_gt.billto_ geography_value#	if US, then SUBSTR(POSTAL_ CODE, 1,5)
BT_GEOCODE	substr(POSTAL_CODE, 6,4)	only if US, ZIP+4
BT_IS_BONDED	UNMAPPED	
BT_LOCATION_TAX_CATEGORY	UNMAPPED	
BT_COMPANY_BRANCH_ID	UNMAPPED	
BP_COUNTRY	UNMAPPED	
BP_PROVINCE	UNMAPPED	
BP_STATE	UNMAPPED	
BP_COUNTY	UNMAPPED	
BP_CITY	UNMAPPED	
BP_DISTRICT	UNMAPPED	
BP_POSTCODE	UNMAPPED	
BP_GEOCODE	UNMAPPED	
BP_IS_BONDED	UNMAPPED	
BP_LOCATION_TAX_CATEGORY	UNMAPPED	
BP_COMPANY_BRANCH_ID	UNMAPPED	
CALCULATION_DIRECTION	UNMAPPED	
COUNTRY_OF_ORIGIN	UNMAPPED	
CURRENCY_CODE	*0*	
CUSTOMER_NAME	zx_ptnr_location_info_gt.bill_to_party_name	
CUSTOMER_NUMBER	zx_ptnr_location_info_gt.bill_to_party_number	
DEPT_OF_CONSIGN	UNMAPPED	
END_USER_NAME	UNMAPPED	
EXEMPT_AMOUNT_COUNTRY	UNMAPPED	
EXEMPT_AMOUNT_PROVINCE	UNMAPPED	
EXEMPT_AMOUNT_STATE	UNMAPPED	
EXEMPT_AMOUNT_COUNTY	UNMAPPED	
EXEMPT_AMOUNT_CITY	UNMAPPED	
EXEMPT_AMOUNT_DISTRICT	UNMAPPED	
EXEMPT_AMOUNT_POSTCODE	UNMAPPED	
EXEMPT_AMOUNT_GEOCODE	UNMAPPED	
EXEMPT_AMOUNT_COUNTRY_FUNC	UNMAPPED	
EXEMPT_AMOUNT_PROVINCE_FUNC	UNMAPPED	
EXEMPT_AMOUNT_STATE_FUNC	UNMAPPED	
EXEMPT_AMOUNT_COUNTY_FUNC	UNMAPPED	
EXEMPT_AMOUNT_CITY_FUNC	UNMAPPED	

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field SABRIX_INVOICE Columns	**Oracle Source**	**Comments**
EXEMPT_AMOUNT_DISTRICT_FUNC	UNMAPPED	
EXEMPT_AMOUNT_POSTCODE_FUNC	UNMAPPED	
EXEMPT_AMOUNT_GEOCODE_FUNC	UNMAPPED	
EXEMPT_CERTIFICATE_COUNTRY	UNMAPPED	
EXEMPT_CERTIFICATE_PROVINCE	UNMAPPED	
EXEMPT_CERTIFICATE_STATE	UNMAPPED	
EXEMPT_CERTIFICATE_COUNTY	UNMAPPED	
EXEMPT_CERTIFICATE_CITY	UNMAPPED	
EXEMPT_CERTIFICATE_DISTRICT	UNMAPPED	
EXEMPT_CERTIFICATE_POSTCODE	UNMAPPED	
EXEMPT_CERTIFICATE_GEOCODE	UNMAPPED	
EXEMPT_REASON_COUNTRY	UNMAPPED	
EXEMPT_REASON_PROVINCE	UNMAPPED	
EXEMPT_REASON_STATE	UNMAPPED	
EXEMPT_REASON_COUNTY	UNMAPPED	
EXEMPT_REASON_CITY	UNMAPPED	
EXEMPT_REASON_DISTRICT	UNMAPPED	
EXEMPT_REASON_POSTCODE	UNMAPPED	
EXEMPT_REASON_GEOCODE	UNMAPPED	
FILTER_GROUP_NAME	UNMAPPED	
FILTER_GROUP_OWNER	UNMAPPED	
FISCAL_DATE	zx_ptnr_location_info_gt.trx_line_gl_date	
FREIGHT_ON_BOARD	UNMAPPED	
GROSS_AMOUNT	UNMAPPED	
FUNCTIONAL_CURRENCY_CODE	UNMAPPED	
DELIVERY_TERMS	UNMAPPED	
INPUT_RECOVERY_TYPE	UNMAPPED	
INVOICE_DATE	zx_lines_det_factors.trx_date	
INVOICE_NUMBER	zx_lines_det_factors.trx_number	
IS_AUDITED	Based on Transaction Type; *Y* if application_id = *222* (AR), *N* if *660* (OM).	
IS_BUSINESS_SUPPLY	UNMAPPED	
IS_CREDIT	'Y' if gross_amount or tax_amount< 0	
IS_EXEMPT_ALL	UNMAPPED	
IS_EXEMPT_COUNTRY	UNMAPPED	
IS_EXEMPT_PROVINCE	UNMAPPED	
IS_EXEMPT_STATE	UNMAPPED	
IS_EXEMPT_COUNTY	UNMAPPED	

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field SABRIX_INVOICE Columns	**Oracle Source**	**Comments**
IS_EXEMPT_CITY	UNMAPPED	
IS_EXEMPT_DISTRICT	UNMAPPED	
IS_EXEMPT_POSTCODE	UNMAPPED	
IS_EXEMPT_GEOCODE	UNMAPPED	
IS_NO_TAX_ALL	UNMAPPED	
IS_NO_TAX_COUNTRY	UNMAPPED	
IS_NO_TAX_PROVINCE	UNMAPPED	
IS_NO_TAX_STATE	UNMAPPED	
IS_NO_TAX_COUNTY	UNMAPPED	
IS_NO_TAX_CITY	UNMAPPED	
IS_NO_TAX_DISTRICT	UNMAPPED	
IS_NO_TAX_POSTCODE	UNMAPPED	
IS_NO_TAX_GEOCODE	UNMAPPED	
IS_REPORTED	UNMAPPED	
IS_REVERSED	UNMAPPED	
IS_ROUNDING	UNMAPPED	
LOCATION_SHIP_FROM	UNMAPPED	
LOCATION_BILL_TO	UNMAPPED	
LOCATION_SUPPLY	UNMAPPED	
LOCATION_MIDDLEMAN	UNMAPPED	
LOCATION_ORDER_ORIGIN	UNMAPPED	
LOCATION_ORDER_ACCEPTANCE	UNMAPPED	
LOCATION_SHIP_TO	UNMAPPED	
LOCATION_SET	UNMAPPED	
MM_COUNTRY	UNMAPPED	
MM_PROVINCE	UNMAPPED	
MM_STATE	UNMAPPED	
MM_COUNTY	UNMAPPED	
MM_CITY	UNMAPPED	
MM_DISTRICT	UNMAPPED	
MM_POSTCODE	UNMAPPED	
MM_GEOCODE	UNMAPPED	
MM_IS_BONDED	UNMAPPED	
MM_LOCATION_TAX_CATEGORY	UNMAPPED	
MM_COMPANY_BRANCH_ID	UNMAPPED	
MODE_OF_TRANSPORT	UNMAPPED	
MOVEMENT_DATE	zx_lines_det_factors.transaction_shipping_date	
MOVEMENT_TYPE	UNMAPPED	

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field SABRIX_INVOICE Columns	**Oracle Source**	**Comments**
NATURE_OF_TRANSACTION_CODE	UNMAPPED	
OA_COUNTRY	zx_ptnr_location_info_gt.poa_ geography_value#	Determination expects E-Business Tax to correctly populate these fields. When the fields are not populated correctly (or at all), the Integration falls back on the HZ_LOCATION and HR_LOCATION tables. The HZ_LOCATION table has precedence over the HR_LOCATION table.
OA_PROVINCE	zx_ptnr_location_info_gt.poa_ geography_value#	
OA_STATE	zx_ptnr_location_info_gt.poa_ geography_value#	
OA_COUNTY	zx_ptnr_location_info_gt.poa_ geography_value#	
OA_CITY	zx_ptnr_location_info_gt.poa_ geography_value#	
OA_DISTRICT	zx_ptnr_location_info_gt.poa_ geography_value#	
OA_POSTCODE	zx_ptnr_location_info_gt.poa_ geography_value#	if US, then SUBSTR(POSTAL_CODE, 1,5)
OA_GEOCODE	substr(POSTAL_CODE, 6,4)	only if US, ZIP+4
OA_IS_BONDED	UNMAPPED	
OA_LOCATION_TAX_CATEGORY	UNMAPPED	
OA_COMPANY_BRANCH_ID	UNMAPPED	
OO_COUNTRY	zx_ptnr_location_info_gt.poo_ geography_value#	Determination expects E-Business Tax to correctly populate these fields. When the fields are not populated correctly (or at all), the Integration falls back on the HZ_LOCATION and HR_LOCATION tables. The HZ_LOCATION table has precedence over the HR_LOCATION table.
OO_PROVINCE	zx_ptnr_location_info_gt.poo_ geography_value#	
OO_STATE	zx_ptnr_location_info_gt.poo_ geography_value#	
OO_COUNTY	zx_ptnr_location_info_gt.poo_ geography_value#	
OO_CITY	zx_ptnr_location_info_gt.poo_ geography_value#	
OO_DISTRICT	zx_ptnr_location_info_gt.poo_ geography_value#	
OO_POSTCODE	zx_ptnr_location_info_gt.poo_ geography_value#	if US, then SUBSTR(POSTAL_CODE, 1,5)
OO_GEOCODE	substr(POSTAL_CODE, 6,4)	only if US, ZIP+4
OO_IS_BONDED	UNMAPPED	
OO_LOCATION_TAX_CATEGORY	UNMAPPED	
OO_COMPANY_BRANCH_ID	UNMAPPED	
ORIGINAL_INVOICE_NUMBER	zx_lines_det_factors.related_ doc_number	
ORIGINAL_DOCUMENT_ID	zx_lines_det_factors.adjusted_ doc_transaction_id	
ORIGINAL_DOCUMENT_ITEM	UNMAPPED	
ORIGINAL_DOCUMENT_TYPE	UNMAPPED	
ORIGINAL_INVOICE_DATE	zx_lines_det_factors.adjusted_ doc_date	

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field SABRIX_INVOICE Columns	**Oracle Source**	**Comments**
ORIGINAL_MOVEMENT_DATE	zx_lines_det_factors.adjusted_ doc_date	
OVERRIDE_AMOUNT_COUNTRY	UNMAPPED	
OVERRIDE_AMOUNT_PROVINCE	UNMAPPED	
OVERRIDE_AMOUNT_STATE	UNMAPPED	
OVERRIDE_AMOUNT_COUNTY	UNMAPPED	
OVERRIDE_AMOUNT_CITY	UNMAPPED	
OVERRIDE_AMOUNT_DISTRICT	UNMAPPED	
OVERRIDE_AMOUNT_POSTCODE	UNMAPPED	
OVERRIDE_AMOUNT_GEOCODE	UNMAPPED	
OVERRIDE_AMOUNT_COUNTRY_FUNC	UNMAPPED	
OVERRIDE_AMOUNT_PROVINCE_FUNC	UNMAPPED	
OVERRIDE_AMOUNT_STATE_FUNC	UNMAPPED	
OVERRIDE_AMOUNT_COUNTY_FUNC	UNMAPPED	
OVERRIDE_AMOUNT_CITY_FUNC	UNMAPPED	
OVERRIDE_AMOUNT_DISTRICT_FUNC	UNMAPPED	
OVERRIDE_AMOUNT_POSTCODE_ FUNC	UNMAPPED	
OVERRIDE_AMOUNT_GEOCODE_FUNC	UNMAPPED	
OVERRIDE_RATE_COUNTRY	UNMAPPED	
OVERRIDE_RATE_PROVINCE	UNMAPPED	
OVERRIDE_RATE_STATE	UNMAPPED	
OVERRIDE_RATE_COUNTY	UNMAPPED	
OVERRIDE_RATE_CITY	UNMAPPED	
OVERRIDE_RATE_DISTRICT	UNMAPPED	
OVERRIDE_RATE_POSTCODE	UNMAPPED	
OVERRIDE_RATE_GEOCODE	UNMAPPED	
POINT_OF_TITLE_TRANSFER	UNMAPPED	
PORT_OF_ENTRY	UNMAPPED	
PORT_OF_LOADING	UNMAPPED	
REGIME	UNMAPPED	
SP_COUNTRY	UNMAPPED	
SP_PROVINCE	UNMAPPED	
SP_STATE	UNMAPPED	
SP_COUNTY	UNMAPPED	
SP_CITY	UNMAPPED	
SP_DISTRICT	UNMAPPED	
SP_POSTCODE	UNMAPPED	
SP_GEOCODE	UNMAPPED	

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field SABRIX_INVOICE Columns	**Oracle Source**	**Comments**
SP_IS_BONDED	UNMAPPED	
SP_LOCATION_TAX_CATEGORY	UNMAPPED	
SP_COMPANY_BRANCH_ID	UNMAPPED	
SF_COUNTRY	zx_ptnr_location_info_gt.ship_ from_geography_value#	Determination expects E-Business Tax to correctly populate these fields. When the fields are not populated correctly (or at all), the Integration falls back on the HZ_LOCATION and HR_LOCATION tables. The HZ_LOCATION table has precedence over the HR_LOCATION table.
SF_PROVINCE	zx_ptnr_location_info_gt.ship_ from_geography_value#	
SF_STATE	zx_ptnr_location_info_gt.ship_ from_geography_value#	
SF_COUNTY	zx_ptnr_location_info_gt.ship_ from_geography_value#	
SF_CITY	zx_ptnr_location_info_gt.ship_ from_geography_value#	
SF_DISTRICT	zx_ptnr_location_info_gt.ship_ from_geography_value#	
SF_POSTCODE	zx_ptnr_location_info_gt.ship_ from_geography_value#	if US, then SUBSTR(POSTAL_ CODE, 1,5)
SF_GEOCODE	substr(POSTAL_CODE, 6,4)	only if US, ZIP+4
SF_IS_BONDED	UNMAPPED	
SF_LOCATION_TAX_CATEGORY	UNMAPPED	
SF_COMPANY_BRANCH_ID	UNMAPPED	
ST_COUNTRY	zx_ptnr_location_info_gt.ship_ to_geography_value#	Determination expects E-Business Tax to correctly populate these fields. When the fields are not populated correctly (or at all), the Integration falls back on the HZ_LOCATION and HR_LOCATION tables. The HZ_LOCATION table has precedence over the HR_LOCATION table.
ST_PROVINCE	zx_ptnr_location_info_gt.ship_ to_geography_value#	
ST_STATE	zx_ptnr_location_info_gt.ship_ to_geography_value#	
ST_COUNTY	zx_ptnr_location_info_gt.ship_ to_geography_value#	
ST_CITY	zx_ptnr_location_info_gt.ship_ to_geography_value#	
ST_DISTRICT	zx_ptnr_location_info_gt.ship_ to_geography_value#	
ST_POSTCODE	zx_ptnr_location_info_gt.ship_ to_geography_value#	if US, then SUBSTR(POSTAL_ CODE, 1,5)
ST_GEOCODE	substr(POSTAL_CODE, 6,4)	only if US, ZIP+4
ST_IS_BONDED	UNMAPPED	
ST_LOCATION_TAX_CATEGORY	UNMAPPED	
ST_COMPANY_BRANCH_ID	UNMAPPED	
STATISTICAL_PROCEDURE	UNMAPPED	
SU_COUNTRY	UNMAPPED	
SU_PROVINCE	UNMAPPED	
SU_STATE	UNMAPPED	
SU_COUNTY	UNMAPPED	

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field SABRIX_INVOICE Columns	**Oracle Source**	**Comments**
SU_CITY	UNMAPPED	
SU_DISTRICT	UNMAPPED	
SU_POSTCODE	UNMAPPED	
SU_GEOCODE	UNMAPPED	
SU_IS_BONDED	UNMAPPED	
SU_LOCATION_TAX_CATEGORY	UNMAPPED	
SU_COMPANY_BRANCH_ID	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_COUNTRY	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_ PROVINCE	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_STATE	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_COUNTY	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_CITY	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_DISTRICT	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_ POSTCODE	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_ GEOCODE	UNMAPPED	
SUPPLY_TYPE	UNMAPPED	
TAX_AMOUNT	UNMAPPED	
TAX_CODE	UNMAPPED	
TAX_TYPE_ALL	UNMAPPED	
TAX_TYPE_COUNTRY	UNMAPPED	
TAX_TYPE_PROVINCE	UNMAPPED	
TAX_TYPE_STATE	UNMAPPED	
TAX_TYPE_COUNTY	UNMAPPED	
TAX_TYPE_CITY	UNMAPPED	
TAX_TYPE_DISTRICT	UNMAPPED	
TAX_TYPE_POSTCODE	UNMAPPED	
TAX_TYPE_GEOCODE	UNMAPPED	
TITLE_TRANSFER_LOCATION	UNMAPPED	
TRANSACTION_TYPE	*GS*	
IS_AUDIT_UPDATE	*Y* if zx_lines_det_factors.event_class_code = *INVOICE_ADJUSTMENT*	
BUYER_EST_BILL_TO	UNMAPPED	
BUYER_EST_BUYER_PRIMARY	UNMAPPED	
BUYER_EST_MIDDLEMAN	UNMAPPED	
BUYER_EST_ORDER_ORIGIN	UNMAPPED	
BUYER_EST_ORDER_ACCEPTANCE	UNMAPPED	
BUYER_EST_SELLER_PRIMARY	UNMAPPED	

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field SABRIX_INVOICE Columns	**Oracle Source**	**Comments**
BUYER_EST_SHIP_FROM	UNMAPPED	
BUYER_EST_SHIP_TO	UNMAPPED	
BUYER_EST_SUPPLY	UNMAPPED	
MIDDLEMAN_EST_BILL_TO	UNMAPPED	
MIDDLEMAN_EST_BUYER_PRIMARY	UNMAPPED	
MIDDLEMAN_EST_MIDDLEMAN	UNMAPPED	
MIDDLEMAN_EST_ORDER_ORIGIN	UNMAPPED	
MIDDLEMAN_EST_ORDER_ ACCEPTANCE	UNMAPPED	
MIDDLEMAN_EST_SELLER_PRIMARY	UNMAPPED	
MIDDLEMAN_EST_SHIP_FROM	UNMAPPED	
MIDDLEMAN_EST_SHIP_TO	UNMAPPED	
MIDDLEMAN_EST_SUPPLY	UNMAPPED	
SELLER_EST_BILL_TO	UNMAPPED	
SELLER_EST_BUYER_PRIMARY	UNMAPPED	
SELLER_EST_MIDDLEMAN	UNMAPPED	
SELLER_EST_ORDER_ORIGIN	UNMAPPED	
SELLER_EST_ORDER_ACCEPTANCE	UNMAPPED	
SELLER_EST_SELLER_PRIMARY	UNMAPPED	
SELLER_EST_SHIP_FROM	UNMAPPED	
SELLER_EST_SHIP_TO	UNMAPPED	
SELLER_EST_SUPPLY	UNMAPPED	
IS_SIMPLIFICATION	UNMAPPED	
UNIQUE_INVOICE_NUMBER	zx_lines_det_factors.transaction_id	
VENDOR_TAX	UNMAPPED	
VENDOR_TAX_FUNC	UNMAPPED	
US_PRODUCT_XREF_GROUP_OWNER	UNMAPPED	
US_PRODUCT_XREF_GROUP_NAME	UNMAPPED	
INTL_PRODUCT_XREF_GROUP_OWNER	UNMAPPED	
INTL_PRODUCT_XREF_GROUP_NAME	UNMAPPED	
US_REGISTRATION_GROUP_OWNER	UNMAPPED	
US_REGISTRATION_GROUP_NAME	UNMAPPED	
INTL_REGISTRATION_GROUP_OWNER	UNMAPPED	
INTL_REGISTRATION_GROUP_NAME	UNMAPPED	
VAT_GROUP_REGISTRATION	UNMAPPED	
USER_ELEMENT_ATTRIBUTE1	UNMAPPED	
USER_ELEMENT_ATTRIBUTE2	UNMAPPED	
USER_ELEMENT_ATTRIBUTE3	UNMAPPED	

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field SABRIX_INVOICE Columns	**Oracle Source**	**Comments**
USER_ELEMENT_ATTRIBUTE4	UNMAPPED	
USER_ELEMENT_ATTRIBUTE5	UNMAPPED	
USER_ELEMENT_ATTRIBUTE6	UNMAPPED	
USER_ELEMENT_ATTRIBUTE7	UNMAPPED	
USER_ELEMENT_ATTRIBUTE8	UNMAPPED	
USER_ELEMENT_ATTRIBUTE9	UNMAPPED	
USER_ELEMENT_ATTRIBUTE10	UNMAPPED	
USER_ELEMENT_ATTRIBUTE11	UNMAPPED	
USER_ELEMENT_ATTRIBUTE12	UNMAPPED	
USER_ELEMENT_ATTRIBUTE13	UNMAPPED	
USER_ELEMENT_ATTRIBUTE14	UNMAPPED	
USER_ELEMENT_ATTRIBUTE15	UNMAPPED	
USER_ELEMENT_ATTRIBUTE16	UNMAPPED	
USER_ELEMENT_ATTRIBUTE17	UNMAPPED	
USER_ELEMENT_ATTRIBUTE18	UNMAPPED	
USER_ELEMENT_ATTRIBUTE19	UNMAPPED	
USER_ELEMENT_ATTRIBUTE20	UNMAPPED	
USER_ELEMENT_ATTRIBUTE21	UNMAPPED	
USER_ELEMENT_ATTRIBUTE22	UNMAPPED	
USER_ELEMENT_ATTRIBUTE23	UNMAPPED	
USER_ELEMENT_ATTRIBUTE24	UNMAPPED	
USER_ELEMENT_ATTRIBUTE25	UNMAPPED	
USER_ELEMENT_ATTRIBUTE26	UNMAPPED	
USER_ELEMENT_ATTRIBUTE27	UNMAPPED	
USER_ELEMENT_ATTRIBUTE28	UNMAPPED	
USER_ELEMENT_ATTRIBUTE29	UNMAPPED	
USER_ELEMENT_ATTRIBUTE30	UNMAPPED	
USER_ELEMENT_ATTRIBUTE31	UNMAPPED	
USER_ELEMENT_ATTRIBUTE32	UNMAPPED	
USER_ELEMENT_ATTRIBUTE33	UNMAPPED	
USER_ELEMENT_ATTRIBUTE34	UNMAPPED	
USER_ELEMENT_ATTRIBUTE35	UNMAPPED	
USER_ELEMENT_ATTRIBUTE36	UNMAPPED	
USER_ELEMENT_ATTRIBUTE37	UNMAPPED	
USER_ELEMENT_ATTRIBUTE38	UNMAPPED	
USER_ELEMENT_ATTRIBUTE39	UNMAPPED	
USER_ELEMENT_ATTRIBUTE40	UNMAPPED	
USER_ELEMENT_ATTRIBUTE41	zx_lines_det_factors.transaction_id	

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field SABRIX_INVOICE Columns	**Oracle Source**	**Comments**
USER_ELEMENT_ATTRIBUTE42	zx_ptnr_location_info_gt.bill_to_party_number	
USER_ELEMENT_ATTRIBUTE43	zx_lines_det_factors.related_ doc_number	
USER_ELEMENT_ATTRIBUTE44	zx_lines_det_factors.event_type_code	
USER_ELEMENT_ATTRIBUTE45	UNMAPPED	
USER_ELEMENT_ATTRIBUTE46	UNMAPPED	
USER_ELEMENT_ATTRIBUTE47	UNMAPPED	
USER_ELEMENT_ATTRIBUTE48	UNMAPPED	
USER_ELEMENT_ATTRIBUTE49	Application ID	
USER_ELEMENT_ATTRIBUTE50	zx_lines_det_factors.event_class_mapping_id	
COMPANY_NAME	UNMAPPED	
HOST_REQUEST_ID	UNMAPPED	
HOST_REQUEST_LOG_ENTRY_ID	UNMAPPED	
AUDIT_MESSAGE_THRESHOLD	UNMAPPED	
CUSTOMER_GROUP_NAME	UNMAPPED	
CUSTOMER_GROUP_OWNER	UNMAPPED	
DOCUMENT_TYPE	UNMAPPED	
END_USE	UNMAPPED	
AUTHORITY_TYPE	UNMAPPED	
FULLY_INCLUSIVE	UNMAPPED	
IS_AUDITING_MESSAGES	UNMAPPED	
TAX_DETERMINATION_DATE	UNMAPPED	
TAX_EXCHANGE_RATE_DATE	UNMAPPED	
TAX_POINT_DATE	UNMAPPED	
VENDOR_TAX	UNMAPPED	

O2C: LINE-LEVEL INPUT XML MAPPED TO ORACLE

Determination Line-Level Input XML Mapped to Oracle Fields		
SABRIX_Line Columns	**Oracle Source**	**Comments**
BATCH_ID	Autogenerated from sequence.	
INVOICE_ID	*1*	
LINE_ID	Increment starting at 1.	
CREATION_DATE	SYSDATE	
ACCOUNTING_CODE	UNMAPPED	
ALLOCATION_GROUP_NAME	UNMAPPED	
ALLOCATION_GROUP_OWNER	UNMAPPED	
ALLOCATION_NAME	UNMAPPED	
IS_ALLOCATABLE	UNMAPPED	
BASIS_PERCENT	UNMAPPED	
BT_COUNTRY	zx_ptnr_location_info_gt.bill_to_geography_value#	Determination expects E-Business Tax to correctly populate thesefields. When the fields are not populated correctly (or at all), the Integration falls back on the HZ_LOCATION and HR_LOCATION tables. The HZ_LOCATION table has precedence over the HR_LOCATION table.
BT_PROVINCE	zx_ptnr_location_info_gt.bill_to_geography_value#	
BT_STATE	zx_ptnr_location_info_gt.bill_to_geography_value#	
BT_COUNTY	zx_ptnr_location_info_gt.bill_to_geography_value#	
BT_CITY	zx_ptnr_location_info_gt.bill_to_geography_value#	
BT_DISTRICT	zx_ptnr_location_info_gt.bill_to_geography_value#	
BT_POSTCODE	zx_ptnr_location_info_gt.bill_to_geography_value#	If US, then SUBSTR(POSTAL_CODE, 1,5)
BT_GEOCODE	substr(POSTAL_CODE, 6,4)	Only if US, ZIP+4
BT_IS_BONDED	UNMAPPED	
BT_LOCATION_TAX_CATEGORY	UNMAPPED	
BT_COMPANY_BRANCH_ID	UNMAPPED	
BP_COUNTRY	UNMAPPED	
BP_PROVINCE	UNMAPPED	
BP_STATE	UNMAPPED	
BP_COUNTY	UNMAPPED	
BP_CITY	UNMAPPED	
BP_DISTRICT	UNMAPPED	
BP_POSTCODE	UNMAPPED	
BP_GEOCODE	UNMAPPED	
BP_IS_BONDED	UNMAPPED	
BP_LOCATION_TAX_CATEGORY	UNMAPPED	
BP_COMPANY_BRANCH_ID	UNMAPPED	
COMMODITY_CODE	UNMAPPED	

Determination Line-Level Input XML Mapped to Oracle Fields		
SABRIX_Line Columns	**Oracle Source**	**Comments**
COUNTRY_OF_ORIGIN	UNMAPPED	
CUSTOMER_NAME	UNMAPPED	
CUSTOMER_NUMBER	UNMAPPED	
DEPT_OF_CONSIGN	UNMAPPED	
DESCRIPTION	zx_lines_det_factors.product_description	
DISCOUNT_AMOUNT	UNMAPPED	
EXEMPT_AMOUNT_COUNTRY	UNMAPPED	
EXEMPT_AMOUNT_PROVINCE	UNMAPPED	
EXEMPT_AMOUNT_STATE	UNMAPPED	
EXEMPT_AMOUNT_COUNTY	UNMAPPED	
EXEMPT_AMOUNT_CITY	UNMAPPED	
EXEMPT_AMOUNT_DISTRICT	UNMAPPED	
EXEMPT_AMOUNT_POSTCODE	UNMAPPED	
EXEMPT_AMOUNT_GEOCODE	UNMAPPED	
EXEMPT_AMOUNT_COUNTRY_FUNC	UNMAPPED	
EXEMPT_AMOUNT_PROVINCE_FUNC	UNMAPPED	
EXEMPT_AMOUNT_STATE_FUNC	UNMAPPED	
EXEMPT_AMOUNT_COUNTY_FUNC	UNMAPPED	
EXEMPT_AMOUNT_CITY_FUNC	UNMAPPED	
EXEMPT_AMOUNT_DISTRICT_FUNC	UNMAPPED	
EXEMPT_AMOUNT_POSTCODE_FUNC	UNMAPPED	
EXEMPT_AMOUNT_GEOCODE_FUNC	UNMAPPED	
EXEMPT_CERTIFICATE_COUNTRY	UNMAPPED	
EXEMPT_CERTIFICATE_PROVINCE	UNMAPPED	
EXEMPT_CERTIFICATE_STATE	UNMAPPED	
EXEMPT_CERTIFICATE_COUNTY	UNMAPPED	
EXEMPT_CERTIFICATE_CITY	UNMAPPED	
EXEMPT_CERTIFICATE_DISTRICT	UNMAPPED	
EXEMPT_CERTIFICATE_POSTCODE	UNMAPPED	
EXEMPT_CERTIFICATE_GEOCODE	UNMAPPED	
EXEMPT_REASON_COUNTRY	UNMAPPED	
EXEMPT_REASON_PROVINCE	UNMAPPED	
EXEMPT_REASON_STATE	UNMAPPED	
EXEMPT_REASON_COUNTY	UNMAPPED	
EXEMPT_REASON_CITY	UNMAPPED	
EXEMPT_REASON_DISTRICT	UNMAPPED	
EXEMPT_REASON_POSTCODE	UNMAPPED	
EXEMPT_REASON_GEOCODE	UNMAPPED	
DELIVERY_TERMS	zx_lines_det_factors.fob_point	

Determination Line-Level Input XML Mapped to Oracle Fields		
SABRIX_Line Columns	**Oracle Source**	**Comments**
FREIGHT_ON_BOARD	1st character of zx_lines_det_factors.fob_point	
GROSS_AMOUNT	zx_lines_det_factors.line_amount	
GROSS_AMOUNT_FUNC	UNMAPPED	
INCLUSIVE_TAX_AUTHORITY_TYPE	UNMAPPED	
INCLUSIVE_TAX_FULLY_INCLUSIVE	UNMAPPED	
INPUT_RECOVERY_TYPE	UNMAPPED	
INPUT_RECOVERY_AMOUNT	UNMAPPED	
INPUT_RECOVERY_AMOUNT_FUNC	UNMAPPED	
INPUT_RECOVERY_PERCENT	UNMAPPED	
IS_BUSINESS_SUPPLY	UNMAPPED	
IS_EXEMPT_ALL	UNMAPPED	
IS_EXEMPT_COUNTRY	UNMAPPED	
IS_EXEMPT_PROVINCE	UNMAPPED	
IS_EXEMPT_STATE	UNMAPPED	
IS_EXEMPT_COUNTY	UNMAPPED	
IS_EXEMPT_CITY	UNMAPPED	
IS_EXEMPT_DISTRICT	UNMAPPED	
IS_EXEMPT_POSTCODE	UNMAPPED	
IS_EXEMPT_GEOCODE	UNMAPPED	
IS_MANUFACTURING	UNMAPPED	
IS_NO_TAX_ALL	UNMAPPED	
IS_NO_TAX_COUNTRY	UNMAPPED	
IS_NO_TAX_PROVINCE	UNMAPPED	
IS_NO_TAX_STATE	UNMAPPED	
IS_NO_TAX_COUNTY	UNMAPPED	
IS_NO_TAX_CITY	UNMAPPED	
IS_NO_TAX_DISTRICT	UNMAPPED	
IS_NO_TAX_POSTCODE	UNMAPPED	
IS_NO_TAX_GEOCODE	UNMAPPED	
ITEM_VALUE	UNMAPPED	
ITEM_VALUE_FUNC	UNMAPPED	
LINE_NUMBER	AR --zx_lines_det_factors.trx_line_number OM -- line_id	
LOCATION_SHIP_FROM	UNMAPPED	
LOCATION_BILL_TO	UNMAPPED	
LOCATION_SUPPLY	UNMAPPED	
LOCATION_MIDDLEMAN	UNMAPPED	
LOCATION_ORDER_ORIGIN	UNMAPPED	

Determination Line-Level Input XML Mapped to Oracle Fields		
SABRIX_Line Columns	**Oracle Source**	**Comments**
LOCATION_ORDER_ACCEPTANCE	UNMAPPED	
LOCATION_SHIP_TO	UNMAPPED	
LOCATION_SET	UNMAPPED	
MASS	UNMAPPED	
MM_COUNTRY	UNMAPPED	
MM_PROVINCE	UNMAPPED	
MM_STATE	UNMAPPED	
MM_COUNTY	UNMAPPED	
MM_CITY	UNMAPPED	
MM_DISTRICT	UNMAPPED	
MM_POSTCODE	UNMAPPED	
MM_GEOCODE	UNMAPPED	
MM_IS_BONDED	UNMAPPED	
MM_LOCATION_TAX_CATEGORY	UNMAPPED	
MM_COMPANY_BRANCH_ID	UNMAPPED	
MODE_OF_TRANSPORT	UNMAPPED	
MOVEMENT_DATE	zx_lines_det_factors.transaction_ shipping_date	
MOVEMENT_TYPE	UNMAPPED	
OVERRIDE_AMOUNT_COUNTRY	UNMAPPED	
OVERRIDE_AMOUNT_PROVINCE	UNMAPPED	
OVERRIDE_AMOUNT_STATE	UNMAPPED	
OVERRIDE_AMOUNT_COUNTY	UNMAPPED	
OVERRIDE_AMOUNT_CITY	UNMAPPED	
OVERRIDE_AMOUNT_DISTRICT	UNMAPPED	
OVERRIDE_AMOUNT_POSTCODE	UNMAPPED	
OVERRIDE_AMOUNT_GEOCODE	UNMAPPED	
OVERRIDE_AMOUNT_COUNTRY_FUNC	UNMAPPED	
OVERRIDE_AMOUNT_PROVINCE_FUNC	UNMAPPED	
OVERRIDE_AMOUNT_STATE_FUNC	UNMAPPED	
OVERRIDE_AMOUNT_COUNTY_FUNC	UNMAPPED	
OVERRIDE_AMOUNT_CITY_FUNC	UNMAPPED	
OVERRIDE_AMOUNT_DISTRICT_FUNC	UNMAPPED	
OVERRIDE_AMOUNT_POSTCODE_FUNC	UNMAPPED	
OVERRIDE_AMOUNT_GEOCODE_FUNC	UNMAPPED	
OVERRIDE_RATE_COUNTRY	UNMAPPED	
OVERRIDE_RATE_PROVINCE	UNMAPPED	
OVERRIDE_RATE_STATE	UNMAPPED	
OVERRIDE_RATE_COUNTY	UNMAPPED	

Determination Line-Level Input XML Mapped to Oracle Fields		
SABRIX_Line Columns	**Oracle Source**	**Comments**
OVERRIDE_RATE_CITY	UNMAPPED	
OVERRIDE_RATE_DISTRICT	UNMAPPED	
OVERRIDE_RATE_POSTCODE	UNMAPPED	
OVERRIDE_RATE_GEOCODE	UNMAPPED	
ORIGINAL_INVOICE_NUMBER	zx_lines_det_factors.adjusted_doc_number	
ORIGINAL_DOCUMENT_ID	zx_lines_det_factors.adjusted_doc_transaction_id	
ORIGINAL_DOCUMENT_ITEM	UNMAPPED	
ORIGINAL_DOCUMENT_TYPE	UNMAPPED	
ORIGINAL_INVOICE_DATE	zx_lines_det_factors.adjusted_doc_date	
ORIGINAL_MOVEMENT_DATE	UNMAPPED	
OA_COUNTRY	zx_ptnr_location_info_gt.poa_geography_value#	Determination expects E-Business Tax to correctly populate thesefields. When the fields are not populated correctly (or at all), the Integration falls back on the HZ_LOCATION and HR_LOCATION tables. The HZ_LOCATION table has precedence over the HR_LOCATION table.
OA_PROVINCE	zx_ptnr_location_info_gt.poa_geography_value#	
OA_STATE	zx_ptnr_location_info_gt.poa_geography_value#	
OA_COUNTY	zx_ptnr_location_info_gt.poa_geography_value#	
OA_CITY	zx_ptnr_location_info_gt.poa_geography_value#	
OA_DISTRICT	zx_ptnr_location_info_gt.poa_geography_value#	
OA_POSTCODE	zx_ptnr_location_info_gt.poa_geography_value#	if US, then SUBSTR(POSTAL_CODE, 1,5)
OA_GEOCODE	substr(POSTAL_CODE, 6,4)	only if US, ZIP+4
OA_IS_BONDED	UNMAPPED	
OA_LOCATION_TAX_CATEGORY	UNMAPPED	
OA_COMPANY_BRANCH_ID	UNMAPPED	
OO_COUNTRY	zx_ptnr_location_info_gt.poo_geography_value#	Determination expects E-Business Tax to correctly populate thesefields. When the fields are not populated correctly (or at all), the Integration falls back on the HZ_LOCATION and HR_LOCATION tables. The HZ_LOCATION table has precedence over the HR_LOCATION table.
OO_PROVINCE	zx_ptnr_location_info_gt.poo_geography_value#	
OO_STATE	zx_ptnr_location_info_gt.poo_geography_value#	
OO_COUNTY	zx_ptnr_location_info_gt.poo_geography_value#	
OO_CITY	zx_ptnr_location_info_gt.poo_geography_value#	
OO_DISTRICT	zx_ptnr_location_info_gt.poo_geography_value#	
OO_POSTCODE	zx_ptnr_location_info_gt.poo_geography_value#	if US, then SUBSTR(POSTAL_CODE, 1,5)
OO_GEOCODE	substr(POSTAL_CODE, 6,4)	only if US, ZIP+4

Determination Line-Level Input XML Mapped to Oracle Fields		
SABRIX_Line Columns	**Oracle Source**	**Comments**
OO_IS_BONDED	UNMAPPED	
OO_LOCATION_TAX_CATEGORY	UNMAPPED	
OO_COMPANY_BRANCH_ID	UNMAPPED	
PART_NUMBER	UNMAPPED	
POINT_OF_TITLE_TRANSFER	UNMAPPED	
PORT_OF_ENTRY	UNMAPPED	
PORT_OF_LOADING	UNMAPPED	
PRODUCT_CODE	zx_lines_det_factors.product_code	
QUANTITY	zx_lines_det_factors.transaction_line_quantity	
REGIME	UNMAPPED	
RELATED_LINE_NUMBER	UNMAPPED	
SP_COUNTRY	UNMAPPED	
SP_PROVINCE	UNMAPPED	
SP_STATE	UNMAPPED	
SP_COUNTY	UNMAPPED	
SP_CITY	UNMAPPED	
SP_DISTRICT	UNMAPPED	
SP_POSTCODE	UNMAPPED	
SP_GEOCODE	UNMAPPED	
SP_IS_BONDED	UNMAPPED	
SP_LOCATION_TAX_CATEGORY	UNMAPPED	
SP_COMPANY_BRANCH_ID	UNMAPPED	
SF_COUNTRY	zx_ptnr_location_info_gt.ship_from_geography_value#	Determination expects E-Business Tax to correctly populate thesefields. When the fields are not populated correctly (or at all), the Integration falls back on the HZ_LOCATION and HR_LOCATION tables. The HZ_LOCATION table has precedence over the HR_LOCATION table.
SF_PROVINCE	zx_ptnr_location_info_gt.ship_from_geography_value#	
SF_STATE	zx_ptnr_location_info_gt.ship_from_geography_value#	
SF_COUNTY	zx_ptnr_location_info_gt.ship_from_geography_value#	
SF_CITY	zx_ptnr_location_info_gt.ship_from_geography_value#	
SF_DISTRICT	zx_ptnr_location_info_gt.ship_from_geography_value#	
SF_POSTCODE	zx_ptnr_location_info_gt.ship_from_geography_value#	if US, then SUBSTR(POSTAL_CODE, 1,5)
SF_GEOCODE	substr(POSTAL_CODE, 6,4)	only if US, ZIP+4
SF_IS_BONDED	UNMAPPED	
SF_LOCATION_TAX_CATEGORY	UNMAPPED	
SF_COMPANY_BRANCH_ID	UNMAPPED	

Determination Line-Level Input XML Mapped to Oracle Fields		
SABRIX_Line Columns	**Oracle Source**	**Comments**
ST_COUNTRY	zx_ptnr_location_info_gt.ship_to_ geography_value#	Determination expects E-Business Tax to correctly populate thesefields. When the fields are not populated correctly (or at all), the Integration falls back on the HZ_LOCATION and HR_LOCATION tables. The HZ_LOCATION table has precedence over the HR_LOCATION table.
ST_PROVINCE	zx_ptnr_location_info_gt.ship_to_ geography_value#	
ST_STATE	zx_ptnr_location_info_gt.ship_to_ geography_value#	
ST_COUNTY	zx_ptnr_location_info_gt.ship_to_ geography_value#	
ST_CITY	zx_ptnr_location_info_gt.ship_to_ geography_value#	
ST_DISTRICT	zx_ptnr_location_info_gt.ship_to_ geography_value#	
ST_POSTCODE	zx_ptnr_location_info_gt.ship_to_ geography_value#	if US, then SUBSTR(POSTAL_CODE, 1,5)
ST_GEOCODE	substr(POSTAL_CODE, 6,4)	only if US, ZIP+4
ST_IS_BONDED	UNMAPPED	
ST_LOCATION_TAX_CATEGORY	UNMAPPED	
ST_COMPANY_BRANCH_ID	UNMAPPED	
SUPPLEMENTARY_UNIT	UNMAPPED	
SU_COUNTRY	UNMAPPED	
SU_PROVINCE	UNMAPPED	
SU_STATE	UNMAPPED	
SU_COUNTY	UNMAPPED	
SU_CITY	UNMAPPED	
SU_DISTRICT	UNMAPPED	
SU_POSTCODE	UNMAPPED	
SU_GEOCODE	UNMAPPED	
SU_IS_BONDED	UNMAPPED	
SU_LOCATION_TAX_CATEGORY	UNMAPPED	
SU_COMPANY_BRANCH_ID	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_COUNTRY	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_ PROVINCE	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_STATE	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_COUNTY	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_CITY	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_DISTRICT	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_ POSTCODE	UNMAPPED	
SUPPLY_EXEMPT_PERCENT_ GEOCODE	UNMAPPED	
SUPPLY_TYPE	UNMAPPED	
TAX_AMOUNT	null	

Determination Line-Level Input XML Mapped to Oracle Fields		
SABRIX_Line Columns	**Oracle Source**	**Comments**
TAX_AMOUNT_FUNC	UNMAPPED	
TAX_TYPE	UNMAPPED	
TAX_CODE	zx_lines_det_factors.input_tax_classification_code	
TAX_TYPE_ALL	UNMAPPED	
TAX_TYPE_COUNTRY	UNMAPPED	
TAX_TYPE_PROVINCE	UNMAPPED	
TAX_TYPE_STATE	UNMAPPED	
TAX_TYPE_COUNTY	UNMAPPED	
TAX_TYPE_CITY	UNMAPPED	
TAX_TYPE_DISTRICT	UNMAPPED	
TAX_TYPE_POSTCODE	UNMAPPED	
TAX_TYPE_GEOCODE	UNMAPPED	
TITLE_TRANSFER_LOCATION	UNMAPPED	
TRANSACTION_TYPE	UNMAPPED	
BUYER_EST_BILL_TO	UNMAPPED	
BUYER_EST_BUYER_PRIMARY	UNMAPPED	
BUYER_EST_MIDDLEMAN	UNMAPPED	
BUYER_EST_ORDER_ORIGIN	UNMAPPED	
BUYER_EST_ORDER_ACCEPTANCE	UNMAPPED	
BUYER_EST_SELLER_PRIMARY	UNMAPPED	
BUYER_EST_SHIP_FROM	UNMAPPED	
BUYER_EST_SHIP_TO	UNMAPPED	
BUYER_EST_SUPPLY	UNMAPPED	
MIDDLEMAN_EST_BILL_TO	UNMAPPED	
MIDDLEMAN_EST_BUYER_PRIMARY	UNMAPPED	
MIDDLEMAN_EST_MIDDLEMAN	UNMAPPED	
MIDDLEMAN_EST_ORDER_ORIGIN	UNMAPPED	
MIDDLEMAN_EST_ORDER_ACCEPTANCE	UNMAPPED	
MIDDLEMAN_EST_SELLER_PRIMARY	UNMAPPED	
MIDDLEMAN_EST_SHIP_FROM	UNMAPPED	
MIDDLEMAN_EST_SHIP_TO	UNMAPPED	
MIDDLEMAN_EST_SUPPLY	UNMAPPED	
SELLER_EST_BILL_TO	UNMAPPED	
SELLER_EST_BUYER_PRIMARY	UNMAPPED	
SELLER_EST_MIDDLEMAN	UNMAPPED	
SELLER_EST_ORDER_ORIGIN	UNMAPPED	
SELLER_EST_ORDER_ACCEPTANCE	UNMAPPED	
SELLER_EST_SELLER_PRIMARY	UNMAPPED	

Determination Line-Level Input XML Mapped to Oracle Fields		
SABRIX_Line Columns	**Oracle Source**	**Comments**
SELLER_EST_SHIP_FROM	UNMAPPED	
SELLER_EST_SHIP_TO	UNMAPPED	
SELLER_EST_SUPPLY	UNMAPPED	
IS_SIMPLIFICATION	UNMAPPED	
GROSS_PLUS_TAX	UNMAPPED	
GROSS_PLUS_TAX_FUNC	UNMAPPED	
MIDDLEMAN_MARKUP_AMOUNT	UNMAPPED	
MIDDLEMAN_MARKUP_AMOUNT_FUNC	UNMAPPED	
MIDDLEMAN_MARKUP_RATE	UNMAPPED	
UNIQUE_LINE_NUMBER	zx_lines_det_factors.transaction_line_id	
UNIT_OF_MEASURE	zx_lines_det_factors.uom_code	
VAT_GROUP_REGISTRATION	UNMAPPED	
VENDOR_NAME	UNMAPPED	
VENDOR_NUMBER	UNMAPPED	
VENDOR_TAX	UNMAPPED	
VENDOR_TAX_FUNC	UNMAPPED	
USER_ELEMENT_ATTRIBUTE1	UNMAPPED	
USER_ELEMENT_ATTRIBUTE2	UNMAPPED	
USER_ELEMENT_ATTRIBUTE3	UNMAPPED	
USER_ELEMENT_ATTRIBUTE4	UNMAPPED	
USER_ELEMENT_ATTRIBUTE5	UNMAPPED	
USER_ELEMENT_ATTRIBUTE6	UNMAPPED	
USER_ELEMENT_ATTRIBUTE7	UNMAPPED	
USER_ELEMENT_ATTRIBUTE8	UNMAPPED	
USER_ELEMENT_ATTRIBUTE9	UNMAPPED	
USER_ELEMENT_ATTRIBUTE10	UNMAPPED	
USER_ELEMENT_ATTRIBUTE11	UNMAPPED	
USER_ELEMENT_ATTRIBUTE12	UNMAPPED	
USER_ELEMENT_ATTRIBUTE13	UNMAPPED	
USER_ELEMENT_ATTRIBUTE14	UNMAPPED	
USER_ELEMENT_ATTRIBUTE15	UNMAPPED	
USER_ELEMENT_ATTRIBUTE16	UNMAPPED	
USER_ELEMENT_ATTRIBUTE17	UNMAPPED	
USER_ELEMENT_ATTRIBUTE18	UNMAPPED	
USER_ELEMENT_ATTRIBUTE19	UNMAPPED	
USER_ELEMENT_ATTRIBUTE20	UNMAPPED	
USER_ELEMENT_ATTRIBUTE21	UNMAPPED	
USER_ELEMENT_ATTRIBUTE22	UNMAPPED	
USER_ELEMENT_ATTRIBUTE23	UNMAPPED	

Determination Line-Level Input XML Mapped to Oracle Fields		
SABRIX_Line Columns	**Oracle Source**	**Comments**
USER_ELEMENT_ATTRIBUTE24	UNMAPPED	
USER_ELEMENT_ATTRIBUTE25	UNMAPPED	
USER_ELEMENT_ATTRIBUTE26	UNMAPPED	
USER_ELEMENT_ATTRIBUTE27	UNMAPPED	
USER_ELEMENT_ATTRIBUTE28	UNMAPPED	
USER_ELEMENT_ATTRIBUTE29	UNMAPPED	
USER_ELEMENT_ATTRIBUTE30	UNMAPPED	
USER_ELEMENT_ATTRIBUTE31	UNMAPPED	
USER_ELEMENT_ATTRIBUTE32	UNMAPPED	
USER_ELEMENT_ATTRIBUTE33	UNMAPPED	
USER_ELEMENT_ATTRIBUTE34	UNMAPPED	
USER_ELEMENT_ATTRIBUTE35	UNMAPPED	
USER_ELEMENT_ATTRIBUTE36	UNMAPPED	
USER_ELEMENT_ATTRIBUTE37	UNMAPPED	
USER_ELEMENT_ATTRIBUTE38	UNMAPPED	
USER_ELEMENT_ATTRIBUTE39	UNMAPPED	
USER_ELEMENT_ATTRIBUTE40	UNMAPPED	
USER_ELEMENT_ATTRIBUTE41	zx_lines_det_factors.transaction_line_id	
USER_ELEMENT_ATTRIBUTE42	zx_lines_det_factors.product_id	
USER_ELEMENT_ATTRIBUTE43	zx_lines_det_factors.applied_ from_line_id	
USER_ELEMENT_ATTRIBUTE44	zx_lines_det_factors.ship_from_location _id	
USER_ELEMENT_ATTRIBUTE45	zx_ptnr_location_info_gt.ship_to_party_ name	
USER_ELEMENT_ATTRIBUTE46	zx_ptnr_location_info_gt.ship_to_party_ number	
USER_ELEMENT_ATTRIBUTE47	zx_lines_det_factors.entity_code	
USER_ELEMENT_ATTRIBUTE48	zx_lines_det_factors.transaction_line_ type	If Freight Line is from OM, where the List_Line_Type_Code = Freight_Charge, then the Charge_Type_Code is populated.
USER_ELEMENT_ATTRIBUTE49	zx_lines_det_factors.event_ class_code	
USER_ELEMENT_ATTRIBUTE50	zx_lines_det_factors.event_type_code	
CALCULATED_UNIT_AMOUNT	UNMAPPED	
CUSTOMER_GROUP_NAME	UNMAPPED	
CUSTOMER_GROUP_OWNER	UNMAPPED	
END_USE	UNMAPPED	
INVOICE_DATE	UNMAPPED	
IS_CREDIT	UNMAPPED	
TAX_DETERMINATION_DATE	UNMAPPED	

Determination Line-Level Input XML Mapped to Oracle Fields		
SABRIX_Line Columns	**Oracle Source**	**Comments**
TAX_EXCHANGE_RATE_DATE	UNMAPPED	
TAX_POINT_DATE	UNMAPPED	

O2C: LICENSE_IN INVOICE-LEVEL INPUT XML MAPPED TO ORACLE

Determination Invoice-Level Input XML Mapped to Oracle Fields		
Input XML Field SABRIX_LICENSE_IN COLUMNS	**Oracle Source**	**Comments**
LICENSE_NUMBER	UNMAPPED	UNMAPPED
LICENSE_TYPE	UNMAPPED	UNMAPPED
LICENSE_NUMBER_TYPE	UNMAPPED	UNMAPPED

O2C: LINE-LEVEL OUTPUT TAX MAPPING

Determination Line-Level Output Tax	
Column Name	**E-Business Tax Result Table**
BATCH_ID	UNMAPPED
INVOICE_ID	UNMAPPED
LINE_ID	tax_lines_result_tbl.trx_line_number
LINE_TAX_ID	tax_lines_result_tbl.tax_line_number, tax_line_id
CREATION_DATE	UNMAPPED
ADDRESS_TYPE	UNMAPPED
AUTHORITY_NAME	tax_lines_result_tbl.tax_jurisdiction
AUTHORITY_TYPE	UNMAPPED
AUTHORITY_CURRENCY_CODE	UNMAPPED
AUTHORITY_FIPS	UNMAPPED
AUTHORITY_OFFICIAL_NAME	UNMAPPED
AUTHORITY_CATEGORY	UNMAPPED
AUTHORITY_UUID	UNMAPPED
LINE_COMMENT	tax_lines_result_tbl.tax_code
ERP_TAX_CODE	tax_lines_result_tbl.tax_code
EXEMPT_CERTIFICATE_EXPIRE_DATE	UNMAPPED
EXEMPT_REASON	UNMAPPED
STEP_1_FROM_CURRENCY_CODE	tax_lines_result_tbl.tax_currency_code, trx_currency_code
STEP_1_TO_CURRENCY_CODE	UNMAPPED
STEP_1_EXCHANGE_RATE	UNMAPPED
STEP_2_FROM_CURRENCY_CODE	UNMAPPED
STEP_2_TO_CURRENCY_CODE	UNMAPPED
STEP_2_EXCHANGE_RATE	UNMAPPED
EXCHANGE_RATE_DATE	UNMAPPED
EXCHANGE_RATE_SOURCE	UNMAPPED
EU_TRANSACTION	UNMAPPED
EXEMPT_AUTHORITY_AMOUNT	UNMAPPED
EXEMPT_DOCUMENT_AMOUNT	UNMAPPED

Determination Line-Level Output Tax	
Column Name	**E-Business Tax Result Table**
EXEMPT_UNROUNDED_AUTHORITY_AMT	UNMAPPED
EXEMPT_UNROUNDED_DOCUMENT_AMT	UNMAPPED
FLAT_FEE	UNMAPPED
AUTHORITY_AMOUNT	UNMAPPED
UNROUNDED_AUTHORITY_AMOUNT	UNMAPPED
INCLUSIVE_TAX	UNMAPPED
INVOICE_DESCRIPTION	tax_lines_result_tbl.legal_justification_Text2
IS_TRIANGULATION	UNMAPPED
JURISDICTION_TEXT	tax_lines_result_tbl.legal_justification_Text1
REVISED_GROSS_AMOUNT	UNMAPPED
RULE_ORDER	UNMAPPED
RULE_REPORTING_CATEGORY	UNMAPPED
TAX_AMOUNT	tax_lines_results_tbl.tax_amount
TAX_AUTHORITY_AMOUNT	UNMAPPED
TAX_UNROUNDED_AUTHORITY_AMOUNT	UNMAPPED
TAX_UNROUNDED_DOCUMENT_AMOUNT	tax_lines_result_tbl.unrounded_tax_amount, tax_current_tax_amount
TAX_DETERMINATION_DATE	tax_lines_result_tbl.tax_determine_date
TAX_POINT_DATE	tax_lines_result_tbl.tax_date
TAX_AMOUNT_FUNC	UNMAPPED
TAX_DIRECTION	UNMAPPED
TAX_RATE	tax_lines_result_tbl.tax_rate_percentage
TAX_TYPE	UNMAPPED
TAX_TREATMENT	UNMAPPED
ZONE_LEVEL	UNMAPPED
ZONE_NAME	UNMAPPED
TAXABLE_BASIS	tax_lines_result_tbl.taxable_amount
BASIS_AUTHORITY_AMOUNT	UNMAPPED
BASIS_UNROUNDED_AUTHORITY_AMT	UNMAPPED
BASIS_UNROUNDED_DOCUMENT_AMT	UNMAPPED
ADMIN_ZONE_LEVEL	UNMAPPED
BASIS_PERCENT	UNMAPPED
CALCULATION_METHOD	UNMAPPED
EFFECTIVE_ZONE_LEVEL	UNMAPPED
EXEMPT_AMOUNT	tax_lines_result_tbl.exempt_amt
EXEMPT_AMOUNT_FUNC	UNMAPPED
EXEMPT_CERTIFICATE	tax_lines_result_tbl.exempt_certificate_number
INPUT_RECOVERY_AMOUNT	UNMAPPED
INPUT_RECOVERY_AMOUNT_FUNC	UNMAPPED
INPUT_RECOVERY_PERCENT	UNMAPPED

Determination Line-Level Output Tax	
Column Name	**E-Business Tax Result Table**
IS_EXEMPT	UNMAPPED
IS_INTRASTAT_REPORTED	UNMAPPED
IS_NOTAX	UNMAPPED
IS_VAT_REPORTED	UNMAPPED
IS_VIES_REPORTED	UNMAPPED
LOCATION_CODE	UNMAPPED
NATURE_OF_TAX	tax_lines_result_tbl.rate_type_code
OVERRIDE_AMOUNT	UNMAPPED
OVERRIDE_AMOUNT_FUNC	UNMAPPED
OVERRIDE_RATE	UNMAPPED
REPORTING_PERIOD	UNMAPPED
SUPPLY_EXEMPT_PERCENT	UNMAPPED
TAX_RATE_CODE	UNMAPPED
TAXABLE_COUNTRY	tax_lines_result_tbl.country_code
TAXABLE_COUNTRY_NAME	UNMAPPED
TAXABLE_PROVINCE	UNMAPPED
TAXABLE_STATE	tax_lines_result_tbl.state
TAXABLE_COUNTY	tax_lines_result_tbl.county
TAXABLE_CITY	tax_lines_result_tbl.city
TAXABLE_DISTRICT	UNMAPPED
TAXABLE_POSTCODE	UNMAPPED
TAXABLE_GEOCODE	UNMAPPED
BUYER_REGISTRATION	UNMAPPED
SELLER_REGISTRATION	UNMAPPED
MIDDLEMAN_REGISTRATION	UNMAPPED
FISCAL_REP_NAME	UNMAPPED
FISCAL_REP_ADDRESS1	UNMAPPED
FISCAL_REP_ADDRESS2	UNMAPPED
FISCAL_REP_CONTACT	UNMAPPED
ERP_ACCOUNT_ID	UNMAPPED
ERP_OFFSET_ACCOUNT_ID	UNMAPPED
UOM_CONV_FACTOR	UNMAPPED
UOM_CONV_OPERATOR	UNMAPPED
FROM_AMOUNT	UNMAPPED
FROM_UOM	UNMAPPED
TO_ROUNDED_AMOUNT	UNMAPPED
TO_ROUNDED_UOM	UNMAPPED
TO_UNROUNDED_AMOUNT	UNMAPPED
TO_UNROUNDED_UOM	UNMAPPED

Determination Line-Level Output Tax	
Column Name	**E-Business Tax Result Table**
VAT_GROUP_REGISTRATION	UNMAPPED
UNIQUE_LINE_NUMBER	UNMAPPED
REGISTRATION_ATTRIBUTE1	UNMAPPED
REGISTRATION_ATTRIBUTE2	UNMAPPED
REGISTRATION_ATTRIBUTE3	UNMAPPED
REGISTRATION_ATTRIBUTE4	UNMAPPED
REGISTRATION_ATTRIBUTE5	UNMAPPED
REGISTRATION_ATTRIBUTE6	UNMAPPED
REGISTRATION_ATTRIBUTE7	UNMAPPED
REGISTRATION_ATTRIBUTE8	UNMAPPED
REGISTRATION_ATTRIBUTE9	UNMAPPED
REGISTRATION_ATTRIBUTE10	UNMAPPED
REGISTRATION_ATTRIBUTE11	UNMAPPED
REGISTRATION_ATTRIBUTE12	UNMAPPED
REGISTRATION_ATTRIBUTE13	UNMAPPED
REGISTRATION_ATTRIBUTE14	UNMAPPED
REGISTRATION_ATTRIBUTE15	UNMAPPED
REGISTRATION_ATTRIBUTE16	UNMAPPED
REGISTRATION_ATTRIBUTE17	UNMAPPED
REGISTRATION_ATTRIBUTE18	UNMAPPED
REGISTRATION_ATTRIBUTE19	UNMAPPED
REGISTRATION_ATTRIBUTE20	UNMAPPED
REGISTRATION_ATTRIBUTE21	UNMAPPED
REGISTRATION_ATTRIBUTE22	UNMAPPED
REGISTRATION_ATTRIBUTE23	UNMAPPED
REGISTRATION_ATTRIBUTE24	UNMAPPED
REGISTRATION_ATTRIBUTE25	UNMAPPED
REGISTRATION_ATTRIBUTE26	UNMAPPED
REGISTRATION_ATTRIBUTE27	UNMAPPED
REGISTRATION_ATTRIBUTE28	UNMAPPED
REGISTRATION_ATTRIBUTE29	UNMAPPED
REGISTRATION_ATTRIBUTE30	UNMAPPED
REGISTRATION_ATTRIBUTE31	UNMAPPED
REGISTRATION_ATTRIBUTE32	UNMAPPED
REGISTRATION_ATTRIBUTE33	UNMAPPED
REGISTRATION_ATTRIBUTE34	UNMAPPED
REGISTRATION_ATTRIBUTE35	UNMAPPED
REGISTRATION_ATTRIBUTE36	UNMAPPED
REGISTRATION_ATTRIBUTE37	UNMAPPED

Determination Line-Level Output Tax	
Column Name	**E-Business Tax Result Table**
REGISTRATION_ATTRIBUTE38	UNMAPPED
REGISTRATION_ATTRIBUTE39	UNMAPPED
REGISTRATION_ATTRIBUTE40	UNMAPPED
REGISTRATION_ATTRIBUTE41	tax_lines_result_tbl.legal_justification_text1
REGISTRATION_ATTRIBUTE42	tax_lines_result_tbl.legal_justification_text2
REGISTRATION_ATTRIBUTE43	UNMAPPED
REGISTRATION_ATTRIBUTE44	UNMAPPED
REGISTRATION_ATTRIBUTE45	UNMAPPED
REGISTRATION_ATTRIBUTE46	UNMAPPED
REGISTRATION_ATTRIBUTE47	UNMAPPED
REGISTRATION_ATTRIBUTE48	UNMAPPED
REGISTRATION_ATTRIBUTE49	UNMAPPED
REGISTRATION_ATTRIBUTE50	UNMAPPED
AUTHORITY_ATTRIBUTE1	UNMAPPED
AUTHORITY_ATTRIBUTE2	UNMAPPED
AUTHORITY_ATTRIBUTE3	UNMAPPED
AUTHORITY_ATTRIBUTE4	UNMAPPED
AUTHORITY_ATTRIBUTE5	UNMAPPED
AUTHORITY_ATTRIBUTE6	UNMAPPED
AUTHORITY_ATTRIBUTE7	UNMAPPED
AUTHORITY_ATTRIBUTE8	UNMAPPED
AUTHORITY_ATTRIBUTE9	UNMAPPED
AUTHORITY_ATTRIBUTE10	UNMAPPED
AUTHORITY_ATTRIBUTE11	UNMAPPED
AUTHORITY_ATTRIBUTE12	UNMAPPED
AUTHORITY_ATTRIBUTE13	UNMAPPED
AUTHORITY_ATTRIBUTE14	UNMAPPED
AUTHORITY_ATTRIBUTE15	UNMAPPED
AUTHORITY_ATTRIBUTE16	UNMAPPED
AUTHORITY_ATTRIBUTE17	UNMAPPED
AUTHORITY_ATTRIBUTE18	UNMAPPED
AUTHORITY_ATTRIBUTE19	UNMAPPED
AUTHORITY_ATTRIBUTE20	UNMAPPED
AUTHORITY_ATTRIBUTE21	UNMAPPED
AUTHORITY_ATTRIBUTE22	UNMAPPED
AUTHORITY_ATTRIBUTE23	UNMAPPED
AUTHORITY_ATTRIBUTE24	UNMAPPED
AUTHORITY_ATTRIBUTE25	UNMAPPED
AUTHORITY_ATTRIBUTE26	UNMAPPED

Determination Line-Level Output Tax	
Column Name	**E-Business Tax Result Table**
AUTHORITY_ATTRIBUTE27	UNMAPPED
AUTHORITY_ATTRIBUTE28	UNMAPPED
AUTHORITY_ATTRIBUTE29	UNMAPPED
AUTHORITY_ATTRIBUTE30	UNMAPPED
AUTHORITY_ATTRIBUTE31	UNMAPPED
AUTHORITY_ATTRIBUTE32	UNMAPPED
AUTHORITY_ATTRIBUTE33	UNMAPPED
AUTHORITY_ATTRIBUTE34	UNMAPPED
AUTHORITY_ATTRIBUTE35	UNMAPPED
AUTHORITY_ATTRIBUTE36	UNMAPPED
AUTHORITY_ATTRIBUTE37	UNMAPPED
AUTHORITY_ATTRIBUTE38	UNMAPPED
AUTHORITY_ATTRIBUTE39	UNMAPPED
AUTHORITY_ATTRIBUTE40	UNMAPPED
AUTHORITY_ATTRIBUTE41	UNMAPPED
AUTHORITY_ATTRIBUTE42	UNMAPPED
AUTHORITY_ATTRIBUTE43	UNMAPPED
AUTHORITY_ATTRIBUTE44	UNMAPPED
AUTHORITY_ATTRIBUTE45	UNMAPPED
AUTHORITY_ATTRIBUTE46	UNMAPPED
AUTHORITY_ATTRIBUTE47	UNMAPPED
AUTHORITY_ATTRIBUTE48	UNMAPPED
AUTHORITY_ATTRIBUTE49	UNMAPPED
AUTHORITY_ATTRIBUTE50	UNMAPPED
AUTHORITY_NAME	UNMAPPED
ADMIN_ZONE_LEVEL	UNMAPPED
AUTHORITY_TYPE	UNMAPPED
DISTANCE_SALES_THRESHOLD	UNMAPPED
EFFECTIVE_ZONE_LEVEL	UNMAPPED
INVOICE_DESCRIPTION	tax_lines_result_tbl.legal_justification_Text2
LOCATION_CODE	UNMAPPED
TAX_AMOUNT	UNMAPPED
TAX_RATE	UNMAPPED
TAX_TYPE	UNMAPPED
AUTHORITY_NAME	UNMAPPED
ADMIN_ZONE_LEVEL	UNMAPPED
AUTHORITY_TYPE	UNMAPPED
DISTANCE_SALES_THRESHOLD	UNMAPPED

O2C: INVOICE-LEVEL OUTPUT XML MAPPED TO ORACLE

Determination Invoice-Level Output XML	
Column Name	**Maps to Oracle Element**
BATCH_ID	UNMAPPED
INVOICE_ID	UNMAPPED
CREATION_DATE	UNMAPPED
STATISTICS_NAME	UNMAPPED
STATISTICS_VALUE	UNMAPPED
BASIS_PERCENT	UNMAPPED
CALLING_SYSTEM_NUMBER	tax_lines_results_tbl.application_id
COMPANY_ID	UNMAPPED
COMPANY_NAME	UNMAPPED
CURRENCY_CODE	UNMAPPED
EXTERNAL_COMPANY_ID	UNMAPPED
INVOICE_NUMBER	tax_lines_results_tbl.trx_number
TOTAL_TAX_AMOUNT	UNMAPPED
CALCULATION_DIRECTION	UNMAPPED
COMPANY_ROLE	UNMAPPED
CURRENCY_NAME	UNMAPPED
CUSTOMER_NAME	UNMAPPED
CUSTOMER_NUMBER	UNMAPPED
DELIVERY_TERMS	UNMAPPED
END_USER_NAME	UNMAPPED
FISCAL_DATE	UNMAPPED
FLAT_FEE	UNMAPPED
FUNCTIONAL_CURRENCY_CODE	UNMAPPED
HOST_SYSTEM	UNMAPPED
INVOICE_DATE	tax_lines_results_tbl.trx_line_date
IS_AUDIT_UPDATE	UNMAPPED
IS_BUSINESS_SUPPLY	UNMAPPED
IS_CREDIT	UNMAPPED
IS_REPORTED	UNMAPPED
MIN_ACCOUNTABLE_UNIT	UNMAPPED
NATURE_OF_TRANSACTION_CODE	UNMAPPED
ORIGINAL_INVOICE_NUMBER	UNMAPPED
ORIGINAL_DOCUMENT_ID	UNMAPPED
ORIGINAL_DOCUMENT_ITEM	UNMAPPED
ORIGINAL_DOCUMENT_TYPE	UNMAPPED
ORIGINAL_INVOICE_DATE	UNMAPPED
ORIGINAL_MOVEMENT_DATE	UNMAPPED

Determination Invoice-Level Output XML	
Column Name	**Maps to Oracle Element**
ROUNDING_PRECISION	UNMAPPED
ROUNDING_RULE	UNMAPPED
STATISTICAL_PROCEDURE	UNMAPPED
TRANSACTION_DATE	UNMAPPED
UNIQUE_INVOICE_NUMBER	tax_lines_results_tbl.transaction_id
VENDOR_NAME	UNMAPPED
VENDOR_NUMBER	UNMAPPED
VENDOR_TAX	UNMAPPED
VENDOR_TAX_FUNC	UNMAPPED
USER_ELEMENT_ATTRIBUTE1	UNMAPPED
USER_ELEMENT_ATTRIBUTE2	UNMAPPED
USER_ELEMENT_ATTRIBUTE3	UNMAPPED
USER_ELEMENT_ATTRIBUTE4	UNMAPPED
USER_ELEMENT_ATTRIBUTE5	UNMAPPED
USER_ELEMENT_ATTRIBUTE6	UNMAPPED
USER_ELEMENT_ATTRIBUTE7	UNMAPPED
USER_ELEMENT_ATTRIBUTE8	UNMAPPED
USER_ELEMENT_ATTRIBUTE9	UNMAPPED
USER_ELEMENT_ATTRIBUTE10	UNMAPPED
USER_ELEMENT_ATTRIBUTE11	UNMAPPED
USER_ELEMENT_ATTRIBUTE12	UNMAPPED
USER_ELEMENT_ATTRIBUTE13	UNMAPPED
USER_ELEMENT_ATTRIBUTE14	UNMAPPED
USER_ELEMENT_ATTRIBUTE15	UNMAPPED
USER_ELEMENT_ATTRIBUTE16	UNMAPPED
USER_ELEMENT_ATTRIBUTE17	UNMAPPED
USER_ELEMENT_ATTRIBUTE18	UNMAPPED
USER_ELEMENT_ATTRIBUTE19	UNMAPPED
USER_ELEMENT_ATTRIBUTE20	UNMAPPED
USER_ELEMENT_ATTRIBUTE21	UNMAPPED
USER_ELEMENT_ATTRIBUTE22	UNMAPPED
USER_ELEMENT_ATTRIBUTE23	UNMAPPED
USER_ELEMENT_ATTRIBUTE24	UNMAPPED
USER_ELEMENT_ATTRIBUTE25	UNMAPPED
USER_ELEMENT_ATTRIBUTE26	UNMAPPED
USER_ELEMENT_ATTRIBUTE27	UNMAPPED
USER_ELEMENT_ATTRIBUTE28	UNMAPPED
USER_ELEMENT_ATTRIBUTE29	UNMAPPED
USER_ELEMENT_ATTRIBUTE30	UNMAPPED

Determination Invoice-Level Output XML	
Column Name	**Maps to Oracle Element**
USER_ELEMENT_ATTRIBUTE31	UNMAPPED
USER_ELEMENT_ATTRIBUTE32	UNMAPPED
USER_ELEMENT_ATTRIBUTE33	UNMAPPED
USER_ELEMENT_ATTRIBUTE34	UNMAPPED
USER_ELEMENT_ATTRIBUTE35	UNMAPPED
USER_ELEMENT_ATTRIBUTE36	UNMAPPED
USER_ELEMENT_ATTRIBUTE37	UNMAPPED
USER_ELEMENT_ATTRIBUTE38	UNMAPPED
USER_ELEMENT_ATTRIBUTE39	UNMAPPED
USER_ELEMENT_ATTRIBUTE40	UNMAPPED
USER_ELEMENT_ATTRIBUTE41	UNMAPPED
USER_ELEMENT_ATTRIBUTE42	UNMAPPED
USER_ELEMENT_ATTRIBUTE43	UNMAPPED
USER_ELEMENT_ATTRIBUTE44	UNMAPPED
USER_ELEMENT_ATTRIBUTE45	UNMAPPED
USER_ELEMENT_ATTRIBUTE46	UNMAPPED
USER_ELEMENT_ATTRIBUTE47	UNMAPPED
USER_ELEMENT_ATTRIBUTE48	UNMAPPED
USER_ELEMENT_ATTRIBUTE49	UNMAPPED
USER_ELEMENT_ATTRIBUTE50	tax_lines_results_tbl.doc_type_id
IS_SUCCESS	UNMAPPED
IS_PARTIAL_SUCCESS	UNMAPPED

O2C: LINE-LEVEL OUTPUT XML MAPPED TO ORACLE

Determination Line-Level Output XML	
Column Name	**E-Business Tax Result Table**
BATCH_ID	UNMAPPED
INVOICE_ID	UNMAPPED
LINE_ID	UNMAPPED
CREATION_DATE	UNMAPPED
LINE_NUMBER	UNMAPPED
TOTAL_TAX_AMOUNT	UNMAPPED
ACCOUNTING_CODE	UNMAPPED
ALLOCATION_NAME	UNMAPPED
BASIS_PERCENT	UNMAPPED
GROSS_AMOUNT	tax_lines_result_tbl.line_amt
GROSS_AMOUNT_FUNC	UNMAPPED
TAX_CODE	UNMAPPED
BILL_TO_BRANCH_ID	UNMAPPED
COMMODITY_CODE	UNMAPPED
COUNTRY_OF_ORIGIN	UNMAPPED
CUSTOMER_NAME	UNMAPPED
CUSTOMER_NUMBER	UNMAPPED
DEPT_OF_CONSIGN	UNMAPPED
DESCRIPTION	UNMAPPED
DISCOUNT_AMOUNT	UNMAPPED
DISCOUNT_AMOUNT_FUNC	UNMAPPED
DELIVERY_TERMS	UNMAPPED
INPUT_RECOVERY_TYPE	UNMAPPED
IS_BUSINESS_SUPPLY	UNMAPPED
IS_REVERSED	UNMAPPED
ITEM_VALUE	UNMAPPED
ITEM_VALUE_FUNC	UNMAPPED
MASS	UNMAPPED
MIDDLEMAN_BRANCH_ID	UNMAPPED
MODE_OF_TRANSPORT	UNMAPPED
MOVEMENT_DATE	UNMAPPED
ORIGINAL_DOCUMENT_ID	UNMAPPED
ORIGINAL_DOCUMENT_ITEM	UNMAPPED
ORIGINAL_DOCUMENT_TYPE	UNMAPPED
ORIGINAL_INVOICE_DATE	UNMAPPED
ORIGINAL_INVOICE_NUMBER	UNMAPPED
ORIGINAL_MOVEMENT_DATE	UNMAPPED

Determination Line-Level Output XML	
Column Name	**E-Business Tax Result Table**
PART_NUMBER	UNMAPPED
POINT_OF_TITLE_TRANSFER	UNMAPPED
PORT_OF_ENTRY	UNMAPPED
PORT_OF_LOADING	UNMAPPED
QUANTITY	tax_lines_result_tbl.quantity
REGIME	UNMAPPED
RELATED_LINE_NUMBER	UNMAPPED
SHIP_FROM_COUNTRY	UNMAPPED
SHIP_FROM_BRANCH_ID	UNMAPPED
SHIP_TO_COUNTRY	UNMAPPED
SHIP_TO_BRANCH_ID	UNMAPPED
SUPPLEMENTARY_UNIT	UNMAPPED
SUPPLY_BRANCH_ID	UNMAPPED
SUPPLY_TYPE	UNMAPPED
TRANSACTION_TYPE	UNMAPPED
MIDDLEMAN_MARKUP_AMOUNT	UNMAPPED
MIDDLEMAN_MARKUP_AMOUNT_FUNC	UNMAPPED
MIDDLEMAN_MARKUP_RATE	UNMAPPED
TITLE_TRANSFER_LOCATION	UNMAPPED
UNIQUE_LINE_NUMBER	UNMAPPED
UNIT_OF_MEASURE	UNMAPPED
VENDOR_NAME	UNMAPPED
VENDOR_NUMBER	UNMAPPED
VENDOR_TAX	UNMAPPED
VENDOR_TAX_FUNC	UNMAPPED
USER_ELEMENT_ATTRIBUTE1	UNMAPPED
USER_ELEMENT_ATTRIBUTE2	UNMAPPED
USER_ELEMENT_ATTRIBUTE3	UNMAPPED
USER_ELEMENT_ATTRIBUTE4	UNMAPPED
USER_ELEMENT_ATTRIBUTE5	UNMAPPED
USER_ELEMENT_ATTRIBUTE6	UNMAPPED
USER_ELEMENT_ATTRIBUTE7	UNMAPPED
USER_ELEMENT_ATTRIBUTE8	UNMAPPED
USER_ELEMENT_ATTRIBUTE9	UNMAPPED
USER_ELEMENT_ATTRIBUTE10	UNMAPPED
USER_ELEMENT_ATTRIBUTE11	UNMAPPED
USER_ELEMENT_ATTRIBUTE12	UNMAPPED
USER_ELEMENT_ATTRIBUTE13	UNMAPPED
USER_ELEMENT_ATTRIBUTE14	UNMAPPED

Determination Line-Level Output XML	
Column Name	**E-Business Tax Result Table**
USER_ELEMENT_ATTRIBUTE15	UNMAPPED
USER_ELEMENT_ATTRIBUTE16	UNMAPPED
USER_ELEMENT_ATTRIBUTE17	UNMAPPED
USER_ELEMENT_ATTRIBUTE18	UNMAPPED
USER_ELEMENT_ATTRIBUTE19	UNMAPPED
USER_ELEMENT_ATTRIBUTE20	UNMAPPED
USER_ELEMENT_ATTRIBUTE21	UNMAPPED
USER_ELEMENT_ATTRIBUTE22	UNMAPPED
USER_ELEMENT_ATTRIBUTE23	UNMAPPED
USER_ELEMENT_ATTRIBUTE24	UNMAPPED
USER_ELEMENT_ATTRIBUTE25	UNMAPPED
USER_ELEMENT_ATTRIBUTE26	UNMAPPED
USER_ELEMENT_ATTRIBUTE27	UNMAPPED
USER_ELEMENT_ATTRIBUTE28	UNMAPPED
USER_ELEMENT_ATTRIBUTE29	UNMAPPED
USER_ELEMENT_ATTRIBUTE30	UNMAPPED
USER_ELEMENT_ATTRIBUTE31	UNMAPPED
USER_ELEMENT_ATTRIBUTE32	UNMAPPED
USER_ELEMENT_ATTRIBUTE33	UNMAPPED
USER_ELEMENT_ATTRIBUTE34	UNMAPPED
USER_ELEMENT_ATTRIBUTE35	UNMAPPED
USER_ELEMENT_ATTRIBUTE36	UNMAPPED
USER_ELEMENT_ATTRIBUTE37	UNMAPPED
USER_ELEMENT_ATTRIBUTE38	UNMAPPED
USER_ELEMENT_ATTRIBUTE39	UNMAPPED
USER_ELEMENT_ATTRIBUTE40	UNMAPPED
USER_ELEMENT_ATTRIBUTE41	tax_lines_result_tbl.transaction_line_id
USER_ELEMENT_ATTRIBUTE42	UNMAPPED
USER_ELEMENT_ATTRIBUTE43	UNMAPPED
USER_ELEMENT_ATTRIBUTE44	UNMAPPED
USER_ELEMENT_ATTRIBUTE45	UNMAPPED
USER_ELEMENT_ATTRIBUTE46	UNMAPPED
USER_ELEMENT_ATTRIBUTE47	tax_lines_result_tbl.entity_code
USER_ELEMENT_ATTRIBUTE48	UNMAPPED
USER_ELEMENT_ATTRIBUTE49	tax_lines_result_tbl.event_class_code
USER_ELEMENT_ATTRIBUTE50	tax_lines_result_tbl.event_type_code
INVOICE_DATE	UNMAPPED
TAXABLE_BASIS	UNMAPPED
NON_TAXABLE_BASIS	UNMAPPED

Determination Line-Level Output XML	
Column Name	**E-Business Tax Result Table**
EXEMPT_AMOUNT	UNMAPPED
TAX_RATE	UNMAPPED
EFFECTIVE_TAX_RATE	UNMAPPED
ADVISORIES	UNMAPPED
RELATED_ALLOCATION_LINE_NUMBER	UNMAPPED
RELATED_LINE_NUMBER	UNMAPPED

O2C: TAX RESULTS MAPPED TO TAX_LINES_RESULT_TBL

O2C Tax Results	
Tax Lines Result Table	**Mapping Source**
INTERNAL_ORGANIZATION_ID	fnd_profile('ORG_ID')
LEGAL_ENTITY_ID	fnd_profile('ORG_ID');
BUSINESS_GROUP_ID	hr_organization_units.business_group_id where organization_id = fnd_profile('ORG_ID')
TRANSACTION_LINE_ID	sabrix_line_out.user_element_attribute41
TAX_LINE_ID	sabrix_line_tax.line_tax_id;
TRANSACTION_ID	sabrix_invoice_out.unique_invoice_number;
TRX_NUMBER	sabrix_invoice_out.invoice_number;
TAX_LINE_NUMBER	sabrix_line_tax.line_tax_id;
TRX_LINE_NUMBER	sabrix_line_tax.line_id;
TAX_APPORTIONMENT_LINE_NUMBER	rowcount
APPLICATION_ID	sabrix_invoice_out.calling_system_number
DOCUMENT_TYPE_ID	sabrix_invoice_out.user_element_attribute50
EVENT_CLASS_CODE	sabrix_line_out.user_element_attribute47
EVENT_TYPE_CODE	sabrix_line_out.user_element_attribute49
ENTITY_CODE	sabrix_line_out.user_element_attribute50
DOC_EVENT_STATUS	NULL;
TAX_EVENT_CLASS_CODE	NULL;
TAX_EVENT_TYPE_CODE	NULL;
TRX_LEVEL_TYPE	'LINE'
COUNTRY_CODE	sabrix_line_tax.taxable_country;
STATE	sabrix_line_tax.taxable_state
COUNTY	sabrix_line_tax.taxable_county;
CITY	sabrix_line_tax.taxable_city;
TAX_CURRENCY_CODE	sabrix_line_tax.step_1_from_currency_code;
TRX_CURRENCY_CODE	sabrix_line_tax.step_1_from_currency_code;
TAX_DATE	sabrix_line_tax.tax_point_date;
TAX_DETERMINE_DATE	sabrix_line_tax.tax_determination_date;
TRX_LINE_DATE	sabrix_invoice_out.invoice_date;
TRX_DATE	sabrix_invoice_out.invoice_date;
LINE_AMT	sabrix_line_out.gross_amount;
TAX_AMOUNT	sabrix_line_tax.tax_amount;
UNROUNDED_TAX_AMOUNT	sabrix_line_tax.tax_unrounded_document_amount;
TAX_CURR_TAX_AMOUNT	sabrix_line_tax.tax_unrounded_document_amount;
TAX_RATE_PERCENTAGE	sabrix_line_tax.tax_rate * 100;
TAXABLE_AMOUNT	sabrix_line_tax.taxable_basis;
TAXABLE_AMT_TAX_CURR	NULL;
EXEMPT_AMT	sabrix_line_tax.exempt_amount;

O2C Tax Results	
Tax Lines Result Table	**Mapping Source**
EXEMPT_CERTIFICATE_NUMBER	sabrix_line_tax.exempt_certificate;
TRX_LINE_QUANTITY	sabrix_line_out.quantity;
UNIT_PRICE	sabrix_line_out.gross_amount / sabrix_line_out.quantity;
TAX_PRECISION	NULL;
MINIMUM_ACCOUNTABLE_UNIT	NULL;
ROUNDING_RULE_CODE	NULL;
TAX	'SABRIX_TAX_AR'
TAX_STATUS_CODE	'SABRIX_TAX_STATUS_AR'
TAX_JURISDICTION	SUBSTR(sabrix_line_tax.authority_name,1,30);
TAX_RATE_CODE	sabrix_line_tax.erp_tax_code
TAX_TYPE_CODE	NULL;
TAX_REGIME_ID	zx_regimes.tax_regime_id where tax_regime_code = 'SABRIX_REGIME' and effective for current processing date
TAX_ID	zx_taxes.tax_id where tax = 'SABRIX_TAX_AR' and effective for current processing date
TAX_STATUS_ID	zx_status.tax_status_id where tax_status_code = 'SABRIX_TAX_STATUS_AR' and effective for current processing date
TAX_JURISDICTION_ID	zx_jurisdictions where tax_jurisdiction_code = 'SABRIX_JURISDICTION_AR' and effective for current processing date
TAX_RATE_ID	zx_rates_b.tax_rate_id where tax_rate_code = sabrix_line_tax.erp_tax_code and tax = 'SABRIX_TAX_AR and tax_regime = 'SABRIX_REGIME' and effective for sabrix_line_tax.tax_determination_date;
LEDGER_ID	zx_accounts.ledger_id where tax_account_entity_code = 'RATES' and tax_account_entity_id = tax_rate_id and internal_organization_id = org_unit_id
RATE_TYPE_CODE	decode(tl.nature_of_tax, 'P', 'PERCENTAGE', 'FIXED')
SYNC_WITH_PRVDR_FLAG	'Y';
TAX_ONLY_LINE_FLAG	'N';
INCLUSIVE_TAX_LINE_FLAG	'N';
LINE_AMT_INCLUDES_TAX_FLAG	'N';
USE_TAX_FLAG	'N';
USER_OVERRIDE_FLAG	'N';
MANUALLY_ENTERED_FLAG	'N';
TAX_PROVIDER_ID	zx_srvc_subscriptions.srvc_provider_id where prod_family_grp_code = 'O2C'
CANCEL_FLAG	'N';
DELETE_FLAG	'N';
THRESHOLD_INDICATOR_FLAG	'N';
OFFSET_FLAG	'N';
PROCESS_FOR_RECOVERY_FLAG	'N';
COMPOUNDING_TAX_FLAG	'N';
TAX_APPORTIONMENT_FLAG	'N';

O2C Tax Results	
Tax Lines Result Table	**Mapping Source**
HISTORICAL_FLAG	'N';
PURGE_FLAG	'N';
REPORTING_ONLY_FLAG	N';
FREEZE_UNTIL_OVERRIDDEN_FLAG	'N';
COPIED_FROM_OTHER_DOC_FLAG	'N';
MRC_TAX_LINE_FLAG	'N';
SITUS	NULL;
EXEMPT_RATE_MODIFIER	NULL;
EXEMPT_REASON	NULL;
LAST_MANUAL_ENTRY	NULL;
REGISTRATION_PARTY_TYPE	NULL;
PARTY_TAX_REG_NUMBER	NULL;
THIRD_PARTY_TAX_REG_NUMBER	NULL;
TAX_CURRENCY_CONVERSION_DATE	NULL;
TAX_CURRENCY_CONVERSION_TYPE	NULL;
TAX_CURRENCY_CONVERSION_RATE	NULL;
APPLIED_FROM_APPLICATION_ID	NULL;
APPLIED_FROM_ENTITY_CODE	NULL;
APPLIED_FROM_EVENT_CLASS_CODE	NULL;
APPLIED_FROM_TRX_ID	NULL;
APPLIED_FROM_LINE_ID	NULL;
APPLIED_FROM_TRX_LEVEL_TYPE	NULL;
APPLIED_FROM_TRX_NUMBER	NULL;
ADJUSTED_DOC_APPLICATION_ID	NULL;
ADJUSTED_DOC_ENTITY_CODE	NULL;
ADJUSTED_DOC_EVENT_CLASS_CODE	NULL;
ADJUSTED_DOC_TRX_ID	NULL;
ADJUSTED_DOC_LINE_ID	NULL;
ADJUSTED_DOC_TRX_LEVEL_TYPE	NULL;
ADJUSTED_DOC_NUMBER	NULL;
ADJUSTED_DOC_DATE	NULL;

APPENDIX F: NON-STANDARDORACLE APIS

Oracle seeded tables into which direct inserts are performed during installation in the absence of known public APIs:

- ZX_API_OWNER_STATUSES
- ZX_SRVC_TYP_PARAMS
- ZX_API_CODE_COMBINATIONS
- ZX_API_REGISTRATIONS
- ZX_SUBSCRIPTION_DETAILS
- ZX_SUBSCRIPTION_OPTIONS
- ZX_SRVC_SUBSCRIPTIONS
- ZX_EVNT_CLS_OPTIONS
- ZX_REGIMES_USAGES
- ZX_ACCOUNTS
- ZX_PRODUCT_OPTIONS

Oracle seeded tables on which updates are performed during installation in the absence of known public APIs:

- ZX_API_CODE_COMBINATIONS
- ZX_API_OWNER_STATUSES
- ZX_EVNT_CLS_MAPPINGS
- ZX_EVNT_CLS_OPTIONS

Oracle seeded table on which a direct update is performed in AP during processing to overcome absent functionality of a public API:

- FND_CONCURRENT_REQUESTS

GLOSSARY

The following terms are used throughout this installation guide:

Country/Regime: In eBTax, this is a set of tax rules for a geography. This is roughly equivalent to Zone in Determination and can contain multiple tax authorities.

Determination: The tax engine, also known as ONESOURCE Indirect Tax Determination, that is connected to Oracle Applications using the ONESOURCE Indirect Tax Integration for Oracle 12. Formerly known as the Sabrix Solution.

ERP Tax Code: A field on a tax authority in Determination.

Jurisdiction: In eBTax, this is a geographic area where tax is levied. This corresponds to Tax Authority in Determination.

Sabrix Solution: See Determination.

Status: In eBTax, this is the category of taxes for a product or service (for example, zero-rate, standard rate, and exempt). This corresponds to Tax Code in Determination.

Tax: A distinct charge imposed by a tax authority, for example, State, County, City, and District. This is mapped to Tax Type in Determination (for example, *23-AC*, *ZR*, *IM*, *CU*, *US*, *SA*, etc.).

Tax Direction: Specifies input/output values for VAT recovery identification.

Tax Expense: A Payables tax account that records tax amounts from invoice distributions; or a Receivables tax account that records taxes collected from customers.

Tax Rate Code: In eBTax, this is the rate for a tax jurisdiction and/or the tax status based on a specific time period, percentage, or quantity. This is mapped in Determination to the ERP Tax Code on authorities (for example, *USWA*).

Tax Recoverable/Liability: An account that records tax recoverable amounts or relieves tax liability amounts.