ORACLE

White Paper- UBL 2.1 Electronic Invoicing in Oracle Payables Cloud

—

Overview, Configurations and Frequently Asked Questions

ORACLE

PURPOSE STATEMENT

This document provides an overview of features, process, configurations and frequently asked questions in implementing electronic invoicing in ERP Cloud in release 20A. It is intended solely to help in implementing electronic invoicing in 20A .

DISCLAIMER

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

This document is for informational purposes only and is intended solely to assist you in planning for the implementation and upgrade of the product features described. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.

Due to the nature of the product architecture, it may not be possible to safely include all features described in this document without risking significant destabilization of the code.

DISCLAIMERS FOR PRE-RELEASE, PRE-GA PRODUCTS

The **revenue recognition disclaimer** on this page is required for any white paper that addresses future functionality or for products that are not yet generally available (GA). If you are unsure whether your statement of direction needs the disclaimer, read the revenue recognition policy. If you have further questions about your content and the disclaimer requirements, e-mail REVREC_US@oracle.com.

The **testing disclaimer** in the copyright section on the last page (highlighted in yellow) is provided by the FCC for hardware products. It must appear in the copyright section for all pre-release, pre-GA hardware products. Be sure to remove the yellow highlighting before publishing. When the product becomes GA, update your collateral by removing the disclaimer from the copyright section. If your product is already GA or if you are writing about a software product, delete the disclaimer from the copyright section.

Important: If your product is not GA, then you cannot include any regulatory compliance information in the statement of direction. Regulatory compliance information may be included for GA products only if you have completed all required safety and emissions testing, and you have received the certificates issued by the testing organization

TABLE OF CONTENTS

INTRODUCTION

In the current business environment, it is imperative for organizations to operate efficiently to remain competitive. Business-To-Business e-commerce becomes an obvious choice to the organizations to transact with the business partners efficiently to improve the operational efficiency. Oracle Applications Cloud support the exchange of business documents like Purchase Orders and Invoices between trading partners electronically using XML standards. The business benefits achieved using electronic invoicing are as follows:

- Reduction in manual invoice entry.
- Streamlined invoice processing.
- Reducing costs by eliminating "non-value-added" tasks involved in paper invoice processing
- Tax and statutory Compliance

SCOPE OF THIS DOCUMENT

This document covers the overview of electronic invoicing in Oracle Applications Cloud and the set-ups required for electronic invoicing. It also answers some of the frequently asked questions by SIs/Partners/Customers of Oracle while setting up electronic invoicing in Oracle Applications Cloud.

It covers the implementation of electronic invoicing using UBL 2.1 standard, with Collaboration Messaging Framework (CMK). However, if you are implementing another standard (e.g. OAGIS and cXML etc.), you will still find this document useful. The focus of this paper is inbound invoicing (Accounts Payable) although some of the CMK set-ups are common/related for inbound (Accounts Payable) and outbound (Accounts Receivable) invoices. Therefore, they are covered, when required to give a holistic view. Unless explicitly stated, this document only refers to inbound invoicing. In the next release, this document will be extended to outbound Invoices as well.

This document does not intend to replace Implementation Guides or User Guides of CMK, Receivables or Payables. Please refer to these documents for more details.

The target audience for this document is the application consultant(s) implementing or planning to implement electronic invoicing.

OVERVIEW OF COLLABORATION MESSAGING FRAMEWORK

You can use Oracle Fusion Collaboration Messaging Framework to enable your Oracle Fusion applications to establish business-to-business (B2B) message exchanging capabilities with your customers or suppliers.

Using this framework, you can send and receive real-time transactional B2B messages. You can exchange messages directly with your customers or suppliers or by using an intermediary such as a B2B service provider. Collaboration Messaging Framework can receive B2B messages from trading partners or service providers through Oracle B2B, an e-commerce gateway that enables the secure and reliable exchange of business documents between an enterprise and its trading partners, or directly using SOAP web services that are available to consumers. For more information, see Collaboration Messaging Web Service.

Collaboration Messaging Framework supports transformation of a B2B document, such as a purchase order, between the Oracle Fusion applications format and a message format supported by your customers or suppliers. When you send messages to partners or receive messages from them, the framework performs the required transformation.

The configuration and set-up tasks you need to perform depend on the business process that you implement B2B message exchange for and whether or not you use a service provider.

This document describes the steps required to set-up message exchange between Oracle Applications Cloud and trading partners using the Collaboration Messaging Framework. Oracle Applications Cloud interact with the CMK layer. CMK interacts either through 1) B2B Protocols or 2) web services to communicate with external parties. CMK can receive/send messages from/to trading partners (with or without service providers) in two ways:

1. SOAP Web services – CMK provides web services which trading partners can invoke to submit business documents. For more details, refer to "Collaboration Messaging Web Service" in Oracle Help Center.

2. Oracle B2B – Oracle B2B is a Middleware component that facilitates exchange of business documents between trading partners.

.

HIGH LEVEL CONFIGURATION

At a high level, the following needs to be performed to enable an Oracle Applications Cloud customer to receive electronic invoices from their suppliers:

1. CMK set-up-
   a. Set up service provider-associate with delivery method and message
   b. Set up Trading partner
   c. Associate supplier site (in case of supplier) and customer account (in case of customer) with service provider/trading partner/collaboration document
   d. Define message
2. Oracle B2B set-up (required if B2B and not web service is used for communication with trading partners) -
   a. Create B2B partner (this will be mapped to CMK Service Provider) with the same name as CMK service provider (defined in 1(a) above)
   b. Add document for the partner (e.g.; UBL invoice)
   c. Add Channel for the partner (this is required only for outbound messages)
   d. Add agreement for the trading partner - map the document and direction. Validate and deploy
3. Customer (for outbound invoices) or Supplier (for inbound invoices) set-up-
   a. Customer (required for outbound invoices) :
      i. Set up XML invoicing in customer account profile. Profile history- preferred delivery method XML. Also enable all relevant transactions for XML invoicing
   b. Supplier (required for inbound invoices):
      i. Set up supplier site- go to edit site – set B2B communication method as CMK
      ii. Map the supplier site with a trading partner and service provider combination
      iii. Associate Collaboration Documents

DETAILS OF CONFIGURATION

Following lists the configuration required for electronic invoicing. Please refer to CMK implementation guide for more details here.

Set up Collaboration Messaging Framework

There are three tasks to be completed for enabling messaging with CMK:

1. Manage Collaboration Messaging Configuration
2. Manage Collaboration Messaging Service Provider
3. Manage Trading Partner

Manage Collaboration Messaging Configuration

1. Navigate to FSM > Define Collaboration Messaging > Manage Collaboration Messaging Configuration (setup task)
2. Review set-up information

   Enable Collaboration Messaging: Ensure that the Procure-to-Pay business process has been enabled

   Message Storage: Select which documents are to be stored by checking the appropriate boxes. You can accept the default (all documents selected)

   Outbound Sender: Capture global sender ID here. If you are using OSN, Enter the username and ID Type that were used to create the Buyer Account within Oracle Supplier Network or that will be used to create the Buyer Account within OSN. It is important that this information perfectly matches what was or will be created in OSN

3. After review, select Save and Close.

Manage Collaboration Messaging Configuration   Save Save and Close Cancel

Enable Collaboration Messaging

External Partner Role	Collaboration Business Process	Enabled
Supplier	Procure To Pay	☑

Message Storage

☑ Store Outbound Source Document   ☑ Store Inbound Source Document

☑ Store Outbound Transformed Document   ☑ Store Inbound Transformed Document

Outbound Sender

Global Sender ID 888877771   Global Sender ID Type DUNS   * Maximum attachment size for outbound messaging 2 MB

Manage Collaboration Messaging Service Provider

There are a few service providers defined out of the box including Oracle Supplier Network (Oracle Business Network). You can define additional service providers.

Service Providers are intermediaries involved in the transmission of messages between your Oracle Applications Cloud and your Trading Partners.

An organization may have multiple such intermediaries, to handle different messaging scenarios. As an example, consider an organization that uses a third party service provider for EDI, an in-house B2B Gateway for most XML messages, and Oracle Supplier Network for communication with a specific set of suppliers. This will be set up as three service providers in the Collaboration Messaging Framework.

Collaboration Messaging Framework receives/sends communication from/to the service provider, and this is used to exchange messages for all Trading Partners that are on the service provider's network.

As an example, in New Zealand, the service providers can be access point providers. You can create new service provider in CMK

1. Navigate to Navigator > Tools > Collaboration Messaging
2. Go to the task "Manage Collaboration Messaging Service Provider". Use the ID type = "Name"

The below screenshots represent setting up of a service provider to be used with B2B adapter. If you plan to use web service, you can change the delivery method accordingly. You can also have email or UCM as delivery method.

Edit Collaboration Messaging Service Provider: exchange   Save Save and Close Cancel

Name exchange
Provider ID exchange
ID Type Name

Delivery Methods Outbound Collaboration Messages Inbound Collaboration Messages

Actions ▾ View ▾ + ✕

* Name	* Delivery Method Type	Service Name	Security Policy	Recipient Email ID	Attach Payload	Endpoint	User Name	Password	UCM File Export Document Account	B2B Routing Enabled
exchangeB2B	B2B Adapter				—					✓

Edit Collaboration Messaging Service Provider: exchange   Save Save and Close Cancel

Last Saved 3/30/20 6:50 AM

Name exchange
Provider ID exchange
ID Type Name

Delivery Methods Outbound Collaboration Messages **Inbound Collaboration Messages**

Actions ▾ View ▾ + □ ✕

* Name	* Collaboration Message Definition	* Status	Application Partner Code XPath	Retention Duration	Retention Duration Unit
UBL_2.1_Invoice_In	UBL_2.1_INVOICE_IN	Active	Invoice/BuyerReference	3	Year

If you use the same service provider for AR Invoices (Outbound Invoices), you will need to define the outbound message as well (as below).

Set up Trading Partner

In collaboration messaging, you need to create a trading partner to identify different entities, such as your customer or supplier, for B2B messaging.

Before you can exchange B2B messages with a trading partner, you need to complete these high-level tasks:

1. Set up the trading partner with or without a service provider.
2. Associate the trading partner with the entity it represents; for example, a customer, customer account, or supplier site.

You can also optionally set up confirmation codes and message processing rules.

Set up B2B

This is required only if you are using B2B adapter (and not web service) as delivery method

1. Create B2B partner (this is mapped to CMK Service Provider) with the same name as CMK service provider. Use ID type = Name
2. Add document for the partner-UBL invoice
3. Add Channel for the partner. Ensure to check "Use Proxy". (Required only for outbound processing. Skip this step if you are using only inbound invoice).

4. Add agreement for the trading partner- map the document and direction. Validate and deploy

CMK Web service

Set it up, if you are using CMK web service for inbound messages.

You can use the access point with the following endpoint.

Error! Hyperlink reference not valid. Replace <pod> with your actual pod.

Supplier Site/Trading Partner/Application Partner set-up

This task must be performed for each supplier that needs to be enabled for B2B messaging using CMK. Navigate to Suppliers:

1. Search for the supplier to be enabled for CMK. Select the supplier site.

2. Under B2B Communication section, select the **B2B Communication method** as "Collaboration Messaging Framework"

3. Select **Manage B2B Trading Partners**. Click on the Create button and select **Create Trading Partner with Service Provider**.

4. Fill in the appropriate information in the **Create Trading Partner** dialog.

5. Select **Actions** > **Add Row**. Select documents to be enabled for collaboration messaging. For receiving invoices from supplier **select** PROCESS_INVOICES_IN. Change the Status to Active. **Select Save and Close**. Select OK. This will navigate you back to the Edit Site page.

6. Select **Edit** under Associated Collaboration Documents on Edit Site page. Under Associated Service Provider, select **Action** > **Add Row** to add the Service Provider created earlier. Under Collaboration Documents, select **Action** > **Add Row** and add the documents selected earlier. Ensure that Association Status is set to "Enabled". Select Save and Close. This step associates the service provider with the supplier and generates an **Application Partner Code**. System defaults it to "Supplier number-Site Name", but you can change it to any other identifier. This code must be communicated to suppliers since they need to send this code as the B2B site code in the XML payload.

PROCESSING XML INVOICES

Once the set-up is completed, the customer can receive XML invoices from the supplier.

- The supplier sends the XML invoice through web service or through B2B adapter etc.
- B2B adapter/Web service submits the XML payload to CMK.
- CMK transforms the XML payload and loads the invoice into the Payables Interface tables.
- Run the "Import Payables Invoices" process with source as "B2B XML Invoice". Optionally, schedule the request set to run as often as you want to process XML invoices.

FREQUENTLY ASKED QUESTIONS

General questions like licensing, supported formats etc.

1. Does Oracle Applications Cloud support UBL 2.1 messages?

Oracle Applications Cloud enables customers to use UBL 2.1 messages. Please see details of data elements mapping section. Further, we also enabled EN16931 standard in Release 20A. The functionality that is supported includes credit notes, payment method (payment means code: BT-21), line level charge reason code etc. Most of the data items that you will need for inbound invoicing are covered. Additionally, you can customize the mapping either by DVM (Domain Value Map) or by modifying the style sheet. Details are covered in another section of this document here or here.

2. What are the other formats that are supported by Oracle Applications Cloud?

As of Release 20A, we support OAGIS 7.2.1 and OAGIS 10.2, cXML and UBL 2.1. The messages can also be customized based on specific customer requirements.

3. Do we need OBN (Oracle Business Network) / OSN (Oracle Supplier Network) registration to use B2B communication?

Using OBN for B2B communication (including electronic invoicing) is not mandatory. OBN is one of the options to implement electronic invoicing. Other options are 1) Web service directly from CMK 2) B2B adapter with CMK. Note that, currently UBL invoice is not one of the supported documents for CMK/OSN combination (we plan to add the ability to override the message definitions of the seeded service providers, so that would allow for different messaging standards such as UBL, in future)

4. We have subscribed to Oracle Applications Cloud. Do we need any other Oracle infrastructure license for using B2B communication?

No, you do not need any other license to make B2B invoicing work. Internally it uses CMK and B2B- but it does not involve any extra licensing.

5. Is B2B / Trading Partner Channel set-up required when using Collaborative Messaging Framework (CMK)?

The B2B / Trading Partner Channel set-up is no longer applicable since the introduction of Collaborative Messaging Framework (CMK). However, certain minimal B2B set-ups are required to make it work with CMK, if you choose to use B2B adapter as delivery method (Details of set-up required are covered in this document separately). No B2B set-ups are required, if you use web service/email as delivery method.

6. Does CMK Support AS2?

CMK supports AS2 from 20B onwards

Set-up related questions

1. Who are the potential parties involved in an e-invoicing scenarios and how are they defined In Oracle Applications Cloud?

The following parties are typically involved in an e-invoicing transaction.

Parties in invoice transaction	CMK Definition	B2B Definition
Service Provider/Access Point Provider	Service Provider	Partner
Supplier	Trading Partner	Not Applicable
Customer Account	Trading Partner	Not Applicable
Supplier Site	Trading partner/Application Partner	Not Applicable

2. What is the role of service Provider?

Service Providers are intermediaries involved in the transmission of messages between your Oracle Applications Cloud and your Trading Partners.

An organization may have multiple such intermediaries, to handle different messaging scenarios. As an example, consider an organization that uses a third party service provider for EDI, an in-house B2B Gateway for most XML messages, and Oracle Supplier Network for communication with a specific set of suppliers. This will be setup as three service providers in the Collaboration Messaging Framework.

Collaboration Messaging Framework receives/sends communication from/to the service provider, and this is used to exchange messages for all Trading Partners that are on the service provider's network.

As an example, in New Zealand, the service providers are access point providers. You can create new service provider in CMK.

3. What is the difference between Trading Partner and Application Partner?

A Trading Partner is a single, consistent way of identifying the different application entities involved in B2B messaging – i.e. within Oracle Applications Cloud; there are different entities such as customer, supplier etc., but they all represent Trading Partners from a B2B point of view.

In Payables, a supplier is a trading partner. Therefore, we need to relate a supplier with a trading partner ID.

Your service provider will identify both you and your customer by the trading partner IDs. You should consider what best identifier to denote a trading partner. You can take guidance from your service provider as trading partner can be in the context of a service provider. It can vary by country as well. For example, one of the options for New Zealand can be using NZBN as trading partner IDs.

Application partner refers to internal business object within Oracle Applications Cloud (e.g.; customer/Supplier etc.). Associating an Application Partner with a Trading Partner establishes the business relationship between Oracle Applications Cloud customer and the Trading Partner. In this context, it is specific to a supplier site (not to supplier). When you set up B2B communication for a supplier, you can assign an application partner code to a supplier site (system by default, assigns "Supplier Number-Supplier Site", you can change it to the right identifier based on the process you have decided to follow). Therefore, your system can distinguish invoices sent by different sites of the same supplier by application partner codes. The data element of the xml invoice that is picked up by the system for application partner code is configurable.


In EN16931 UBL 2.1 the suggested data element for this is "Invoice/BuyerReference". This data element should match application partner code for any given Supplier Site in a BU. Hence, the leading practice is to use Invoice/BuyerReference as the data element to map application partner.

4. How can I convert the Unit of Measures in XML file to the Unit of Measures we have defined in our system?

You can map the values in DVM. DVM is a generic solution that can be used for such mappings. Other options currently available are currency code, payment terms etc.

For Unit of Measure, the template already incorporates a domain-value map (DVM). The DVM maps the values from supplier invoice to the corresponding Oracle application values. A DVM named UOM_CODE is seeded in the system and customers can add the mapping by following the steps below. (For more details, refer to the documentation here

3. Go to Setup and Maintenance and search for the task "Manage Collaboration Messaging Domain Value Map"
4. It will open the SOA Composer UI. Select Open > Open DVM and select FusionCode.DVM
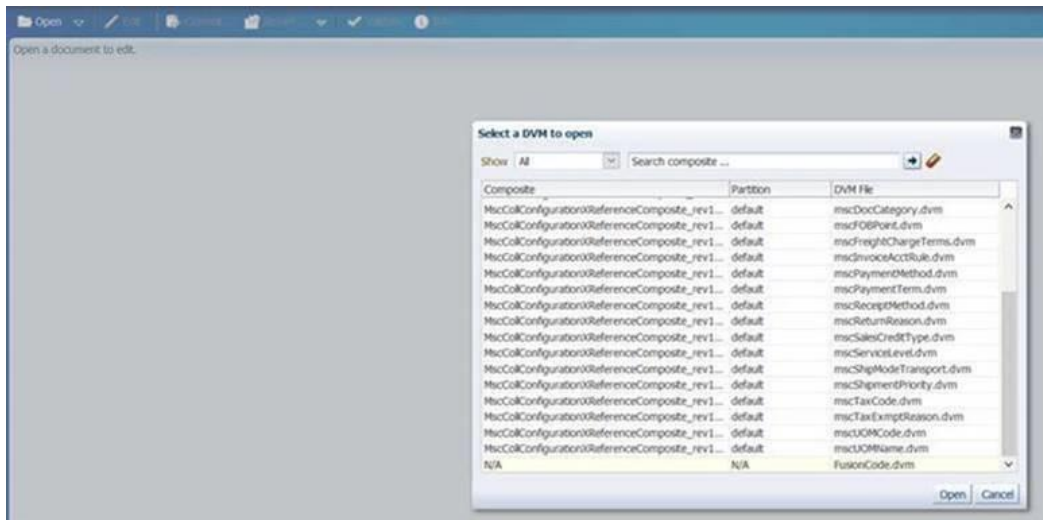
5. Click on Edit button
6. Add an entry for DVM Name UOM_CODE. Enter the value configured in Oracle Applications (Fusion Value), the value received from supplier (Trading Partner Value) and the Trading Partner Qualifier. The below example is for converting the trading partner value "C62" to Oracle applications value "EA". The Trading Partner Qualifier (PEPPOL) is used to associate one or more suppliers with this particular mapping as explained in the following steps.
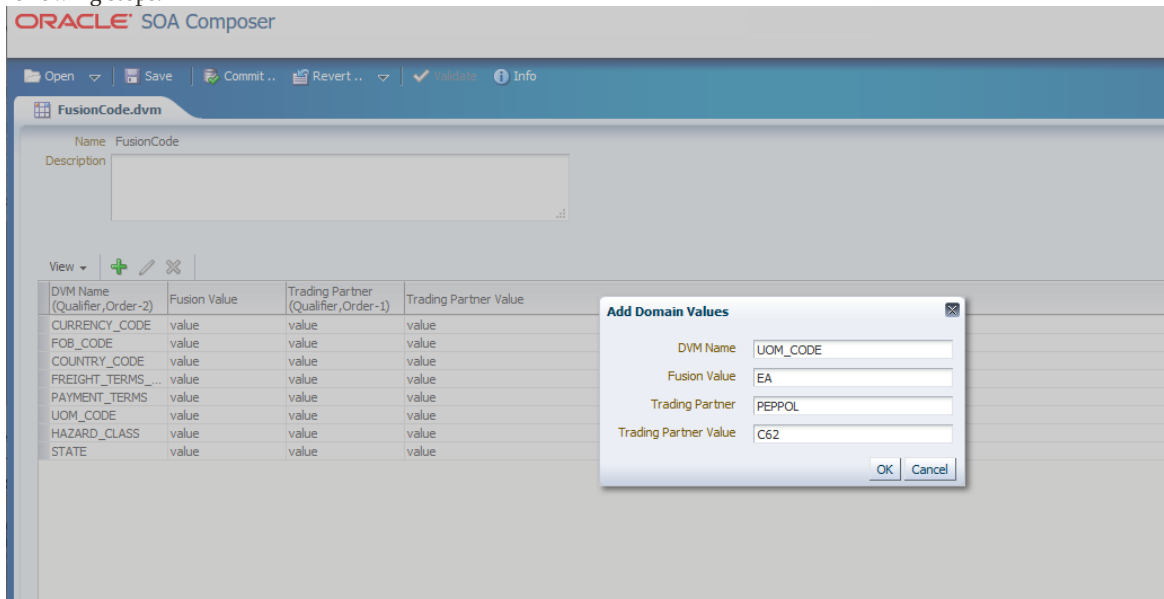
7. Make sure to Save and Commit the changes
8. On the Supplier Site, Edit the Associated Collaboration Documents
9. In the Domain Value Map field, enter the same value entered for Trading Partner in step 4 (PEPPOL in this example)
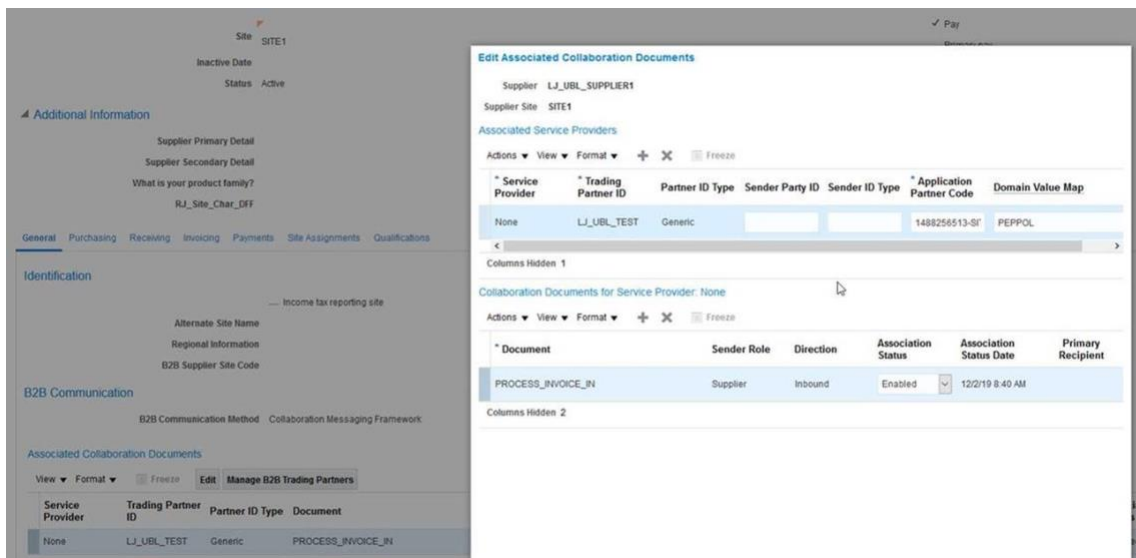
10. Now test by sending an invoice and check if the UOM gets updated

5. Does DVM always need to be defined in the context of a trading partner? Can we define DVM mapping only once that is valid for all trading partners?

DVM is always in the context of a trading partner. However, you can reuse the same DVM for multiple TPs (Trading Partner). Specify the same value for "Domain Value Map" (i.e. the TP Qualifier) for multiple TPs.

6. What are the configurations required to implement inbound electronic invoicing?

The details of set-ups and configurations are detailed in a separate section here. Additionally, please refer to implementation guide of CMK and Payables for details

Invoice processing related questions

1. Invoice payload failed with the error "Application partner not found". How can I resolve this?

This occurs when the supplier identification provided in the payload does not match with the Application Partner Code for the supplier site in Oracle Applications Cloud. Verify the Application Partner Code for the site and ensure that the same is populated in the payload.

2. Can we handle negative invoices?

Yes, your Oracle Cloud can handle negative invoices. Negative invoices are created as credit memos. It can handle credit note as well (only UBL 2.1 at present).

3. Can users within my organization be notified when e-invoices are processed?

We do not support sending notifications to the users when e-invoices are received/processed/rejected until they are imported to Payables base tables, consider using the dashboards of CMK for this. However, notifications to suppliers are sent by email, if an invoice is rejected and supplier's email is available in our system. Additionally, job completion status notifications can be sent to users. Standard invoice processing notifications are anyways available for e-invoices as well.

4. Can the e-invoices be stored in my system as historical records?

Yes, they can be stored as CMK messages. The duration of storage can be specified in the set-up of service provider.

5. Can I print or make pdf out of the e-invoice in human readable form, so that I can send it to others within my organization for offline approval, vetting etc.?

As the e-invoice is primarily for machine-to-machine communication-, it is an xml file without a layout. We do not support printing it in a human readable form with a specific layout. The xml file as such can be printed. In addition, Oracle standard approval processes can be used for e-invoices as well.

6. My auditor needs a printout of the e-invoice in human readable form.

As the e-invoice is primarily for machine-to-machine communication-, it is an xml file without a layout. We do not support printing it in a human readable form. The xml file as such can be printed.

7. The invoice XML file does not have Purchase Order reference. Will it be rejected?

No, the system will import it as Non-PO invoice. However, if there is any incorrect PO reference, it will fail validation and get rejected.

8. Can Oracle Applications Cloud send automated feedback response to the supplier after processing the invoices?

The supplier gets an email with appropriate feedback if the supplier email details are available. Currently a separate XML response message is not sent through the service provider.

9. The supplier's invoice contains incorrect payment means code. Can we just use default Supplier's payment method for non-PO invoices?

The payment means code of the XML file can be mapped to a payment method by DVM (Domain Value Map). If you prefer getting it from the supplier's default payment method, you can modify the style sheet to ignore the values coming from the XML file.

10. How are Freight and Miscellaneous lines allocated?

Freight and Miscellaneous lines are prorated across all item invoice lines for e-invoices. They cannot be allocated to specific lines.

Troubleshooting

1. We need to change the mapping for our specific requirement. Can we do that?

You can customize the mapping easily by modifying the default stylesheet (XSLT) associated with the message definition.

CMK delivers pre-seeded message definitions that map the XML payload fields to the Oracle Applications Cloud application objects such as Invoices using XSLT files. Customers can change the mappings delivered out of the box, to accommodate either trading partner specific variations, or specific requirements that are not addressed by the out-of-the-box mappings.

The steps for updating the mapping are as below:

1) Navigate to Collaboration Messaging work area(Navigator-Tools-Collaboration Messaging) and click on Manage Collaboration Message Definitions

2) Search for the message by selecting Direction as Inbound and Document as PROCESS_INVOICE_IN

3) From the search results, select the message UBL_2.1_INVOICE_IN and select Actions > Duplicate Collaboration Message.

You can edit the defaulted Message Name if required. Select the modified XSLT attached. Save and Close.

4) The new message created must be associated with the B2B Trading Partner or the Service Provider

1. From the task Manage Collaboration Messaging Service Providers.

2. Select the Inbound Collaboration Messages tab.

3. Mark the existing Message as Inactive and duplicate the Message

4. Enter Name and select the new Message definition that you created.

5. Make the new message Active.

6. Save and Close.

2. Invoice payload failed with the error "Application partner not found". How can I resolve this?

This occurs when the supplier identification provided in the payload does not match with the Application Partner Code for the supplier site in Oracle Applications Cloud. Verify the Application Partner Code generated for the site and ensure that the same is populated in the payload.

3. Invoices get rejected during import with "Insufficient tax information" rejection. How can I resolve this?

If invoice payload contains tax lines, they will be processed as manual tax lines on the invoice.

Below is a background on how taxes are calculated.

There are multiple ways to handle taxes on XML invoices.

1. System calculated taxes
2. Manual tax lines

System calculated taxes

When an invoice is validated, the system calculates tax based on the tax rules configured. If you want the tax to be calculated automatically, you can configure the XSLT to ignore the tax lines sent by the supplier. The Item lines will be populated in the interface and imported. Tax will be calculated automatically during validation.

However, some customers may have a requirement of creating invoice with the exact tax amount provided by the supplier. The recommended way is to take the tax amount provided in the XML invoice and populate it in the tax control amount fields. The system calculates tax as per the configured rules. Oracle Fusion Tax adjusts the calculated taxes based on the specified tax control amount in proportion to the tax amounts determined for the applicable taxes. So the invoice will have the same tax amount as provided in XML payload.

This requires updating the default mapping of tax lines from XML to AP interface.

CMK delivers pre-seeded message definitions that map the XML payload fields to the Oracle Applications Cloud application objects such as Invoices using XSLT files. Customers can change the mappings delivered out of the box, to accommodate either trading partner specific variations, or specific requirements that are not addressed by the out-of-the-box mappings.

You can update the XSLT file to map the tax amount from the XML invoice to the tax control amount on the invoice. The steps for updating the mapping are as below:

1. Navigate to Collaboration Messaging(Navigator-Tools-Collaboration Messaging) work area and click on Manage Collaboration Message Definitions

2. Search for the message by selecting Direction as Inbound and Document as PROCESS_INVOICE_IN

From the search results, select the message ORCL_CMK_PROCESS_INVOICE_002 for OAG 7.2.1 or OAGIS_10.1_PROCESS_INVOICE_COLLAB_MSG_IN for OAG 10.1. Download the transformation package by selecting Actions > Export Transformation Package.

3. Transformation package gets downloaded as a zip file with source and target XSDs and the XSLT. Customers can update mappings of the downloaded XSLT and save it to local system, so that his can be uploaded while duplicating a message.

4. The seeded message definitions cannot be modified. You can duplicate the seeded message and upload the modified XSLT. Select the message to be modified and select Actions > Duplicate Collaboration Message.

Edit the defaulted Message Name if required and select the modified XSLT to be uploaded. Save and Close.

5. The new message created must be associated with the B2B Trading Partner or the Service Provider.
6. If the trading partner sends invoices through a service provider, add the new message definition to the service provider from the task Manage Collaboration Messaging Service Providers.
   a. Select the Inbound Collaboration Messages tab.
   b. Mark the existing Message as Inactive and duplicate the Message
   c. Enter Name and select the new Message definition that you created.
   d. Make the new message Active.
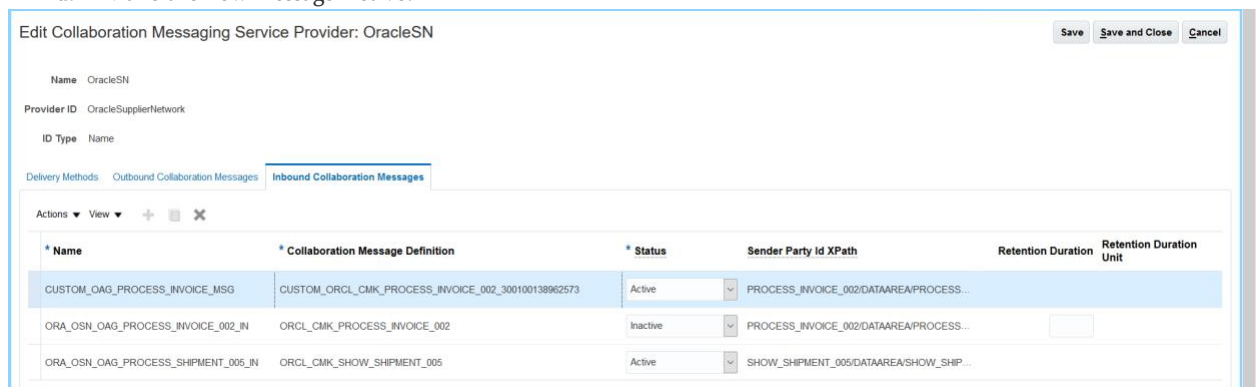
7. If the trading partner sends invoices directly to CMK web service, then add the new message definition to the trading partner from Manage B2B Trading Partners
   a. Select the Inbound Collaboration Messages tab.
   b. Mark the existing Message (if any) as Inactive and Add new message.
   c. Enter Name and select the new Message definition that you created.
   d. Make the new message Active.
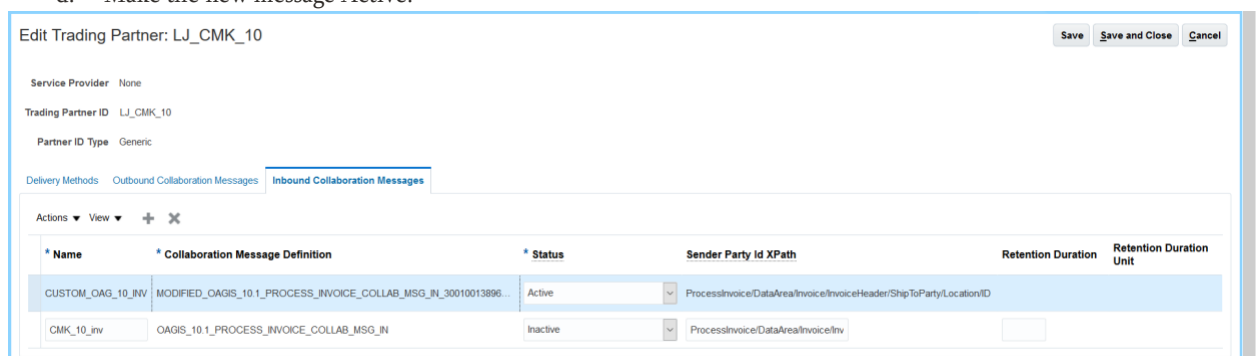
8. Save and Close.

Manual tax lines

Tax lines on XML invoice payload can be imported into Payables as manual tax lines with the amount in the XML invoice. In order to import manual tax lines, the tax classification codes are required. However, suppliers are not aware of the tax codes

setup in the customer's internal system. In this case, tax code can be defaulted on the tax lines using Application Tax Options setup. The instructions to setup the tax defaulting are as below:

1. Go to "Manage Application Tax Options" in FSM
2. Search for the application "Payables"
3. Select the Business Unit and click on Edit button
4. Set the Defaulting Order 1 as "Supplier Site". Leave the Defaulting Order 2-6 as blank.

5. Save and Close
6. Go to "Manage Party Tax Profiles".
7. Select "Third-Party Site Tax Profiles" in the LOV at the top and search for the supplier

8. The supplier sites will be listed in the search results. Select the site and click on Edit.
9. Select the Tax Classification Code

10. Save and Close

4. The supplier sends XML invoice in UBL standard with alphanumeric line identifiers. Can we process the invoice?

Typically, invoice line identifiers are numeric in the industry. Oracle Applications Cloud needs line identifiers with numeric values for processing invoices (This is true for any invoice, including paper invoices) . However, if a supplier sends an XML invoice with line identifiers as string values (alphanumeric), you can change the style sheet to ignore these values and let Oracle generate them automatically. The procedure for using a new style sheet is as below

11. Save the revised XSLT on local system
12. Navigate to Collaboration Messaging work area and click on Manage Collaboration Message Definitions
13. Search for the message by selecting Direction as Inbound and Document as PROCESS_INVOICE_IN
14. Select the message definition that was created earlier and click on Edit.
15. Select the updated XSLT from the saved location
16. Test with a new invoice

5. How can I search for suppliers based on their National Unique Identifier (e.g.; Tax Registration number etc.)

You can search on some of the identifiers (like DUNS etc.) , on the UI screen of "Manage Supplier". However, if you have any other national identifier (e.g.; NZBN in New Zealand, ABN in Australia etc.), you can define them in the Party Tax Profile (with country context but without any reference to any specific Tax Regime) making them available for search through "Manage Supplier" UI.

Below is a screenshot for a supplier with multiple tax identifiers. The one with the country (e.g. Country - New Zealand, Registration Number - XXX) is defined as PTP (Party Tax Profile) without any reference to tax regimes and can be searched on from the Manage Supplier screen. The tax registration number provided here is validated per the setups done at the Country Default Controls.

You can define it on the Manage Supplier Screen.

Please note that any identifier defined at Tax Regime level is not searchable from Manage Supplier UI.

FIELD MAPPING

The following table shows the field in the EN16931 XML payload which is used by Oracle Applications Cloud Payables to process the invoice payload.

Invoice Header Attributes

ATTRIBUTE	BT CODE	UBL ELEMENT	PAYABLES INTERFACE COLUMN
Invoice number	BT-1	/Invoice/ID	AP_INVOICES_INTERFACE .INVOICE_NUM
Invoice date	BT-2	/Invoice/IssueDate	AP_INVOICES_INTERFACE .INVOICE_DATE
Invoice type code	BT-3	/Invoice/InvoiceTypeCode	AP_INVOICES_INTERFACE.INVOICE_TYPE_LOOKUP_CODE
Description	BT-22	/Invoice/Note	AP_INVOICES_INTERFACE.DESCRIPTION
Invoice currency code	BT-5	/Invoice/DocumentCurrencyCode	AP_INVOICES_INTERFACE.INVOICE_CURRENCY_CODE
Buyer reference	BT-10	/Invoice/BuyerReference	comes from configuration
Purchase order number	BT-13	/Invoice/OrderReference/ID	AP_INVOICES_INTERFACE.PO_NUMBER
Receipt number	BT-15	/Invoice/ReceiptDocumentReference/ID	AP_INVOICES_INTERFACE.RECEIPT_NUMBER
Supporting document reference	BT-122	/Invoice/AdditionalDocumentReference/ID	AP_INVOICES_INTERFACE.EXTERNAL_DOC_REF
URL attachment	BT-124	/Invoice/AdditionalDocumentReference/ Attachment/ExternalReference/URI	AP_INVOICES_INTERFACE.IMAGE_DOCUMENT_URI
Supplier name	BT-28	/Invoice/AccountingSupplierParty/ Party/PartyName/Name	AP_INVOICES_INTERFACE.VENDOR_NAME
Seller tax identifier	BT-31	/Invoice/AccountingSupplierParty/ Party/PartyTaxScheme/CompanyID	AP_INVOICES_INTERFACE.THIRD_PARTY_REGISTRATION_NUM We map BT-31 to the Supplier Tax Registration number column out of the box. If customer wants to map BT-29 or BT-30 instead, they can customize it.
Supplier email	BT-43	/Invoice/AccountingSupplierParty/ Party/Contact/ElectronicMail	AP_INVOICES_INTERFACE.VENDOR_EMAIL_ADDRESS

Buyer tax identifier	BT-48	/Invoice/AccountingCustomerParty/ Party/PartyTaxScheme/CompanyID	AP_INVOICES_INTERFACE.FIRST_PARTY_REGISTRATION_NUM
Buyer name	BT-44	/Invoice/AccountingCustomerParty/ Party/PartyLegalEntity/RegistrationN ame	AP_INVOICES_INTERFACE.LEGAL_ENTITY_NAME
Buyer legal registration ID	BT-47	/Invoice/AccountingCustomerParty/ Party/PartyLegalEntity/CompanyID	AP_INVOICES_INTERFACE.CUST_REGISTRATION_NUMBER
Goods received date	BT-72	/Invoice/Delivery/ActualDeliveryDat e	AP_INVOICES_INTERFACE.GOODS_RECEIVED_DATE
Ship to location	BT-71	/Invoice/Delivery/DeliveryLocation/I D	AP_INVOICES_INTERFACE.SHIP_TO_LOCATION
Payment method	BT-81	/Invoice/PaymentMeans/ PaymentMeansCode	AP_INVOICES_INTERFACE.PAYMENT_METHOD_CODE
Remittance information	BT-83	/Invoice/PaymentMeans/PaymentID	AP_INVOICES_INTERFACE.UNIQUE_REMITTANCE_IDENTIFIER
Payment Terms	BT-20	/Invoice/PaymentTerms/Note	AP_INVOICES_INTERFACE.TERMS_NAME
Invoice amount	BT-115	/Invoice/LegalMonetaryTotal/Payable Amount	AP_INVOICES_INTERFACE.INVOICE_AMOUNT While applying prepayment, user has to check the flag "Included on Invoice".

Header Level Charges

ATTRIBUTE	BT CODE	UBL ELEMENT	PAYABLES INTERFACE COLUMN
Charge indicator		/Invoice/AllowanceCharge/ChargeIndi cator	True = Charge. Create line with positive amount. False = Discount. Create line with negative amount.
Charge reason code	BT-98	/Invoice/AllowanceCharge/Allowance ChargeReasonCode,	AP_INVOICE_LINES_INTERFACE.LINE_TYPE_LOOKUP_CODE Charge Reason Code will be translated to the Payable line type as below: Charge Reason Code / Meaning / Payables value FC / Freight service / FREIGHT SAA / Shipping and handling / FREIGHT Any other / Any other / MISC If the charge indicator is False (discount) then Line Type Lookup Code will be MISC and the line amount will be negative.
Charge reason	BT-97	/Invoice/AllowanceCharge/Allowance ChargeReason	AP_INVOICE_LINES_INTERFACE.DESCRIPTION
Charge amount	BT-92	/Invoice/AllowanceCharge/Amount	AP_INVOICE_LINES_INTERFACE.AMOUNT

Invoice Line Attributes

ATTRIBUTE	BT CODE	UBL ELEMENT	PAYABLES INTERFACE COLUMN
Invoice line number	BT-126	/Invoice/InvoiceLine/ID	AP_INVOICE_LINES_INTERFACE.LINE_NUMBER
Line Description	BT-127	/Invoice/InvoiceLine/Note	AP_INVOICE_LINES_INTERFACE.DESCRIPTION
Quantity	BT-129	/Invoice/InvoiceLine/InvoicedQuantity	AP_INVOICE_LINES_INTERFACE.QUANTITY_INVOICED
Line amount	BT-131	/Invoice/InvoiceLine/LineExtensionAmount	AP_INVOICE_LINES_INTERFACE.AMOUNT
Distribution account	BT-133	/Invoice/InvoiceLine/AccountingCost	AP_INVOICE_LINES_INTERFACE.DIST_CODE_CONCATENATED
PO line number	BT-132	/Invoice/InvoiceLine/OrderLineReference/LineID	AP_INVOICE_LINES_INTERFACE.PO_LINE_NUMBER
Item Description	BT-154	/Invoice/InvoiceLine/Item/Description	AP_INVOICE_LINES_INTERFACE.ITEM_DESCRIPTION
Item identifier	BT-156	/Invoice/InvoiceLine/Item/BuyersItemIdentification/ID	AP_INVOICE_LINES_INTERFACE.INVENTORY_ITEM_ID
Vendor Item number	BT-155	/Invoice/InvoiceLine/Item/SellersItemIdentification/ID	AP_INVOICE_LINES_INTERFACE.VENDOR_ITEM_NUMBER
Unit Price	BT-146	/Invoice/InvoiceLine/Price/PriceAmount	AP_INVOICE_LINES_INTERFACE.UNIT_PRICE

Line Level Charges

ATTRIBUTE	BT CODE	UBL ELEMENT	PAYABLES INTERFACE COLUMN
Charge indicator		/Invoice/InvoiceLine/AllowanceCharge/ChargeIndicator	True = Charge False = Discount
Charge reason code	BT-98	/Invoice/InvoiceLine/AllowanceCharge/AllowanceChargeReasonCode,	AP_INVOICE_LINES_INTERFACE.LINE_TYPE_LOOKUP_CODE Charge Reason Code will be translated to the Payable line type as below: Charge Reason Code / Meaning / Payables value FC / Freight service / FREIGHT SAA / Shipping and handling / FREIGHT Any other / Any other / MISC
Charge reason	BT-97	/Invoice/InvoiceLine/AllowanceCharge/AllowanceChargeReason	AP_INVOICE_LINES_INTERFACE.DESCRIPTION
Charge amount	BT-92	/Invoice/InvoiceLine/AllowanceCharge/Amount	AP_INVOICE_LINES_INTERFACE.AMOUNT

GLOSSARY OF TERMS

- A **collaboration document** represents a business document exchanged with a specific Trading Partner Role. Each document can be represented in many external message types, and a specific message type has to be selected for exchange with the partner.
- An **external message type** is the representation of a collaboration document in a specific format (standard or custom) for exchange with the Trading partner (e.g. OAGIS 7.2.1, OAGIS 10.1, UBL 2.1 etc.) .
- A **collaboration message definition** identifies a specific transformation (XSLT) applied between an Oracle Applications Cloud business object and an external message type, and some additional configuration information that is used for processing messages.
- A **Service Provider** is an intermediary involved in the transmission of messages between Oracle Applications Cloud and Trading Partners.
- A **Trading Partner** is a supplier or customer with whom B2B messages are exchanged (also known as External Partner) is a single, consistent way of identifying the different application entities involved in B2B messaging– i.e. within Oracle Applications Cloud, there are different entities such as customer, supplier etc., but they all represent Trading Partners from a B2B point of view.
- An **Application Partner** is a B2B Trading Partner defined in Oracle Applications Cloud such as supplier site, customer account or customer
- A **Delivery Method** specifies how messages are transmitted electronically to the Trading Partner.
- An **outbound message** is a message sent to a Trading Partner using a specific delivery method.
- An **inbound message** is a message that was received from a Trading Partner.
- A **collaboration event** is a unique name for a business action that triggers an outbound collaboration document.
- **CMK- Collaboration Messaging Framework (CMK)** supports messaging with Trading Partners using Service Providers (e.g.; Oracle Business Network or any other network). In this model, the service provider acts as an intermediary between the Trading Partners and the Oracle Applications Cloud customers.
  Once a Trading partner is on-boarded to a service provider network, the partner can exchange messages with any Oracle Applications Cloud client.
- **OBN (Oracle Business Network) /OSN (Oracle Supplier Network)** is an open community for Oracle Cloud Applications, E-Business Suite, and PeopleSoft customers to exchange business documents with their Trading Partners. It provides out-of-the-box connectivity for Oracle customers and easy connection options for your Trading Partners.
- **DVM (Domain value map)** operates on actual data values that transit through the infrastructure at runtime. They enable you to map from one vocabulary used in a given domain to another vocabulary used in a different domain. For example, one domain may represent a city with a long name (Boston), while another domain may represent a city with a short name (BO). In such cases, you can directly map the values by using domain value maps.

CONNECT WITH US

Call +1.800.ORACLE1 or visit oracle.com.
Outside North America, find your local office at oracle.com/contact.

blogs.oracle.com   facebook.com/oracle   twitter.com/oracle

White Paper- UBL 2.1 Electronic Invoicing in Oracle Payables Cloud
May 2222