# OUM

# TA.070-4 DISEÑO TÉCNICO DE EXTENSIONES E INTERFACES



Autor:	Oracle Consulting
Fecha de creación:	enero 15, 2018
Última actualización:	enero 15, 2018
Código de referencia OUM:	DS.140 Design Specification
Versión:	1.0
Aprobadores:	
Ivan Muñoz	
José Ángel Flores	



# **Control de Documento**

#### 1.1 Bitácora de Cambios

Fecha	Autor	Versión	Referencia del cambio
15-ene-2018	Edgar Cruz	1.0	No Previous Document

# 1.2 Revisores

Nombre	Posición

# Contenido

Control de Documento	<i>i</i>
1.1 Bitácora de Cambios	i
Resumen técnico	1
2.1 Diagrama de la integración	3
Lógica de implementación	
3.1 InvoicesEnt	
3.1.2 LoadInvoices	10
3.2 InvoicesBiz	17
3.2.2.2 Pruebas UpdateStatusInvoices	24
3.2.3.1 Lista de objetos GetComplementInvoice	25
3.2.4 UpdateUuld	27
3.2.4.1 Lista de objetos UpdateUuld	30
3.2.4.2 Pruebas UpdateUuld	31
3.3 Resultados de prueba de notificaciones de carga y actualización de transacciones_	32
Diseño de Datos	34
4.1 Origen de Datos	34
	35
	39
	1.1 Bitácora de Cambios  Resumen técnico  2.1 Diagrama de la integración  Lógica de implementación  3.1 InvoicesEnt  3.1.1 Getinvoices  3.1.1.1 Lista de objetos Getinvoices  3.1.2.1 Loadinvoices  3.1.2.2 Loadinvoices  3.1.2.3 Pruebas (Getinvoices)  3.2.1 Lista de objetos Loadinvoices  3.2.1 Buildinvoices  3.2.1.1 Lista de objetos Buildinvoices  3.2.1.2 Pruebas Buildinvoices  3.2.1.2 Pruebas Buildinvoices  3.2.1.3 Lista de objetos Buildinvoices  3.2.1.2 Pruebas Buildinvoices  3.2.1.3 Lista de objetos UpdateStatusInvoices  3.2.2.1 Pruebas UpdateStatusInvoices  3.2.2.2 Pruebas UpdateStatusInvoices  3.2.3.3 GetComplementToUpdateInvoices  3.2.3.1 Lista de objetos GetComplementInvoice  3.2.3.2 Pruebas GetComplementInvoices  3.2.3.1 Lista de objetos UpdateUuld  3.2.4.1 Lista de objetos UpdateUuld  3.2.4.2 Pruebas UpdateUuld  3.2.4.3 Resultados de prueba de notificaciones de carga y actualización de transacciones  Diseño de Datos  4.1 Origen de Datos  4.2 Lógica de Validación  Diseño SQL

6.1 Diseño del Servicio	39
7 Consideraciones de Rendimiento	40
7.1 Estrategia de Reinicio	40
7.2 Seguridad	40
7.3 Personalización	40
7.4 Catálogo de Errores	40
8 Consideraciones de Instalación	41
9 URL de Acceso y seguridad	43
10 Temas abiertos y cerrados	44
10.1 Temas Abiertos	44
10.2 Temas Cerrados	44

#### Resumen técnico

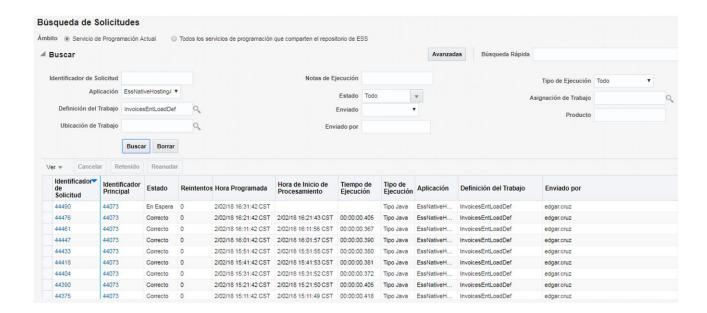
Este documento hace referencia a las especificaciones descritas en el documento "TA 020-4 ER05 AR Creación de Transacciones"; para la automatización de ejecuciones periódicas se utilizó el Oracle Enterprise Scheduler. La integración se construyó con base en el modelo de tres capas, estas son las siguientes:

- Ent (Empresarial)
- Biz (De negocio)
- Tec (Técnica)
- Ent (InvoicesEnt) es la capa empresarial en la que se encuentra el servicio iniciador del proceso, en este servicio se invocan operaciones de la capa Biz para complementar información y enviar datos a otras capas.
- Biz (InvoicesBiz) se encarga de recibir la información del Ent y utilizarla para poder crear archivos con los que cargaran las facturas y notas de crédito, registrando sus procesos exitosos o erróneos dentro de la base de datos (con ayuda de aplicaciones en la capa Tec) y a la vez notificarlos vía correo electrónico.
- Tec se hace uso de diversos servicios que ayudan a atender peticiones de la capa Biz, la capa TEC interactúa con diversos orígenes de información ajenos a los componentes de SOA.

En la capa ENT se encuentra la operación GetInvoices, esta operación se encarga de iniciar el proceso de carga de las facturas y notas de crédito por lo tanto es la operación que se manda a ejecutar con el ESS, a continuación, se muestra información de la configuración que se hizo para la ejecución periódica de la carga.

Componente	Propiedad	Valor	
Trabajo	Nombre	InvoicesEntLoadDef	
	Nombre Mostrado	InvoicesEntLoadDef	
	Paquete	/oracle/apps/ess/custom/soa	
	Descripción	Definición de trabajo para la carga de facturas y notas de crédito	
	Tipo de trabajo	SyncWebserviceJobType	
WSDL		http:// <host>:<port>/soa- infra/services/default/InvoicesEnt/InvoicesEnt?WSDL</port></host>	
	Tipo de puerto	InvoicesEntPort	
	Operación	GetInvoices	
Planificación	Nombre	InvoicesEntGetInvoicesSch	
	Nombre Mostrado	InvoicesEntGetInvoicesSch	
	Paquete	/oracle/apps/ess/custom/soa	
	Descripción	Ejecución diaria a las 8:30 am	
	Frecuencia	Diariamente	
	Zona horaria	(UTC-06:00) Ciudad de México	

Desde las pantallas del ESS se pueden buscar las ejecuciones actuales con lo cual se puede validar el estatus de las de acuerdo a la planificación con la que se ejecutó.

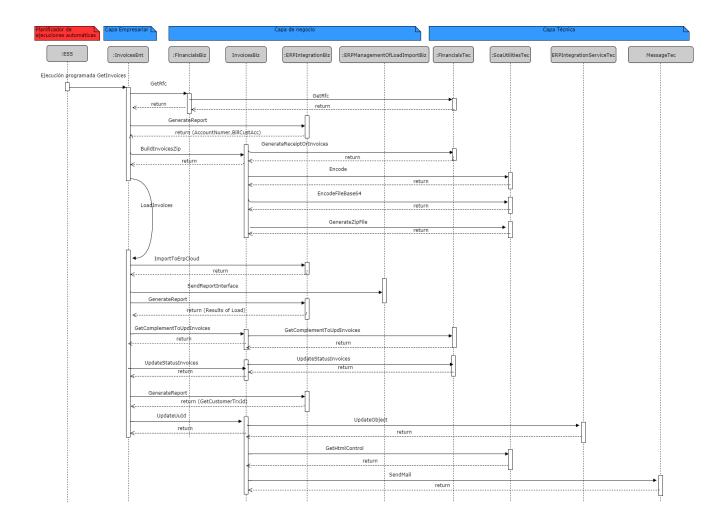


#### Existen dos tipos de integraciones:

- Inbound: En este tipo de integraciones, las aplicaciones legadas llamarán a servicios web SOAP expuestos en la nube, enviando los archivos para ser cargados en ERP Cloud por los mecanismos de integración de datos y estándares que requiera la aplicación.
- Outbound: En el ERP Cloud se dispondrán de reportes a través de BI Publisher, que mediante servicios web SOAP serán leídos y la información se procesa según corresponda.

#### Diagrama de la integración 2.1

En el siguiente diagrama de secuencias se muestra con más detalle el flujo del proceso para la carga de facturas y notas de crédito en donde la capa Ent con la operación InvoicesEnt.GetInvoices inician el flujo y este se comunica con las demás capas y diferentes servicios.

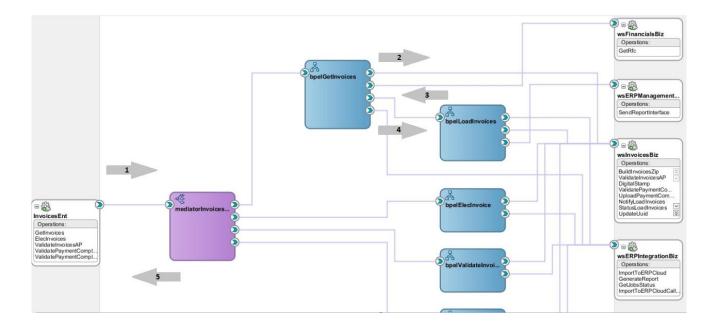


# 3 Lógica de implementación

En esta sección se describe cada uno de las fases que son necesarias para hacer la carga de facturas y notas de crédito, en donde se incluye la recuperación de información desde la base de datos, la creación de archivos CSV con las facturas o notas de crédito, la carga al ERP-Cloud, actualización y notificaciones de resultados. El proceso inicia con la aplicación compuesta InvoicesEnt, donde esta aplicación, es invocada por el ESS.

#### 3.1 InvoicesEnt

Dentro de la capa Ent se expone el servicio **InvoicesEnt** el cual expone la operación GetInvoices la cual consume la operación LoadInvoices y otros servicios comunes para completar el proceso de inicio del proceso de transacciones.



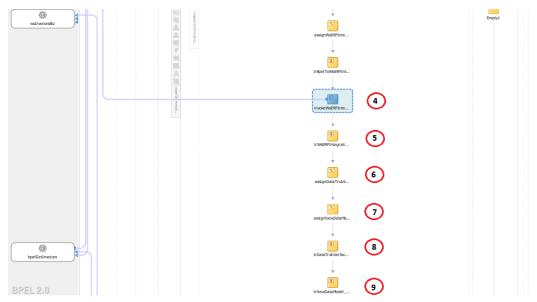
#### 3.1.1 GetInvoices

**GetInvoices**: Esta es la primera operación en ser invocada para el proceso de carga, el BPEL que se usa es de tipo síncrono a continuación se detallan los componentes que integran esta operación:

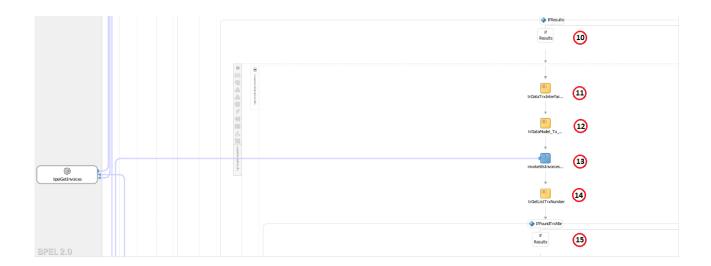
- 1. Se invoca la operación "FinancialsTec"-"GetCustomerRfc" para recuperar los RFC de los clientes que estarán disponibles para la carga de facturas y notas de crédito.
- 2. Se valida que la invocación para recuperar los RFC, haya traído resultados para poder continuar con el flujo.
- 3. Se hace uso de una transformación para establecer valores a la variable de entrada para la invocación de la siguiente operación que es consumir un reporte.
  - Se integra el control de excepciones para capturar aquellas excepciones más comunes que lleguen a pasar en el flujo completo de la operación GetInvoices.



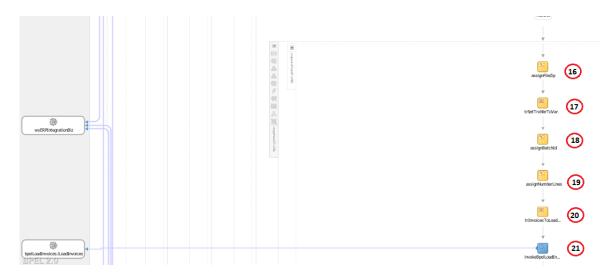
- 4. Se hace invocación al servicio "ERPIntegrationBiz"-"GenerateReport" para consumir el Data Model "XXER\_AR\_TRX\_INTERFACE" que devolverá información complementaria para poder importar las Facturas o notas de crédito.
- 5. En este paso se hace uso de una transformación (trWsERPIntegrationBizGenerateReportRs\_To\_WsERPIntegrationBizGenerateReportRs) para poder tomar la información resultante de la llamada al servicio "ERPIntegrationBiz"-"GenerateReport" y transfórmala a información en donde a los nodos se les pueda establecer determinado Namespace.
- 6. En este paso se agrega una asignación (assignDataTrxInterfaces) en la que se toma la información del paso anterior y se convierte a información de tipo XML.
- 7. Este proceso es similar al anterior solo que la información es asignada a otra variable.
- 8. En este paso se toma la información del paso 6, y ya que es información de tipo XML, se utiliza una transformación para asignar los datos a una variable con la estructura del reporte OTBI que se consultó en la invocación de servicio anterior y así poder usar la información para complementar los datos de carga.
- 9. En este paso se hace el mismo procedimiento que el paso anterior, solo que la información se establece a otra variable.



- 10. En este paso se usa el componente IF para evaluar si hubo datos devueltos del reporte OTBI anterior, si hubo datos se continua con el flujo, si no es así se entra al ELSE del IF y se termina el flujo.
- 11. Se agrega transformación para tomar datos de la variable que contiene la información del reporte y pasarlos a variable de entrada para la ejecución de Invoke.
- 12. En este paso se hace el mismo procedimiento que el paso anterior, solo que el origen es de distinta variable.
- 13. Se hace la invocación del servicio "InvoicesEnt"-"BuildInvoicesZip" que es la operación que construirá el CSV que contendrá las facturas o facturas a importar.
- 14. Se agrega transformación para obtener la lista de TRX\_NUMBER de las facturas que se tienen en el archivo CSV también se recupera el BATCH\_ID.
- 15. Se evalúa si la lista de TRX\_NUMBER no es nula para poder continuar con el flujo de importación de facturas.



- 16. En este paso se agregan valores a los parámetros de la variable de entrada para invocar el bpelLoadInvoices.
- 17. Se Utiliza transformación para pasar la lista de TRX\_NUMBER y otras variables a una variable que incluye los datos de entrada para la ejecución del servicio "ERPIntegrationBiz"-"GenerateReport".
- 18. Se agrega Asignación para recuperar el BATCH ID.
- 19. Se agrega asignación para calcular el número de líneas que incluye el CSV, que se ha de cargar.
- 20. Se agrega transformación (trInvoicesToLoadToBpelLoadInvoices) para establecer valores a variable de entrada de invocación de bpel asíncrono (bpelLoadInvoices)



21. Se invoca al bpel "bpelLoadInvoices" para poder llamar al bpel asíncrono que se encargará de ejecutar las operaciones para importar las facturas o notas de crédito al ERP-Cloud.

#### Lista de objetos GetInvoices 3.1.1.1

La siguiente tabla contiene los objetos que fueron usados para la funcionalidad de obtener información para cargar las facturas y notas de crédito al ERP-Cloud.

#	Nombre	Tipo	Descripción
1	receiveInput	Entrada	Recibe los parámetros de entrada
2	bpelGetInvoices	Alcance	Define un nuevo alcance
3	sequenceMain	Secuencia	Secuencia de asignaciones
4	assignParamsToGetRFCs	Asignación	Asignación de datos
5	InvokeWsFinancialsBizGetRfc	Invocación	Invocación de servicio
6	scopeGetInvoices	Alcance	Define un nuevo alcane
7	sequenceGetInvoices	Secuencia	Secuencia de asignaciones
8	IfRFCs	Condición	Condicionante de datos
9	assignWsERPIntegrationBizReferenceOperation Rq	Asignación	Asignación de datos
10	scopeGetInvoices_	Alcance	Define un nuevo alcance
11	sequenceGetInvoices_	Secuencia	Secuencia de asignaciones

12	trBpelToWsERPIntegrationBizGenerateReportRq	Transformación	Transformación de esquemas
13	invokeWsERPIntegrationBizReferenceOperation	Invocación	Invocación de servicio
14	trWsERPIntegrationBizGenerateReportRs_To_W sERPIntegrationBizGenerateReportRs	Transformación	Transformación de esquemas
15	assignDataTrxInterfaces	Asignación	Asignación de datos
16	assignNewDataModelTest	Asignación	Asignación de datos
17	trDataTrxInterface_To_DataTrxInterfaceResult	Transformación	Transformación de esquemas
18	trNewDataModel_To_DataModelResult	Transformación	Transformación de esquemas
19	IfResults	Condición	Condicionante de datos
20	trDataTrxInterface_To_BuildInvoicesZipRq	Transformación	Transformación de esquemas
21	scopeDumpResults	Alcance	Define un nuevo alcance
22	sequenceDumpResults	Secuencia	Secuencia de asignaciones
23	trDataModel_To_BuildInvoicesZipRq	Transformación	Transformación de esquemas
24	invokeWsInvoicesBizToBpel	invocación	Invocación de servicio
25	trGetListTrxNumber	Transformación	Transformación de esquemas
26	IfFoundTrxNbr	Condición	Condicionante de datos
27	assignFileZip	Asignación	Asignación de datos
28	scopeFoundTrxNbr	Alcance	Define un nuevo alcance
29	sequenceFoundTrxNbr	Secuencia	Secuencia de asignaciones
30	trSetTrxNbrToVarGenerateReport	Transformación	Transformación de esquemas
31	assignBatchId	Asignación	Asignación de datos
32	assignNumberLines	Asignación	Asignación de datos
33	trInvoicesToLoadToBpelLoadInvoices	Transformación	Transformación de esquemas
34	invokeBpelLoadInvoices	Invocación	Invocación de proceso
35	replyOutput	Salida	Contiene la respuesta de la operación

#### 3.1.1.2 Pruebas (GetInvoices)

Al hacer una prueba del proceso en general para la carga, actualización y notificación de facturas y notas de crédito es necesario llamar primero al servicio InvoicesEnt.GetInvoices.

Las pruebas se ejecutarán en el ambiente de desarrollo tomando la dirección del siguiente WSDL:

http://ersoacsprod-wls-1:9073/soa-infra/services/default/InvoicesEnt/InvoicesEnt?WSDL

A continuación, se muestran los parámetros de entrada y la respuesta que se presentan en esta operación:

Request: No se necesitan parámetros de entrada para la ejecución de esta operación:

Response: El servicio no requiere algún dato de respuesta por lo que no se recibe nada:

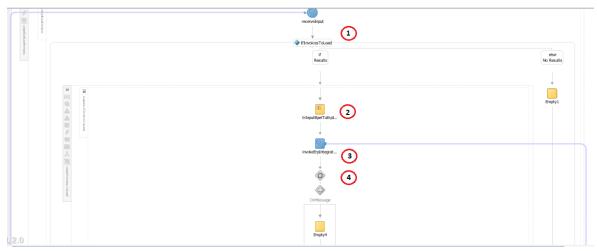
```
→
◎ replyOutput

  [2018/02/07 14:08:28]
 Reply to partner "bpelGetInvoices".
    - <outputVariable>
       - <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="GetInvoicesRs">
         - <GetInvoicesRs xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/
          xmlns:ns2="http://soa.estrellaroja.com.mx/ERPIntegrationBiz/types"
          xmlns:ns1="http://soa.estrellaroja.com.mx/EstrellaRojaCommons" xmlns:plnk="http://docs.oasis-
          open.org/wsbpel/2.0/plnktype" xmlns:ns4="http://soa.estrellaroja.com.mx/FinancialBiz"
          xmlns:ns3="http://schemas.oracle.com/bpel/extension"
xmlns:ns6="http://soa.estrellaroja.com.mx/InvoicesBiz"
          xmlns:ns5="http://soa.estrellaroja.com.mx/StructureTrxInterface"
          xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
          xmlns:client="http://soa.estrellaroja.com.mx/InvoicesEnt/bpelGetInvoices"
          xmlns:tns="http://soa.estrellaroja.com.mx/InvoicesEnt"
          xmlns="http://soa.estrellaroja.com.mx/InvoicesEnt"
            <ns1:Succe
          </GetInvoicesRs>
        </part>
     </outputVariable>
```

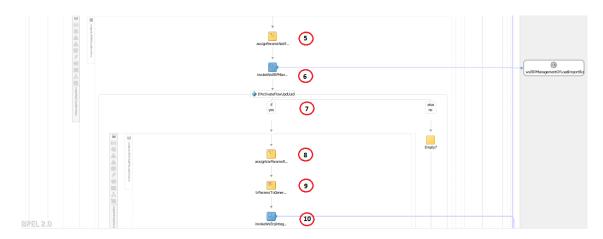
#### 3.1.2 LoadInvoices

**LoadInvoices**: Es la operación que conforma el bpel asíncrono bpelLoadInvoices, este bpel es encargado de hacer la carga de facturas o notas de crédito al ERP-Cloud, se eligió este tipo de servicio ya que el proceso de importación demora el tiempo suficiente para que un bpel síncrono presente problemas de Timeout, problema que no se presenta en este tipo de bpel, los componentes que integran este bpel se detallan a continuación.

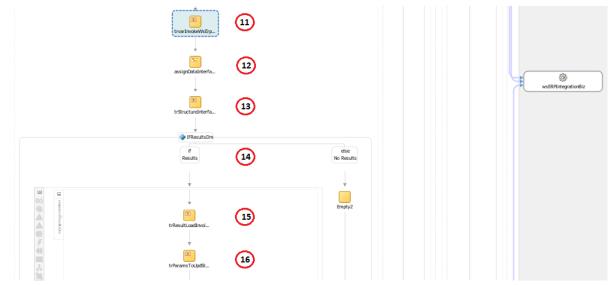
- 1. Se agrega IF para evaluar que los parámetros no sean nulos para poder avanzar con el flujo
- 2. Se agrega transformación (trInputBpelToErpIntegrationBizImportErpCloud) para asignar los valores de los parámetros a la variable request de la invocación del servicio para importar las facturas o notas de crédito al ERP-Cloud.
- Se invoca el servicio "ERPIntegrationBiz"-"ImportToErpCloud", para poder cargar las facturas y notas de crédito al ERP-Cloud.
- 4. Se agrega Pick para poder poner en pausa el flujo hasta la operación de cargar al ERP-Cloud finalice.



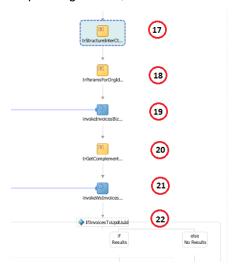
- Se asigna valores a la variable request para la invocación del servicio "ERPManagementOfLoadImportBiz"-"SendReportInterface"
- 6. Se invoca el servicio "ERPManagementOfLoadImportBiz"-"SendReportInterface" para poder recuperar los resultados de carga y notificarlos vía correo electrónico.
- 7. "IF" para evaluar si se activara el flujo de actualización del "UUID" de las facturas y continuar con el flujo.
- 8. Asignación (assignVarParamsRepIntCtrl) para establecer parte de los valores a la variable request para la invocación del servicio para consumir un reporte OTBI
- 9. Transformación para establecer los valores restantes a la variable de request para la invocación del servicio para consumir reporte.
- 10. Se invoca el servicio "ERPIntegrationBiz"-"GenerateReport", para consumir el reporte OTBI "XXER\_FND\_INTERFACE\_CONTROL" con el que se recuperara los resultados de carga junto con información adicional que ayudara actualizar el UUID de las facturas o notas de crédito.



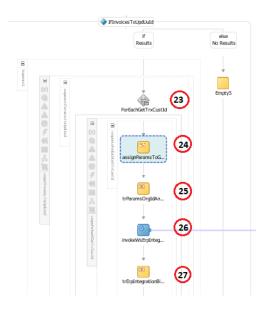
- 11. Se agrega transformación (trvarInvokeWsErpIntegrationBizGenerateReportRsTovarInvokeWsErpIntegrationBizGenerateReportRs) para tomar la información resultante de carga de facturas que se hizo en el paso anterior y poner a los nodos el namespace correspondiente.
- 12. Se agrega asignación (assignDataInterfaceCtrl) para convertir la información resultante del paso anterior a información en formato XML.
- 13. Se agrega transformación (trStructureInterfaceCtrlToStructureInterfaceCtrlResult) para recuperar la información en formato XML y con estructura del reporte "XXER\_FND\_INTERFACE\_CONTROL" para poder hacer uso de la información recuperada, además de agregar funciones a los valores para filtrar información recuperada.
- 14. Se agrega IF para evaluar si hubo registros devueltos por el reporte OTBI consultado anteriormente, si hay resultados se puede continuar con el flujo.
- 15. Se agrega la transformación (trResultLoadInvoicesToparamsToUpdStatusInvoices) para recuperar parte de los valores a establecer para ejecutar el servicio que actualizará el estado de las variables que pudieron importarse al ERP-Cloud y aquella que marcaron error.
- 16. Se agrega transformación (trParamsToUpdStatusInvToWsInvoicesBizUpdStatusInvRq) para asignar valores a la variable request para invocar el servicio que actualizará en BD aquellas facturas o notas de crédito que intentaron cargarse al ERP-Cloud.



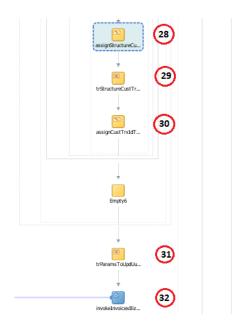
- 17. Se agrega transformación (trStructureInterCtrlToParamsToGetOrgId) para recuperar valores que usaran como parámetros para invocar el servicio que obtendrá complementos CUSOMER\_TRX\_ID.
- 18. Se agrega transformación (trParamsForOrgIdToInvoicesBizGetComplementToUpdateInvoices) para asignar valores a la variable request para invocar el servicio que recuperará la información complementaria (ORGANIZATION ID) para poder ir por el CUSTOMER TRX ID posteriormente.
- 19. Se invoca al servicio "InvoicesBiz"-"GetComplementToUpdInvoices" para recuperar el ORGANIZATION ID y UUID que se usarán para actualizar los registros de facturas o notas de crédito ya cargados en el ERP-Cloud.
- 20. Se agrega transformación (trGetComplementToUpdInvToParamsTrxNumAndOrgId) para recuperar la información resultante de la invocación del servicio anterior, información que servirá para hacer la actualización del UUID de las facturas cargadas en el ERP.
- 21. Se hace la invocación del servicio "InvoicesBiz"-"UpdateStatusInvoices", este servicio actualizará en BD el status que se tubo de intentar cargar las facturas o notas de crédito al ERP-Cloud estableciendo "S" si la carga fue correcta o "E" si es que la carga tubo errores.
- 22. Se agrega IF para evaluar si la invocación del servicio que recupera el complemento (ORGANIZATION\_ID y UUID) para actualizar el UuId, recupero algún dato, si es así se continua con el flujo.



- 23. FOR EACH para iterar la lista de valores que contiene las facturas y sus complementos para recuperar el CUSTOMER TRX ID necesario para la actualización del Uuld.
- 24. Se agrega asignación (assignParamsToGenerateReportCustTrxId) para pasar valores de datos complementarios a variable con estructura similar a la variable que se utilizara como request para la invocación del servicio que consulta el reporte que recupera el CUSTOMER\_TRX\_ID.
- 25. Se agrega transformación (trParamsOrgIdAndTrxNumToGenerateReportCustTrxId) para asignar valores a la variable request que se usará para invocar el servicio que consume reportes del Data Model.
- 26. Se invoca el servicio "ERPIntegrationBiz"-"GenerateReport" para ejecutar el reporte OTBI" XXER AR TRX INTRF CUST TRX ID." El cual recuperar el CUSTOMER TRX ID de cada factura, este valor que se recupera será necesario para poder actualizar el Uuld de las facturas o notas de crédito ya cargadas en el ERP-Cloud.
- 27. Se agrega transformación para tomar los datos recuperados de la invocación del servicio del paso anterior y asignar el namespace correspondiente a los nodos devueltos.



- 28. Se utiliza asignación (assignStructureCustTrxId) para tomar la información resultante del paso anterior y convertirla a formato XML.
- 29. Se usa transformación (trStructureCustTrxIdToStructureCustTrxIdResult) para pasar la información resultante del paso anterior a una variable con la estructura similar a la del reporte XXER\_AR\_TRX\_INTRF\_CUST\_TRX\_ID, esto para que la información pueda ser accesible y usada para los siguientes pasos del flujo.
- 30. Se agrega asignación (assignCustTrxIdToParmsUpdUuId) para pasar la información recuperada en los pasos anteriores a una variable similar a la variable request que se usará para invocar el servicio que actualizará el UuId.
- 31. Se agrega transformación (trParamsToUpdUuldToInvoicesBizUpdateUuldRq) para pasar información a variable request para invocar el servicio para actualizar el Uuld.
- 32. Se invoca el servicio "InvoicesBiz"-"UpdateUuld" para mandar a actualizar el Uuld de aquellas facturas o notas de crédito que pudieron cargarse correctamente en el ERP-Cloud.



# 3.1.2.1 Lista de objetos LoadInvoices

La siguiente tabla contiene los objetos que se utilizaron para la funcionalidad de mandar a cargar las facturas y notas de crédito al ERP-Cloud.

#	Nombre	Tipo	Descripción
1	receiveInput	Entrada	Recibe los parámetros del bpel
2	scopeBpelLoadInvoices	Alcance	Define un nuevo alcance
3	sequenceBpelLoadInvoices	Secuencia	Secuencia de asignaciones
4	IfInvoicesToLoad	Condición	Condicionante de datos
5	scopelfInvoicesToLoad	Alcance	Define un nuevo alcance
6	sequencelfInvoicesToLoad	Secuencia	Secuencia de asignaciones
7	trInputBpelToErpIntegrationBizImportErpCloud	Transformación	Transformación de esquemas
8	invokeErpIntegrationBizImportToErpCloud	Invocación	Invocación de servicio
9	PickWaitLoad	Espera de finalización de proceso	Pausa el proceso hasta que termine el proceso anterior
10	assignParamsNotifyLoad	Asignación	Asignación de datos
11	invokeWsERPManagementOfLoadImportBiz	Invocación	Invocación de servicio
12	IfActivateFlowUpdUuid	Condición	Condicionante de datos
13	assignVarParamsRepIntCtrl	Asignación	Asignación de datos
14	scopeNotifyLoadInvoices	Alcance	Define un nuevo alcance
15	sequenceNotifyLoadInvoices	Secuencia	Secuencia de asignaciones
16	trParamsToGenerateReport	Transformación	Transformación de esquemas
17	invokeWsErpIntegrationBizGenerateReport	Invocación	Invocación de servicio
18	trVarInvokeWsErpIntegrationBizGenerateReportR sToVarInvokeWsErpIntegrationBizGenerateRepo rtRs	Transformación	Transformación de esquemas
19	assignDataInterfaceCtrl	Asignación	Asignación de datos
20	trStructureInterfaceCtrlToStructureInterfaceCtrlRe sult	Transformación	Transformación de esquemas
21	IfResultsDm	Condición	Condición de datos
22	trResultLoadInvoicesToparamsToUpdStatusInvoi ces	Transformación	Transformación de esquemas

23	scopePickWaitForLoadInvoices	Alcance	Define un nuevo alcance
24	sequencePickWaitForLoadInvoices	Secuencia	Secuencia de asignaciones
25	trParamsToUpdStatusInvToWsInvoicesBizUpdSt atusInvRq	Transformación	Transformación de esquemas
26	trStructureInterCtrlToParamsToGetOrgId	Transformación	Transformación de esquemas
27	trParamsForOrgIdToInvoicesBizGetComplement ToUpdateInvoices	Transformación	Transformación de esquemas
28	invokeInvoicesBizGetComplementToUpdInvoices	Invocación	Invocación de servicio
29	trGetComplementToUpdInvToParamsTrxNumAn dOrgld	Transformación	Transformación de esquemas
30	invokeWsInvoicesBizUpdateStatusInvoices	Invocación	Invocación de servicio
31	IfInvoicesToUpdUuld	Condición	Condicionante de datos
32	scopelfResultsDm	Alcance	Define un nuevo alcance
33	sequenceIfResultsDm	Secuencia	Secuencia de asignaciones
34	ForEachGetTrxCustId	Ciclo	Ciclo de registros
35	scopelfInvoicesToUpdUuld	Alcance	Define un nuevo alcance
36	sequencelfInvoicesToUpdUuld	Secuencia	Secuencia de asignaciones
37	assignParamsToGenerateReportCustTrxId	Asignación	Asignación de datos
38	trParamsOrgIdAndTrxNumToGenerateReportCus tTrxId	Transformación	Transformación de esquemas
39	invokeWsErpIntegrationBizGenerateReportCustT rxId	Invocación	Invocación de servicios
40	trErpIntegrationBizGenRepCustTrxIdRsToErpInte grationBizGenRepCustTrxIdRs	Transformación	Transformación de esquemas
41	assignStructureCustTrxId	Asignación	Asignación de datos
42	scopeForEachGetTrxCustId	Alcance	Define un nuevo alcance
43	sequenceForEachGetTrxCustId	Secuencia	Secuencia de asignaciones
44	trStructureCustTrxIdToStructureCustTrxIdResult	Transformación	Transformación de esquemas
45	assignCustTrxIdToParmsUpdUuld	Asignación	Asignación de datos
46	trParamsToUpdUuldToInvoicesBizUpdateUuldRq	Transformación	Transformación de esquemas
47	invokelnvoicesBizUpdateUuid	Invocación	Invocación de servicio
48	callbackClient	Respuesta	Respuesta Asíncrona

#### 3.1.2.2 Pruebas LoadInvoices

Conforme a la prueba general que se hizo, se mostrará a continuación los parámetros de entrada y la respuesta que se presentan en esta operación:

#### Request:

```
[2018/02/07 14:08:28]

Received "LoadInvoices" call from partner "ILoadInvoices"

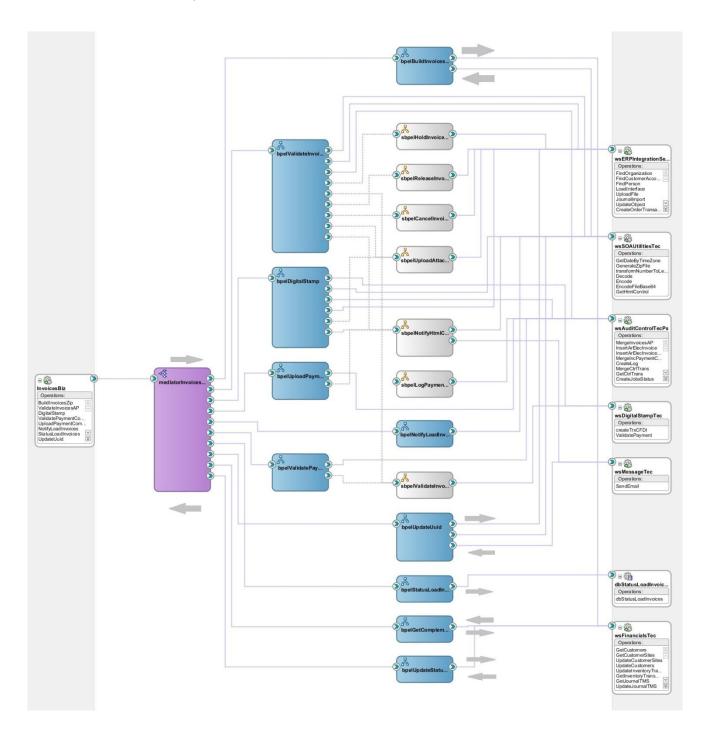
- <inputVariable>
- <inputVariable>
- <inputVariable>
- <inst.coadInvoicesRq xmins:plnke"http://docs.oasis-open.org/wsbpel/2.0/plnktype" xmins:ns4="http://soa.estrellaroja.com.mx/ERPIntegrationBiz/types" xmins:ns5="http://soa.estrellaroja.com.mx/EstrellaRoja.Com.mos" xmins:ns5="http://soa.estrellaroja.com.mx/FinancialBiz" xmins:ns7="http://soa.estrellaroja.com.mx/ERPIntegrationBiz/types" xmins:ns7="http://soa.estrellaroja.com.mx/ERPIntegrationBiz/types" xmins:ns7="http://soa.estrellaroja.com.mx/ERPIntegrationBiz/types" xmins:ns7="http://soa.estrellaroja.com.mx/ERPIntegrationBiz/types" xmins:ns7="http://soa.estrellaroja.com.mx/InvoicesEnt/bpelGetInvoices" xmins:ns912="http://soa.estrellaroja.com.mx/InvoicesEnt/bpelGetInvoices" xmins:ns912="http://schemas.xmisoap.org/wsdl/soap12/" xmins:tins="http://soa.estrellaroja.com.mx/InvoicesEnt/bpelGetInvoices" xmins:ns912="http://soa.estrellaroja.com.mx/InvoicesEnt/bpelGetInvoices" xmins:ns1ententerface>
- <tns:TargetInterface>TrxInterfaceFs</tns:TargetInterface>
- <tns:TargetInterface>TrxInterfaceFs</tns:TargetInterface>
- <tns:TargetInterface>TrxInterfaceFs</tns:TargetInterface>
- <tns:TargetInterface>TrxInterfaceFs</tn>
- <tns:TargetInterface>TrxInterfaceFs</tn>
- <tns:TargetInterfaceFs</tn>
- <tns:TargetInterfaceFs</tn>
- <tns:TargetInterfaceFs</tn>
- <tns:TargetInterfaceFs</tn>
- <tn>
- <tns:TargetInterfaceFs</tn>
- <tns:TargetInterfaceFs</tn>
- <tn>
- <tn>
- <tns:TargetInterfaceFs</tn>
- <tn>
- <t
```

#### Response:

#### ⇒ ⊕ callbackClient

#### 3.2 InvoicesBiz

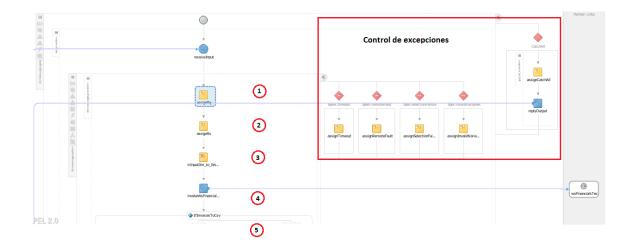
InvoicesBiz es el servicio que contiene la operación de la capa de negocio, esta capa es consumida por la capa Enterprise que en este caso es InvoicesEnt y la capa Biz hace uso de recursos de la capa Tec. En esta capa Biz se encuentran los procesos para ayudar a obtener información de la base de datos con la ayuda de la capa Tec, también en esta capa se usa para ayudar a hacer notificaciones de las cargas y actualizaciones de Facturas y notas de crédito. El siguiente es el compuesto del bpel InvoicesBiz, para el flujo de carga, actualización y notificación de transacciones serán los BPEL que estén incluidos en las flechas.



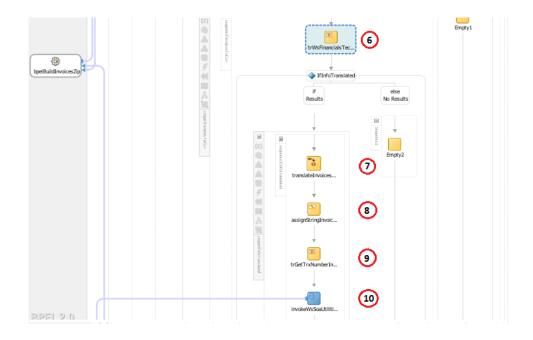
#### 3.2.1 BuildInvoices

Este es la operación que ayuda a construir el archivo Zip que contiene al archivo CSV donde se encuentran los registros de facturas o notas de crédito, para sus procesos se hace uso de la capa Tec. A continuación, se detallan los componentes que incluye este bpel.

- 1. Se agrega asignación para establecer valor de algunas literales comunes para el flujo.
- 2. Se agrega asignación para establecer valor de algunas literales comunes para el flujo.
- Se agrega la transformación (trInputDm\_to\_WsGetInvoicesReq) para asignar los parámetros a una variable y también establecer un valor como parámetro para la invocación del siguiente servicio que ayudará a generar el archivo CSV.
- 4. Se invoca el servicio "FinancialsTec"-"GetReciptOrInvoices" para obtener la colección de datos de la base de datos para construir el archivo CSV.
- 5. Se agrega IF para validar que la variable que almacena los datos consultados a la base de datos, no venga nula, sí es así se podrá continuar con el flujo. En la imagen se puede apreciar también el uso de control de excepciones.



- 6. Se usa la transformación (trWsFinancialsTecGetReceiptOrInvoiceToBpel) para pasar los valores de la variable de salida de la ejecución del servicio anterior.
- 7. Se usa Translate (translateInvoicesToCsvToString) para convertir los datos recuperados en el paso anterior a contenido tipo cadena.
- 8. Se usa asignación (assignStringInvoicesToWsSoaUtils) para establecer valores a la variable request para invocar el siguiente servicio que es para convertir la cadena a contenido base 64.
- Se usa Transformación (trGetTrxNumberInProcess) para recuperar la lista de TRX\_NUMBER que incluirá el archivo CSV.
- 10. Se invoca el servicio "SOAUtilitiesTec"-"Encode" para tomar la cadena que se obtuvo en el paso 7 y convertirlo a contenido Base 64 bits.



- 11. Se agrega transformación (trInvoicesBase64ToEncode64Zip) para tomar el contenido devuelto de la invocación del paso anterior y establecerlo a la variable request para la invocación del próximo servicio.
- 12. Se invoca el servicio "SoaUtiltiesTe"-"EncodeFileBase64" para crear el archivo CSV y codificarlo en base 64.
- 13. Se usa asignación (assignFileBase64) para pasar el contenido recuperado en la invocación de servicio de paso anterior y pasarlo a la variable request para la invocación del siguiente servicio.
- 14. Se invoca el servicio "SoaUtilitiesGenerateFileZip"-"GenerateZipFile" para genera el archivo Zip que contiene al archivo CSV, y también el Zip es convertido en Base 64 bits.
- 15. Se usa asignación (trSetDatToOutput) para pasar los resultados del paso anterior a la variable de respuesta del bpel.



#### Lista de objetos BuildInvoices 3.2.1.1

La siguiente tabla contiene la lista de objetos que se utilizaron para la funcionalidad de la construcción de del archivo CSV que contiene las facturas y notas de crédito y construye también el Zip que comprime al CSV.

#	Nombre	Tipo	Descripción
1	receiveInput	Entrada	Recibe los parámetros del bpel
2	sequenceMain	Secuencia	Secuencia de asignaciones
3	scopeBuildInvoicesZip	Alcance	Define un nuevo alcance
4	sequenceBuildInvoicesZip	Secuencia	Secuencia de asignaciones
5	assignRq	Asignación	Asignación de datos
6	AssignRs	Asignación	Asignación de datos
7	trInputDm_to_WsGetInvoicesReq	Transformación	Transformación de esquemas
8	InvokeWsFinancialsTecGetReciptOrInv oices	Invocación	Invocación de servicio
9	IfInvoicesToCsv	Condición	Condición de datos
10	scopelfInvoicesToCsv	Alcance	Define un nuevo alcance
11	sequencelfInvoicesToCsv	Secuencia	Secuencia de asignaciones
12	trWsFinancialsTecGetReceiptOrInvoice ToBpel	Transformación	Transformación de esquemas
13	IfInfoTranslated	Condición	Condicionante de datos
14	scopelfInfoTranslated	Alcance	Define un nuevo alcance
15	sequencelfInfoTranslated	Secuencia	Secuencia de asignaciones
16	translateInvoicesToCsvToString	Traductor	Traduce el contenido a cadena (en este caso)
18	assignStringInvoicesToWsSoaUtils	Asignación	Asignación de datos
19	trGetTrxNumberInProcess	Transformación	Transformación de esquemas
20	invokeWsSoaUtilitiesTecToBpel	Invocación	Invocación de servicio
21	trInvoicesBase64ToEncode64Zip	Transformación	Transformación de esquemas
22	invokeWsSoaUtiltiesTecEncodeFileBas e64Rq	Invocación	Invocación de servicio
23	assignFileBase64	Asignación	Asignación de datos

24	InvokeWsSoaUtilitiesGenerateFileZip	Invocación	Invocación de servicio
25	trSetDatToOutput	Transformación	Transformación de esquemas
26	replyOutput	Respuesta	Respuesta de bpel síncrono

#### 3.2.1.2 Pruebas BuildInvoices

Con base en la prueba general que se hizo, se presentara a continuación los parámetros de entrada y respuesta de la operación:

#### Request:

```
    receiveInput
```

```
<tns:PARTY_ID>300000003550199</tns:PARTY_ID>
                <tns:ACCOUNT_NUMBER>17002</tns:ACCOUNT_NUMBER>
                <tns:CUST_ACCOUNT_ID>300000003550201</tns:CUST_ACCOUNT_ID>
                <tns:CUST_ACCT_SITE_ID>30000003622742</tns:CUST_ACCT_SITE_ID>
                <tns:PARTY_SITE_NUMBER>20131</tns:PARTY_SITE_NUMBER>
                <tns:SITE_USE_ID>30000003622745</tns:SITE_USE_ID>
                <tns:SITE_USE_CODE>BILL_TO</tns:SITE_USE_CODE>
                <tns:PRIMARY_FLAG>N</tns:PRIMARY_FLAG>
                <tns:LOCATION>27052</tns:LOCATION>
                <tns:LOCATION_ID>300000003622744</tns:LOCATION_ID>
                <tns:PAYMENT TERMS>15 DIAS</tns:PAYMENT TERMS>
              </tns:Q CUSTOMER>
             - <tns:Q_CUSTOMER>
                <tns:RFC>xaxx010101000</tns:RFC>
                <tns:PARTY_NAME>PATITO</tns:PARTY_NAME>
                <tns:PARTY NUMBER>20041</tns:PARTY NUMBER>
                <tns:PARTY ID>300000003550199</tns:PARTY ID>
                <tns:ACCOUNT_NUMBER>17002</tns:ACCOUNT_NUMBER>
                <tns:CUST_ACCOUNT_ID>300000003550201</tns:CUST_ACCOUNT_ID>
                <tns;CUST_ACCT_SITE_ID>30000003622742</tns;CUST_ACCT_SITE_ID>
                <tns:PARTY_SITE_NUMBER>20131/tns:PARTY_SITE_NUMBER>
                <tns:SITE_USE_ID>300000003622746</tns:SITE_USE_ID>
                <tns:SITE_USE_CODE>SHIP_TO</tns:SITE_USE_CODE>
                <tns:PRIMARY FLAG>N</tns:PRIMARY FLAG>
                <tns:LOCATION>27053</tns:LOCATION>
                <tns:LOCATION_ID>300000003622744</tns:LOCATION_ID>
                <tns:PAYMENT_TERMS>15 DIAS</tns:PAYMENT_TERMS>
              </tns:Q_CUSTOMER>
            </tns:DATA_DS>
          </tns:InputDm>
        </BuildInvoicesZipRa>
      </part>
    </inputVariable>
Copy details to clipboard
```

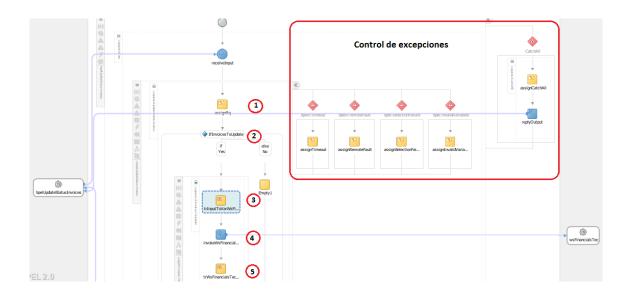
#### Response:

## 3.2.2 UpdateStatusInvoices

UpdateStatusInvoices es otra operación de la capa Biz, esta operación se encarga de modificar el estado de las facturas y notas de crédito, dependiendo del éxito que haya tenido la importación al ERP-Cloud se actualizará en BD el estado, "S" para indicar que la carga fue exitosa y "E" para indicar que se tuvieron errores en la carga. A continuación, se detallan los componentes del bpel bpelUpdateStatusInvoices.

- 1. Se agrega asignación para establecer valor a literal común del bpel.
- 2. Se agrega IF para validar que los parámetros para el bpel no vengan nulos, si no están nulos entonces se puede continuar con el flujo.
- 3. Se usa la transformación (trInputToVarWsFinancialsTecUpdateStatusInvoice)

- 4. Se invoca el servicio "FinancialsTecUpdateStatusInvoices"-"UpdateStatusInvoice" para ejecutar el servicio de la capa Tec que interactúa directamente con la BD y actualiza el estado de las facturas o notas de crédito asignando "S" en caso de que se haya hecho correctamente la importación y "E" en caso de que hayan surgido errores.
- 5. Se usa transformación (trWsFinancialsTecUpdateStatusInvoices) para asignar los resultados de éxito de la actualización a la variable de salida del bpel. El bpel incluye manejador de excepciones, se puede apreciar en el recuadro rojo de la imagen.



#### 3.2.2.1 Lista de objetos UpdateStatusInvoices

La siguiente tabla contiene la lista de objetos que se utilizaron para la funcionalidad de actualizar el estado de las facturas en base de datos con base en el éxito de carga al ERP-Cloud.

#	Nombre	Tipo	Descripción
1	ReceiveInput	Entrada	Recibe los parámetros de entrada
2	scopeUpdateStatusInvoices	Alcance	Define un nuevo alcance
3	seqenceUpdateStatusInvoices	Secuencia	Secuencia de asignaciones
4	AssignRq	Asignación	Asignación de datos
5	IfInvoicesToUpdate	Condición	Condicionante de datos
6	scopelfInvoicesToUpdate	Alcance	Define un nuevo alcance
7	sequencelfInvoicesToUpdate	Secuencia	Secuencia de asignaciones
8	TrInputToVarWsFinancialsTecUpdateStatusInvoic e	Transformación	Transformación de esquemas

9	invokeWsFinancialsTecUpdateStatusInvoices	Invocación	Invocación de servicio
10	trWsFinancialsTecUpdateStatusInvoices	Transformación	Transformación de esquemas
11	assignRs	Asignación	Asignación de datos
12	replyOutput	Salida	Respuesta de bpel síncrono

#### 3.2.2.2 Pruebas UpdateStatusInvoices

Con base en la prueba general que se hizo, se presentara a continuación los parámetros de entrada y respuesta de la operación:

#### Request:

```
[2018/02/07 14:09:41]

Received "UpdateStatusInvoices" call from partner "bpelUpdateStatusInvoices"

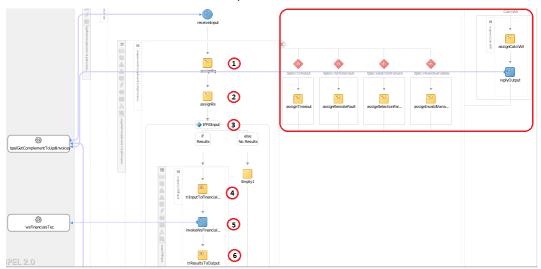
- <inputVariable>
- <-part xmlnsxissi="http://www.w3.org/2001/XMLSchema-instance" name="UpdateStatusInvoicesRq">
- 
            - < UpdateStatusInvoicesRq xmlnsxosp12="http://schemas.xmlsoap.org/wsdl/soap12/" xmlnsxiss="http://soa.estrellaroja.com.mx/ERPIntegrationBiz* xmlnsxips-lphk="http://docs.oasis-open.org/wsbpel/2.0/plnktype" xmlnsxins1="http://soa.estrellaroja.com.mx/ERPIntegrationServiceTeC" xmlnsxnsd="http://soa.estrellaroja.com.mx/ERPIntegrationServiceTeC" xmlnsxisd="http://soa.estrellaroja.com.mx/ERPIntegrationServiceTeC" xmlnsxisd="http://soa.estrellaroja.com.mx/InvoicesEnt/bpelLoadInvoices" xmlnsxisd="http://soa.estrellaroja.com.mx/InvoicesEnt/bpelLoadInvoices" xmlnsxis="http://soa.estrellaroja.com.mx/InvoicesEnt/bpelLoadInvoices" xmlnsxis="http://soa.estrellaroja.com.mx/InvoicesEnt/bpelLoadInvoicesTedt/bpelLoadInvoicesTedt/bpelLoadInvoicesTedt/bp
```

#### Response:

#### 3.2.3 GetComplementToUpdateInvoices

GetComplementToUpdateInvoices es otra operación del servicio InvoicesBiz, esta operación se encarga de recuperar campos necesarios para actualizar el Uuld de las facturas que pudieron cargarse correctamente al ERP-Cloud, los campos a recuperar es el ORGANIZATION\_ID y el UUID. A continuación, se detallan los componentes del bpel.

- 1. Se asignan valores a literales comunes del bpel.
- 2. Se asignan valores a literales comunes del bpel.
- 3. Se usa IF para validar que los parámetros del bpel no vengan nulos, si no son nulos entonces se puede continuar con el flujo.
- 4. Se usa transformación (trInputToFinancialsTecGetComplementInvoices) para pasar los parámetros del bpel a variable request para invocar el servicio del siguiente paso.
- 5. Se invoca el servicio "FinancialsTec"-"GetComplementInvoices", para interactuar con la BD y obtener la información complementaria (ORGANIZATION\_ID y UUID) para actualizar el Uuld de las facturas y notas de crédito que pudieron cargarse al ERP-Cloud.
- 6. Se usa transformación (trResultsToOutput) para pasar los datos de respuesta de la invocación del servicio del paso anterior a la variable de salida del bpel.



## 3.2.3.1 Lista de objetos GetComplementInvoice

La siguiente tabla muestra la lista de objetos que fueron utilizados para la funcionalidad de la recuperación de información complementaria para la actualización del Uuld de las facturas y notas de crédito en el ERP-Cloud.

#	Nombre	Tipo	Descripción
1	receiveInput	Entrada	Recibe los parámetros del bpel
2	scopeGetComplementToUpdInvoices	Alcance	Define un nuevo alcance

3	sequenceGetComplementToUpdInvoices	Secuencia	Secuencia de asignaciones
4	assignRq	Asignación	Asignación de datos
5	assignRs	Asignación	Asignación de datos
6	IfFillInput	Condición	Condicionante de datos
7	trInputToFinancialsTecGetComplementInvoices	Transformación	Transformación de esquemas
8	scopelfFillInput	Alcance	Define un nuevo alcance
9	SequencelfFillInput	Secuencia	Secuencia de asignaciones
10	invokeWsFinancialsTecGetComplementInvoices	Invocación	Invocación de servicio
11	trResultsToOutput	Transformación	Transformación de esquemas
12	replyOutput	Respuesta	Respuesta de bpel síncrono

#### 3.2.3.2 Pruebas GetComplementInvoices

Con base en la prueba general que se hizo, se presentara a continuación los parámetros de entrada y respuesta de la operación:

#### Request:

```
receiveInput
[2018/02/07 14:09:41]
Received "GetComplementToUpdInvoices" call from partner "bpelGetComplementToUpdInvoices"
   - <inputVariable>
     - <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      name="GetComplementToUpdInvoicesRq"
       - < GetComplementToUpdInvoicesRq xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
        xmlns:ns2="http://soa.estrellaroja.com.mx/ERPIntegrationBiz" xmlns:plnk="http://docs.oasis-open.org/wsbpel/2.0/plnktype" xmlns:ns1="http://soa.estrellaroja.com.mx/InvoicesEnt"
         xmlns:ns4="http://soa.estrellaroja.com.mx/ERPIntegrationServiceTec"
         xmlns:ns3="http://soa.estrellaroja.com.mx/EstrellaRojaCommons'
         xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
         xmlns:client="http://soa.estrellaroja.com.mx/InvoicesEnt/bpelLoadInvoices"
         xmlns:tns="http://soa.estrellaroja.com.mx/InvoicesBiz"
         xmlns="http://soa.estrellaroja.com.mx/InvoicesBiz"
         xmlns:wsa="http://www.w3.org/2005/08/addressing">
          - <tns:GetComplementInvoiceList
             <tns:TRX_NUMBER>ERJGR31120185BA</tns:TRX_NUMBER>
              <tns:STATUS>s</tns:STATUS>
           </tns:GetComplementInvoiceList>
          - <tns:GetComplementInvoiceList>
              <tns:TRX_NUMBER>ERJGR31120185BA</tns:TRX_NUMBER>
              <tns:STATUS>s</tns:STATUS>
           </tns:GetComplementInvoiceList>
         </GetComplementToUpdInvoicesRq>
       </part>
    </inputVariable>
Copy details to clipboard
```

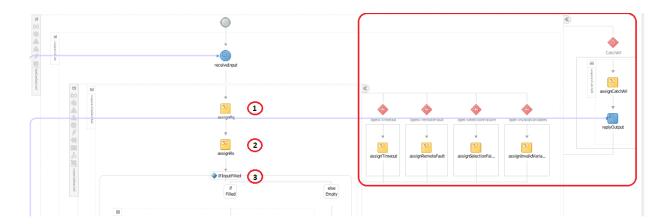
#### Response:

```
🐗 replyOutput
     [2018/02/07 14:09:41]
   Reply to partner "bpelGetComplementToUpdInvoices".
                    - <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="GetComplementToUpdInvoicesRs">
                                part xmins:xsi= nttp://www.ws.org/.co.r/xrits.tienia-instance inside decomposition of control of the control of
                               xmins:nsz="http://soa.estreliaroja.com.mx/Financialsi etc/types
xmins:nsl="http://soa.estreliaroja.com.mx/EstreliaRojaCommons" xmins:plnk="http://docs.oasis-
open.org/wsbpel/2.0/plnktype" xmins:ns4="http://soa.estreliaroja.com.mx/ERPIntegrationServiceTec"
xmins:ns3="http://soa.estreliaroja.com/bpel/extension" xmins:wsdl="http://schemas.xmisoap.org/wsdl/"
xmins:client="http://soa.estreliaroja.com.mx/InvoicesBiz/bpelGetComplementToUpdInvoices"
                                 xmlns:tns="http://soa.estrellaroja.com.mx/InvoicesBiz" xmlns="http://soa.estrellaroja.com.mx/InvoicesBiz" >
                                          - <tns:GetComplementInvoiceList>
                                                        <tns:TRX_NUMBER>ERJGR31120185BA</tns:TRX_NUMBER>
                                                         <tns:ORG_ID>30000001049792</tns:ORG_ID
                                                        <tns:UUID>194EB89E-6690-5118-879D-F1608FFC8061/tns:UUID>
                                                 </tns:GetComplementInvoiceList>
                                          <tns:ORG_ID>30000001049792</tns:ORG_ID
                                                        <tns:UUID>194E889E-6690-5118-879D-F1608FFC8061</tns:UUID>
                                                <tns:P_ERROR/>
                                               <tns:P ERROR DESC/>
                                         </tns:Return>
                                 </GetComplementToUpdInvoicesRs>
                          </part>
    Copy details to clipboard
```

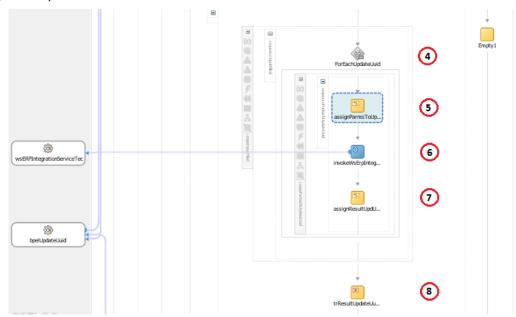
#### 3.2.4 UpdateUuld

UpdateUuld es otra operación del servicio InvoicesBiz, que se utiliza para el flujo de la solución, esta operación se encarga de actualizar el Uuld de las facturas y notas de crédito que pudieron cargarse en el ERP-Cloud y de notificar el éxito de la actualización vía correo electrónico. A continuación, se detallan los componentes del bpel.

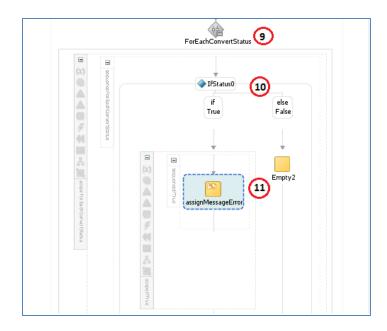
- 1. Se asignan valores a literales comunes del bpel.
- Se asignan valores a literales comunes del bpel.
- 3. Se agrega IF para validar que los parámetros del BPEL no vengan nulos, si no están nulos se puede continuar con el flujo. Se agrega manejador de excepciones al alcance principal.



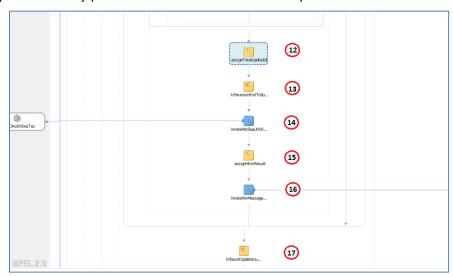
- 4. FOR EACH para iterar la lista de parámetro de facturas a actualizar.
- 5. Se usa asignación (assignParmsToUpdateObjectRq) para establecer valores a variable request para invocar servicio del siguiente paso.
- 6. Se invoca servicio "ErpIntegrationServiceTec"-"UpdateObject", para mandar a actualizar el Uuld la factura o nota de crédito.
- 7. Asignación (assignResultUpdUuIdToListResults) para recuperar en una variable los resultados de actualización del UuId.
- 8. Se agrega transformación (trResultUpdateUuldToResultUpdateUuldResult) para tomar los datos de paso anterior y pasarlo a una variable de similar estructura y se transforman algunos campos que se utilizaran en los siguientes pasos.



- 9. FOR EACH para iterar la colección de datos que se recuperó en el paso anterior, para posteriormente agregar funcionalidad de identificar los mensajes de respuesta del servicio anterior y determinar si se trata de un error o de un éxito.
- 10. IF para evaluar la respuesta del servicio del paso 6, si es "0" entonces se trata de un error si es "1" se trata de un éxito, si es "0" se entra al alcance del IF para hacer conteo de los errores.
- 11. Asignación (assignMessageError) para agregar mensaje de error a variable de lista de errores y agrega una unidad de error a la variable que hace el conteo de los errores de actualización de Uuld.



- 12. Asignación (assignTotalUpdUuld) para asignar a variable de tipo lista, los mensajes de error que llegaron a presentarse en los pasos anteriores y asigna también el valor de la variable del conteo de errores de actualización de Uuld.
- 13. Se usa transformación (trParamsHtmlToSoaUtilitesTecGetHtmlControl) para asignar los valores recuperados en el paso anterior y establecerlos a la variable request para la invocación del servicio del próximo paso.
- 14. Se invoca el servicio "SoaUtilitiesTec"-"GetHtmlControl", para obtener el los resultados de actualización en etiquetas HTML, contenido que se mandará en correo electrónico para notificar sobre la actualización.
- 15. Asignación (assignHtmlResult) para recuperar los datos devueltos por el servicio del paso anterior y establecerlos a la variable request para la invocación del servicio del siguiente paso.
- 16. Se invoca el servicio "MessageTec"-"SendMail", para notificar vía correo electrónico, los resultados de la actualización del Uuld
- 17. Transformación (trResultUpdateUuldToOutput) para recuperar los mensajes de resultado de la ejecución del servicio del paso anterior y pasarlo a la variable de salida del bpel.



#### Lista de objetos UpdateUuld 3.2.4.1

La siguiente tabla muestra la lista de objetos que se ocuparon para la funcionalidad de actualizar el Uuld de las facturas y notas de crédito que se cargaron al ERP-Cloud.

#	Nombre	Tipo	Descripción
1	ReceiveInput	Entrada	Recibe los parámetros de entrada del bpel
2	scopeUpdateUuid	Alcance	Define un nuevo alcance
3	sequenceUpdateUuid	secuencia	Secuencia de asignaciones
4	assignRq	Asignación	Asignación de datos
5	assignRs	Asignación	Asignación de datos
6	IfInputFilled	Condición	Condición de datos
7	scopeInputFilled	Alcance	Define un nuevo alcance
8	sequenceInputFilled	Secuencia	Secuencia de asignaciones
9	ForEachUpdateUuid	Ciclo	Ciclo de registros
10	scopeForEachUpdateUuid	Alcance	Define un nuevo alcance
11	sequenceForEachUpdateUuid	Secuencia	Secuencia de asignaciones
12	assignParmsToUpdateObjectRq	Asignación	Asignación de datos
13	invokeWsErpIntegrationService TecUpdateObject	Invocación	Invocación de servicio
14	assignResultUpdUuldToListRes ults	Asignación	Asignación de datos
15	trResultUpdateUuldToResultUp dateUuldResult	Transformación	Transformación de esquemas
16	ForEachConvertStatus	Ciclo	Ciclo de registros
17	scopeForEachConvertStatus	Alcance	Define un nuevo alcance
18	sequenceForEachConvertStatus	Secuencia	Secuencia de asignaciones
19	assignMessageError	Asignación	Asignación de datos
20	assignTotalUpdUuld	Asignación	Asignación de datos
21	trParamsHtmlToSoaUtilitesTecG etHtmlControl	Transformación	Transformación de esquemas
22	invokeWsSoaUtilitiesTecGetHtm IControl	Invocación	Invocación de servicio

File Ref: TA.070-4 ER01\_AR\_Creacion\_de\_Transacciones.docx

23	assignHtmlResult	Asignación	Asignación de datos
24	invokeWsMessageTecSendMail	Invocación	Invocación de servicio
25	trResultUpdateUuIdToOutput	Transformación	Transformación de esquemas
26	replyOut	Respuesta	Respuesta síncrona de bpel

#### 3.2.4.2 Pruebas UpdateUuld

Con base en la prueba general que se hizo, se presentara a continuación los parámetros de entrada y respuesta de la operación:

#### Request:

```
neceiveInput
 [2018/02/07 14:09:45]
Received "UpdateUuid" call from partner "bpelUpdateUuid"
   - <inputVariable>
     - <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="UpdateUuidRq">
        - < UpdateUuidRq xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/
         xmlns:ns2="http://soa.estrellaroja.com.mx/ERPIntegrationBiz" xmlns:plnk="http://docs.oasis-
         open.org/wsbpel/2.0/plnktype" xmlns:ns1="http://soa.estrellaroja.com.mx/InvoicesEnt" xmlns:ns4="http://soa.estrellaroja.com.mx/ERPIntegrationServiceTec"
         xmlns:ns3="http://soa.estrellaroja.com.mx/EstrellaRojaCommons'
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
         xmlns:client="http://soa.estrellaroja.com.mx/InvoicesEnt/bpelLoadInvoices"
         xmlns:tns="http://soa.estrellaroja.com.mx/InvoicesBiz"
         xmlns="http://soa.estrellaroja.com.mx/InvoicesBiz"
         xmlns:wsa="http://www.w3.org/2005/08/addressing">
          - <tns:ListUpdateUuidRq>
              <tns:OperationMode>single</tns:OperationMode>
            - <tns:Object>
                <ns4:EntityName>Receivables Invoice</ns4:EntityName>
                <ns4:ContextValue>Contabilidad Electronica (MX)
                <ns4:UserKeyA>ERJGR31120185BA</ns4:UserKeyA>
                <ns4:UserKeyB>#NULL</ns4:UserKeyB>
                <ns4:UserKeyC>#NULL</ns4:UserKeyC>
                <ns4:UserKeyD>29002</ns4:UserKeyD>
                <ns4:UserKeyE>#NULL</ns4:UserKeyE>
                <ns4:DFFAttributes>{"ATTRIBUTE3":"194EB89E-6690-5118-879D-F1608FFC8061"}</ns4:DFFAttributes>
              </tns:Object>
              <tns:NotificationCode>10</tns:NotificationCode>
           </tns:ListUpdateUuidRq>
          - <tns:ListUpdateUuidRq>
              <tns:OperationMode>single</tns:OperationMode>
            - <tns:Object>
                <ns4:EntityName>Receivables Invoice</ns4:EntityName>
                <ns4:ContextValue>Contabilidad Electronica (MX)</ns4:ContextValue>
                <ns4:UserKeyA>ERJGR31120185BA</ns4:UserKeyA>
                <ns4:UserKeyB>#NULL</ns4:UserKeyB>
                <ns4:UserKeyC>#NULL</ns4:UserKeyC>
                <ns4:UserKeyD>29002</ns4:UserKeyD>
                <ns4:UserKeyE>#NULL</ns4:UserKeyE>
                <ns4:DFFAttributes>{"ATTRIBUTE3":"194EB89E-6690-5118-879D-F1608FFC8061"}</ns4:DFFAttributes>
              </tns:Object>
              <tns:NotificationCode>10</tns:NotificationCode>
            </tns:ListUpdateUuidRq>
         </UpdateUuidRq>
       </part>
     </inputVariable>
```

#### Response:

```
🧠 replyOutput
 [2018/02/07 14:09:46]
 Reply to partner "bpelUpdateUuid".
   - <outputVariable>
      - <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="UpdateUuidRs">
        - <UpdateUuidRs xmlns:ns2="http://schemas.oracle.com/bpel/extension"</p>
         xmlns:plnk="http://docs.oasis-open.org/wsbpel/2.0/plnktype"
         xmlns:ns1="http://soa.estrellaroja.com.mx/EstrellaRojaCommons"
         xmlns:ns4="http://soa.estrellaroja.com.mx/SOAUtilitiesTec"
         xmlns:ns3="http://soa.estrellaroja.com.mx/ERPIntegrationServiceTec"
         xmlns:ns5="http://soa.estrellaroja.com.mx/MessageTec"
         xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
         xmlns:client="http://soa.estrellaroja.com.mx/InvoicesBiz/bpelUpdateUuid"
         xmlns:ns0="http://soa.estrellaroja.com.mx/InvoicesBiz"
         xmlns="http://soa.estrellaroja.com.mx/InvoicesBiz">
          - <ns0:Return>
              <ns0:TrxNumber>ERJGR31120185BA</ns0:TrxNumber>
             <ns0:UuId>194EB89E-6690-5118-879D-F1608FFC8061</ns0:UuId>
             <ns0:Status>1</ns0:Status>
           </ns0:Return>
          - <ns0:Return>
             <ns0:TrxNumber>ERJGR31120185BA</ns0:TrxNumber>
              <ns0:UuId>194EB89E-6690-5118-879D-F1608FFC8061</ns0:UuId>
             <ns0:Status>1</ns0:Status>
           </ns0:Return>
         </UpdateUuidRs>
       </part>
     </outputVariable>
 Copy details to clipboard
```

# 3.3 Resultados de prueba de notificaciones de carga y actualización de transacciones

Notificación de Proceso de importación exitoso:



#### Proceso de Actualización exitosa:



### 4 Diseño de Datos

## 4.1 Origen de Datos

Tabla	Columna	Comentarios	
XXER_TBL_PUENTE_HEADER	RFC_DESTINO	RFC del cliente a agregar Transacción	
	FECHA_MOVIMIENTO	Fecha de transacción	
	NUMERO_DOCUMENTO	Numero de transacción	
	ADICIONAL2	Campo utilizado para almacenar el BATCH_ID	
	TABLA_PUENTE_ID	Clave Primaria de la tabla	
XXER_CNF_INTERFASE_INGRESOS_A	UNIDAD_NEGOCIO_ORIGEN	Unidad de negocio	
	TIPO_TRANSACCION_AR	Tipo de transacción	
	MEMO_LINE	Cadena MEMOLINE	
	ORGANIZATION_ID	Numero de organización	
XXER_TBL_PUENTE_LINES	LINEA_ARTICULO	Línea	
	SUBTOTAL	Subtotal de monto de transacción	
	TABLA_PUENTE_LINEA_ID	Clave primaria de la tabla	
XXER_TBL_PUENTE_AUX	UUID	CDFI de la transacción	

# 4.2 Lógica de Validación

Las principales validaciones de la integración de transacciones son las siguientes

- El proceso solo toma en cuenta a aquellas facturas que no tengan el status "S" (Success) que significa éxito de carga, pudiendo tomar a aquellas facturas o notas de crédito que tengan el status "E" (Error) para reintentar cargarlas o bien status en nulo para procesar por primera vez la carga.
- Ya que se utilizan reportes del OTBI de Cloud, aquellos RFC a los que no se les encuentre cliente cargado en la nube, no se considerará para la carga de facturas o notas de crédito
- La notificación de carga de facturas y notas de crédito se hace por línea cargada.
- Aquellas facturas o notas de crédito que hayan logrado cargarse al ERP-Cloud pero no tenga ORGANIZATION\_ID válido, no se les podrá actualizar su Uuld.
- No se tomará en cuenta la actualización de Uuld de aquellas facturas o notas de crédito que hayan presentado algún error de carga.
- La actualización de status de facturas y notas de crédito se hará ya sea por factura o por Nota de crédito, no por línea.

## 5 Diseño SQL

En el diseño SQL se abarca principalmente las consultas utilizadas para la extracción de la información.

### 5.1 Sentencias SQL

Los Data Models que se utilizan se ubican en las siguientes rutas:

/Custom/ER\_INTEGRATION/ER01\_AR\_Creacion\_de\_transacciones/Data Models/

- 1. XXER\_AR\_TRX\_INTERFACE
  - a. STATUS\_LOAD\_INVOICES
  - b. Q\_QUSTOMER
- 2. XXER\_AR\_TRX\_INTRF\_CUST\_TRX\_ID
  - a. GET\_CUSTOMER\_TRX\_ID

#### /Custom/ER\_INTEGRATION/Commons/Data Models/

- 1. XXER\_FND\_INTERFACE\_CONTROL
  - a. Q\_DATA

#### Data Set: STATUS\_LOAD\_INVOICES

```
SELECT STATUS ,
 TRX_NUMBER ,
 REQUEST ID ,
 MESSAGE_TEXT ,
  INVALID VALUE ,
  TRX_DATE_ERROR
FROM
  (SELECT 'LOADED' STATUS ,
    CUST.TRX_NUMBER ,
   CUST.REQUEST ID ,
   NULL MESSAGE TEXT ,
   NULL INVALID VALUE ,
   NULL TRX DATE ERROR
  FROM RA_CUSTOMER_TRX_ALL CUST
  WHERE 1 = 1
  UNION
  SELECT 'ERROR' STATUS ,
    RAI.TRX NUMBER ,
```

```
RAI.REQUEST_ID ,

RAIE.MESSAGE_TEXT ,

RAIE. INVALID_VALUE ,

RAI. TRX_DATE TRX_DATE_ERROR

FROM RA_INTERFACE_LINES_ALL RAI ,

RA_INTERFACE_ERRORS_ALL RAIE

WHERE RAI.INTERFACE_LINE_ID = RAIE.INTERFACE_LINE_ID (+)

)

WHERE 1 = 1

AND TRX_NUMBER IN (:P_TRX_NUMBER)

ORDER BY REQUEST_ID DESC
```

### Data Set: Q\_QUSTOMER

```
SELECT hp.jgzz fiscal code rfc,
 hp.party name,
 hp.party number,
 hp.party id,
 hca.account number,
 hca.cust account id,
 hcas.cust acct site id,
 hps.party site number,
 hcsu.site use id,
  hcsu.site use code,
 hcsu.primary flag,
 hcsu.location,
 hl.location id,
  tb.name payment terms
FROM hz cust accounts hca,
 hz parties hp,
 hz cust acct sites all hcas,
 hz_cust_site_uses_all hcsu,
 hz_party_sites hps,
 hz_locations hl,
 hz customer profiles f pf,
 ra terms tl tb
WHERE 1
AND hca.party id
                          = hp.party id
AND hca.cust account id
                          = hcas.cust account id
```

```
AND hcsu.cust_acct_site_id = hcas.cust_acct_site_id

AND hps.party_site_id = hcas.party_site_id

AND hps.location_id = hl.location_id

AND pf.cust_account_id = hca.cust_account_id

AND tb.term_id = pf.standard_terms

AND tb.language = userenv('LANG')

AND hp.jgzz_fiscal_code IN (:p_rfc)
```

#### Data Set: GET\_CUSTOMER\_TRX\_ID

```
SELECT CUSTOMER_TRX_ID ,

TRX_NUMBER ,

ATTRIBUTE3 ,

ORG_ID

FROM RA_CUSTOMER_TRX_ALL

WHERE 1 = 1

AND TRX_NUMBER = :P_TRX_NUMBER

AND ORG_ID = :P_ORG_ID
```

#### Data Set: Q DATA

```
SELECT
   "Status",
   "InterfaceType",
   "BatchId",
   "HeaderId"
   "LineId",
   "ErrorMsg",
   "DataValue",
   "LoadRequestId",
   "RequestId"
FROM
   /**********
   -- Error Dummy
   ****************
       SELECT
           'E' AS "Status",
           'DummyTemplate' AS "InterfaceType",
           '54682' AS "BatchId",
           '98461' AS "HeaderId",
           '487596' AS "LineId",
           'Error: Dummy Error' AS "ErrorMsg",
           'Dat Value Error Dummy' AS "DataValue",
           95623 AS "LoadRequestId" -- Job Id Interfaces
           0 AS "RequestId"
```

```
FROM
       dual
   UNION ALL
/***********
-- Messages Error GL Interfaces
******** SELECT
       'E' AS "Status",
       'JournalImportTemplate' AS "InterfaceType",
       TO CHAR(gi.group id) AS "BatchId",
       TO CHAR(nvl(gi.je header id,0) ) AS "HeaderId",
       TO_CHAR(nvl(gi.je_line_num,0) ) AS "LineId",
       ( gi.status
        | | ' = '
        || gi.status description ) AS "ErrorMsg",
       ( 'USER JE SOURCE NAME: '
        || gi.user je source name
       || '|ACC DATE: '
        || gi.accounting date ) AS "DataValue",
       gi.load request id AS "LoadRequestId" -- Job Id Interfaces
       gi.request id AS "RequestId"
   FROM
       gl interface gi
   UNION ALL
/**********
-- Messages Error AR Interfaces Auto Invoices
```

Como la consulta es algo extensa se agrega un archivo adjunto en donde está la consulta completa:



Consulta Completa.sql

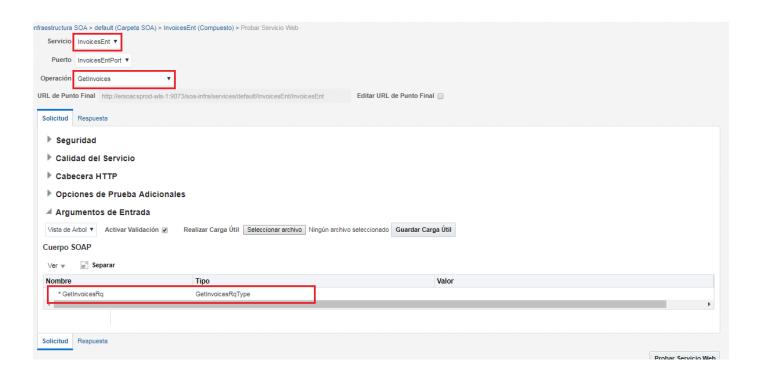
## 6 Reglas de Negocio

Las reglas de negocio para esta integración son las siguientes:

- Validar el estado de las facturas por cargar, ya que solo se cargarán aquellas que no se hayan cargado correctamente antes.
- Validar que el cliente con el que se tiene referencia en BD On Premise, exista en la base de Cloud, ya que si no se encuentra no se podrán derivar datos y no podrá cargarse la información.
- No se podrán cargar Facturas o notas de crédito repetidas.
- Es necesario importar primero la factura o nota de crédito para poder asignar el Uuld o CDFI, ya que si se intenta cargar desde la primera importación esto no será posible

### 6.1 Diseño del Servicio

Para poder iniciar la integración es necesario ejecutar el compuesto **InvoicesEnt – GetInvoices**; para esta ejecución no se requieren parámetros.



## 7 Consideraciones de Rendimiento

Este requerimiento ha sido probado con un conjunto de facturas y notas de crédito pequeñas, tener contemplado que el compuesto puede presentar demora con una cantidad de registros muy grande a procesar.

# 7.1 Estrategia de Reinicio

- Se pude reiniciar la integración, siempre y cuando no existan instancias de SOA corriendo, pues de lo contrario serán abortadas.
- En caso de existir instancias en ejecución, esperar hasta su terminación.

# 7.2 Seguridad

N/A

## 7.3 Personalización

- En caso de requerir modificar el proyecto, tomar la versión más reciente del controlador de versiones con el que se cuente.
- Revisar que los servicio comunes estén desplegados.
- La versión del IDE de desarrollo de JDeveloper con la que se implementó dicha solución es JDEVADF\_12.2.1.2.0\_GENERIC\_161008.1648.S

# 7.4 Catálogo de Errores

El código de error que se maneja para esta integración se especifica en el documento "CatalogosErroresEstrellaRoja.xlsx".

### 8 Consideraciones de Instalación

Como Prerrequisitos se requiere un esquema de base de datos (DBCS) previamente configurado, con los orígenes de datos correspondientes, para más detalle consultar el manual de configuración del ambiente.

En este esquema deben existir las siguientes tablas:

- XXER TBL PUENTE HEADER
- XXER\_TBL\_PUENTE\_LINES
- XXER\_TBL\_PUENTE\_AUX
- XXER\_CNF\_INTERFASE\_INGRESOS\_AR

En este esquema debe existir el siguiente paquete:

XXER\_UTILITIES\_PKG

En este esquema deben existir los tipos:

- XXER AR GETDATA INVOICES TB TYP
- XXER\_AR\_GETDATA\_INVOICES\_TYP
- XXER\_AR\_PARAMS\_UPD\_INV\_TYPE
- XXER\_AR\_PARAMS\_UPD\_INV\_TTYPE
- XXER AR RECEIPT INVOICE TTYPE
- XXER\_AR\_RECEIPT\_INVOICE\_TYPE
- XXER\_COMPLEMENT\_UPD\_UUID\_TTYPE
- XXER\_COMPLEMENT\_UPD\_UUID\_TYPE
- XXER ERROR MESSAGES INV TTYPE
- XXER\_ERROR\_MESSAGES\_INV\_TYPE
- XXER\_INVOICE\_LOAD\_STATUS\_TTYPE
- XXER INVOICE LOAD STATUS TYPE
- XXER INVOICES TO UPDATE TTYPE
- XXER\_INVOICES\_TO\_UPDATE\_TYPE

Deben de existir las siguientes vistas:

- XXER\_AR\_INVOICES\_V
- XXER\_AR\_CUSTOMER\_RFC\_V

En caso de no contar con los objetos de base de datos mencionados en los puntos anteriores, se podrá tomar el siguiente archivo para poder crearlos:



Objetos de BD creación de transacciones.sql

#### Instalación de integración creación de transacciones

Para poder instalar los servicios que componen la solución "Creación de Transacciones" es necesario contar con los siguientes instalables:

sca\_InvoicesEnt.jar sca\_InvoicesBiz.jar  $sca\_ERPM an agement Of Load Import Biz. jar$ 

EL orden no es requerido, es necesario revisar que los servicios ajenos e los que se hace uso, estén instalados correctamente.

#### URL de Acceso y seguridad 9

En la siguiente tabla se define las URL que se utilizan en esta integración:

Servicio	URL	
InvoicesEnt	http:// <host>:<puerto>/soa-infra/services/default/InvoicesEnt/InvoicesEnt?WSDL</puerto></host>	
InvoicesBiz	http:// <host>:<puerto>/soa-infra/services/default/InvoicesBiz/InvoicesBiz?WSDL</puerto></host>	
ERPManagementOfLoadImpo rtBiz	http:// <host>:<puerto>/soa- infra/services/default/ERPManagementOfLoadImportBiz/ERPManagementOfLoadImp ortBiz?WSDL</puerto></host>	

### Temas abiertos y cerrados 10

# 10.1 Temas Abiertos

ID	Tema	Solución	Responsabilidad	Fecha Objetivo	Fecha impacto

## 10.2 Temas Cerrados

ID	Tema	Solución	Responsabilidad	Fecha Objetivo	Fecha impacto