

OUM

TA.070-4 DISEÑO TÉCNICO DE EXTENSIONES E INTERFACES



ER01_AR Creación de Transacciones con 4% de retención, Recibos y Notas de Crédito de TRATECSA, DLIVER y PAQUER

Autor:	In-Tek
Fecha de creación:	julio 1, 2019
Última actualización:	agosto 2, 2019
Código de referencia OUM:	DS.140 Design Specification
Versión:	1.0

Aprobadores:

Edith Ramírez

1. Control de Documento

1.1 Bitácora de Cambios

Fecha	Autor	Versión	Referencia del cambio
1-jul-2019	Arturo Sanchez	1.0	No Previous Document

1.2 Revisores

Nombre	Posición

Contenido

1.	Control de Documento.....	2
1.1	Bitácora de Cambios	2
1.2	Revisores	2
2.	Resumen técnico.....	6
2.1	Programación de ejecuciones	6
2.1.1	PROGRAMACION FACTURAS Y NOTAS DE CREDITO	6
2.1.2	PROGRAMACION APLICACIÓN NOTAS DE CREDITO	7
2.1.3	PROGRAMACION RECIBOS.....	8
3.	Diagrama de la integración.....	10
3.1	DIAGRAMA FACTURAS Y NOTAS DE CREDITO	10
3.2	DIAGRAMA APLICACIÓN NOTAS DE CREDITO	10
3.3	DIAGRAMA RECIBOS	11
4.	Lógica de implementación	11
4.1	Facturas y Notas de Crédito	11
4.1.1	InvoicesEnt	11
4.1.1.1	GetInvoices	11
4.1.1.2	Lista de objetos GetInvoices.....	15
4.1.1.3	Pruebas (GetInvoices)	17
4.1.2	InvoiceBiz	18
4.1.2.1	BuildInvoices.....	19
4.1.2.2	Lista de objetos BuildInvoices	21
4.1.2.3	Pruebas BuildInvoices	22
4.1.2.4	UpdateStatusInvoices	23
4.1.2.4.1	Lista de objetos UpdateStatusInvoices	24
4.1.2.4.2	Pruebas UpdateStatusInvoices	25
4.1.2.5	GetComplementToUpdateInvoices	26
4.1.2.5.1	Lista de objetos GetComplementInvoice.....	26
4.1.2.5.2	Pruebas GetComplementInvoices.....	27
4.1.2.6	UpdateUuld.....	28
4.1.2.6.1	Lista de objetos UpdateUuld	31
4.1.2.6.2	Pruebas UpdateUuld	32
4.2	Aplicación de Notas de Crédito.....	33
4.2.1	CreditMemoApplyEnt	33
4.2.2	Pruebas (process).....	35
4.3	Creación y aplicación de Recibos.....	37

4.3.1	ReceiptEnt	37
4.3.1.1	Lista de Objetos ReceiptEnt	16
	bpelGetRfc.bpel:	16
	bpelCreateReceipts.bpel:	17
	bpelDigitalStampReceipts.bpel:	18
	bpelReceiptApply.bpel:	20
	sbpelNotifyHtmlControl.sbpel:	22
4.3.2	ReceiptBiz	22
4.3.2.1	DigitalStampReceipts	23
4.3.2.2	ReceiptApply	23
	sbpelNotifyHtmlControl	23
	bpelDigitalStampReceipts.bpel	24
	bpelReceiptApply.bpel	26
	sbpelNotifyHtmlControl.sbpel	29
4.3.3	Pruebas (CreateReceiptsRq)	22
5.	Diseño de Datos	23
5.1	Origen de Datos	23
5.2	Lógica de Validación	24
5.2.1	Facturas y Notas de Crédito	24
5.2.2	Recibos	24
6.	Diseño SQL	27
6.1	Objetos de Base de Datos	27
7.	Reglas de Negocio	28
7.1	Facturas y Notas de Crédito	28
7.2	Aplicación de Notas de Crédito	28
7.3	Recibos	28
8.	Diseño del Servicio	29
8.1	Pantalla de configuración	29
8.2	Registros de configuración	29
8.3	Facturas y Notas de Crédito	30
8.4	Aplicación de Notas de Crédito	31
8.5	Recibos	31
9.	Consideraciones de Rendimiento	32
9.1	Estrategia de Reinicio	32
9.2	Seguridad	32
9.3	Personalización	32
10.	Consideraciones de Instalación	33
10.1	Instalación de integración creación de transacciones	34
11.	URL de Acceso y seguridad	35

12. Temas abiertos y cerrados.....36

12.1 Temas Abiertos36

12.2 Temas Cerrados.....36

2. Resumen técnico

Este documento hace referencia a las especificaciones descritas en las reuniones previas de alcance 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.

2.1 Programación de ejecuciones

2.1.1 PROGRAMACION FACTURAS Y NOTAS DE CREDITO

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	INERTransaccionesJob
	Nombre Mostrado	INER Transacciones Job
	Paquete	/oracle/apps/ess/custom/soa
	Descripción	INER Transacciones Job
	Tipo de trabajo	SyncWebserviceJobType
	WSDL	http://ersoacsprod2019-wls-1:9073/soa-infra/services/INER/InvoicesEnt/InvoicesEnt?WSDL
	Tipo de puerto	InvoicesEntPort
	Operación	GetInvoices
Planificación	Nombre	INERTransaccionesSch
	Nombre Mostrado	INER Transacciones Sch
	Paquete	/oracle/apps/ess/custom/soa
	Descripción	INER Transacciones Sch
	Frecuencia	Diariamente
	Zona horaria	(UTC-06:00) Ciudad de México
	Fecha de Inicio	01:00:00 p. m.

Componente	Propiedad	Valor
Trabajo	Nombre	INERNotasCreditoJob
	Nombre Mostrado	INER Notas Credito Job
	Paquete	/oracle/apps/ess/custom/soa
	Descripción	INER Notas Credito Job
	Tipo de trabajo	SyncWebserviceJobType
	WSDL	http://ersoacsprod2019-wls-1:9073/soa-infra/services/INER/InvoicesEnt/InvoicesEnt?WSDL
	Tipo de puerto	InvoicesEntPort
	Operación	GetInvoices
Planificación	Nombre	INERNotasCreditoSch
	Nombre Mostrado	INER Notas Credito Sch
	Paquete	/oracle/apps/ess/custom/soa
	Descripción	INER Notas Credito Sch
	Frecuencia	Diariamente
	Zona horaria	(UTC-06:00) Ciudad de México
	Fecha de Inicio	01:30:00 p. m.

2.1.2 PROGRAMACION APLICACIÓN NOTAS DE CREDITO

En la capa **ENT** se encuentra la operación **process**, esta operación se encarga de iniciar el proceso de aplicación de notas de credito 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	INERAplicarNCTRAJob
	Nombre Mostrado	INER Aplicar NC TRA Job
	Paquete	/oracle/apps/ess/custom/soa
	Descripción	INER Aplicar NC TRA Job
	Tipo de trabajo	SyncWebserviceJobType
	WSDL	http://ersoacsprod2019-wls-1:9073/soa-infra/services/INER/CreditMemoApplyEnt/creditmemoapplybpel_client_ep?WSDL
	Tipo de puerto	CreditMemoApplyBPEL_pt
	Operación	process
Planificación	Nombre	INERAplicarNCTRASch
	Nombre Mostrado	INER Aplicar NC TRA Sch
	Paquete	/oracle/apps/ess/custom/soa
	Descripción	INER Aplicar NC TRA Sch
	Frecuencia	Diariamente
	Zona horaria	(UTC-06:00) Ciudad de México

	Fecha de Inicio	03:00:00 p. m.
--	-----------------	----------------

Componente	Propiedad	Valor
Trabajo	Nombre	INERAplicarNCDLVJob
	Nombre Mostrado	INER AplicarNC DLV Job
	Paquete	/oracle/apps/ess/custom/soa
	Descripción	INER AplicarNC DLV Job
	Tipo de trabajo	SyncWebserviceJobType
	WSDL	http://ersoacsprod2019-wls-1:9073/soa-infra/services/INER/CreditMemoApplyEnt/creditmemoapplybpel_client_ep?WSDL
	Tipo de puerto	CreditMemoApplyBPEL_pt
	Operación	process
Planificación	Nombre	INERAplicarNCDLVSch
	Nombre Mostrado	INER Aplicar NC DLV Sch
	Paquete	/oracle/apps/ess/custom/soa
	Descripción	INER Aplicar NC DLV Sch
	Frecuencia	Diariamente
	Zona horaria	(UTC-06:00) Ciudad de México
	Fecha de Inicio	03:10:00 p. m.

2.1.3 PROGRAMACION RECIBOS

En la capa **ENT** se encuentra la operación **CreateReceipts**, esta operación se encarga de iniciar el proceso de carga y aplicación de los recibos 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.

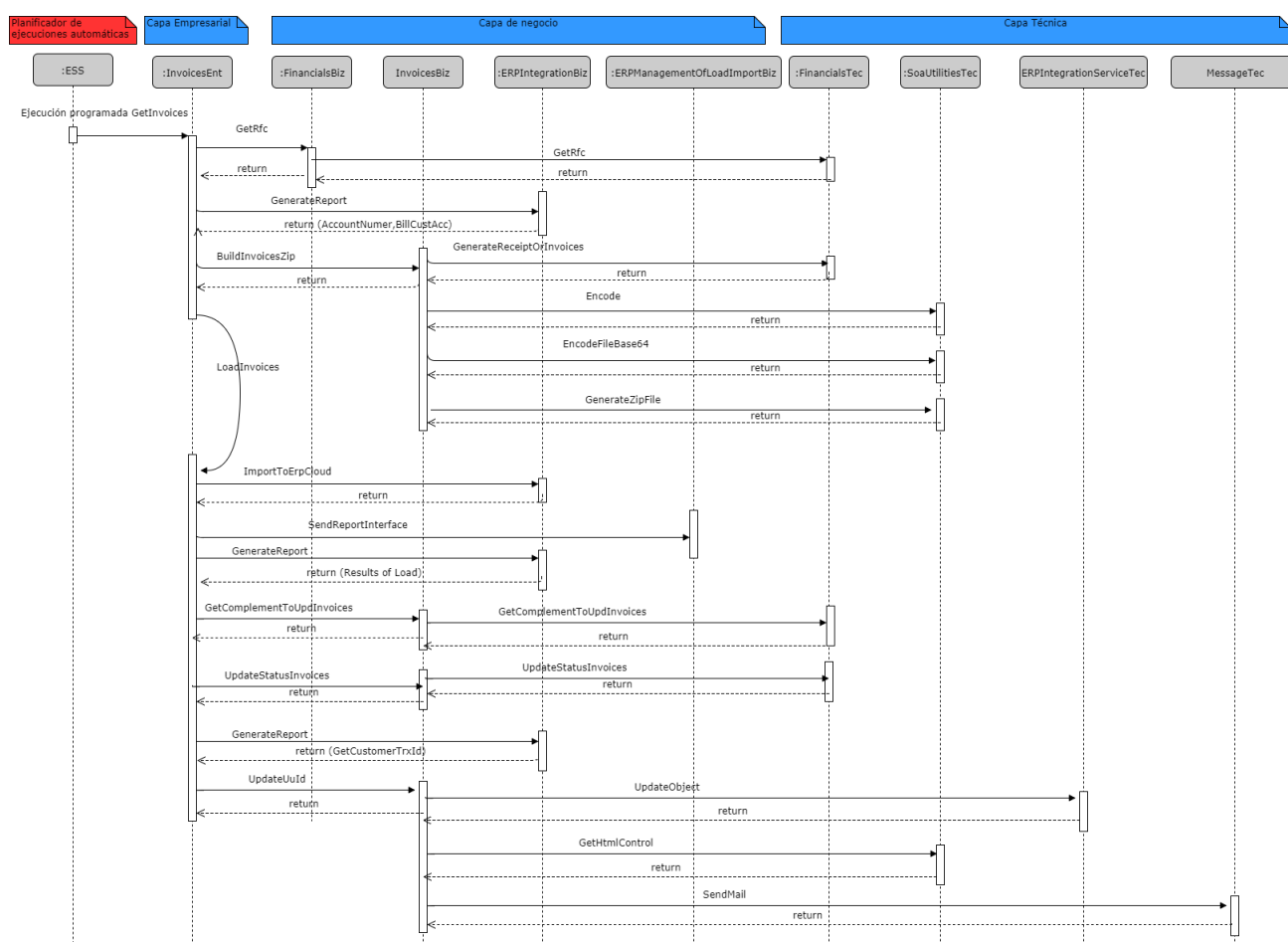
Componente	Propiedad	Valor
Trabajo	Nombre	INERRecibosJob
	Nombre Mostrado	INER Recibos Job
	Paquete	/oracle/apps/ess/custom/soa
	Descripción	INER Recibos Job
	Tipo de trabajo	SyncWebserviceJobType
	WSDL	http://ersoacsprod2019-wls-1:9073/soa-infra/services/INER/ReceiptEnt/ReceiptEnt?WSDL
	Tipo de puerto	ReceiptEntPort
	Operación	CreateReceipts
Planificación	Nombre	INERRecibosSch
	Nombre Mostrado	INER Recibos Sch
	Paquete	/oracle/apps/ess/custom/soa
	Descripción	INER Recibos Sch

	Frecuencia	Diariamente
	Zona horaria	(UTC-06:00) Ciudad de México
	Fecha de Inicio	01:45:00 p. m.

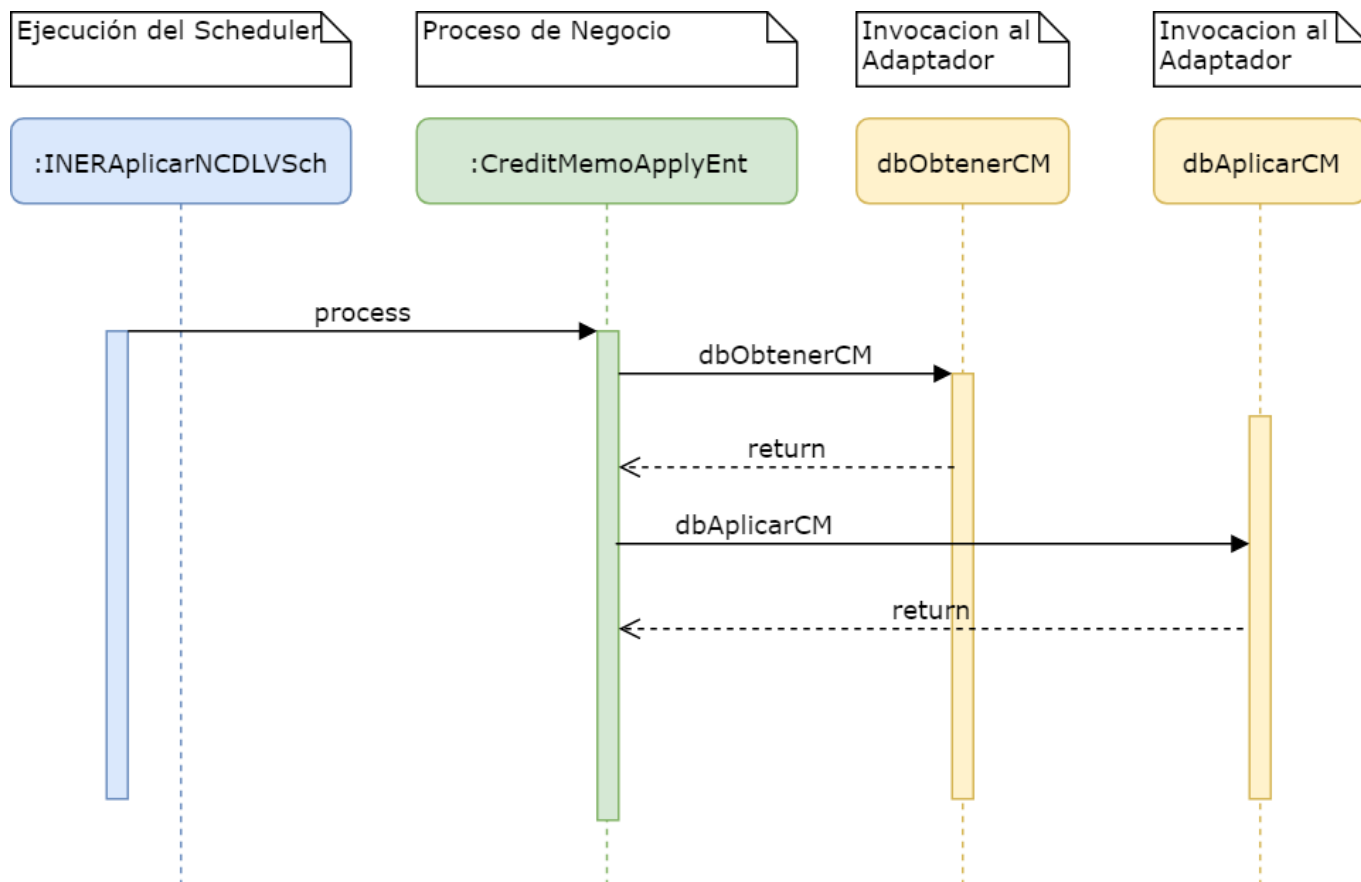
3. Diagrama de la integración

3.1 DIAGRAMA FACTURAS Y NOTAS DE CREDITO

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.

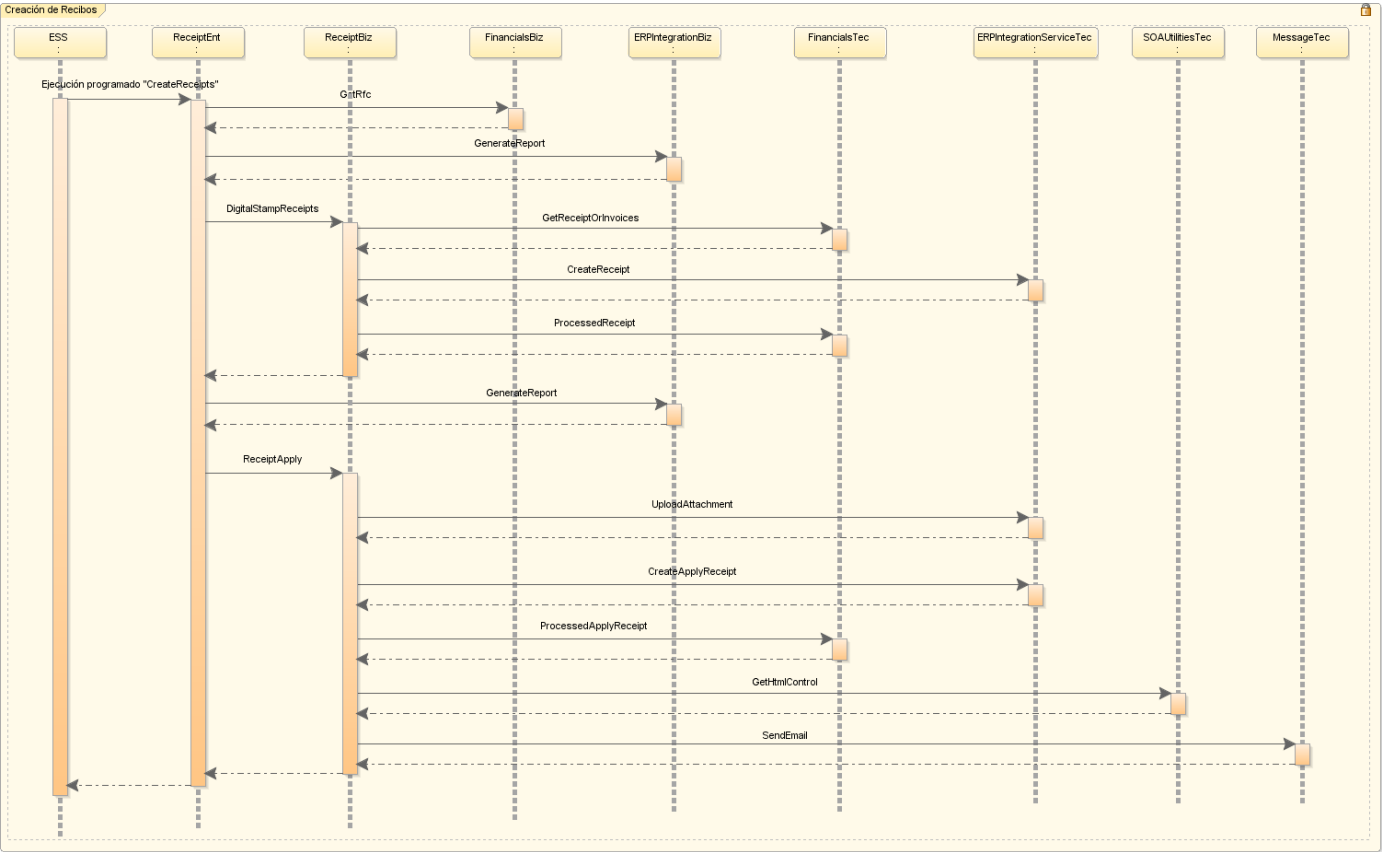


3.2 DIAGRAMA APLICACIÓN NOTAS DE CREDITO



3.3 DIAGRAMA RECIBOS

En el siguiente diagrama se muestra el flujo para la Creación y Aplicación de Recibos el cual inicia por la ejecución automática desde el ESS seguido por la capa **Ent** después **Biz**, para terminar en el **Tec**.



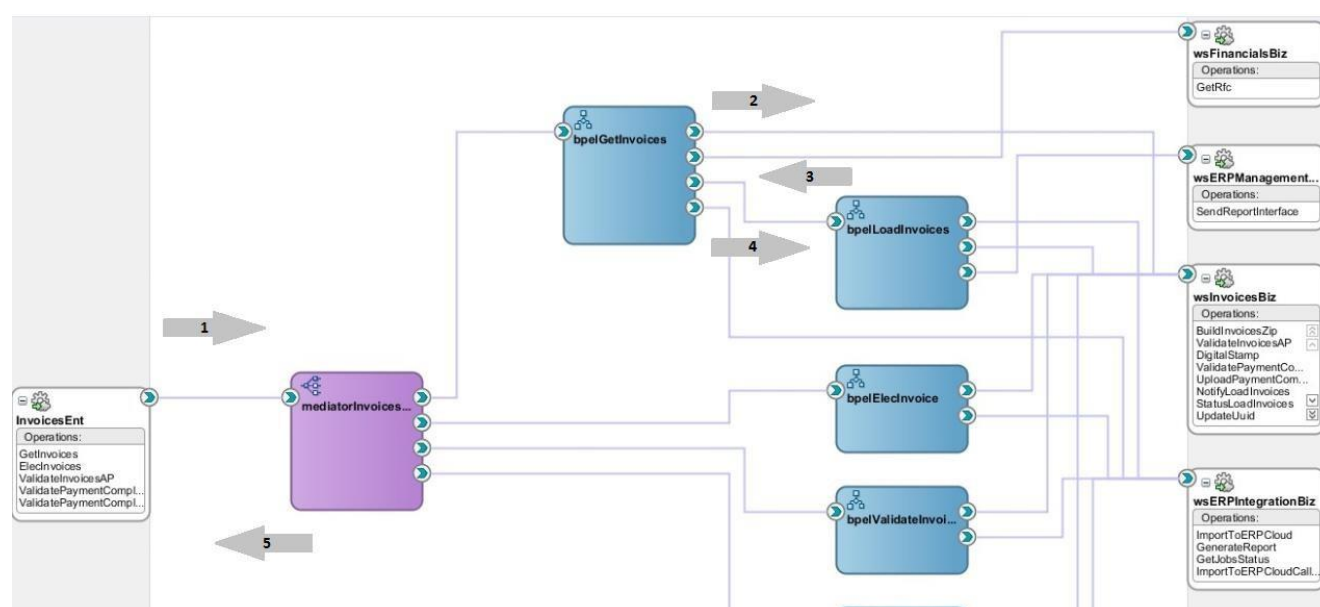
4. Lógica de implementación

En esta sección se describe cada una 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.

4.1 Facturas y Notas de Crédito

4.1.1 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.



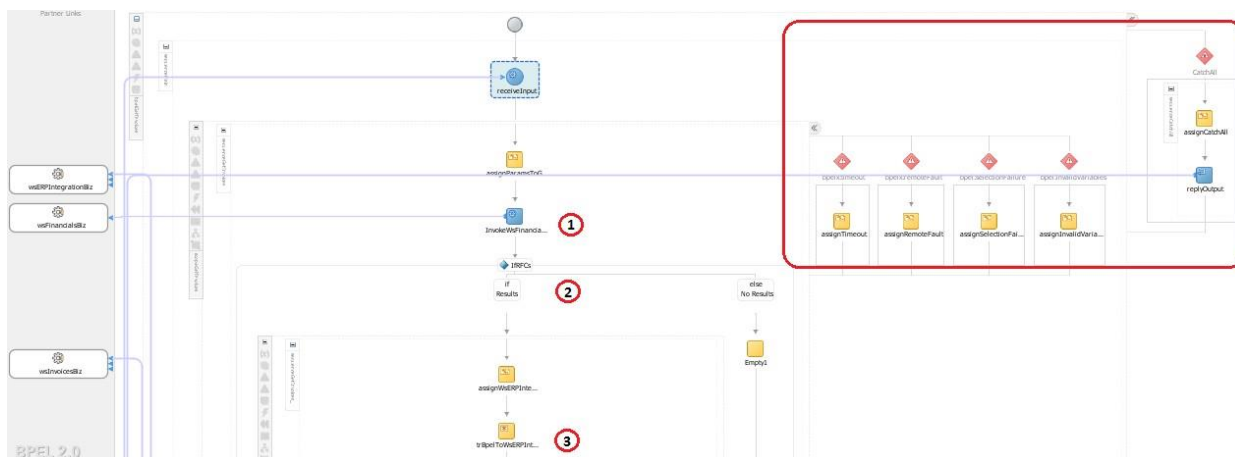
Actualmente el servicio InvoiceEnt expone otros endPoints además de GetInvoices. La integración que describe este documento solo considera la operación GetInvoices ya que es la que provee los flujos de trabajo en el alcance de esta integración. El resto de las operaciones no debe ser utilizada en ninguna circunstancia.

4.1.1.1 GetInvoices

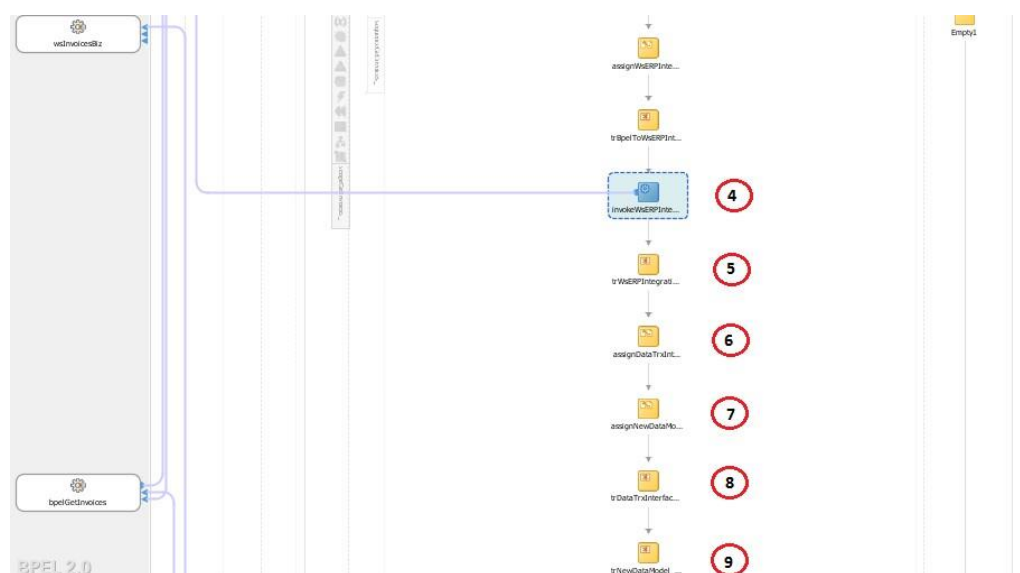
GetInvoices: Esta es la primera operación en ser invocada para el proceso de carga, el BPEL que se usa es de tipo síncrono a continuación se detallan los componentes que integran esta operación:

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.

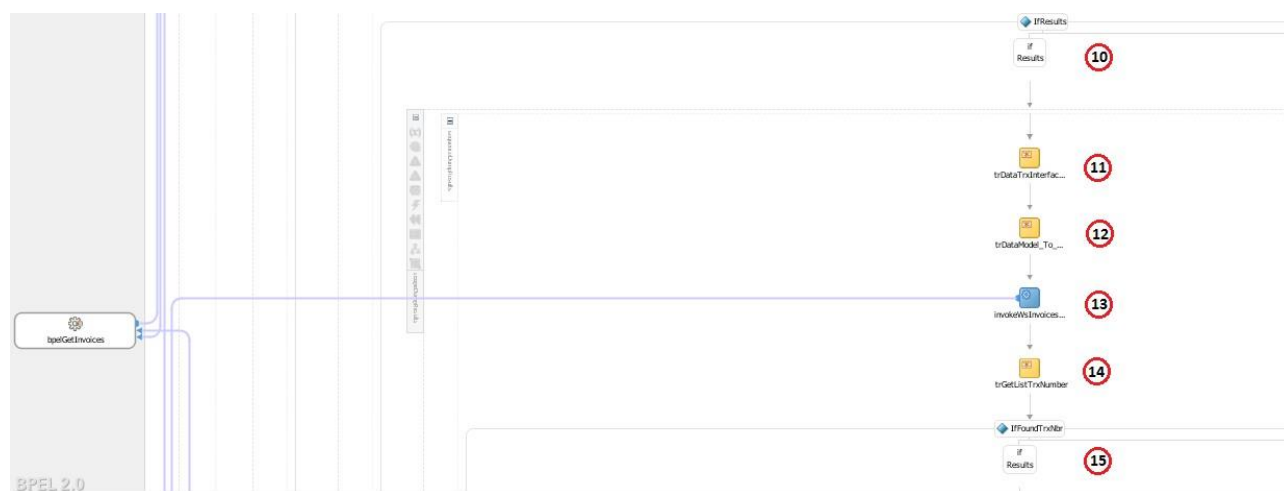
3. Se hace uso de una transformación para establecer valores a la variable de entrada para la invocación de la siguiente operación que es consumir un reporte.
 - a. Se integra el control de excepciones para capturar aquellas excepciones más comunes que lleguen a pasar en el flujo completo de la operación GetInvoices.



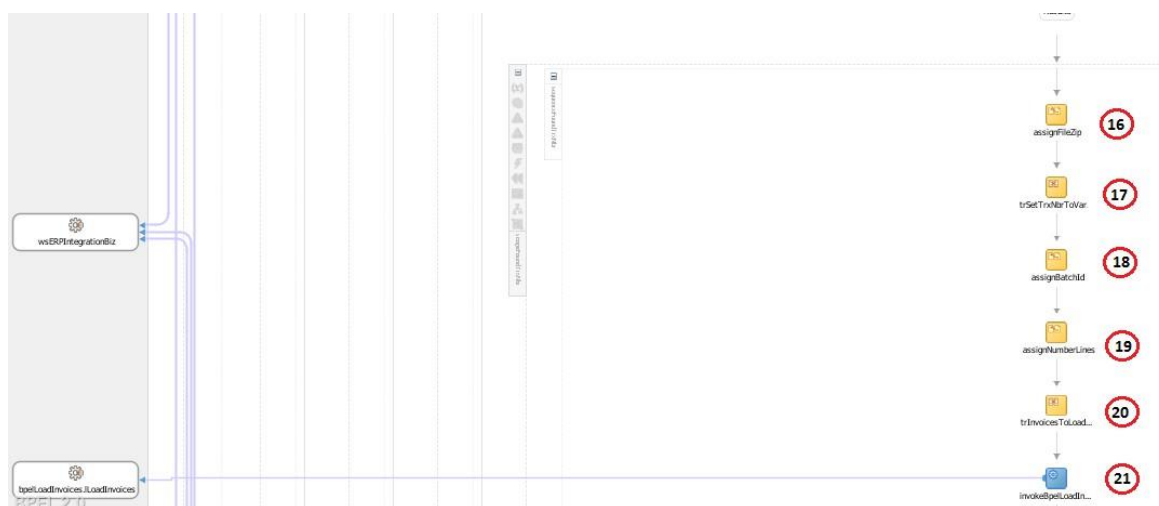
4. Se hace invocación al servicio **“ERPIntegrationBiz”-“GenerateReport”** para consumir el Data Model **“XXER_AR_TRX_INTERFACE”** que devolverá información complementaria para poder importar las Facturas o notas de crédito.
5. En este paso se hace uso de una transformación (**trWsERPIntegrationBizGenerateReportRs_To_WsERPIntegrationBizGenerateReportRs**) 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.

4.1.1.2 Lista de objetos GetInvoices

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 alcance
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_WsERPIntegrationBizGenerateReportRs	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

4.1.1.3 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://ersoacsprod2019-wls-1:9073/soa-infra/services/INERDEV/InvoicesEnt/InvoicesEnt?WSDL>

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

Request: Se necesitan parámetros de entrada por OPERACION para la ejecución de esta operación:

FACTURACION

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:GetInvoicesRq xmlns:ns1="http://soa.estrellaroja.com.mx/InvoicesEnt">
      <ns1:DocumentType>FACTURA</ns1:DocumentType>
    </ns1:GetInvoicesRq>
  </soap:Body>
</soap:Envelope>
```

NOTAS DE CRÉDITO

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:GetInvoicesRq xmlns:ns1="http://soa.estrellaroja.com.mx/InvoicesEnt">
      <ns1:DocumentType>NOTA DE CREDITO</ns1:DocumentType>
    </ns1:GetInvoicesRq>
  </soap:Body>
</soap:Envelope>
```

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

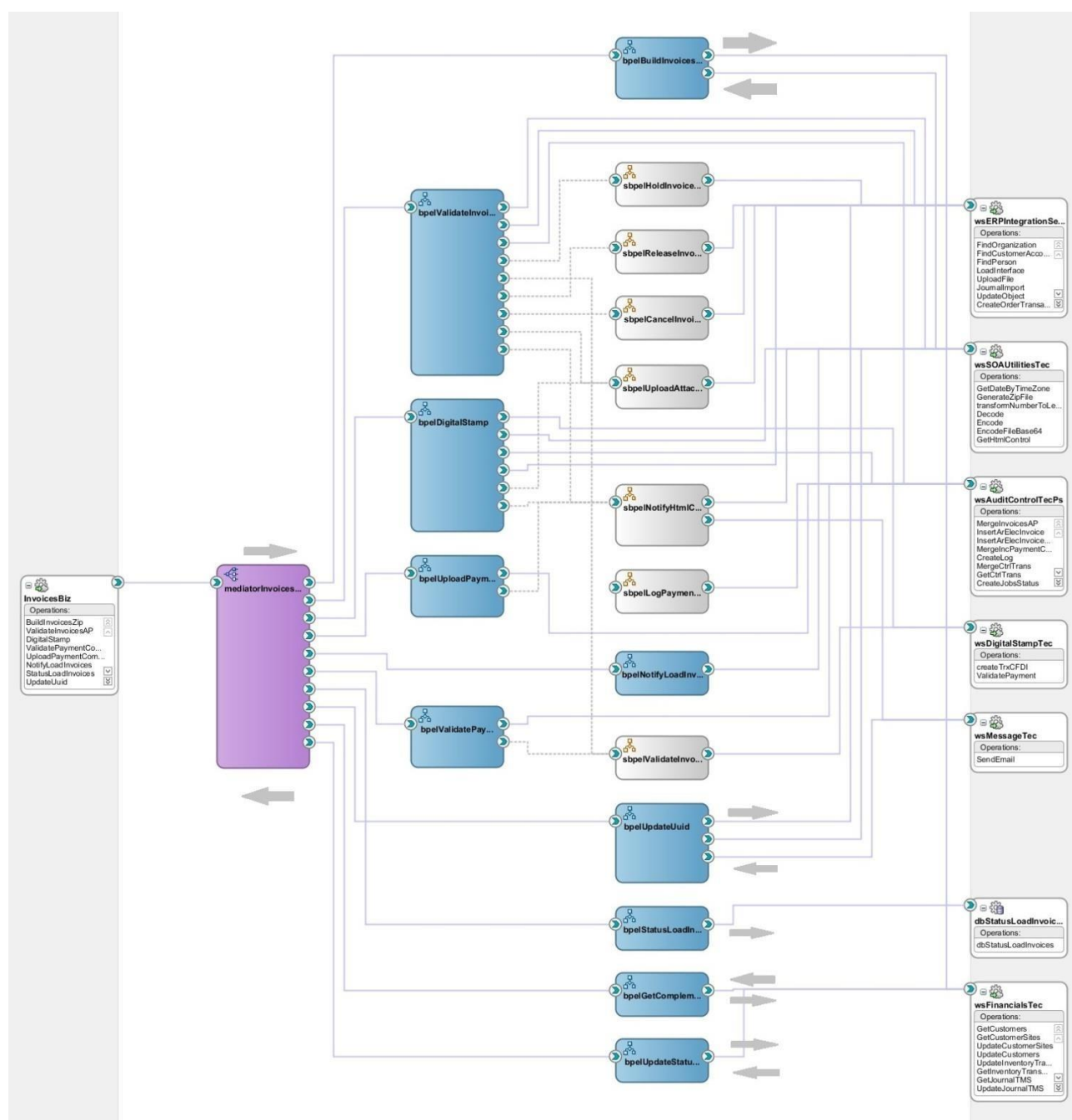
replyOutput

```
[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/EstrellaRojCommons" 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:Success/>
    </GetInvoicesRs>
  </part>
</outputVariable>
```

4.1.2 InvoiceBiz

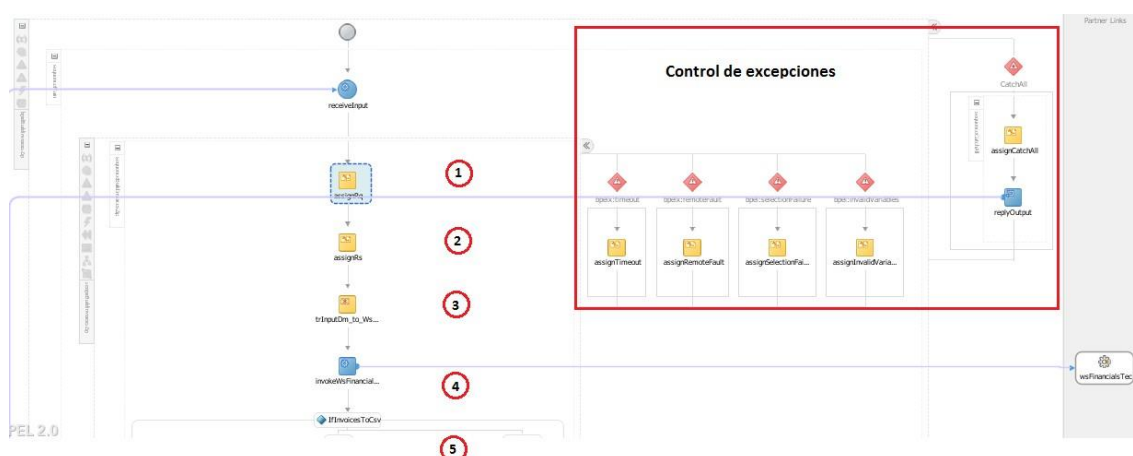
InvoicesBiz es el servicio que contiene la operación de la capa de negocio, esta capa es consumida por la capa Enterprise que en este caso es InvoicesEnt y la capa Biz hace uso de recursos de la capa Tec. En esta capa Biz se encuentran los procesos para ayudar a obtener información de la base de datos 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.



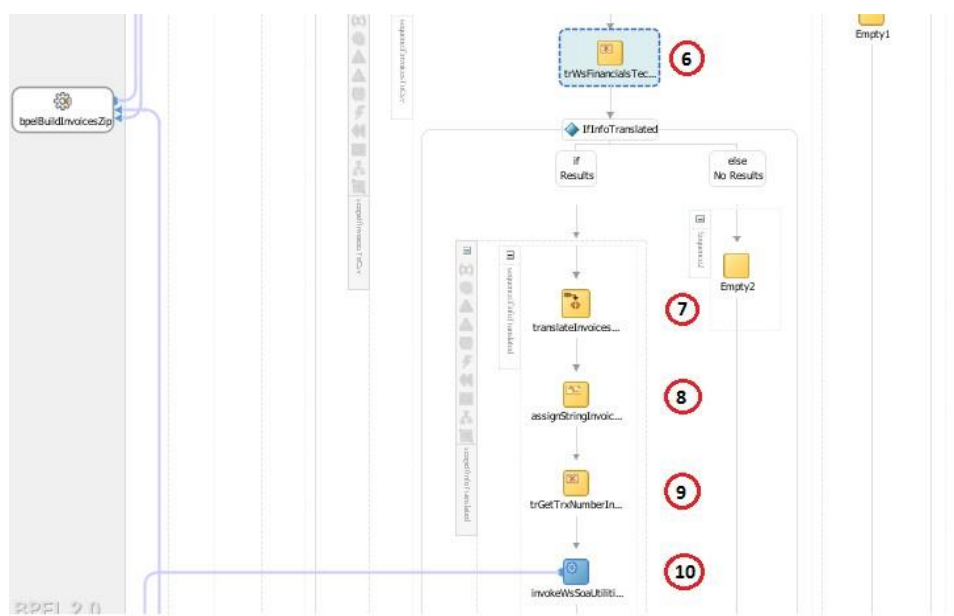
4.1.2.1 BuildInvoices

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.

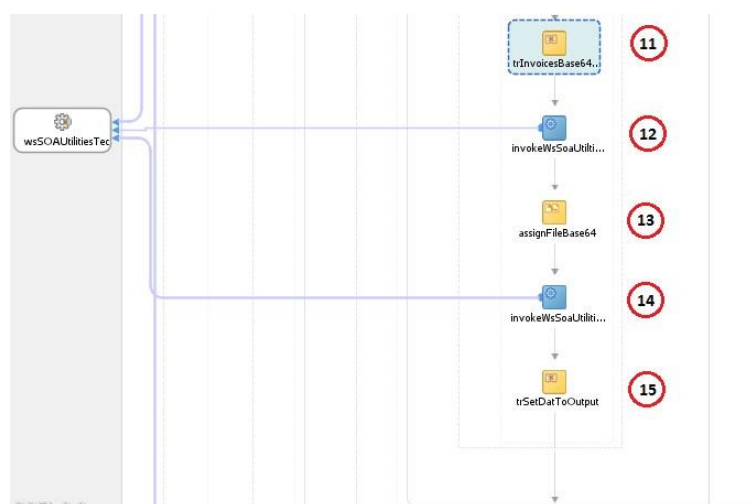
1. Se agrega asignación para establecer valor de algunas literales comunes para el flujo.
2. Se agrega asignación para establecer valor de algunas literales comunes para el flujo.
3. Se agrega la transformación (trInputDm_to_WsGetInvoicesReq) para asignar los parámetros a una variable y también establecer un valor como parámetro para la invocación del siguiente servicio que ayudará a generar el archivo CSV.
4. Se invoca el servicio “FinancialsTec”-“GetReceiptOrInvoices” 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, si 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.
9. 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 “SoaUtilitiesTe”-“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 bpeI.



4.1.2.2 Lista de objetos BuildInvoices

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	InvokeWsFinancialsTecGetReciptOrInvoices	Invocación	Invocación de servicio
9	IfInvoicesToCsv	Condición	Condición de datos
10	scopeIfInvoicesToCsv	Alcance	Define un nuevo alcance
11	sequenceIfInvoicesToCsv	Secuencia	Secuencia de asignaciones
12	trWsFinancialsTecGetReceiptOrInvoiceToBpel	Transformación	Transformación de esquemas
13	IfInfoTranslated	Condición	Condicionante de datos
14	scopeIfInfoTranslated	Alcance	Define un nuevo alcance
15	sequenceIfInfoTranslated	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	invokeWsSoaUiltiesTecEncodeFileBase64Rq	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

4.1.2.3 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

```
[2018/02/07 14:08:27]
Received "BuildInvoicesZip" call from partner "bpelBuildInvoicesZip"

- <inputVariable>
- <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="BuildInvoicesZipRq">
- <BuildInvoicesZipRq xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/" xmlns:ns2="http://schemas.oracle.com/bpel/extension" xmlns:plink="http://docs.oasis-open.org/wsbpel/2.0/plinktype"
  xmlns:ns1="http://soa.estrellaroja.com.mx/InvoicesEnt" xmlns:ns4="http://soa.estrellaroja.com.mx/EstrellaRojCommons" xmlns:ns3="http://soa.estrellaroja.com.mx/FinancialBiz"
  xmlns:ns5="http://soa.estrellaroja.com.mx/ERPIntegrationServiceTec" xmlns:ns6="http://soa.estrellaroja.com.mx/ERPIntegrationBiz" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:client="http://soa.estrellaroja.com.mx/InvoicesEnt/bpelGetInvoices" 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:InputData>
- <tns:DATA_DS>
- <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>300000003550202</tns:CUST_ACCT_SITE_ID>
- <tns:PARTY_SITE_NUMBER>19020</tns:PARTY_SITE_NUMBER>
- <tns:SITE_USE_ID>300000003550226</tns:SITE_USE_ID>
- <tns:SITE_USE_CODE>BILL_TO</tns:SITE_USE_CODE>
- <tns:PRIMARY_FLAG>Y</tns:PRIMARY_FLAG>
- <tns:LOCATION>20087</tns:LOCATION>
- <tns:LOCATION_ID>300000003550204</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>300000003550202</tns:CUST_ACCT_SITE_ID>
- <tns:PARTY_SITE_NUMBER>19020</tns:PARTY_SITE_NUMBER>
- <tns:SITE_USE_ID>300000003622737</tns:SITE_USE_ID>
- <tns:SITE_USE_CODE>SHIP_TO</tns:SITE_USE_CODE>
- <tns:PRIMARY_FLAG>Y</tns:PRIMARY_FLAG>
- <tns:LOCATION>27051</tns:LOCATION>
- <tns:LOCATION_ID>300000003550204</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>300000003622742</tns:CUST_ACCT_SITE_ID>
<tns:PARTY_SITE_NUMBER>20131</tns:PARTY_SITE_NUMBER>
<tns:SITE_USE_ID>300000003622745</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>XAXX0101000</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>300000003622742</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>
</BuildInvoicesZipRq>
</part>
</inputVariable>

```

[Copy details to clipboard](#)

Response:

 replyOutput

```

[2018/02/07 14:08:28]
Reply to partner "bpelBuildInvoicesZip".

- <outputVariable>
- <part xmlns:ns1="http://www.w3.org/2001/XMLSchema-instance" name="BuildInvoicesZipRs">
  - <BuildInvoicesZipRs xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/" xmlns:ns2="http://soa.estrellaroja.com.mx/EstrellaRojCommons" xmlns:plnktype="http://docs.oasis-open.org/wsbpel/2.0/plnktype"
    xmlns:ns4="http://schemas.oracle.com/bpel/extension" xmlns:ns3="http://soa.estrellaroja.com.mx/SOAUtilitiesTec/types" xmlns:ns5="http://soa.estrellaroja.com.mx/ERPIntegrationServiceTec"
    xmlns:ns6="http://soa.estrellaroja.com.mx/FinancialsTec" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:client="http://soa.estrellaroja.com.mx/InvoicesBiz/bpelBuildInvoicesZip"
    xmlns:ns1="http://soa.estrellaroja.com.mx/InvoicesBiz" xmlns="http://soa.estrellaroja.com.mx/InvoicesBiz">
    - <ns1:Return>
      - <ns1:ContentBase64>
        <base64>UEsDBBQACgIAA1R0uAAAAAAAAAAAAAAAAAAUmfJbml1c2h7ZWVhah51c0F5bC53bklTtwaz4WheeQgegtFG0Izblb0BCKtx9Siyeeof2v8Dtrv8UzhkgNE0/ZefjPgPCrUT1Rcxb8B8I9k50k0AG2Yn5UQ5vsdgp3urBPOT2du04ltg84E501NkAvdLqphcyBjijCvWbI4U3quDP45oL6/vm217088
        </base64>
      </ns1:ContentBase64>
    - <ns1:ListTrnNumbers>
      - <ns1:ListTrnNumber>
        <ns1:Trx_Number>ER3GR31120185BA</ns1:Trx_Number>
      </ns1:ListTrnNumber>
      - <ns1:ListTrnNumber>
        <ns1:Trx_Number>ER3GR31120185BA</ns1:Trx_Number>
      </ns1:ListTrnNumber>
      - <ns1:ListTrnNumber>
        <ns1:Trx_Number>ER3GR31120185BA</ns1:Trx_Number>
      </ns1:ListTrnNumber>
      <ns1:BatchId>2018-02-07 14:08:27</ns1:BatchId>
    - <ns1:ListTrnNumbers>
      </ns1:Return>
    </BuildInvoicesZipRs>
  </part>
</outputVariable>

```

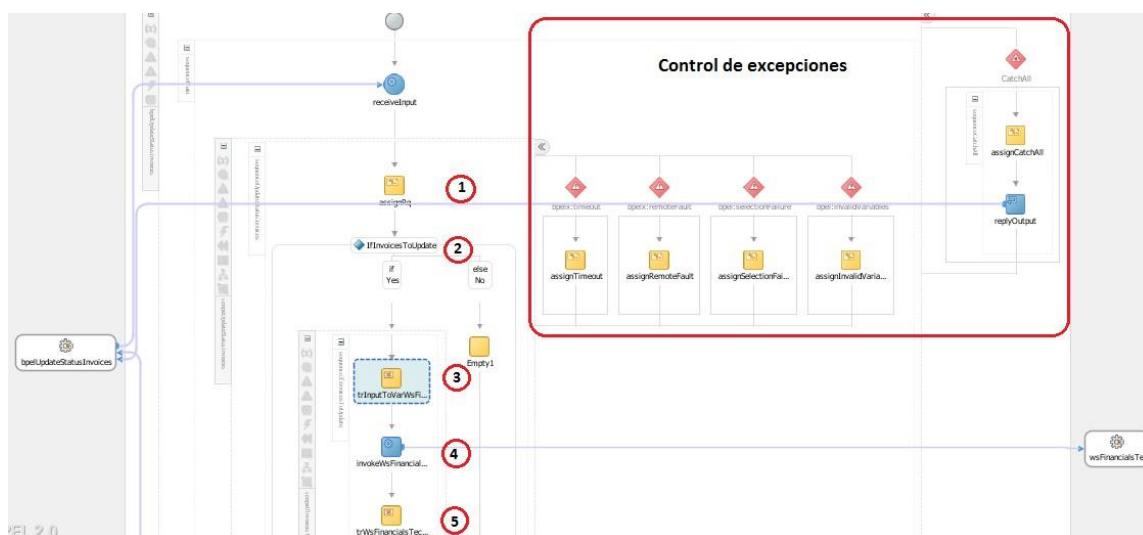
[Copy details to clipboard](#)

4.1.2.4 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.

16. Se agrega asignación para establecer valor a literal común del bpel.
17. 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.
18. Se usa la transformación (trlInputToVarWsFinancialsTecUpdateStatusInvoice)

19. 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.
20. 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.**



4.1.2.4.1 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	sequenceUpdateStatusInvoices	Secuencia	Secuencia de asignaciones
4	AssignRq	Asignación	Asignación de datos
5	IfInvoicesToUpdate	Condición	Condicionante de datos
6	scopeIfInvoicesToUpdate	Alcance	Define un nuevo alcance
7	sequenceIfInvoicesToUpdate	Secuencia	Secuencia de asignaciones
8	TrInputToVarWsFinancialsTecUpdateStatusInvoice	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

4.1.2.4.2 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:

receiveInput

[2018/02/07 14:09:41]
Received "UpdateStatusInvoices" call from partner "bpelUpdateStatusInvoices"

```

<inputVariable>
  <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="UpdateStatusInvoicesRq">
    <UpdateStatusInvoicesRq xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
      xmlns:ns2="http://soa.estrellaraja.com.mx/ERPIntegrationBiz" xmlns:plnk="http://docs.oasis-open.org/wsbpel/2.0/plnktype" xmlns:ns1="http://soa.estrellaraja.com.mx/InvoicesEnt"
      xmlns:ns4="http://soa.estrellaraja.com.mx/ERPIntegrationServiceTec"
      xmlns:ns3="http://soa.estrellaraja.com.mx/EstrellaRajaCommons"
      xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
      xmlns:client="http://soa.estrellaraja.com.mx/InvoicesEnt/bpelLoadInvoices"
      xmlns:tns="http://soa.estrellaraja.com.mx/InvoicesBiz"
      xmlns="http://soa.estrellaraja.com.mx/InvoicesBiz"
      xmlns:wsa="http://www.w3.org/2005/08/addressing">
      <tns:ListElements>
        <tns:BatchId>2018-02-07 14:08:27</tns:BatchId>
        <tns:TransactionNumber>ER2GR311201858A</tns:TransactionNumber>
        <tns:Status>$</tns:Status>
      </tns:ListElements>
      <tns:ListElements>
        <tns:BatchId>2018-02-07 14:08:27</tns:BatchId>
        <tns:TransactionNumber>ER2GR311201858A</tns:TransactionNumber>
        <tns:Status>$</tns:Status>
      </tns:ListElements>
    </UpdateStatusInvoicesRq>
  </part>
</inputVariable>

```

[Copy details to clipboard](#)

Response:

replyOutput

[2018/02/07 14:09:41]
Reply to partner "bpelUpdateStatusInvoices".

```

<outputVariable>
  <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="UpdateStatusInvoicesRs">
    <UpdateStatusInvoicesRs xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
      xmlns:ns2="http://soa.estrellaraja.com.mx/FinancialsTec/types"
      xmlns:ns1="http://soa.estrellaraja.com.mx/EstrellaRajaCommons" xmlns:plnk="http://docs.oasis-open.org/wsbpel/2.0/plnktype"
      xmlns:ns4="http://soa.estrellaraja.com.mx/ERPIntegrationServiceTec"
      xmlns:ns3="http://schemas.oracle.com/bpel/extension"
      xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
      xmlns:client="http://soa.estrellaraja.com.mx/InvoicesBiz/bpelUpdateStatusInvoices"
      xmlns:tns="http://soa.estrellaraja.com.mx/InvoicesBiz"
      xmlns="http://soa.estrellaraja.com.mx/InvoicesBiz">
      <tns:Return>
        <tns:NumberError/>
        <tns>ErrorDesc/>
      </tns:Return>
      <Success xmlns="http://soa.estrellaraja.com.mx/EstrellaRajaCommons"/>
    </UpdateStatusInvoicesRs>
  </part>
</outputVariable>

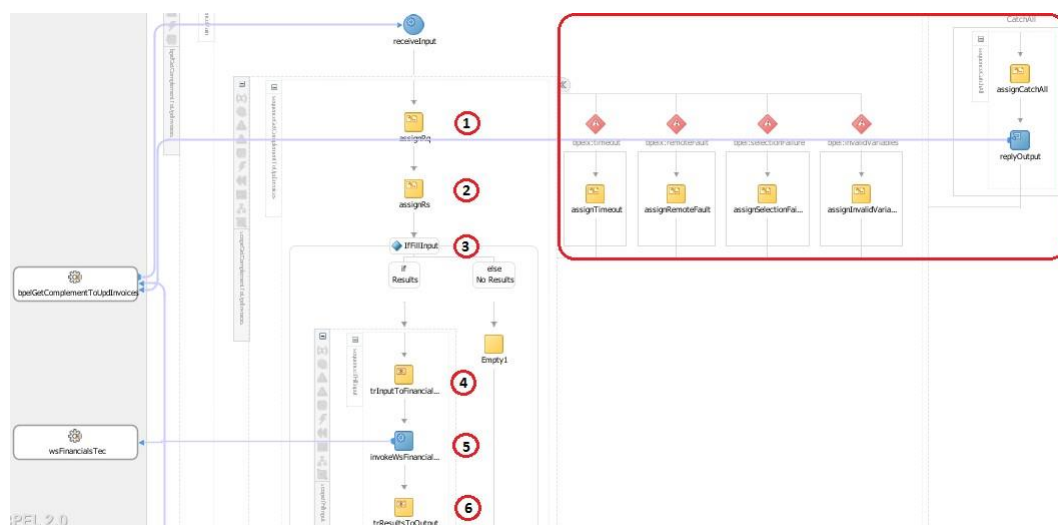
```

[Copy details to clipboard](#)

4.1.2.5 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.

21. Se asignan valores a literales comunes del bpel.
22. Se asignan valores a literales comunes del bpel.
23. Se usa IF para validar que los parámetros del bpel no vengan nulos, si no son nulos entonces se puede continuar con el flujo.
24. Se usa transformación (trInputToFinancialsTecGetComplementInvoices) para pasar los parámetros del bpel a variable request para invocar el servicio del siguiente paso.
25. 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.
26. 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.



4.1.2.5.1 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	lIfFillInput	Condición	Condicionante de datos
7	trInputToFinancialsTecGetComplementInvoices	Transformación	Transformación de esquemas
8	scopeIfFillInput	Alcance	Define un nuevo alcance
9	SequenceIfFillInput	Secuencia	Secuencia de asignaciones
10	invokeWsFinancialsTecGetComplementInvoices	Invocación	Invocación de servicio
11	trResultsToOutput	Transformación	Transformación de esquemas
12	replyOutput	Respuesta	Respuesta de bpel síncrono

4.1.2.5.2 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/ws-bpel/2.0/plinktype" 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>5</tns:STATUS>
    </tns:GetComplementInvoiceList>
    - <tns:GetComplementInvoiceList>
      <tns:TRX_NUMBER>ERJGR31120185BA</tns:TRX_NUMBER>
      <tns:STATUS>5</tns:STATUS>
    </tns:GetComplementInvoiceList>
    </GetComplementToUpdInvoicesRq>
  </part>
</inputVariable>

Copy details to clipboard

```


Response:

```

replyOutput

[2018/02/07 14:09:41]
Reply to partner "bpelGetComplementToUpdInvoices".
- <outputVariable>
- <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="GetComplementToUpdInvoicesRs">
- <GetComplementToUpdInvoicesRs xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:ns2="http://soa.estrellaroja.com.mx/FinancialsTec/types"
xmlns:ns1="http://soa.estrellaroja.com.mx/EstrellaRojCommons" xmlns:plnk="http://docs.oasis-
open.org/wsbpel/2.0/plnktype" xmlns:ns4="http://soa.estrellaroja.com.mx/ERPIntegrationServiceTec"
xmlns:ns3="http://schemas.oracle.com/bpel/extension" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:client="http://soa.estrellaroja.com.mx/InvoicesBiz/bpelGetComplementToUpdInvoices"
xmlns:tns="http://soa.estrellaroja.com.mx/InvoicesBiz"
xmlns="http://soa.estrellaroja.com.mx/InvoicesBiz">
- <tns:Return>
- <tns:GetComplementInvoiceList>
<tns:TRX_NUMBER>ERJGR31120185BA</tns:TRX_NUMBER>
<tns:ORG_ID>300000001049792</tns:ORG_ID>
<tns:UUID>194EB89E-6690-5118-8790-F1608FFC8061</tns:UUID>
</tns:GetComplementInvoiceList>
- <tns:GetComplementInvoiceList>
<tns:TRX_NUMBER>ERJGR31120185BA</tns:TRX_NUMBER>
<tns:ORG_ID>300000001049792</tns:ORG_ID>
<tns:UUID>194EB89E-6690-5118-8790-F1608FFC8061</tns:UUID>
</tns:GetComplementInvoiceList>
<tns:P_ERROR/>
<tns:P_ERROR_DESC/>
</tns:Return>
</GetComplementToUpdInvoicesRs>
</part>
</outputVariable>

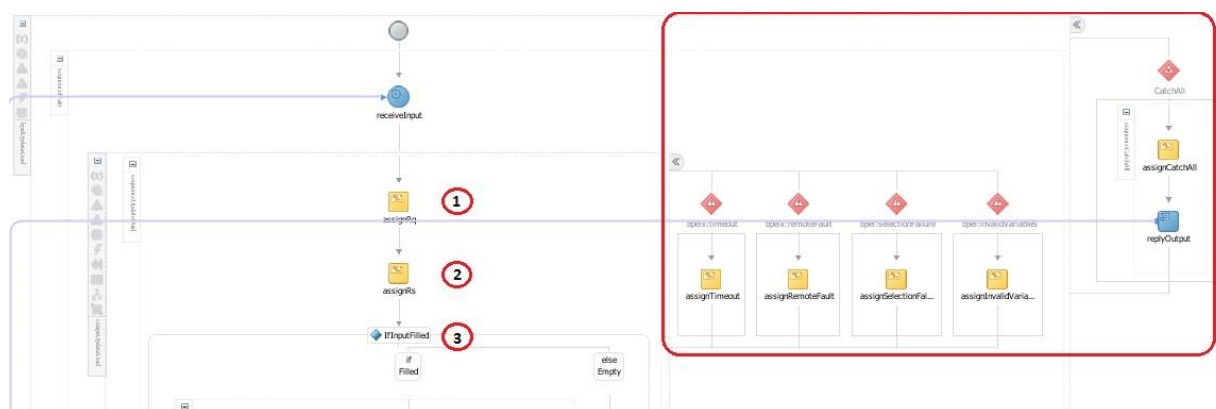
```

[Copy details to clipboard](#)

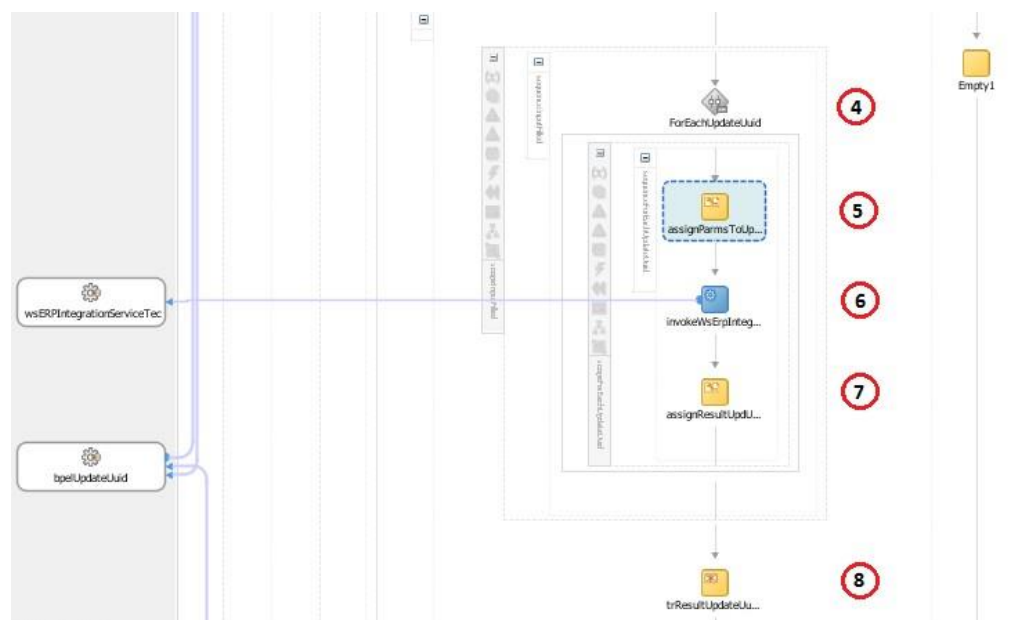
4.1.2.6 UpdateUuld

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.

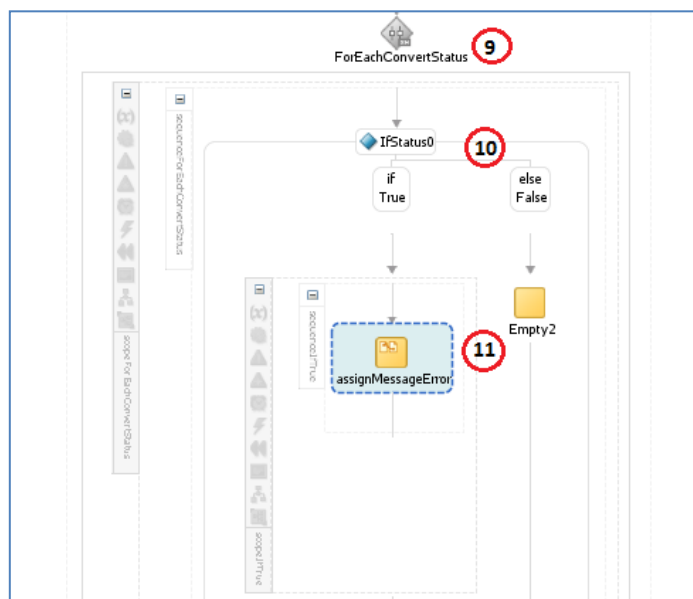
27. Se asignan valores a literales comunes del bpel.
28. Se asignan valores a literales comunes del bpel.
29. Se agrega IF para validar que los parámetros del BPEL no vengan nulos, si no están nulos se puede continuar con el flujo. **Se agrega manejador de excepciones al alcance principal.**



30. FOR EACH para iterar la lista de parámetro de facturas a actualizar.
31. Se usa asignación (assignParamsToUpdateObjectRq) para establecer valores a variable request para invocar servicio del siguiente paso.
32. Se invoca servicio “ErpIntegrationServiceTec”-“UpdateObject”, para mandar a actualizar el Uuld la factura o nota de crédito.
33. Asignación (assignResultUpdUuldToListResults) para recuperar en una variable los resultados de actualización del Uuld.
34. 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.



35. 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.
36. 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.
37. 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.



38. 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.
39. 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.
40. 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.
41. 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.
42. Se invoca el servicio “MessageTec”-“SendMail”, para notificar vía correo electrónico, los resultados de la actualización del Uuld
43. Transformación (trResultUpdateUuldToOutput) para recuperar los mensajes de resultado de la ejecución del servicio del paso anterior y pasarlo a la variable de salida del bpm.



4.1.2.6.1 Lista de objetos UpdateUuld

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	scopeUpdateUuld	Alcance	Define un nuevo alcance
3	sequenceUpdateUuld	secuencia	Secuencia de asignaciones
4	assignRq	Asignación	Asignación de datos
5	assignRs	Asignación	Asignación de datos
6	lIfInputFilled	Condición	Condición de datos
7	scopeInputFilled	Alcance	Define un nuevo alcance
8	sequenceInputFilled	Secuencia	Secuencia de asignaciones
9	ForEachUpdateUuld	Ciclo	Ciclo de registros
10	scopeForEachUpdateUuld	Alcance	Define un nuevo alcance
11	sequenceForEachUpdateUuld	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 lControl	Invocación	Invocación de servicio

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

4.1.2.6.2 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:

```


receiveInput

[2018/02/07 14:09:45]
Received "UpdateUuld" call from partner "bpelUpdateUuld"

- <inputVariable>
- <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="UpdateUuldRq">
- <UpdateUuldRq 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:ListUpdateUuldRq>
- <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:ListUpdateUuldRq>
- <tns:ListUpdateUuldRq>
- <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:ListUpdateUuldRq>
</UpdateUuldRq>
</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"
  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>ERJGR311201858A</ns0:TrxNumber>
  <ns0:Uuid>194E889E-6690-5118-879D-F1688FFC8061</ns0:Uuid>
  <ns0:Status>1</ns0:Status>
</ns0:Return>
- <ns0:Return>
  <ns0:TrxNumber>ERJGR311201858A</ns0:TrxNumber>
  <ns0:Uuid>194E889E-6690-5118-879D-F1688FFC8061</ns0:Uuid>
  <ns0:Status>1</ns0:Status>
</ns0:Return>
</UpdateUuidRs>
</part>
</outputVariable>

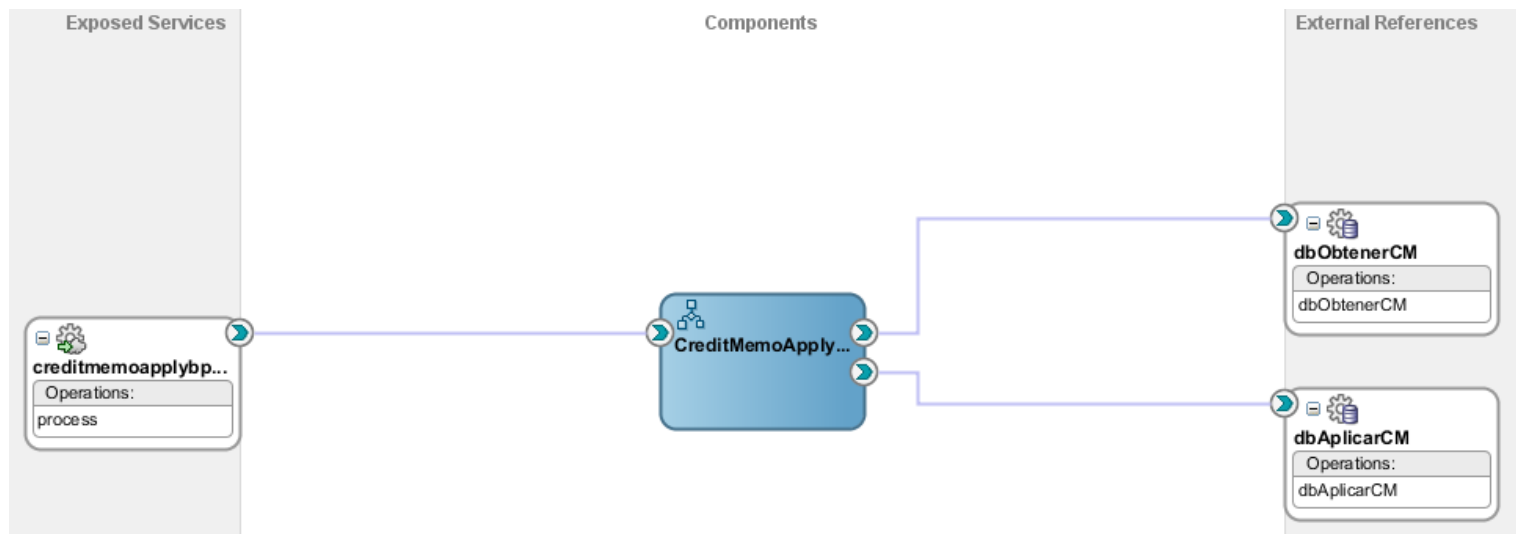
```

[Copy details to clipboard](#)

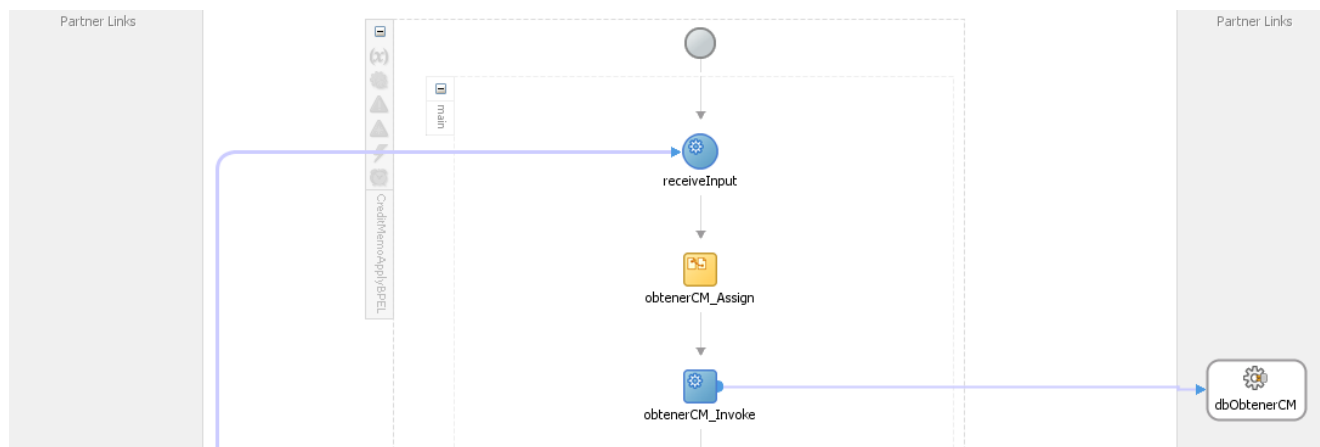
4.2 Aplicación de Notas de Crédito

Dentro de esta sección se describe cada una de las capas requeridas para la Aplicación de Notas de Crédito. Comenzando por la capa de CreditMemoApplyEnt en la cual se obtiene parte de la información requerida que se usará en para los adaptadores de base de datos.

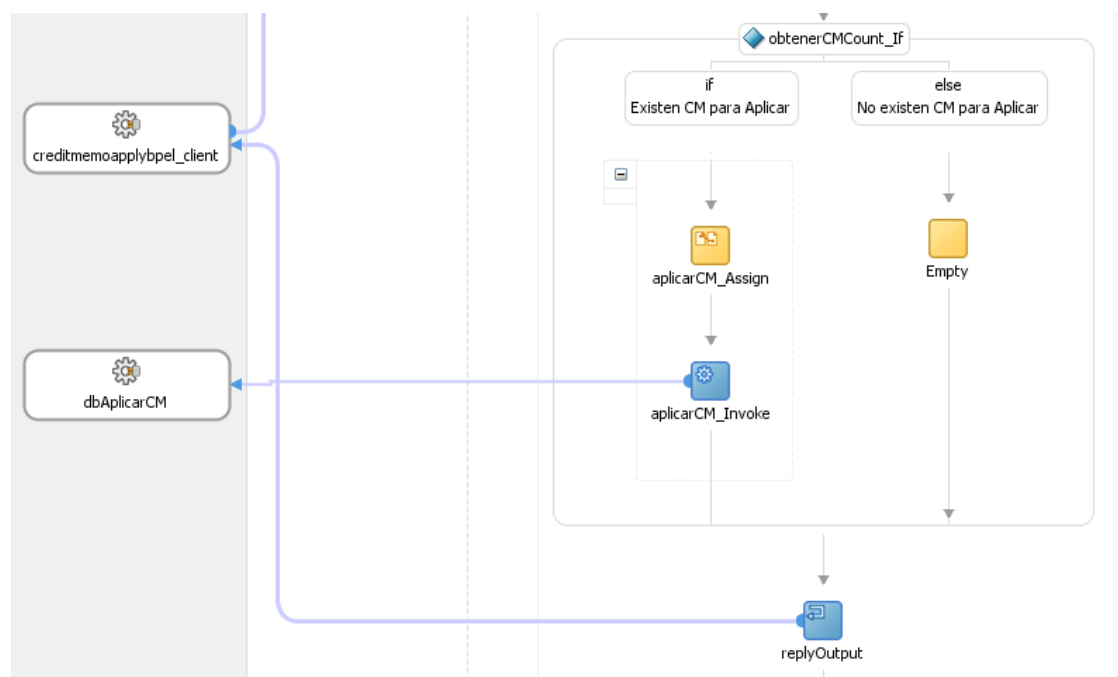
4.2.1 CreditMemoApplyEnt



Primero se manda a llamar el BPEL de CreditMemoApplyBPEL que contiene la lógica de negocio para la extracción de notas de crédito y posterior realiza la aplicación con en fusión.



- 1.- Asignación de la variable de Empresa
- 2.- Invocación al proceso de dbObtenerCM del PartnerLink dbObtenerM



- 3.- Condicional para conocer si existen Notas de Crédito a procesar
- 4.- En caso positivo, se asigna las Notas de Credito a la variable de entrada para el proceso de invocación
- 5.- En caso negativo, se continua con el proceso para terminar.
- 6.- Se invoca la operación dbAplicaCM del PartnerLink dbAplicaCM para enviar los operaciones a Fusion

4.2.2 Pruebas (process)

Al hacer una prueba del proceso en general para la aplicación de notas de crédito es necesario llamar primero al servicio CreditMemoApplyEnt.process.

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

http://ersoacsprod2019-wls-1:9073/soa-infra/services/INERDEV/CreditMemoApplyEnt/creditmemoapplybpel_client_ep?WSDL

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

Request: Se necesitan parámetros de entrada por EMPRESA para la ejecución de esta operación:

DLIVER

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:process
      xmlns:ns1="http://xmlns.oracle.com/CreditMemoApplyEnt/CreditMemoApplyEnt/CreditMemoApplyBPEL">
      <ns1:input>DLIVER</ns1:input>
    </ns1:process>
  </soap:Body>
</soap:Envelope>
```

TRATECSA

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
```

```

    <soap:Body>
      <ns1:process
xmlns:ns1="http://xmlns.oracle.com/CreditMemoApplyEnt/CreditMemoApplyEnt/CreditMemoApplyBPEL">
        <ns1:input>TRATECSA</ns1:input>
      </ns1:process>
    </soap:Body>
  </soap:Envelope>

```

Response:

```

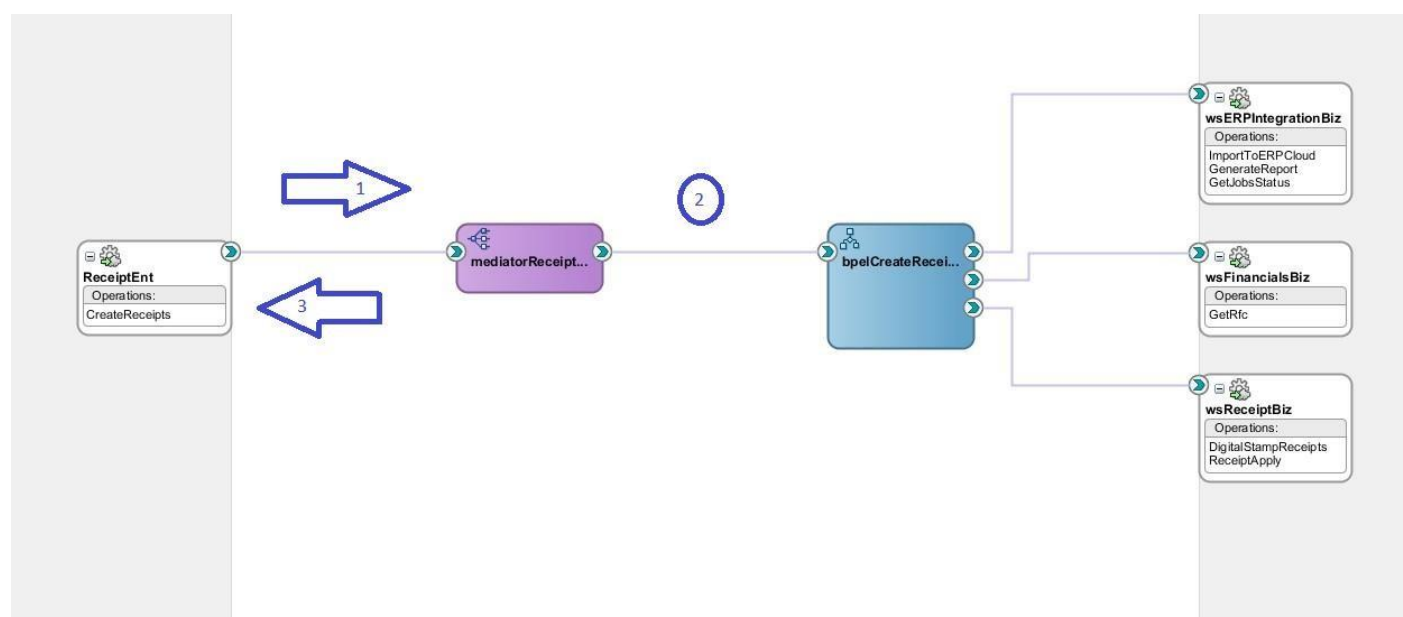
<?xml version="1.0" encoding="UTF-8"?><outputVariable>
  <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="payload">
    <processResponse
xmlns="http://xmlns.oracle.com/CreditMemoApplyEnt/CreditMemoApplyEnt/CreditMemoApplyBPEL"/>
  </part>
</outputVariable>

```

4.3 Creación y aplicación de Recibos

Dentro de esta sección se describe cada una de las capas requeridas para la Creación y Aplicación de Recibos. Comenzando por la capa de ReceiptEnt en la cual se obtiene parte de la información requerida que se usará en la siguiente capa ReceiptBiz, tomando en cuenta que dentro de la capa ReceiptEnt se estará haciendo uso de FinancialsBiz para poder encontrar la información que servirá como filtro de los DataModels.

4.3.1 ReceiptEnt



Dentro de la capa de “**ReceiptEnt**” se expone el servicio web ReceiptEnt con la operación “**CreateReceipts**” en la cual al consumirlo pasará por el mediator que dirige hacia “**bpelCreateReceipts**”, este se encarga de obtener la información necesaria para la creación y aplicación de Recibos.

1. Asigna variables necesarias para invocar el Servicio de **“FinancialsBiz”-“GetRfc”**, se requiere solo el valor **“RECIBO”** el cual se obtiene de **dvmCreateReceiptEnt.dvm**
2. Se invoca el Servicio de **“FinancialsBiz”-“GetRfc”**
3. Valida la respuesta del Servicio de **“FinancialsBiz”-“GetRfc”** que al menos contenga un elemento.



4. Asigna variables para la invocación del servicio **“ERPIntegrationServicesBiz”-“GenerateReport”**.
5. Transforma los datos para complementar los parámetros de entrada para invocar el servicio **“ERPIntegrationServicesBiz”-“GenerateReport”** con **dvmCreateReceiptEnt.dvm** junto con la respuesta del servicio **“FinancialsBiz”-“GetRfc”**.
6. Se invoca el Servicio **“ERPIntegrationServicesBiz”-“GenerateReport”**.
7. Se transforma el resultado del Servicio para inyectar namespace al data set **“Q_CUSTOMER”**.
8. Se asigna el resultado del Servicio a una variable del mismo tipo.
9. Se transforma cada uno de los datos de la variable anterior por otra variable del mismo tipo para poder tener acceso a ellos.
10. Se valida que contenga al menos un registro de los recibos que se deberán aplicar.

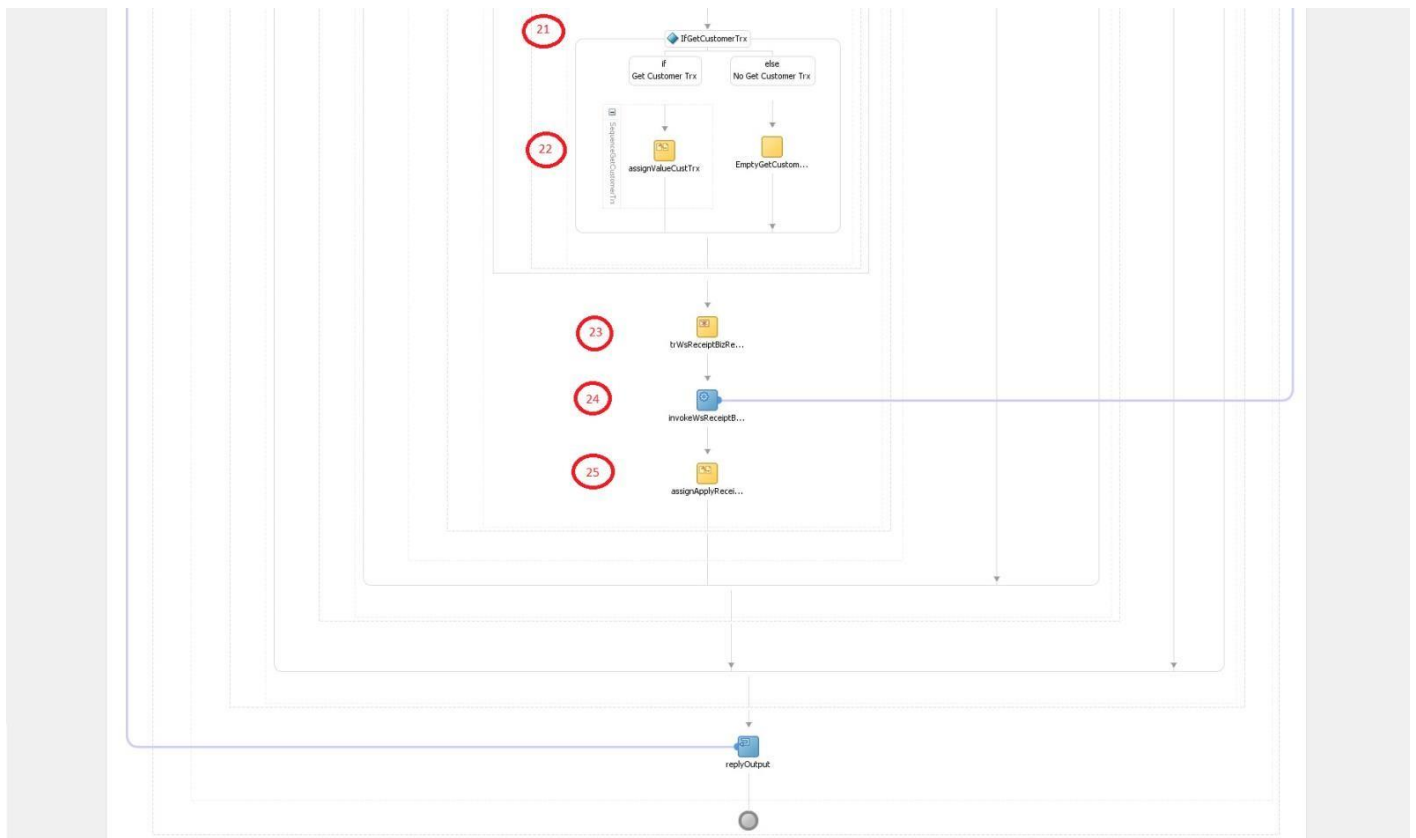


11. Se transforma los datos requeridos para el Servicio **“ReceiptBiz”-“DigitalStampReceipts”**.

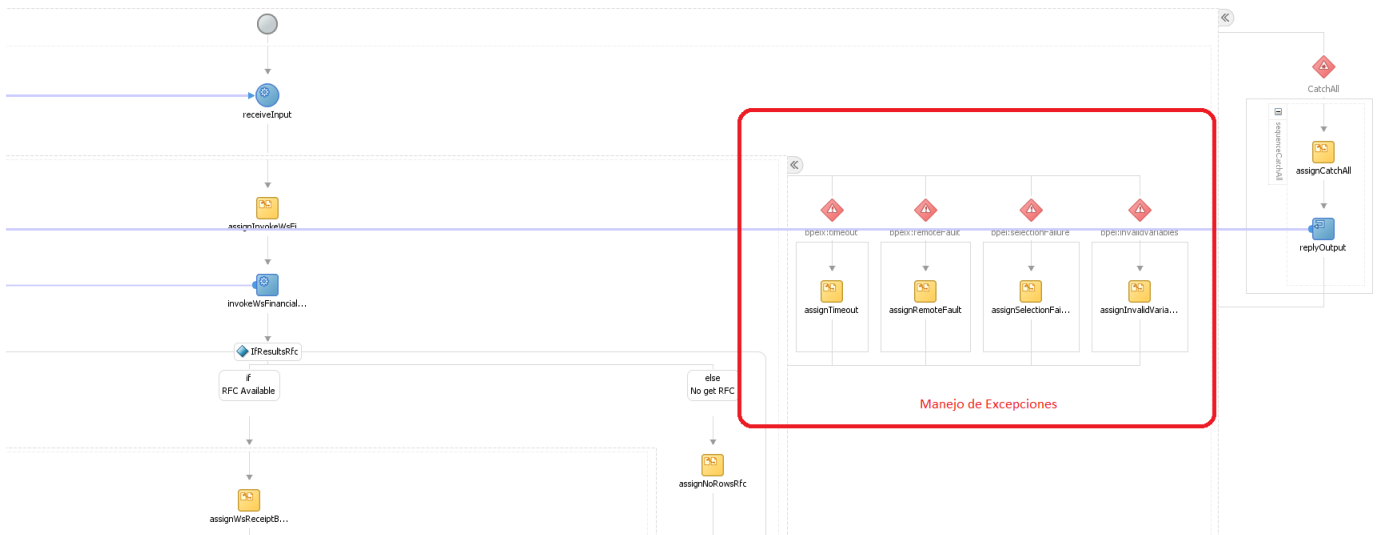
12. Invocación al Servicio **“ReceiptBiz”-“DigitalStampReceipts”** en donde se maneja la creación de Recibos.
13. Inicia **“FOR”** para iterar el resultado del Servicio.
14. Asigna el valor del contador a una variable.
15. Se asignan todos los parámetros necesarios para el Servicio **“ERPIntegrationServicesBiz”-“GenerateReport”**.
16. Se invoca el Servicio **“ERPIntegrationServicesBiz”-“GenerateReport”** para consultar Data Model y complementar la información necesaria para aplicar Recibos.
17. Dentro de la Transformación se inyecta el namespace para el data set **“CUSTOMER_TRX”**.
18. Dentro de la Transformación se inyecta el namespace para el data set **“ORG_UNIT”**.
19. Se asigna el resultado del Servicio a una variable del mismo tipo.
20. Se transforma cada uno de los datos de la variable anterior por otra variable del mismo tipo para poder tener acceso a ellos.



21. Se valida que exista resultado del Servicio.
22. Se asignan las variables que hacen falta para completar la información de respuesta del Servicio **“ReceiptBiz”-“DigitalStampReceipts”**.
23. Dentro de la transformación se establecen los parámetros necesarios para consumir el Servicio **“ReceiptBiz”-“ReceiptApply”** estos datos son los requeridos para aplicar recibos y los errores y cantidad de procesos exitosos.
24. Invoca el Servicio **“ReceiptBiz”-“ReceiptApply”** para los Attachment de Recibos y Aplicación de Recibos.
25. Asigna los valores de respuesta del Servicio **“ReceiptBiz”-“ReceiptApply”** a la salida del Servicio actual.



26. Dentro de este bpel se cuenta con una región en la cual se manejan las excepciones comunes presentadas en los Web Services.



4.3.1.1 Lista de Objetos ReceiptEnt

Los bpels que se usan, están compuestos por los siguientes componentes.

bpelGetRfc.bpel:

[ER05_AR_Creación y Aplicación de Recibos](#)

File Ref: TA.070-4 ER05_AR_Creacion_de_recibos_versiónFinal.docx

#	Nombre	Tipo	Descripción
1	assignWsFinancialTecReferenceOperationRq	assign	Asignación de valores.
2	invokeWsFinancialTecReferenceOperation	invoke	Invocación de Servicio.
3	trWsFinancialTecGetRfcRsToBpelGetRfcRs	transformation	Transformación de datos.

bpelCreateReceipts.bpel:

#	Nombre	Tipo	Descripción
1	assignInvokeWsFinancialsBiz	assign	Asignación de valores.
2	invokeWsFinancialsBizGetRfc	invoke	Invocación de Servicio.
3	IfResultsRfc	If	Validación de datos.
4	assignWsReceiptBizGenerateReportRq	assign	Asignación de valores.
5	assignNoRowsRfc	assign	Asignación de valores.
6	trBpelRqToWsERPIntegrationBizGenerateReportRq	transformation	Transformación de datos.
7	invokeWsERPIntegrationBizGenerateReport	invoke	Invocación de Servicio.
8	trWsERPIntegrationBizGenerateReportRs_To_WsERPIntegrationBizGenerateReportRs	transformation	Transformación de datos.
9	assignDataReceipt	assign	Asignación de valores.
10	trDataPaymentsReceipt_To_DataPaymentsReceiptResult	transformation	Transformación de datos.
11	IfCashInterfaceDM	If	Validación de datos.
12	trDataPaymentsReceiptResult_To_DataPaymentsReceiptBiz	transformation	Transformación de datos.
13	assignNoRows	assign	Asignación de valores.
14	invokeWsReceiptBizCreateReceipt	invoke	Invocación de Servicio.
15	ForGetCustomerTrx	forEach	Componente de repetición.

16	AssignVarTrxCount	assign	Asignación de valores.
17	assignInvokeWsIntegrationGenerateReportTrxCustomer	assign	Asignación de valores.
18	invokeWsIntegrationGenerateReportTrxCustomer	invoke	Invocación de Servicio.
19	trWsErpIntegrationBizGenerateReportRS_To_GenerateCustTrx	transformatio n	Transformació n de datos.
20	trWsErpIntegrationBizGenerateReportRS_To_GenerateOrgUnit	transformatio n	Transformació n de datos.
21	assignDataReceiptApply	assign	Asignación de valores.
22	trPaymentsReceiptApply_To_PaymentsReceiptApplyResult	transformatio n	Transformació n de datos.
23	IfGetCustomerTrx	If	Validación de datos.
24	assignValueCustTrx	assign	Asignación de valores.
25	EmptyGetCustomerTrx	Empty	Componente vacío.
26	trWsReceiptBizReceiptApplyRq	transformatio n	Transformació n de datos.
27	invokeWsReceiptBizReceiptApply	invoke	Invocación de Servicio.
28	assignApplyReceiptRs	assign	Asignación de valores.

bpelDigitalStampReceipts.bpel:

#	Nombre	Tipo	Descripción
1	IfDMIInput	If	Validación de datos.
2	assignWsReceiptTecRq	assign	Asignación de valores.
3	assignNoRowsRfc	assign	Asignación de valores.
4	trBpelRqToWsFinancialsTecGetInformationReceiptRq	transformation	Transformación de datos.
5	invokeWsFinancialsTecProcedure	invoke	Invocación de Servicio.
6	IfProcResult	If	Validación de datos.
7	assignErrorProcedure	assign	Asignación de valores.
8	IfResultProcReceipt	If	Validación de datos.
9	assignWsCreateReceiptRq	assign	Asignación de valores.
10	assignNoRowReceipt	assign	Asignación de valores.

11	ForCreateReceipt	forEach	Componente de repetición.
12	assignVarCount	assign	Asignación de valores.
13	assignInvokeWsStandardReceiptService	assign	Asignación de valores.

14	invokeWsErpIntegrationTecStandardReceipt	invoke	Invocación de Servicio.
15	IfReceiptProcessed	If	Validación de datos.
16	assignSuccessCreateReceipt	assign	Asignación de valores.
17	assignErrorCreateReceipt	assign	Asignación de valores.
18	assignCreateReceiptNotifyHtml	assign	Asignación de valores.
19	invokeWsProcessedReceipt	invoke	Invocación de Servicio.
20	TrBpelRsToWsProcessedReceipt	transformation	Transformación de datos.
21	assignTotalError	assign	Asignación de valores.
22	trBpelCopyRegErrorSuccess	transformation	Transformación de datos.

bpelReceiptApply.bpel:

#	Nombre	Tipo	Descripción
1	trBpelWsReceiptApplyRq	transformation	Transformación de datos.
2	IfContentReceipt	If	Validación de datos.
3	assignWsReceiptApplyRq	assign	Asignación de valores.
4	EmptyContentReceipt	Empty	Componente vacío.
5	forEachAttachment	forEach	Componente de repetición.
6	IfContentUUID	If	Validación de datos.
7	assignUploadAttachment	assign	Asignación de valores.
8	emptyUUID	Empty	Componente vacío.
9	invokeWsErpIntegrationServiceTecUploadAttachment	invoke	Invocación de Servicio.
10	IfResponseAttachment	If	Validación de datos.
11	assignSuccessAttachment	assign	Asignación de valores.
12	assignErrorAttachment	assign	Asignación de valores.
13	assignAttachmentNotifyHtml	assign	Asignación de valores.
14	assignWsErpIntegrationServiceTecCreateApplyReceiptRq	assign	Asignación de valores.
15	invokeWsErpIntegrationServiceTecCreateApplyReceipt	invoke	Invocación de

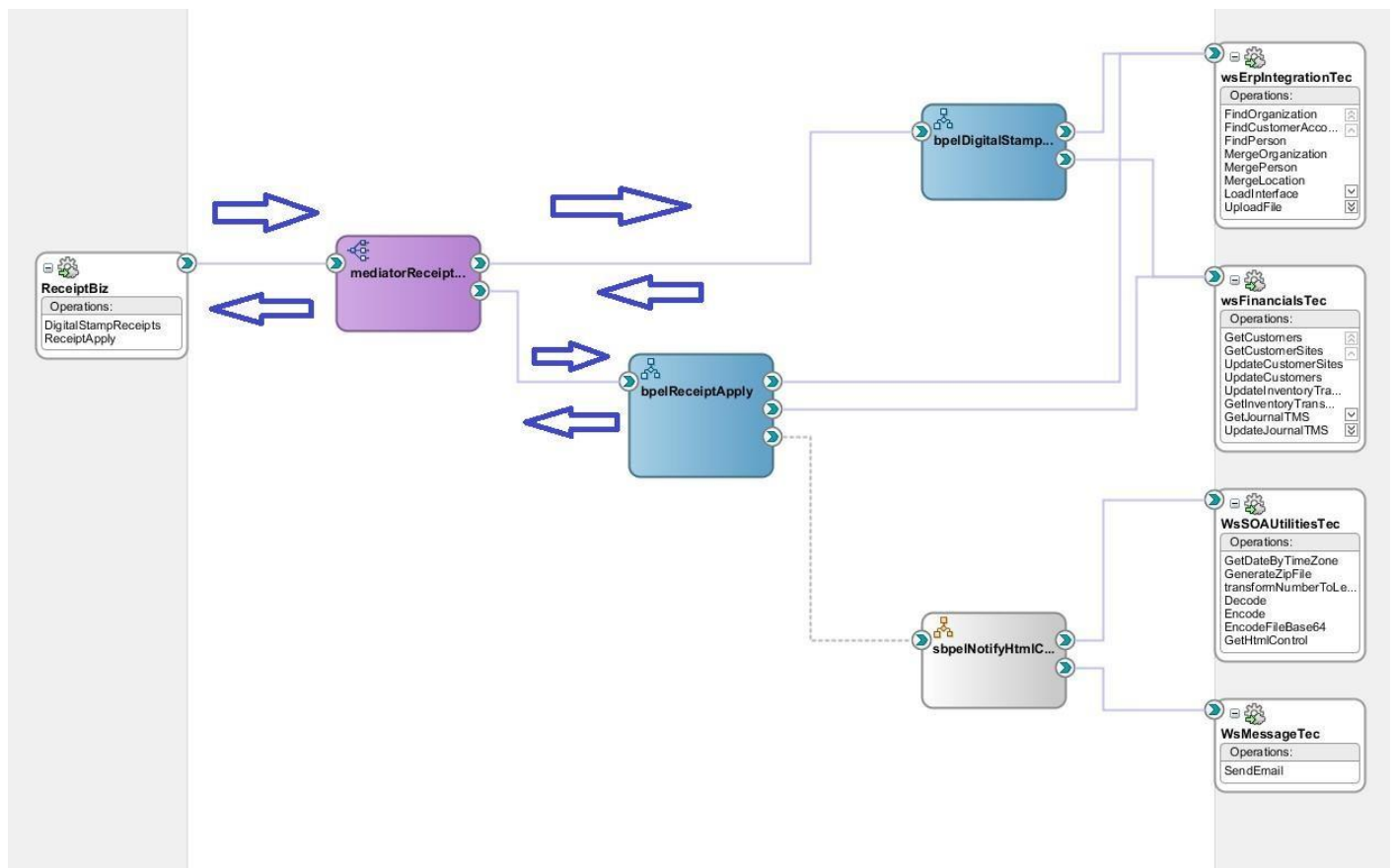
			Servicio.
16	IfResponseApply	If	Validación de datos.
17	assignSuccessApply	assign	Asignación de valores.
18	assignErrorApply	assign	Asignación de valores.

19	assignApplyNotifyHtml	assign	Asignación de valores.
20	invokeWsFinancialsTecRegApplyReceipt	invoke	Invocación de Servicio.
21	IfErrorsExist	If	Validación de datos.
22	sbpelNotifyHtmlControl	Subprocess	Componente subproceso bpel
23	EmptySendMail	Empty	Componente vacío.

sbpelNotifyHtmlControl.sbpel:

#	Nombre	Tipo	Descripción
1	invokeHtmlControl	invoke	Invocación de Servicio.
2	IfCorrectInvocation	If	Validación de datos.
3	assignParams	assign	Asignación de valores.
4	EmptyInvocation	Empty	Componente vacío.
5	invokeSendMail	invoke	Invocación de Servicio.

4.3.2 ReceiptBiz



4.3.2.1 DigitalStampReceipts

En esta operación se estarán creando Recibos recibiendo como parámetros la información requerida para crear cada uno de ellos. Como respuesta de este Servicio es la información de cada Recibo creado junto con los errores y mensajes de errores de los Recibos que no se crearon.

4.3.2.2 ReceiptApply

Dentro de esta operación se Aplicarán los Recibos, recibiendo la información requerida para poder aplicarlos y los errores y mensajes de error devueltos por la operación DigitalStampReceipts. La respuesta del Servicio son los errores globales.

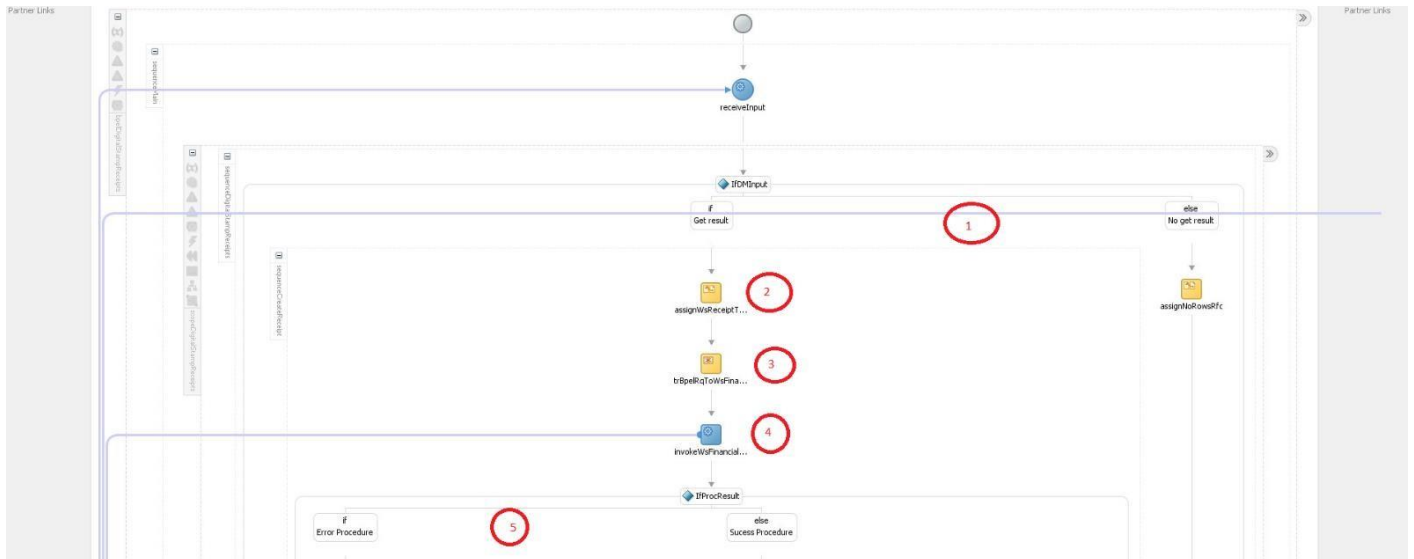
Dentro de este bpel se estarán manejando el envío de las notificaciones con el subproceso sbpelNotifyHtmlControl.

sbpelNotifyHtmlControl

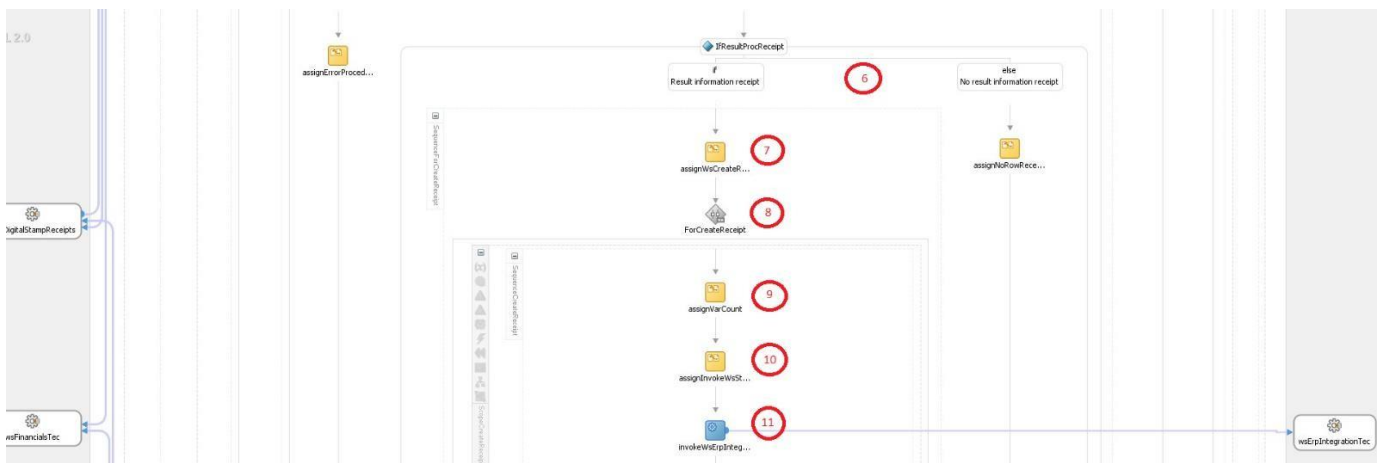
Este Bpel es llamado como subproceso dentro del cual se convierte a HTML la información a notificar y después se envía por correo electrónico.

bpelDigitalStampReceipts.bpel

27. Se valida que la entrada del Servicio contenga información requerida para los Recibos.
28. Si no contiene información se termina el flujo del Servicio. En caso contrario establece variable Global.
29. Transforma los datos de entrada hacia la variable de entrada del Servicio **“FinancialsTec”-“GetReceiptOrInvoices”**.
30. Se invoca el Servicio **“FinancialsTec”-“GetReceiptOrInvoices”** para complementar información en la creación de Recibos.
31. Se valida si existen errores en la respuesta del Servicio **“FinancialsTec”-“GetReceiptOrInvoices”**. Si los hay se termina el flujo del servicio. En caso contrario continua con el flujo.



32. Se valida que exista al menos un registro en la respuesta del Servicio anterior. Si no lo hay se termina el flujo. En caso contrario continua con el flujo.
33. Se inicializa variables globales.
34. Ciclo For en el cual se va iterar la respuesta del Servicio anterior.
35. Se asigna el valor del contador del For a variable local.
36. Se asignan valores a los parámetros de entrada del Servicio **“ErplIntegrationServiceTec”-“CreateReceipt”**.
37. Se invoca el Servicio **“ErplIntegrationServiceTec”-“CreateReceipt”** para la creación de Recibos.



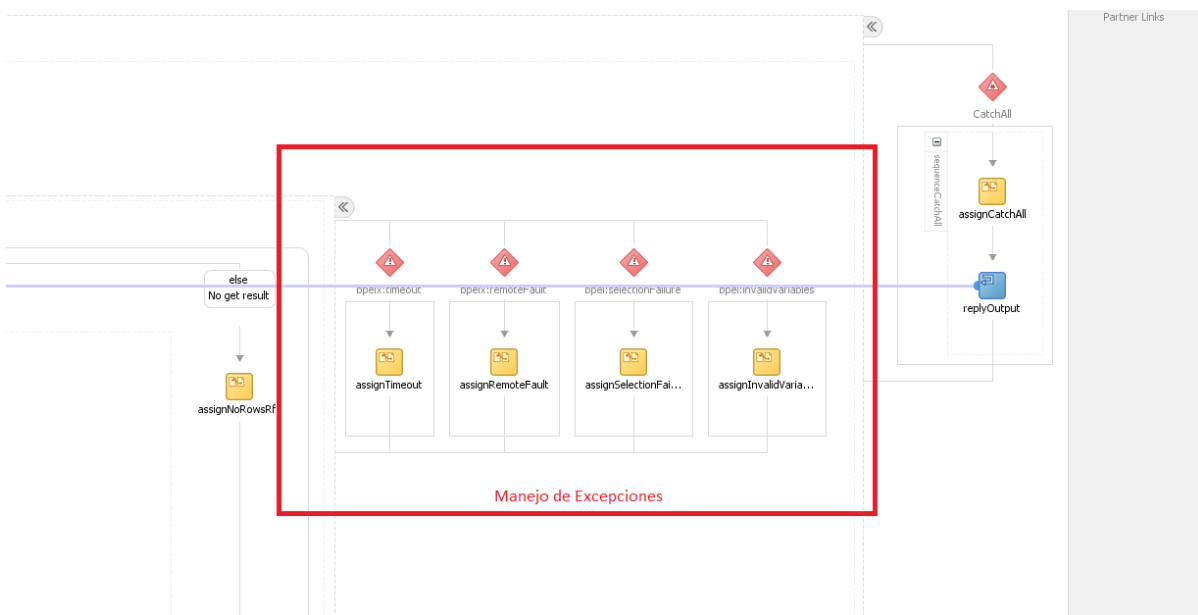
38. Valida la respuesta del Servicio **“ErpIntegrationServiceTec”-“CreateReceipt”**.
39. Si la respuesta del Servicio es correcta se incrementa la variable de procesos exitosos, en caso contrario si incrementa la variable de procesos con error.
40. Se establecen a la variable global los errores devueltos por el Servicio anterior. Aparte se llena la variable de entrada del Servicio **“FinancialsTec”-“ProcessedReceipt”** usado para registrar en base de datos la respuesta de la creación de los Recibos.



41. Se invoca el Servicio **“FinancialsTec”-“ProcessedReceipt”** para registrar la respuesta de la creación de cada Recibo.
42. La respuesta del Servicio anterior incluye la información de los Recibos que se aplicaran, esta información se transforma para ser regresada al proyecto ReceiptEnt.
43. Se asigna el valor de la variable global donde se registra la cantidad de errores hacia la variable de retorno de cantidad de procesos con errores.
44. Se transforma los datos de retorno para el total de procesos exitosos, total de errores, total de procesos y la descripción de los errores junto con sus códigos.



45. Esta región del bpel es utilizada para el manejo de Excepciones comunes en los Servicios Web.

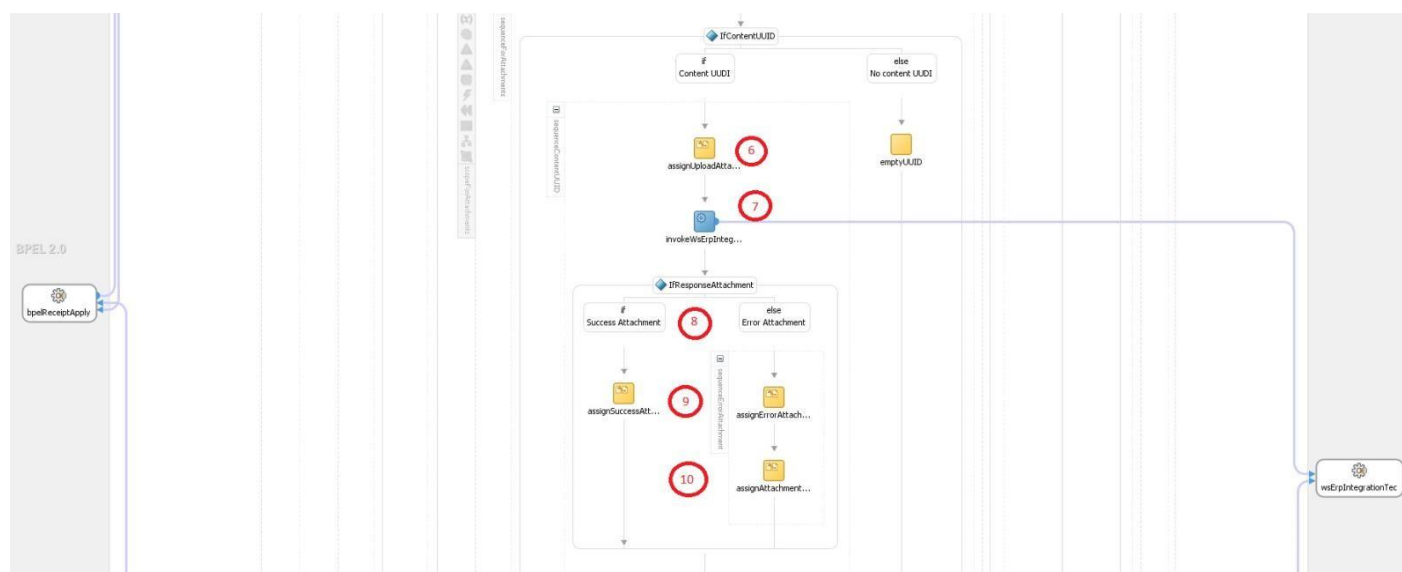


bpelReceiptApply.bpel

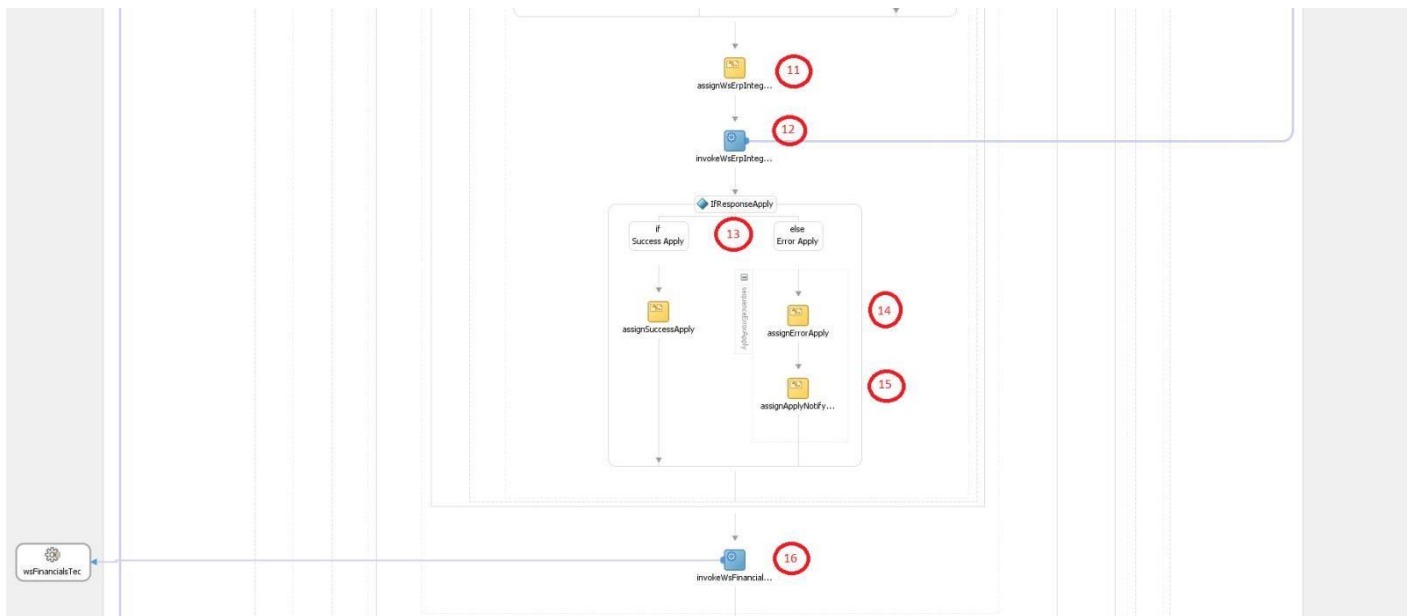
1. Transformación en la que se establecen los valores de entrada, necesarios para enviar la notificación. Estos valores se han enriquecido con los Servicios anteriores.
2. Se valida que existan valores para enviar Recibos a aplicar. Si no los hay el flujo se dirige a enviar la solicitud.
3. Establece variables globales.
4. Ciclo For el cual itera cada Recibo a aplicar.
5. Validación si el recibo a aplicar contiene valor UUID, se deberá agregar como Attachment, en caso contrario pasa al bloque en que se aplica el Recibo.



6. Utilizando los valores de entrada y de **dvmReceiptBiz.dvm** se asigna los valores de entrada para la invocación el Servicio **“ErpIntegrationServiceTec”.**”**UploadAttachment”**.
7. Se invoca el Servicio **“ErpIntegrationServiceTec”.**”**UploadAttachment”** en el cual se agregan los UUID por medio de los Attachment.
8. Se valida la respuesta del Servicio **“ErpIntegrationServiceTec”.**”**UploadAttachment”**.
9. Si la respuesta del Servicio es SUCCEED, incrementa la variable de procesos exitosos. En caso contrario incrementa la variable de procesos con error
10. Se establece el código de error y el mensaje de error para la variable global de las notificaciones.

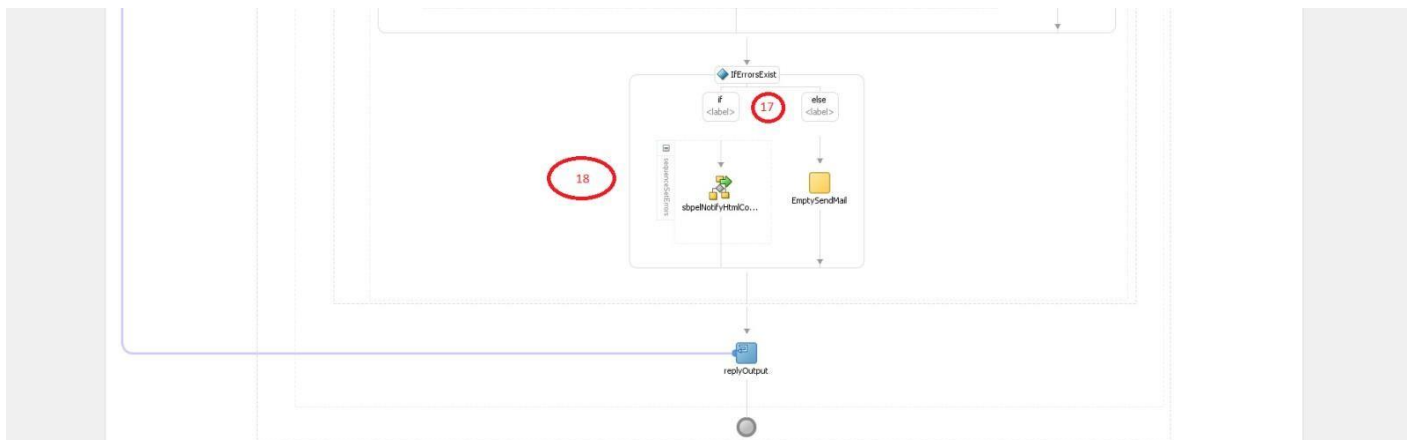


11. Asigna las variables de entrada para la invocación del Servicio **“ErpIntegrationServiceTec”-** **“CreateApplyReceipt”**.
12. Se invoca el Servicio **“ErpIntegrationServiceTec”-** **“CreateApplyReceipt”** el cual aplica los Recibos.
13. Se valida la respuesta del Servicio **“ErpIntegrationServiceTec”-** **“CreateApplyReceipt”**.
14. Si el Servicio **“ErpIntegrationServiceTec”-** **“CreateApplyReceipt”** crea correctamente la aplicación del Recibo este pasa a la asignación en donde se incrementa la variable de procesos exitosos. En caso contrario se asigna el incremento a la variable de procesos con error.
15. Se asigna el código del error y el mensaje del error a la variable global usada para las notificaciones, aparte se asigna la descripción del error a la variable usada para consumir el servicio **“FinancialsTec”-** **“ProcessedApplyReceipt”** usado para registrar en la base de datos cuales recibos no se pudieron aplicar.
16. Se invoca el Servicio **“FinancialsTec”-** **“ProcessedApplyReceipt”** para registrar si los recibos se pudieron o no aplicar.

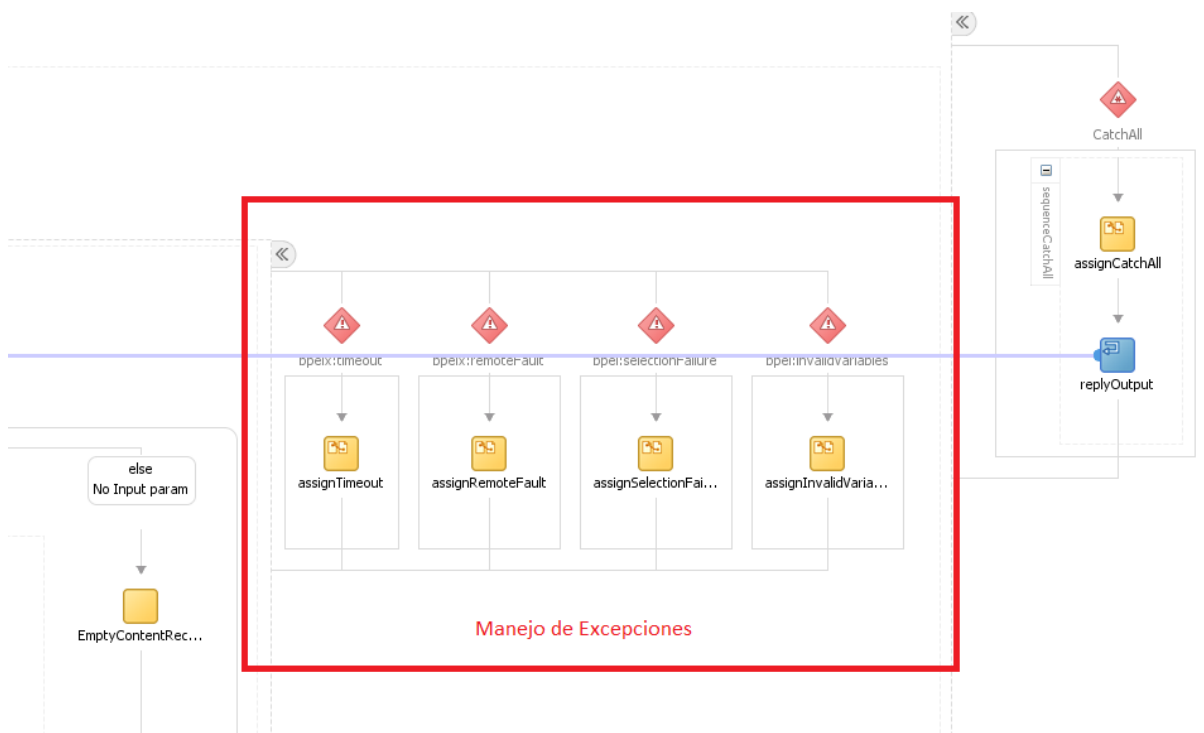


17. Se valida que existan transacciones exitosas y con errores para poder mandar la notificación.

18. Si se cumple la validación anterior se llama al subproceso **sbpelNotifyHtmlControl** dentro del cual se le pasan el total de procesos exitosos y los errores para ser enviados por correo electrónico.

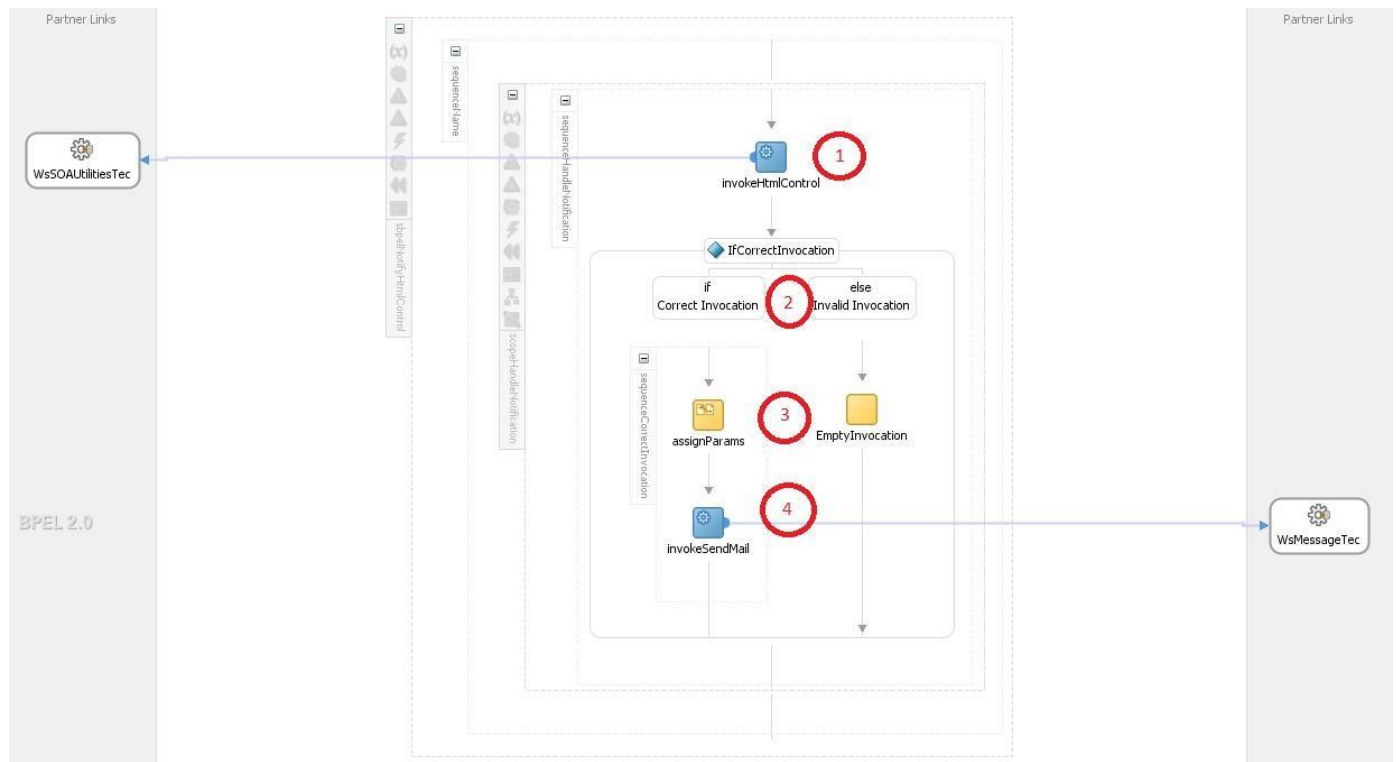


19. Esta región del bpm es utilizada para el manejo de Excepciones comunes en los Servicios Web.



sbpelNotifyHtmlControl.sbpel

1. Invoca al Servicio **“SOAUtilitiesTec”-“GetHtmlControl”** para Obtener un formato en Html con la descripción de los errores.
2. Se valida que no devuelva errores la respuesta de Servicio anterior.
3. Si la respuesta del Servicio es correcta se asigna la respuesta del servicio que es un Html y el valor de la DVM para obtener los destinatarios a los parámetros de entrada del Servicio **“MessageTec”-“SendEmail”**.
4. Se invoca el Servicio **“MessageTec”-“SendEmail”** el cual envía la notificación.



4.3.3 Pruebas (CreateReceiptsRq)

Para ejecutar una prueba manual es necesario ejecutar el Servicio de ReceiptEnt. URL del servicio en el ambiente de desarrollo:

<http://ersoacsprod2019-wls-1:9073/soa-infra/services/INNERDEV/ReceiptEnt/ReceiptEnt?WSDL>

REQUEST: El servicio no contiene parámetros.

```

<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Body>
    <ns1:CreateReceiptsRq
      xmlns:ns1="http://soa.estrellaroja.com.mx/ReceiptEnt"/>
    </soap:Body>
  </soap:Envelope>
  
```


5. Diseño de Datos

5.1 Origen de Datos

EMPRESA	NOMBRE DE TABLA	DESCRIPCION
TRATECSA	ZAMTRA_DESP_FLOTAS	Catálogo de Flotas
TRATECSA	ZAMTRA_GENERAL_IVAS	Catálogo de IVAS
TRATECSA	ZAMTRA_GENERAL_RETENCIONES	Catálogo de RETENCIONES
TRATECSA	ZAMTRA_MTTTO_UNIDADES	Catálogo de Unidades
TRATECSA	ZAMTRA_TRAFICO_BANCO	Catálogo de Bancos
TRATECSA	ZAMTRA_TRAFICO_CLIENTE	Catálogo de Clientes
TRATECSA	ZAMTRA_TRAFICO_DEPOSITO	Transacciones de depósitos
TRATECSA	ZAMTRA_TRAFICO_FACTURA_ELEC	Transacciones de Facturas Electrónicas
TRATECSA	ZAMTRA_TRAFICO_GUIA	Transacciones de guías
TRATECSA	ZAMTRA_TRAFICO_GUIA_OTRO	Detalle de guías
TRATECSA	ZAMTRA_TRAFICO_LIQUIDACION	Transacciones de liquidación
TRATECSA	ZAMTRA_TRAFICO_MOVIMIENTO	Transacciones de movimientos
TRATECSA	ZAMTRA_TRAFICO_NOTA_CREDITO	Transacciones de notas de crédito
TRATECSA	ZAMTRA_TRAFICO_PRODUCTO	Catálogo de Productos
TRATECSA	ZAMTRA_TRAFICO_RENGLON_LIQ	Detalle de Liquidaciones
TRATECSA	ZAMTRA_TRAFICO_RENGLON_MOV	Detalle de Movimientos
TRATECSA	ZAMTRA_TRAFICO_RENGLON_NCREDI	Detalle de Notas de Crédito
TRATECSA	ZAMTRA_TRAFICO_VIAJE_LIQ	Transacciones de viajes
DLIVER	ZAM_DESP_FLOTAS	Catálogo de Flotas
DLIVER	ZAM_GENERAL_IVAS	Catálogo de IVAS
DLIVER	ZAM_GENERAL_RETENCIONES	Catálogo de RETENCIONES
DLIVER	ZAM_MTTTO_UNIDADES	Catálogo de Unidades
DLIVER	ZAM_TRAFICO_BANCO	Catálogo de Bancos
DLIVER	ZAM_TRAFICO_CLIENTE	Catálogo de Clientes
DLIVER	ZAM_TRAFICO_DEPOSITO	Transacciones de depósitos
DLIVER	ZAM_TRAFICO_FACTURA_ELECTRON	Transacciones de Facturas Electrónicas
DLIVER	ZAM_TRAFICO_GUIA	Transacciones de guías
DLIVER	ZAM_TRAFICO_GUIA_OTRO	Detalle de guías
DLIVER	ZAM_TRAFICO_LIQUIDACION	Transacciones de liquidación
DLIVER	ZAM_TRAFICO_MOVIMIENTO	Transacciones de movimientos
DLIVER	ZAM_TRAFICO_NOTA_CREDITO	Transacciones de notas de crédito

DLIVER	ZAM_TRAFICO_PRODUCTO	Catálogo de Productos
DLIVER	ZAM_TRAFICO_RENGLON_LIQ	Detalle de Liquidaciones
DLIVER	ZAM_TRAFICO_RENGLON_MOVIMIENTO	Detalle de Movimientos
DLIVER	ZAM_TRAFICO_RENGLON_NCREDITO	Detalle de Notas de Crédito
DLIVER	ZAM_TRAFICO_VIAJE_LIQ	Transacciones de viajes
PAQUER	ER_FACTURAS_TBL	Facturas de Crédito
PAQUER	ER_PRODUCTOS_FACTURAS_TBL	Detalles de Productos por Factura
PAQUER	paq_tipos_pago_tbl	Catálogo de Tipos de Pago
PAQUER	paq_guias_tbl	Transacciones de guías
PAQUER	paq_cortes_tbl	Transacciones de cortes
PAQUER	paq_guias_prepagadas_tbl	Transacciones de Guías de Prepagadas
PAQUER	paq_cortes_tbl	Transacciones de cortes
PAQUER	paq_go_pagos_tbl	Transacciones de GO de Pagos
PAQUER	paq_pagos_tbl	Transacciones de pagos
PAQUER	PAQ_SUCURSALES_TBL	Catálogo de Sucursales

5.2 Lógica de Validación

5.2.1 Facturas y Notas de Crédito

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.

5.2.2 Recibos

Las validaciones principales que se encuentran dentro de esta integración tenemos las siguientes:

- Se crearán los recibos que estén con estatus **Null** y **E**.
- Se aplicarán todos los recibos que ya fueron antes creados

- El la sumatoria del monto de cada factura no deberá pasar el monto del recibo.

6. Diseño SQL

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

6.1 Objetos de Base de Datos

NOMBRE DE OBJETO	TIPO	DESCRIPCION
INER_CARGA_PAQUER_PKG	PAQUETE	Paquete de extracción de datos para PAQUER
INER_XXER_CALL_FUSION_SOAP_PKG	PAQUETE	Paquete de llamada a web services Fusion SOAP
INER_XXER_STAGE_EXTRACT_PKG	PAQUETE	Paquete de extracción de datos desde ZAM
INER_XXER_UTILITIES_PKG	PAQUETE	Paquete de control para procesos de importación a Fusion
XXER_CARGA_ZAMDLV_PKG	PAQUETE	Paquete de extracción de datos para DLIVER
XXER_CARGA_ZAMTRA2_PKG	PAQUETE	Paquete de extracción de datos para TRATECSA
INER_XXER_AP_CONCEPTOS_LIQ_TBL	TABLA	Tabla de conceptos de liquidación válidos
INER_XXER_CNF_INTERFASE_INGRESOS_AR_TBL	TABLA	Tabla de configuración de interfaz
INER_XXER_DATA_SOAP_TBL	TABLA	Tabla de credenciales para conexión SOAP
INER_XXER_GL_CODE_COMB_TBL	TABLA	Tabla de combinaciones contables validas para Fusion
INER_XXER_PUENTE_AUX_TBL	TABLA	Tabla de interfaz con datos de facturas electronicas
INER_XXER_PUENTE_HEADER_TBL	TABLA	Tabla de interfaz con datos de encabezados
INER_XXER_PUENTE_LINES_TBL	TABLA	Tabla de interfaz con datos de lineas
INER_XXER_AR_INVOICES_V	VISTA	Vista de extracción de datos de facturas y notas de credito
INER_XXER_CNF_INT_INGRESOS_SEQ	SECUENCIA	Secuencia para tabla de configuración
INER_XXER_PUENTE_AUX_SEQ	SECUENCIA	Secuencia para tabla auxiliar
INER_XXER_PUENTE_HEADER_SEQ	SECUENCIA	Secuencia para tabla de encabezados
INER_XXER_PUENTE_LINES_SEQ	SECUENCIA	Secuencia para tabla de lineas
INER_XXER_STAGE_EXTRACT_JOB	JOB	Para extraer los datos desde ZAM
XXER_CONSOLIDA_LIQ_DLIVER_JOB	JOB	Para procesar los datos de Liquidacion de DLIVER
XXER_CONSOLIDA_LIQ_TRATECSA_JOB	JOB	Para procesar los datos de Liquidacion de TRATECSA
XXER_EJECUTAR_TODOS_DLIVER_JOB	JOB	Para procesar las Facturas, NC y Recibos de DLIVER y los datos de PAQUER
XXER_EJECUTAR_TODOS_TRATECSA_JOB	JOB	Para procesar las Facturas, NC y Recibos de TRATECSA

7. Reglas de Negocio

7.1 Facturas y Notas de Crédito

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

7.2 Aplicación de Notas de Crédito

- Validar el estado de las notas de crédito por aplicar. Estas deben tener estado exitoso (S) y el numero de operación de aplicación debe estar en nulo (adicional3)

7.3 Recibos

Las reglas de negocio para esta integración es la siguiente:

- Validar la creación de un recibo y su respectiva aplicación y que se aplique a varias facturas siempre y cuando se verifique el estatus del recibo.
- Validar el caso de error en el cual no se genere el recibo y se envíe la notificación del error y en la tabla de superinterfaz se actualice el status E
- Validar un recibo que haya tenido un error y se corrige el registro en la tabla se puede reprocesar
- El recibo se esté creando con las cuentas correctas de acuerdo al método de recibo de origen(Superinterfaz)
- Las siguientes columnas dentro de estas tablas son requeridas para la Creación y Aplicación de los recibos.
 - ✓ INER_XXER_CNF_INTERFASE_INGR
 ESOS_AR:
 RECEIPT_METHOD_ID
 ORGANIZATION_ID
 TIPO_DOCUMENTO =
 'RECIBO'
 - ✓ INER_XXER_TBL_PUENTE_HEADER:
 TABLA_PUENTE_ID
 SISTEMA_ORIGEN
 TIPO_DOCUMENTO =
 'RECIBO' TIPO_MOVIMIENTO
 UNIDAD_NEGOCIO_ORIGEN
 RFC_DESTINO
 REFERENCIA_APLICACION

ESTATUS

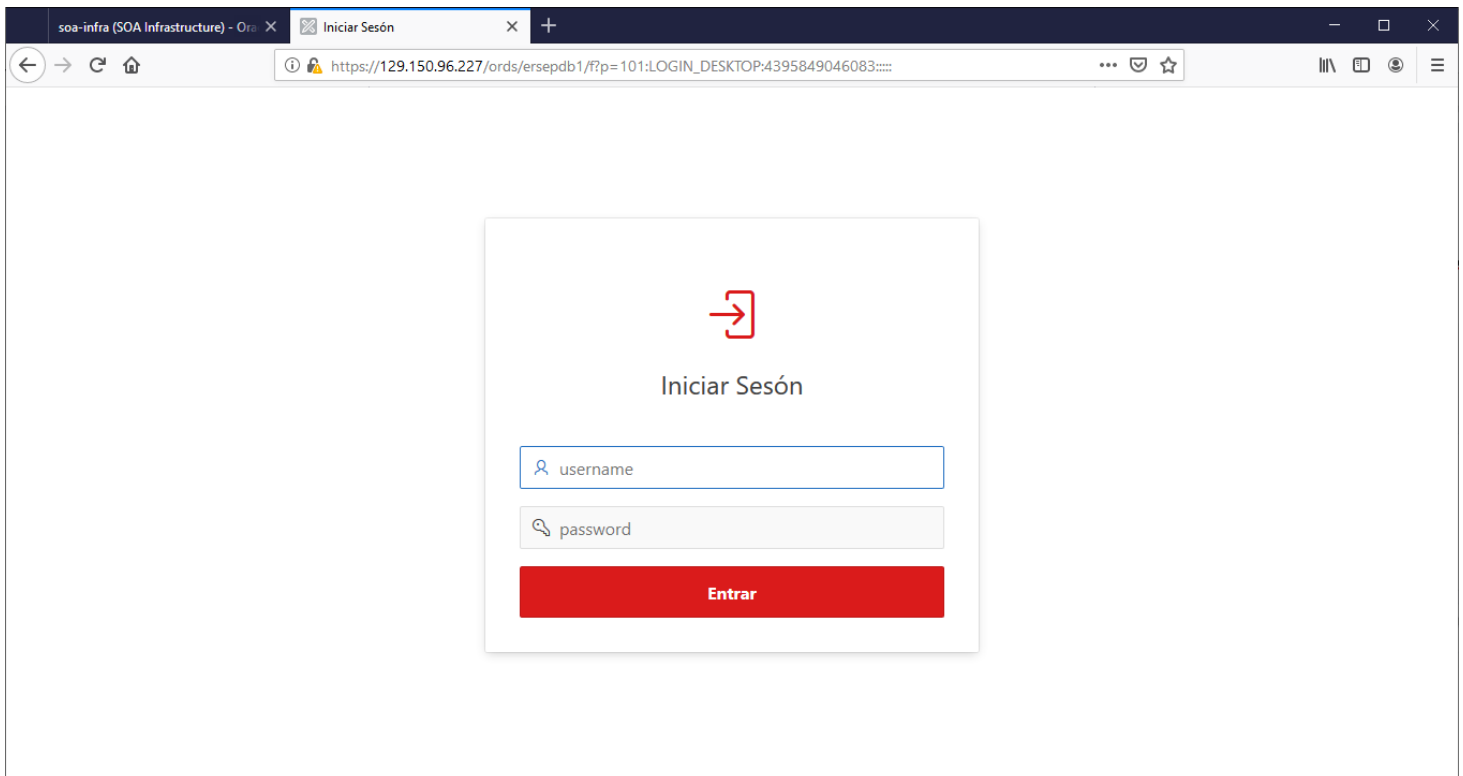
- ✓ INER_XXER_TBL_PUENTE_LINES:
 - TABLA_PUENTE_LIN
 - EA_ID
 - TABLA_PUENTE_ID
 - TOTAL

8. Diseño del Servicio

8.1 Pantalla de configuración

Para poder cargar reglas de extracción es necesario configurar los registros desde la pantalla de configuración que se estableció para ello:

https://129.150.96.227/ords/ersepdb1/f?p=107:LOGIN_DESKTOP:4395849046083:::



8.2 Registros de configuración

	Id Interfaz	Unidad Operativa	Sistema Origen	Tipo Documento	Tipo Movimiento	Coordinado	Impuesto	Origen Transaccion	Tipo Transaccion Ar	Metodo Recibo	Unidad Negocio Origen	Nombre Fiscal Origen	Empresa Origen
	7250	VERSA_BU	DLIVER	LIQUIDACION	PROPIO	-	TAS-IVAAP16%	-	-	-	-	-	-
	7249	VERSA_BU	DLIVER	LIQUIDACION	PROPIO	-	TAS-IVAAP16%	-	-	-	-	-	-
	7248	TRATECSA_BU	TRATECSA	LIQUIDACION	PROPIO	-	TAS-IVAAP16%	-	-	-	-	-	-
	3784	VERSA_BU	DLIVER	LIQUIDACION	PROPIO	-	TAS-IVAAP16%	-	-	-	-	-	-
	3783	VERSA_BU	DLIVER	LIQUIDACION	PROPIO	-	TAS-IVAAP16%	-	-	-	-	-	-
	3782	VERSA_BU	DLIVER	LIQUIDACION	PROPIO	-	TAS-IVAAP16%	-	-	-	-	-	-
	3781	VERSA_BU	DLIVER	LIQUIDACION	PROPIO	-	TAS-IVAAP16%	-	-	-	-	-	-
	3780	VERSA_BU	DLIVER	LIQUIDACION	PROPIO	-	TAS-IVAAP16%	-	-	-	-	-	-

8.3 Facturas y Notas de Crédito

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

nfraestructura SOA > default (Carpeta SOA) > InvoicesEnt (Compuesto) > Probar Servicio Web

Servicio: InvoicesEnt

Puerto: InvoicesEntPort

Operación: GetInvoices

URL de Punto Final: http://ersoacsprod-wls-1:9073/soa-infra/services/default/InvoicesEnt/InvoicesEnt

Editar URL de Punto Final

Solicitud Respuesta

Seguridad

Calidad del Servicio

Cabecera HTTP

Opciones de Prueba Adicionales

Argumentos de Entrada

Vista de Árbol Activar Validación Realizar Carga Útil Seleccionar archivo Ningún archivo seleccionado Guardar Carga Útil

Cuerpo SOAP

Ver Separar

Nombre	Tipo	Valor
* GetInvoicesRq	GetInvoicesRqType	

Solicitud Respuesta

Probar Servicio Web

8.4 Aplicación de Notas de Crédito

The screenshot displays the Oracle Enterprise Manager Fusion Middleware Control 12c interface. The breadcrumb navigation shows the path: SOA Infrastructure > INERDEV (SOA Folder) > CreditMemoApplyEnt (Composite) > Test Web Service. The 'Service' dropdown is set to 'creditmemoapplybpel_client_ep' and the 'Port' is 'CreditMemoApplyBPel_pt'. The 'Operation' dropdown is set to 'process'. The 'Endpoint URL' is 'http://ersocspod2019-wls-1:9073/soa-infra/services/INERDEV/CreditMemo'. The 'Request' tab is active, showing a tree view of the request structure. The 'SOAP Body' section is expanded, showing a table with columns 'Name', 'Type', and 'Value'. The 'Value' column for the 'input' row is highlighted with a red box.

Name	Type	Value
* payload	payload	
* input	string	

8.5 Recibos

Test Web Service: soa-infra [SOA In] X

https://129.150.99.167:7002/em/faces/as-wsmgt-wstest?inputWSDLURL=http%3A%2F%2Fersoacsprod2019-wls-1%3A9073%2Fsoa-infra%

ORACLE Enterprise Manager Fusion Middleware Control 12c

WebLogic Domain weblogic

Jul 11, 2019 5:26:04 PM UTC

soa-infra SOA Infrastructure

SOA Infrastructure > INERDEV (SOA Folder) > ReceiptEnt (Composite) > Test Web Service

Service ReceiptEnt

Port ReceiptEntPort

Operation CreateReceipts

Endpoint URL http://ersoacsprod2019-wls-1:9073/soa-infra/services/INERDEV/ReceiptEnt/ Edit Endpoint URL

Request Response

Security

Quality of Service

HTTP Header

Additional Test Options

Input Arguments

Tree View

Enable Validation

Load Payload

Browse

No file selected.

Save Payload

SOAP Body

View Detach

Name	Type	Value
* CreateReceiptsRq	CreateReceiptsRqType	

Request Response

Test Web Service

9. 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.

9.1 Estrategia de Reinicio

- Se puede 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.

9.2 Seguridad

N/A

9.3 Personalización

- En caso de requerir modificar el proyecto, tomar la versión de código fuente adjunta en la documentación.
 - Revisar que los servicios 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
-

10. 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:

- 10.1 INER_XXER_PUENTE_HEADER_TBL
- 10.2 INER_XXER_PUENTE_LINES_TBL
- 10.3 INER_XXER_PUENTE_AUX_TBL
- 10.4 INER_XXER_CNF_INTERFASE_INGRESOS_AR_TBL

En este esquema debe existir los siguientes paquetes:

- 10.5 INER_CARGA_PAQUER_PKG
- 10.6 INER_XXER_CALL_FUSION_SOAP_PKG
- 10.7 INER_XXER_STAGE_EXTRACT_PKG
- 10.8 INER_XXER_UTILITIES_PKG
- 10.9 XXER_CARGA_ZAMDLV_PKG
- 10.10 XXER_CARGA_ZAMTRA2_PKG

En este esquema deben existir los tipos:

- 10.11 XXER_AR_GETDATA_INVOICES_TB_TYP
- 10.12 XXER_AR_GETDATA_INVOICES_TYP
- 10.13 XXER_AR_PARAMS_UPD_INV_TYPE
- 10.14 XXER_AR_PARAMS_UPD_INV_TTYPE
- 10.15 XXER_AR_RECEIPT_INVOICE_TTYPE
- 10.16 XXER_AR_RECEIPT_INVOICE_TYPE
- 10.17 XXER_COMPLEMENT_UPD_UUID_TTYPE
- 10.18 XXER_COMPLEMENT_UPD_UUID_TYPE
- 10.19 XXER_ERROR_MESSAGES_INV_TTYPE
- 10.20 XXER_ERROR_MESSAGES_INV_TYPE
- 10.21 XXER_INVOICE_LOAD_STATUS_TTYPE
- 10.22 XXER_INVOICE_LOAD_STATUS_TYPE
- 10.23 XXER_INVOICES_TO_UPDATE_TTYPE
- 10.24 XXER_INVOICES_TO_UPDATE_TYPE
- 10.25 XXER_AR_APPLY_CM_TYPE
- 10.26 XXER_AR_APPLY_CM_COLLTYPE

Deben de existir las siguientes vistas:

- 10.27 INER_XXER_AR_INVOICES_V
- 10.28 INER_XXER_AR_CUSTOMER_RFC_V

10.1 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:

Facturas y Notas de Crédito

sca_InvoicesEnt.jar
sca_InvoicesBiz.jar
sca_ERPManagementOfLoadImportBiz.jar

Aplicación de Notas de Crédito

sca_CreditMemoApplyEnt

Recibos

sca_ReceiptEnt.jar

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

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:



Script SQL.zip



SOA Composite.zip

11. URL de Acceso y seguridad

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

Servicio	URL
InvoicesEnt	http://ersoacsprod2019-wls-1:9073/soa-infra/services/INER/InvoicesEnt/InvoicesEnt?WSDL
ReceiptEnt	http://ersoacsprod2019-wls-1:9073/soa-infra/services/INER/ReceiptEnt/ReceiptEnt?WSDL
CreditMemoApplyEnt	http://ersoacsprod2019-wls-1:9073/soa-infra/services/INER/CreditMemoApplyEnt/creditmemoapplybpel_client_ep?WSDL

12. Temas abiertos y cerrados

12.1 Temas Abiertos

ID	Tema	Solución	Responsabilidad	Fecha Objetivo	Fecha impacto

12.2 Temas Cerrados

ID	Tema	Solución	Responsabilidad	Fecha Objetivo	Fecha impacto