

OUM

TA.070-4 DISEÑO TÉCNICO DE EXTENSIONES E INTERFACES

Grupo Estrella Roja

ER01_Lectura de Kilómetros

Autor:	Jesús Alberto Suárez Xelo
Fecha de creación:	abril 17, 2020
Última actualización:	mayo 12, 2020
Código de referencia OUM:	NA
Versión:	2.3

Aprobadores:

Rodrigo Pardo Cue

ORACLE

i. Control de Documento

i.1 Registro de cambios

Fecha	Autor	Versión	Referencia del cambio
17-abr-2020	Jesús Alberto Suárez Xelo	Draft	Sin documento previo
23-abr-2020	Josué Ayala Carro	1.0	Draft
23-abr-2020	Uriel Díaz Austria	1.1	Revisión
29-abr-2020	Jesús Alberto Suárez Xelo	2.0	Integración componentes Service Bus
29-abr-2020	Uriel Díaz Austria	2.1	Revisión
04-may-2020	Jesús Alberto Suárez Xelo	2.2	Integración componentes Service Bus
04-may-2020	Uriel Díaz Austria	2.2	Revisión
08-may-2020	Jesús Alberto Suárez Xelo	2.3	Integración flujos Service Bus
12-may-2020	Uriel Díaz Austria	2.3	Revisión

i.2 Revisores

Nombre	Cargo
Jose Pedro Hernández Romero	Project Manager
María Edith Ramírez Ramos	Líder funcional
Gregorio Flores Cabanzo	Consultor funcional
Juan González Ramírez	Gerencia de desarrollo

Contenido

i.	Control de Documento.....	2
i.1	Registro de cambios.....	2
i.2	Revisores.....	2
1.	Resumen técnico.....	4
1.1	Diagrama de la integración.....	5
2	Lógica de implementación.....	6
2.1	RecordUnitMetrics.....	6
2.1.1	RecordUnitMetrics.....	8
2.2	Resultados de prueba de bloqueo/desbloqueo de unidades ZAM.....	22
2.3	Componentes Service Bus.....	23
2.4	APIUnitMetricsTec.....	23
2.5	UtilityTec.....	32
2.6	ERPUnitMetricsTec.....	35
2.7	SCMWorkOrdersTec.....	37
2.8	EAMUnitMetricsTec.....	38
3	Diseño de Datos.....	42
3.1	Origen de Datos.....	42
4	Reglas de Negocio.....	43
5	Consideraciones de Rendimiento.....	43
5.1	Estrategia de Reinicio.....	43
5.2	Seguridad.....	43
5.3	Personalización.....	43
6	Consideraciones de Instalación.....	43
6.1	Instalación de integración creación de transacciones.....	44
7	URL de Acceso y seguridad.....	44
8	Temas abiertos y cerrados.....	45
8.1	Temas Abiertos.....	45
8.2	Temas Cerrados.....	45

1. Resumen técnico

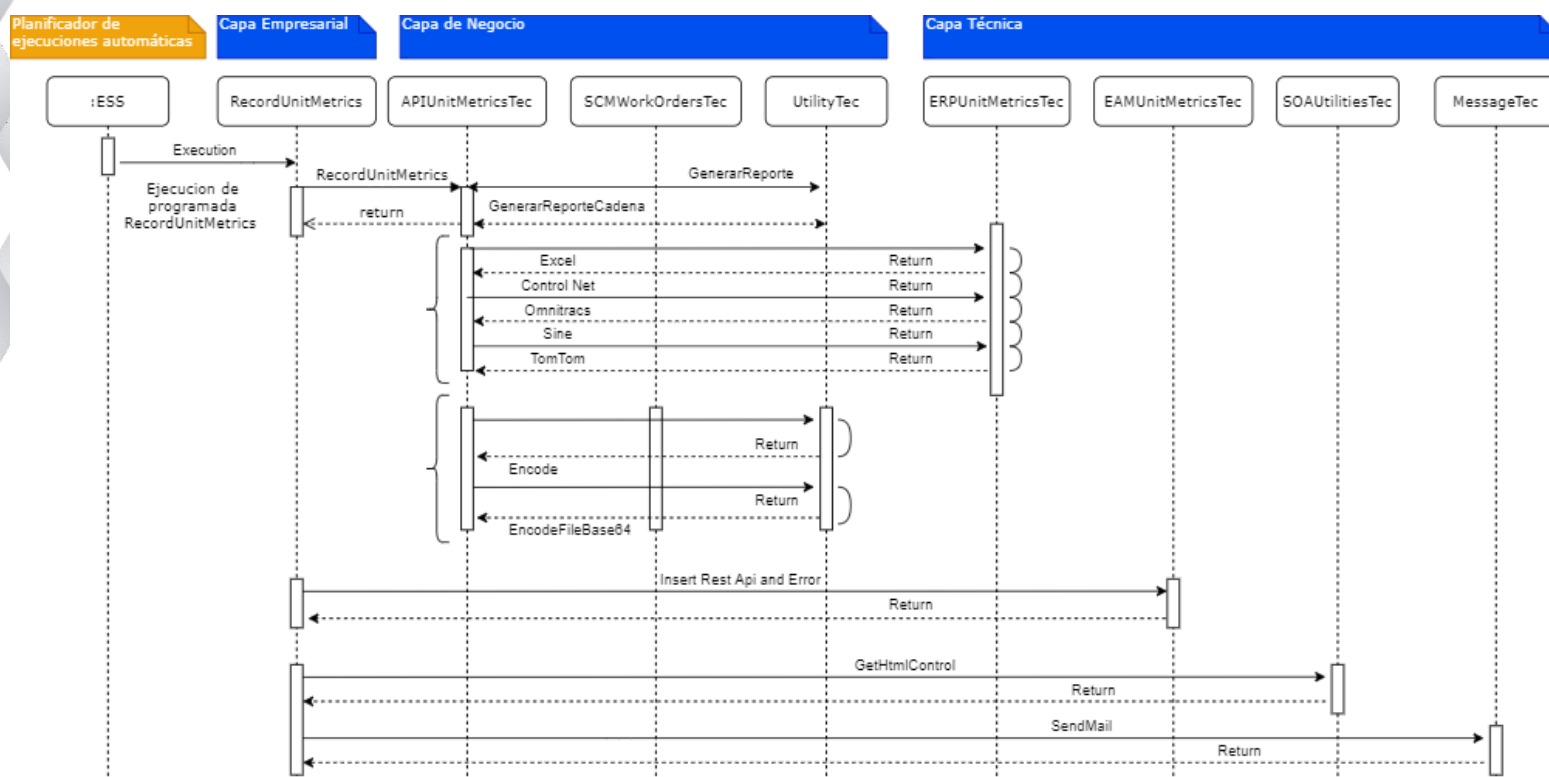
Este documento define los componentes técnicos requeridos para implementar el proceso personalizado de “Lectura de Kilómetros”. Este documento es una extensión técnica del documento de diseño funcional “MD050–Lectura de Kilómetros” y se deberá considerar para completar el diseño detallado.

El requerimiento de Estrella Roja (ER) referente a la Lectura de kilómetros es identificar mediante la consulta a los diferentes orígenes de datos como lo son; Excel, Omnitrac, TomTom, Sine y ControlNet el respectivo kilometraje por día que recorren las unidades para así registrarlo en Oracle Fusion y puedan ser planificados o alertados cuando una unidad requiera de mantenimiento.

1.1 Diagrama de la integración

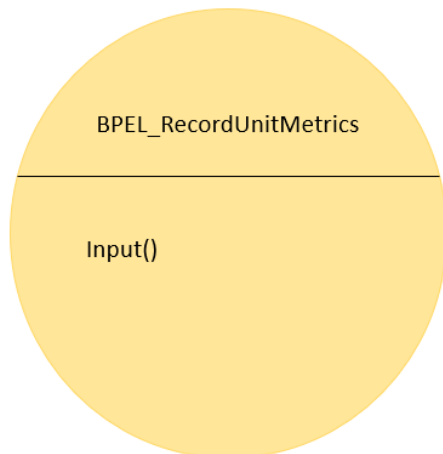
La interfaz **RecordUnitMetrics** concentrará las funciones de negocio principales para registro de la lectura de kilómetros en Oracle Fusion las cuales se obtienen de los diferentes orígenes de datos.

El principal objetivo de la interfaz es orquestar y llamado y validaciones de cada lectura obtenida para así poder registrarse en Fusion y determinar cuándo una unidad de transporte requiere de un mantenimiento, el intercambio de mensajes entre las principales capas de la arquitectura se muestra en el siguiente flujo:



2 Lógica de implementación

El nombre de este servicio es **BPEL_RecordUnitMetrics**, por su función de negocio se clasifica como un servicio de orquestación.

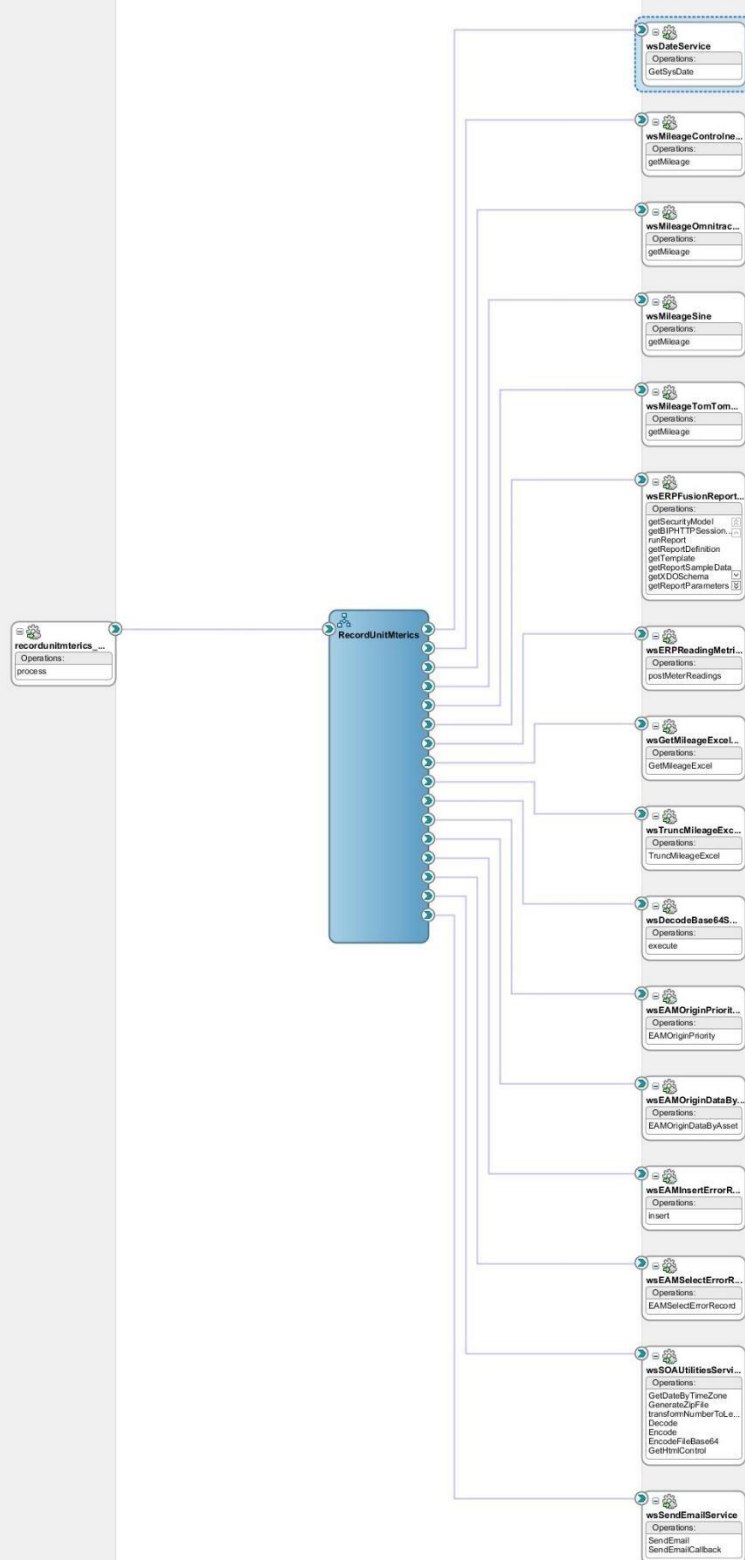


Las operaciones que provee el servicio son las siguientes:

input (): esta operación se encarga de recibir con su lógica tecnológica los datos de entrada para filtrar las lecturas de kilómetros.

2.1 RecordUnitMetrics

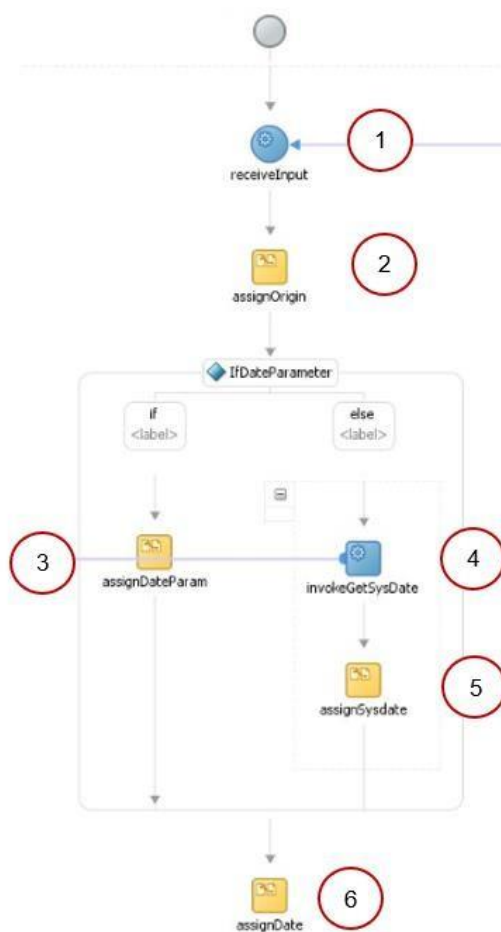
Dentro de la capa SOA se expone el servicio **RecordUnitMetrics** el cual expone la operación process, ésta a su vez, consume otros servicios comunes para completar el proceso de inicio del proceso de transacciones.



2.1.1 RecordUnitMetrics

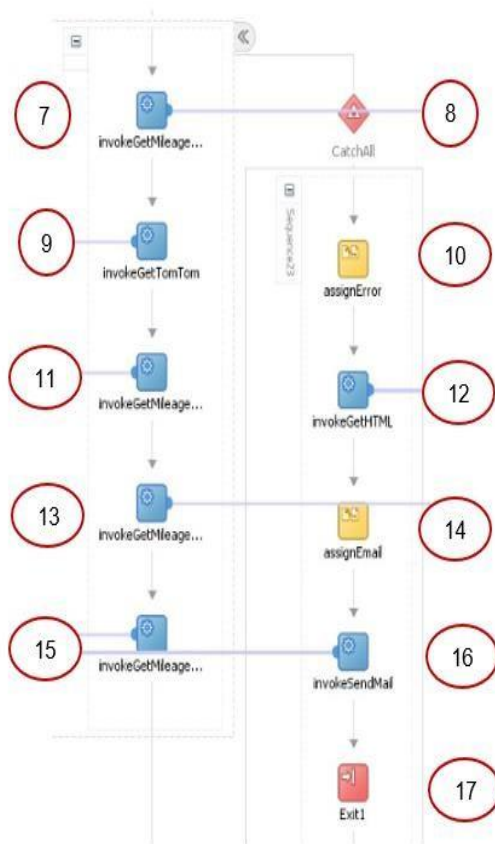
RecordUnitMetrics: Esta es la operación que se invoca en el proceso de lectura de Kilómetros, el BPEL que se usa es de tipo oneway a continuación se detallan los componentes que integran esta operación:

1. Se recibe mediante la operación process, los parámetros para la lectura de Kilómetros, Origen, FechaInicio y FechaFin.
2. Se asigna el origen el cual se consultará para obtener las lecturas de kilómetros.
3. Si en los parámetros de entrada del servicio vienen los parámetros de fechas se asignan a las variables para la consulta de lecturas.
4. Si en los datos de entrada no viene establecido la fecha de inicio y fin, el proceso invoca al servicio **GetSysDatePS** para obtener la fecha de sistema.
5. Se asignan las fechas que se obtienen de la consulta de sysdate.
6. Se asignan las fechas de inicio y fin a las variables de entrada para la consulta de lecturas por origen.



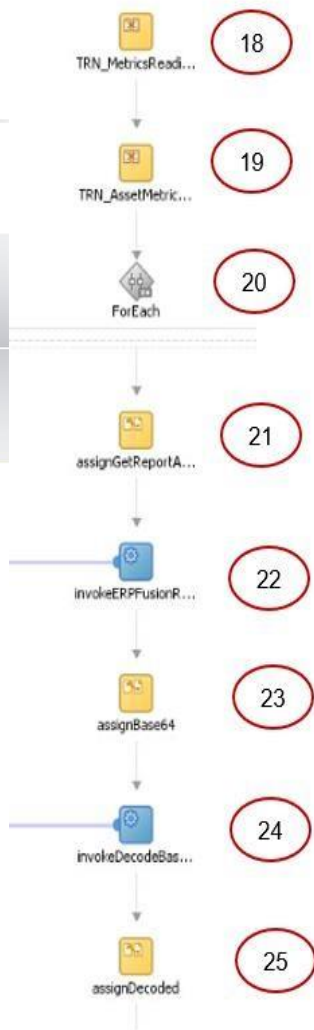
7. Se invoca el servicio GetMileageControlNetPS para obtener las lecturas de kilómetros.

8. Si ocurre una excepción en el llamado de los servicios se obtiene el error.
9. Se invoca el servicio **GetMileageTomTomPS** para obtener las lecturas de kilómetros.
10. Se asigna el mensaje de error para notificar que ocurrió un fallo en el procesamiento.
11. Se invoca el servicio **GetMileageOmnitracsPS** para obtener las lecturas de kilómetros.
12. Se invoca el servicio **SOAUtilitiesTec** para obtener la plantilla de correo que se enviará con el detalle del fallo.
13. Se invoca el servicio **GetMileageSinePS** para obtener las lecturas de kilómetros.
14. Se asigna la plantilla de correo HTML para el envío de notificación de error.
15. Se invoca el servicio **GetMileageExcelPS** para obtener las lecturas de kilómetros.
16. Se invoca el servicio **MessageTec** para el envío de correo electrónico con el detalle de procesamiento.
17. Se interrumpe el procesamiento de las lecturas al recibir un fallo en la consulta de las mismas en alguno de los orígenes.



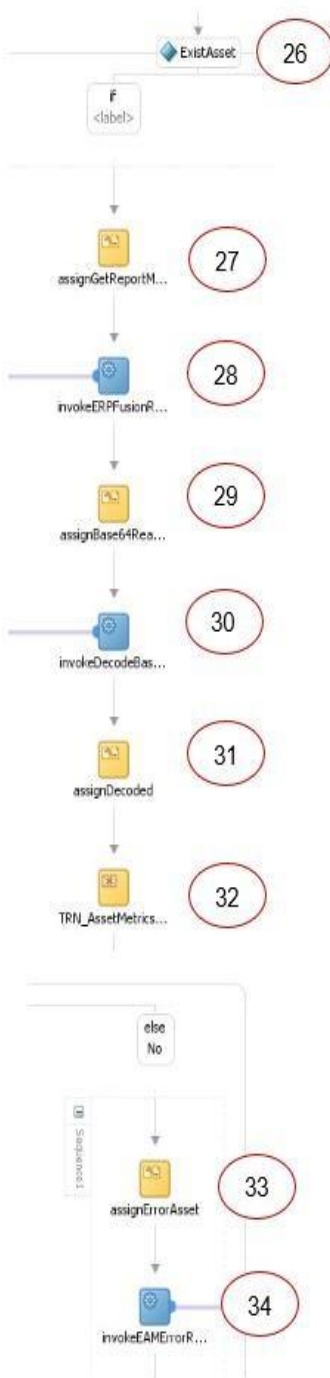
18. Se realiza la preparación de datos mediante la transformación **Tr_MetricsReadingResponse.xsl** la cual permite ordenar los valores máximos por activo de cada lectura recibida de los orígenes consultados.
19. Se asignan los datos en una estructura de mensaje XML estándar para el procesamiento de lecturas mediante la transformación **Tr_AssetMetricReadingsResponse.xsl**.
20. Se inicia el ciclo de procesamiento para cada una de las lecturas.
21. Se asignan los valores de consulta en Oracle Fusion para obtener el activo que se está procesando.
22. Se invoca el servicio **WorkOrdersPS** para obtener el activo consultado.

23. Se asigna el mensaje Base64 que se obtuvo de la consulta del activo.
24. Se invoca el servicio **DecodeBase64PS** para decodificar el mensaje Base64.
25. Se asigna el mensaje de tipo string del activo consultado.



26. Mediante un IF se valida si el activo consultado existe en Oracle Fusion.
27. Si el activo existe en Oracle Fusion se asignan los parámetros de consulta para obtener las lecturas que tiene asociadas el activo en Oracle Fusion.
28. Se invoca el servicio **WorkOrdersPS** para obtener las lecturas del activo.
29. Se asigna el mensaje Base64 que se obtuvo de la consulta del activo.
30. Se invoca el servicio **DecodeBase64PS** para decodificar el mensaje Base64.
31. Se asigna el mensaje de tipo string del activo consultado.
32. Se asigna el mensaje a una estructura XML de las lecturas obtenidas **Tr_AssetMetricsResponse.xml**.
33. Si el activo no existe en Oracle Fusion se asignan los parámetros para guardar el activo procesado en la bitácora.

34. Se invoca el servicio **EAMInsertErrorRecordPS** para guardar el estado de procesamiento del activo.



35. Mediante un IF se valida si el activo consultado está asociado a un medidor.

36. Si el activo está asociado a un medidor, mediante un segundo IF se valida que el medidor se encuentre activo en Oracle Fusion.

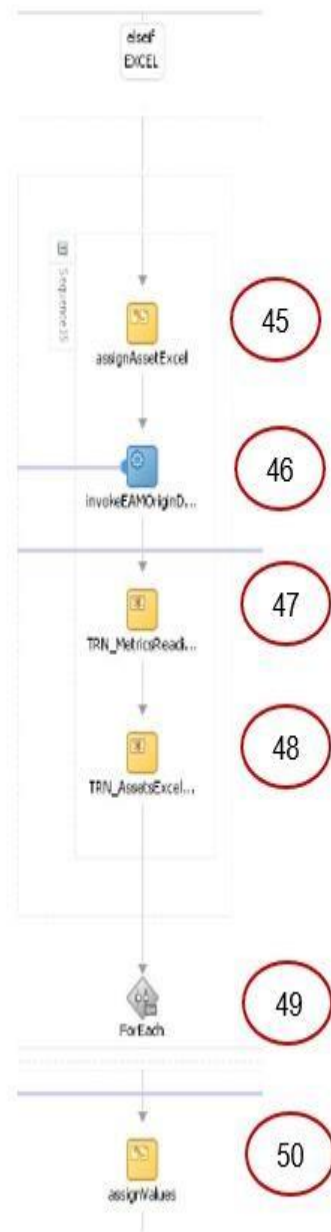
37. Si el medidor no se encuentra activo se asignan los parámetros para guardar el activo procesado en la bitácora.
38. Se invoca el servicio **EAMInsertErrorRecordPS** para guardar el estado de procesamiento del activo.
39. Si el medidor se encuentra activo, se asignan los parámetros de consulta para obtener la prioridad del origen.
40. Se invoca el servicio **EAMOriginPriorityPS** para obtener la prioridad del origen del activo procesado.



41. Si el medidor no se encuentra activo se asignan los parámetros para guardar el activo procesado en la bitácora.
42. Se invoca el servicio **EAMInsertErrorRecordPS** para guardar el estado de procesamiento del activo.
43. Mediante un IF se valida si el activo se encuentra asociado a un origen con prioridad.
44. Si el activo está asociado a un origen con prioridad, se valida que solo este asignado a un origen con prioridad.

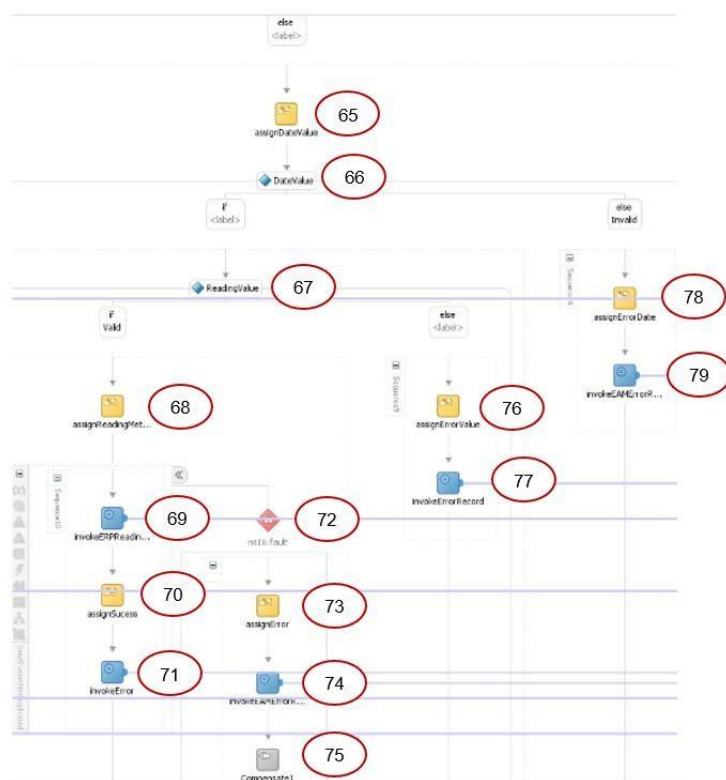


45. Si el activo no se encuentra asociado a un origen con prioridad y su origen es Excel se asignan los valores para consultar el activo en la BBDD.
46. Se consulta en la BBDD el activo.
47. Se asigna mediante una transformación **Tr_MetricsReadinExcelResponse.xsl** los datos consultados.
48. Mediante la transformación **Tr_AssetsExcelResponse.xsl** se asigna el mensaje a una estructura XML estándar para el procesamiento de lecturas.
49. Se inicia un ciclo para procesar cada una de las lecturas obtenidas.
50. Se asignan los valores a variables de los registros máximos de las lecturas por activo.



51. Mediante un IF se valida que la fecha procesada sea mayor a la última registrada en Oracle Fusion.
52. Si la fecha es mayor, se valida mediante un segundo IF que la lectura procesada sea mayor a la última registrada en Oracle Fusion.
53. Si la lectura es mayor a la registrada en Fusion, se asignan los valores de la lectura procesada para registrarla en Oracle Fusion.
54. Se invoca el servicio **ERPMeterReadingsPS** para registrar la lectura procesada en Oracle Fusion.
55. Si el registro de la lectura se realiza con éxito se asigna los valores para insertarlos en la bitácora.

67. Si la fecha es mayor, se valida mediante un segundo IF que la lectura procesada sea mayor a la última registrada en Oracle Fusion.
68. Si la lectura es mayor a la registrada en Fusion, se asignan los valores de la lectura procesada para registrarla en Oracle Fusion.
69. Se invoca el servicio **ERPMeterReadingsPS** para registrar la lectura procesada en Oracle Fusion.
70. Si el registro de la lectura se realiza con éxito se asigna los valores para insertarlos en la bitácora.
71. Se invoca el servicio **EAMInsertErrorRecordPS** para registrar la lectura procesada en la bitácora.
72. Se maneja un control de excepción cuando el servicio **ERPMeterReadingsPS** responda con error en el registro de la lectura en Oracle Fusion.
73. Se asignan los valores de la lectura con el error de registro en Oracle Fusion para registrarlo en la bitácora de procesamiento.
74. Se invoca el servicio **EAMInsertErrorRecordPS** para registrar la lectura procesada en la bitácora.
75. Se añade un punto de deshidratación en el servicio para que aún con el fallo siga procesando las demás lecturas por activo.
76. Si la lectura procesada es menor a la última registrada en **Fusion** se asignan los valores de la lectura con el error de validación en la fecha para registrarlo en la bitácora de procesamiento.
77. Se invoca el servicio **EAMInsertErrorRecordPS** para registrar la lectura procesada en la bitácora.
78. Si la fecha de la lectura procesada es menor a la última registrada en **Fusion** se asignan los valores de la lectura con el error de validación en la lectura para registrarlo en la bitácora de procesamiento.
79. Se invoca el servicio **EAMInsertErrorRecordPS** para registrar la lectura procesada en la bitácora.

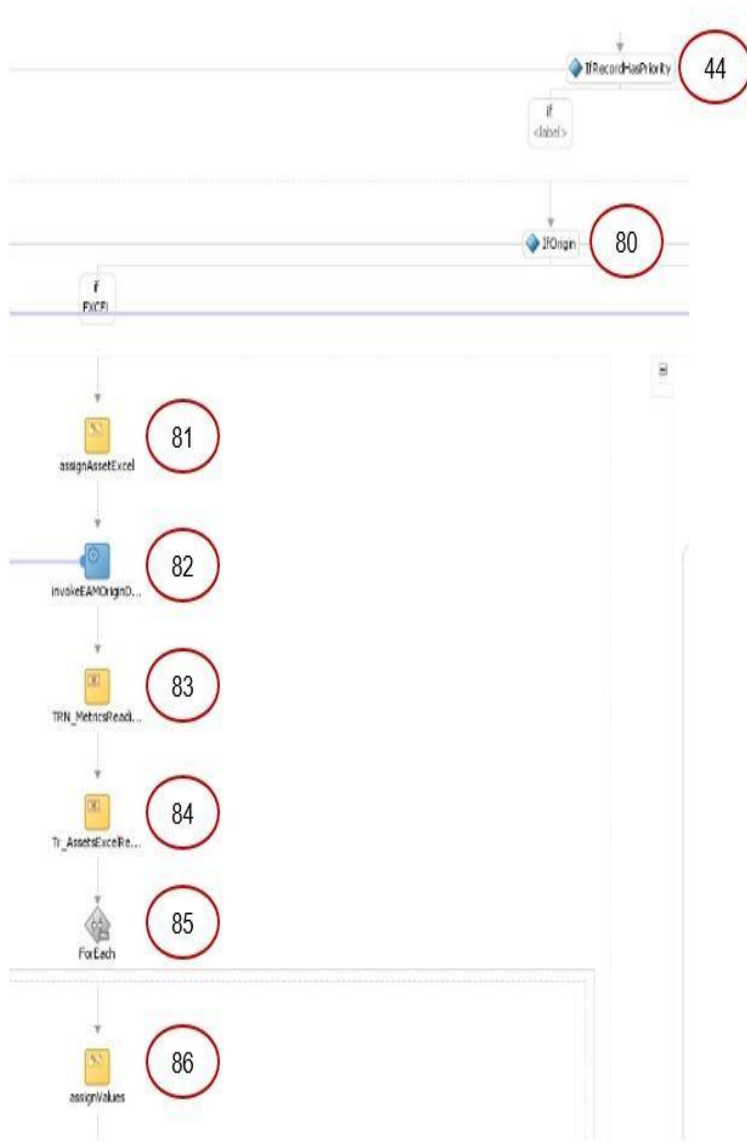


80. Mediante un IF se valida si el origen con prioridad es Excel.
81. Si el origen con prioridad del activo es Excel se asignan los valores para consultar el activo en la BBDD.
82. Se consulta en la BBDD el activo.
83. Se asigna mediante una transformación **Tr_MetricsReadinExcelResponse.xsl** los datos consultados.
84. Mediante la transformación **Tr_AssetsExcelResponse.xsl** se asigna el mensaje a una estructura XML estándar.

para el procesamiento de lecturas.

85. Se inicia un ciclo para procesar cada una de las lecturas obtenidas.

86. Se asignan los valores a variables de los registros máximos de las lecturas por activo.



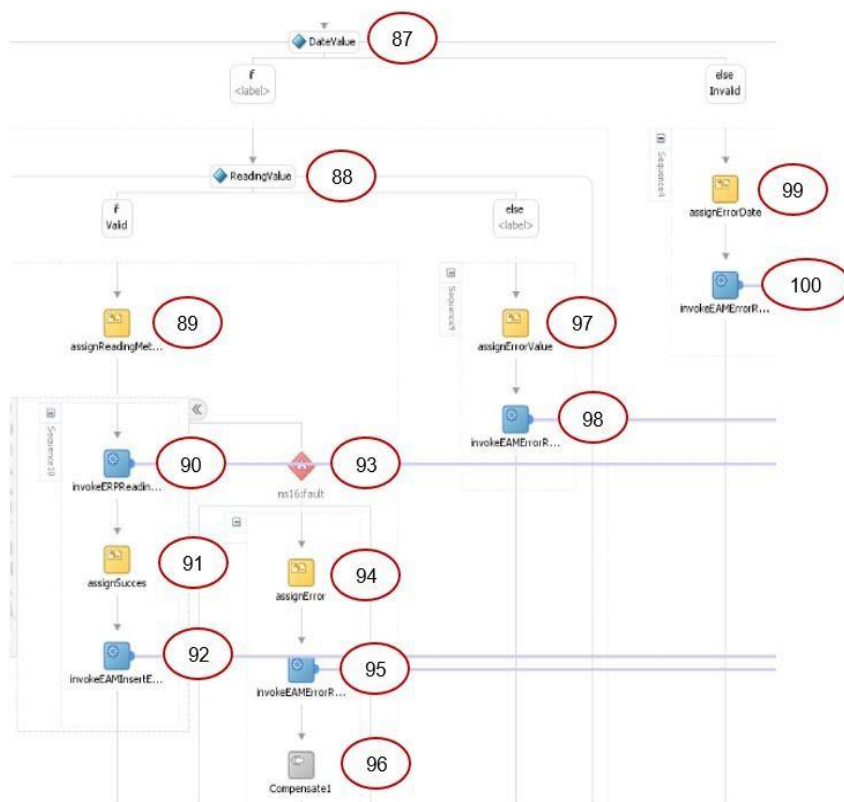
87. Mediante un IF se valida que la fecha procesada sea mayor a la última registrada en Oracle Fusion.

88. Si la fecha es mayor, se valida mediante un segundo IF que la lectura procesada sea mayor a la última registrada en Oracle Fusion.

89. Si la lectura es mayor a la registrada en Fusion, se asignan los valores de la lectura procesada para registrarla en Oracle Fusion.

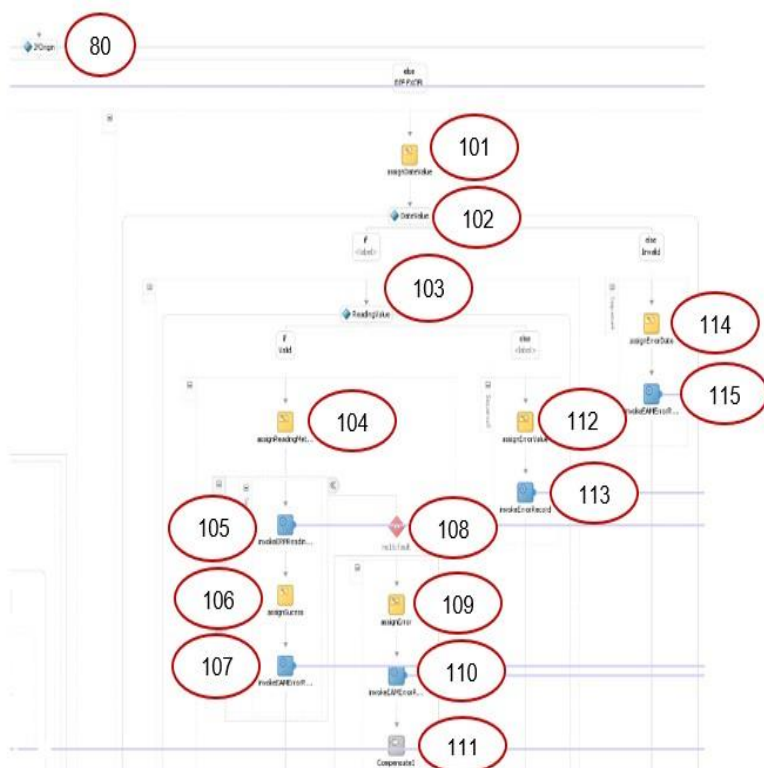
90. Se invoca el servicio **ERPMeterReadingsPS** para registrar la lectura procesada en Oracle Fusion.

91. Si el registro de la lectura se realiza con éxito se asigna los valores para insertarlos en la bitácora.
92. Se invoca el servicio **EAMInsertErrorRecordPS** para registrar la lectura procesada en la bitácora.
93. Se maneja un control de excepción cuando el servicio **ERPMeterReadingsPS** responda con error en el registro de la lectura en Oracle Fusion.
94. Se asignan los valores de la lectura con el error de registro en Oracle Fusion para registrarlo en la bitácora de procesamiento.
95. Se invoca el servicio **EAMInsertErrorRecordPS** para registrar la lectura procesada en la bitácora.
96. Se añade un punto de deshidratación en el servicio para que aún con el fallo siga procesando las demás lecturas por activo.
97. Si la lectura procesada es menor a la última registrada en **Fusion** se asignan los valores de la lectura con el error de validación en la fecha para registrarlo en la bitácora de procesamiento.
98. Se invoca el servicio **EAMInsertErrorRecordPS** para registrar la lectura procesada en la bitácora.
99. Si la fecha de la lectura procesada es menor a la última registrada en **Fusion** se asignan los valores de la lectura con el error de validación en la lectura para registrarlo en la bitácora de procesamiento.
100. Se invoca el servicio **EAMInsertErrorRecordPS** para registrar la lectura procesada en la bitácora.

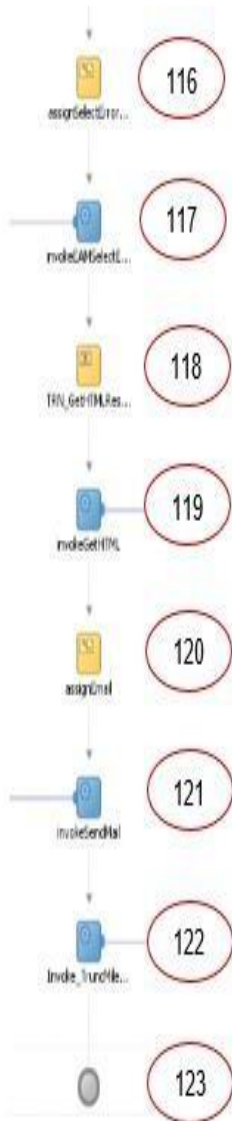


101. Mediante un IF se valida si el origen con prioridad es diferente de Excel.
102. Mediante un IF se valida que la fecha procesada sea mayor a la última registrada en Oracle Fusion.
103. Si la fecha es mayor, se valida mediante un segundo IF que la lectura procesada sea mayor a la última registrada en Oracle Fusion.
104. Si la lectura es mayor a la registrada en Fusion, se asignan los valores de la lectura procesada para registrarla en Oracle Fusion.
105. Se invoca el servicio **ERPMeterReadingsPS** para registrar la lectura procesada en Oracle Fusion.
106. Si el registro de la lectura se realiza con éxito se asigna los valores para insertarlos en la bitácora.
107. Se invoca el servicio **EAMInsertErrorRecordPS** para registrar la lectura procesada en la bitácora.

108. Se maneja un control de excepción cuando el servicio **ERPMeterReadingsPS** responda con error en el registro de la lectura en Oracle Fusion.
109. Se asignan los valores de la lectura con el error de registro en Oracle Fusion para registrarlo en la bitácora de procesamiento.
110. Se invoca el servicio **EAMInsertErrorRecordPS** para registrar la lectura procesada en la bitácora.
111. Se añade un punto de deshidratación en el servicio para que aún con el fallo siga procesando las demás lecturas por activo.
112. Si la lectura procesada es menor a la última registrada en **Fusion** se asignan los valores de la lectura con el error de validación en la fecha para registrarlo en la bitácora de procesamiento.
113. Se invoca el servicio **EAMInsertErrorRecordPS** para registrar la lectura procesada en la bitácora.
114. Si la fecha de la lectura procesada es menor a la última registrada en **Fusion** se asignan los valores de la lectura con el error de validación en la lectura para registrarlo en la bitácora de procesamiento.
115. Se invoca el servicio **EAMInsertErrorRecordPS** para registrar la lectura procesada en la bitácora.



116. Se asignan la fecha de inicio y fin del procesamiento para obtener las lecturas procesadas.
117. Se invoca el servicio **EAMSelectErrorRecordPS** para obtener las lecturas procesadas.
118. Se prepara mediante la transformación **Tr_GetHTMLResponse.xsl** el número de lecturas procesadas con su respectivo resumen para generar el template HTML de envío de correo.
119. Se invoca el servicio **SOAUtilitiesTec** para generar el template HTML con el resumen de procesamiento.
120. Se asignan los valores con el template HTML generado.
121. Se invoca el servicio **MessageTec** para el envío de correo electrónico con el resumen de procesamiento.
122. Marca el fin del procesamiento.
123. Se añade un control de excepción general en el procesamiento por si ocurre un error inesperado.



2.1.1.1 Lista de Web Services

A continuación, se describen los webs services que intervienen en la orquestación del servicio para la extracción y envío de datos.

Web Service	Descripción
GetSysDatePS	Servicio utilitario para obtener la fecha de sistema
GetMileageControlnetPS	Extracción de la información capturada de los viajes realizados por activos con parámetros de fecha inicio y fecha fin.
GetMileageOmnitracsPS	Extracción de la información capturada de los viajes realizados por activos con parámetros de fecha inicio y fecha fin.
GetMileageSinePS	Extracción de la información capturada de los viajes realizados por activos con parámetros de fecha inicio y fecha fin.
GetMileageTomTomPS	Extracción de la información capturada de los viajes realizados por activos con parámetros de fecha inicio y fecha fin.
WorkOrdersPS	Servicio para validar si las unidades existen en Fusion.
ERPMeterReadingsPS	Servicio para el registro de lecturas de kilómetros en Fusion.
GetMileageExcelPS	Extracción de la información capturada de los viajes realizados por activos con parámetros de fecha inicio y fecha fin.
TruncMileageExcelPS	Servicio de datos para eliminarlos registros que se obtuvieron de la lectura de origen Excel.
DecodeBase64PS	Servicio utilitario para decodificar el reporte obtenido de Fusion.
EAMOriginPriorityPS	Servicio de datos que permite validar la prioridad de los activos.
EAMInsertErrorRecordPS	Servicio de datos que permite el registro en la bitácora de lectura de kilómetros.
EAMSelectErrorRecordPS	Servicio de datos que provee la información de las lecturas procesadas.
SOAUtilitiesTec	Servicio utilitario para generar el formato de correo electrónico.
MessageTec	Servicio utilitario que permite el envío de correo electrónico con el detalle de la lectura procesada.

2.2 Resultados de prueba de bloqueo/desbloqueo de unidades ZAM

Bitácora de lectura de kilometros

Notificación de Proceso exitoso:

Estimado Usuario(a)

Te informamos que hemos procesado tu solicitud con el siguiente resultado:

Total de éxitos: 0

Total de errores: 340

Total de registros procesados: 340

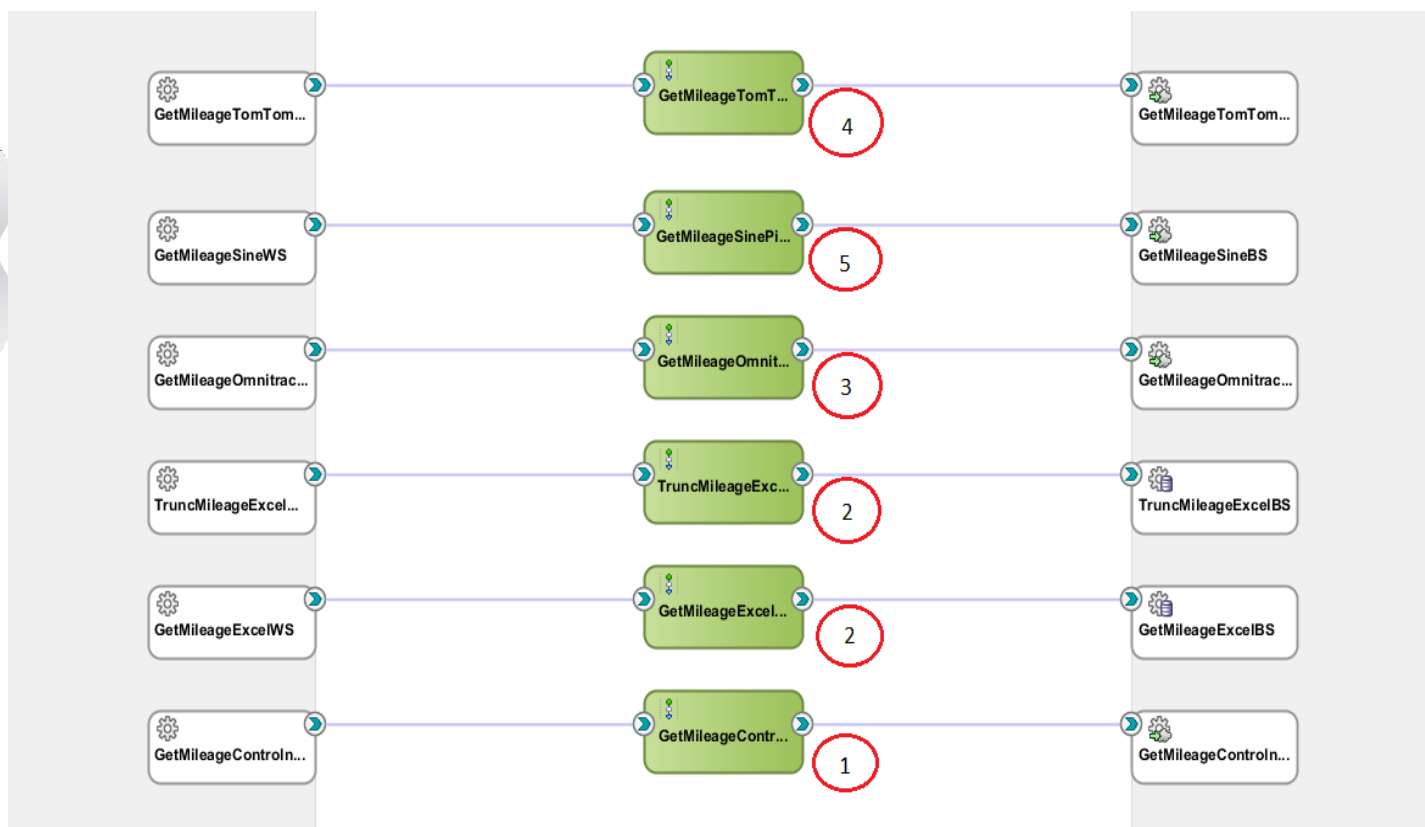
Llave	Mensaje de error
TSA016	Error de validación, la fecha de carga es menor al último registro existente en Oracle Fusion
TSA014	El activo no esta asociado a un medidor
904	El activo no se encuentra registrado en Oracle Fusion
9016	El activo no esta asociado a un medidor
9015	El activo no esta asociado a un medidor
9014	El activo no esta asociado a un medidor
9013	El activo no esta asociado a un medidor
9011	El activo no esta asociado a un medidor
9010	El activo no esta asociado a un medidor

2.1 Componentes Service Bus

En este apartado se hace mención de los Bus de servicios que fueron empleados para la integración de “Lectura de Kilómetros” así mismo como los componentes asociados a los mismos y su EndpointURI correspondiente.

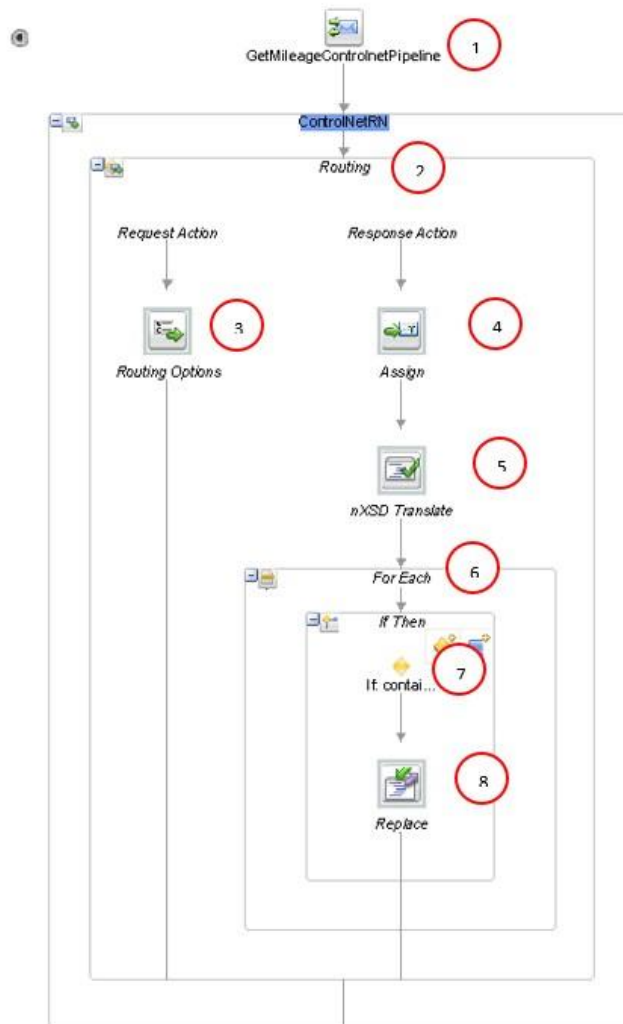
2.2 APIUnitMetricsTec

Este servicio permite realizar el llamado de los rest api como la invocacion de excel para generar los registros en Oracle Fusion para ser procesadas por el servicio APIUnitMetricsTec. Los componentes que conforman este servicio son los siguientes:



1. **GetMileageControlnet:** Este servicio está encargado de recuperar todas las lecturas registradas en este servicio web, a través de parámetros de fecha inicial y fecha final. Los componentes que conforman este servicio son los siguientes:

GetMileageControlnetBS: Servicio de negocio mediante el cual se virtualiza el API para la consulta de lecturas registradas en un tiempo determinado en el origen ControlNet. El endpoint del servicio es: <http://200.0.91.59:9090/controlnet/api/v1/kilometraje>



1. Componente que marca el inicio de la lógica del servicio para consultar la lectura de kilómetros en el origen ControlNet.
2. Elemento donde se realiza el llamado al servicio de negocio (Business Service) que representa el servicio final en ControlNet.
3. Componente que permite configurar las opciones de direccionamiento en el llamado al servicio de negocio final de ControlNet, las opciones configuradas son: la URI que contiene los parámetros de consulta y el verbo http por el cual se invocará el servicio, en este caso GET.
4. Asignación de mensaje de respuesta desde la variable body a una variable local para traducir el mensaje.
5. Traducción del mensaje JSON a formato XML mediante una estructura definida en el NXSD del servicio.
6. Inicio de ciclo foreach para iterar las lecturas obtenidas en ControlNet.
7. Validación de tipo IF donde se filtran los nodos que contengan la fecha de lectura.
8. Componente que permite reemplazar el mensaje de respuesta concatenando una "T" entre la fecha y la hora de lectura.

Algunos componentes extras del llamado Oracle Service Bus:

GetMileageControlnetWS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el API de ControlNet.

GetMileageControlnetWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación de consulta de las lecturas en ControlNet.

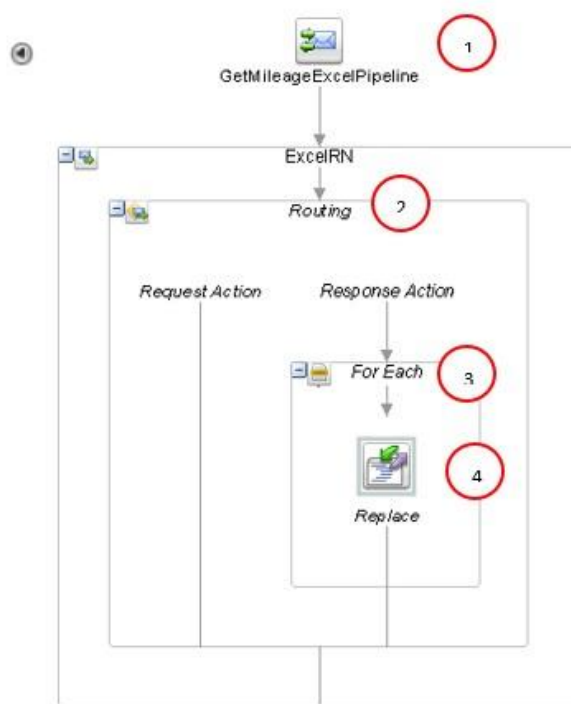
MileageControlnetTecNXSD: Archivo que permite traducir la estructura de mensaje de un lenguaje Nativo (JSON) a XML.

El EndpointURI correspondiente a este servicio es: GetMileageControlnetPS

2. **GetMileageExcelPipeline:** En este servicio se incluye la recuperación de lecturas registradas a través del archivo Excel que es cargado mediante una aplicación de Oracle Apex, de igual manera que el anterior y los demás servicios funciona mediante parámetros que se le asignan, correspondientes a la fecha inicial y fecha final, además de un servicio extra que se encarga de borrar datos temporales para posterior a ello puedan ser procesados. Los componentes que conforman este servicio son los siguientes: Componente mediante el cual se realiza el ruteo de mensaje hacia el servicio de negocio para la consulta de lecturas en origen Excel dentro de la Base de Datos Middleware.

GetMileageExcelBS: Servicio de negocio mediante el cual se virtualiza el llamado a base de datos Middleware para obtener las lecturas ingresadas en Apex de origen Excel, al ser un componente utilizado mediante un DBAdapter este se accede mediante un pool de conexiones (JNDI): **jca://eis/DB/DEV/ERSEPDB1**

TruncMileageExcelBS: Servicio de negocio mediante el cual se virtualiza el llamado a base de datos Middleware para realizar el borrado de lecturas procesadas de origen Excel, al ser un componente utilizado mediante un DBAdapter este se accede mediante un pool de conexiones (JNDI): **jca://eis/DB/DEV/ERSEPDB1**

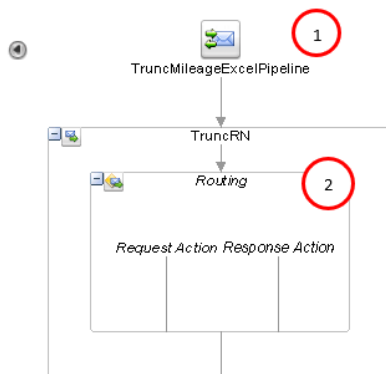


1. Componente que marca el inicio de la lógica del servicio para consultar la lectura de kilómetros en el origen Excel.
2. Elemento donde se realiza el llamado al servicio de negocio (Business Service) que representa el adaptador de base de datos de tipo JCA que permite la conexión a la base de datos Middleware para obtener las lecturas de Excel.
3. Inicio de ciclo foreach para iterar las lecturas obtenidas en Excel.
4. Componente que permite reemplazar el mensaje de respuesta concatenando una "T" entre la fecha y la hora de lectura.

Algunos componentes extras del llamado Oracle Service Bus:

GetMileageExcelWS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el servicio final que permite la consulta de lecturas del origen Excel.

TruncMileageExcelPipeline: Componente mediante el cual se realiza el ruteo de mensaje hacia el servicio de negocio para el borrado de lecturas en origen Excel dentro de la Base de Datos Middleware.



1. Componente que marca el inicio de la lógica del servicio para eliminar los registros de lecturas procesadas en el origen Excel.
2. Elemento donde se realiza el llamado de tipo **pass-through** al servicio de negocio (Business Service) que representa el adaptador de base de datos de tipo JCA que permite la conexión a la base de datos Middleware.

Algunos componentes extras del llamado Oracle Service Bus:

TruncMileageExcelWS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el servicio final que permite el borrado de lecturas procesadas del origen Excel.

GetMileageExcelWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación de consulta de las lecturas en Excel.

TruncMileageExcelWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación de borrado de lecturas del origen Excel.

GetMileageExcelJCA: Archivo de configuración mediante el cual se define la consulta de lecturas registradas en Excel mediante la configuración de un DBAdapter, la sentencia ejecutada es la siguiente: **select ASSET_NUMBER, ORIGIN, to_date (READING_DATE,'DD/MM/YYYY HH24:MI:SS') AS READING_DATE, READING_VALUE AS READING_VALUE, STATUS, ERROR_MESSAGE from XXER_EAM_DATA_EXCEL_TBL where to_date (reading_date,'DD/MM/YYYY HH24:MI:SS') between #fechaInicio and #fechaFin ORDER BY ASSET_NUMBER**

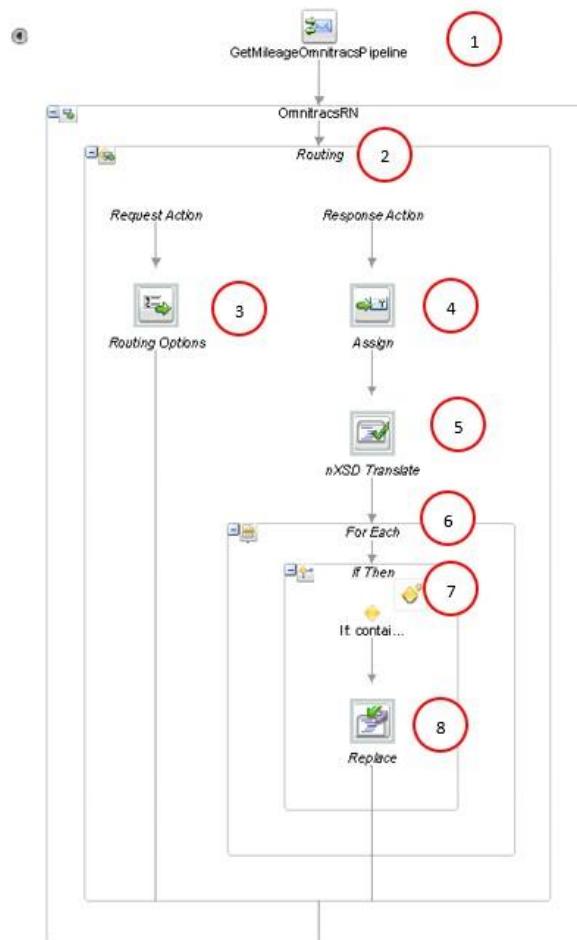
TruncMileageExcelJCA: Archivo de configuración mediante el cual se define el borrado de lecturas procesadas del origen Excel mediante la configuración de un DBAdapter, la sentencia ejecutada es la siguiente: **TRUNCATE TABLE xerintuser.XXER_EAM_DATA_EXCEL_TBL**

El EndpointURI correspondiente a este servicio es: **GetMileageExcelPS** y **TruncMileageExcelPS**.

3. **GetMileageOmnitracs:** Este servicio es encargado de realizar la extracción de lecturas correspondientes a este origen, así mismo como realizando esta recuperación de datos de acuerdo a los parámetros de fecha inicial y final. Los componentes que conforman este servicio son los siguientes: Componente donde se encapsula la lógica de ruteo que permite la consulta de lecturas, así como la traducción de mensaje tipo JSON a XML el cual contiene la respuesta de la consulta ejecutada.

GetMileageOmnitracsBS: Servicio de negocio mediante el cual se virtualiza el API para la consulta de lecturas registradas en un tiempo determinado en el origen Omnitracs. El endpoint del servicio es:

<http://200.0.91.59:9090/omnitracs/api/v1/kilometraje>



1. Componente que marca el inicio de la lógica del servicio para consultar la lectura de kilómetros en el origen Omnitacs.
2. Elemento donde se realiza el llamado al servicio de negocio (Business Service) que representa el servicio final en Omnitacs.
3. Componente que permite configurar las opciones de direccionamiento en el llamado al servicio de negocio final de Omnitacs, las opciones configuradas son: la URI que contiene los parámetros de consulta y el verbo http por el cual se invocará el servicio, en este caso GET.
4. Asignación de mensaje de respuesta desde la variable body a una variable local para traducir el mensaje.
5. Traducción del mensaje JSON a formato XML mediante una estructura definida en el NXSD del servicio.
6. Inicio de ciclo foreach para iterar las lecturas obtenidas en Omnitacs.
7. Validación de tipo IF donde se filtran los nodos que contengan la fecha de lectura.
8. Componente que permite reemplazar el mensaje de respuesta concatenando una "T" entre la fecha y hora de lectura.

Algunos componentes extras del llamado Oracle Service Bus:

GetMileageOmnitracsWS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el API de Omnitacs.

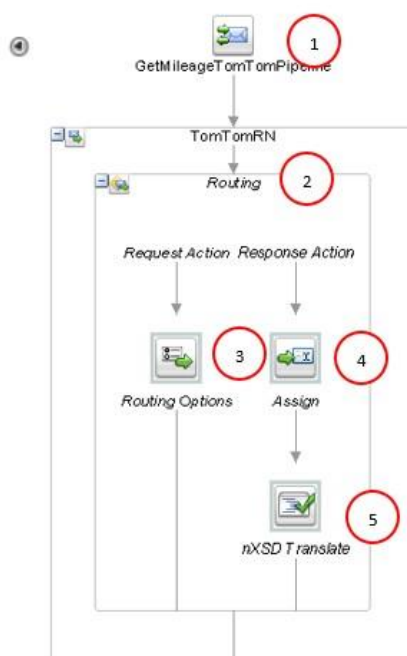
GetMileageOOmnitracsWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación de consulta de las lecturas en Omnitacs.

MileageOmnitracsTecNXSD: Archivo que permite traducir la estructura de mensaje de un lenguaje Nativo (JSON) a XML.

El EndpointURI correspondiente a este servicio es: GetMileageOmnitracsPS.

4. **GetMileageTomTom:** El siguiente servicio recupera los datos correspondientes a las lecturas de kilómetros registradas, las cuales se encuentren dentro del rango de fechas establecidas como fecha inicio y fecha fin. Los componentes que conforman este servicio son los siguientes: Componente donde se encapsula la lógica de ruteo que permite la consulta de lecturas, así como la traducción de mensaje tipo JSON a XML el cual contiene la respuesta de la consulta ejecutada.

GetMileageTomTomBS: Servicio de negocio mediante el cual se virtualiza el API para la consulta de lecturas registradas en un tiempo determinado en el origen TomTom. El endpoint del servicio es: <http://170.247.130.132:5151/api/consultas/consumoCombustible>



1. Componente que marca el inicio de la lógica del servicio para consultar la lectura de kilómetros en el origen TomTom.
2. Elemento donde se realiza el llamado al servicio de negocio (Business Service) que representa el servicio final en TomTom.
3. Componente que permite configurar las opciones de direccionamiento en el llamado al servicio de negocio final de TomTom, las opciones configuradas son: la URI que contiene los parámetros de consulta y el verbo http por el cual se invocará el servicio, en este caso GET.
4. Asignación de mensaje de respuesta desde la variable body a una variable local para traducir el mensaje.
5. Traducción del mensaje JSON a formato XML mediante una estructura definida en el NXSD del servicio.

Algunos componentes extras del llamado Oracle Service Bus

GetMileageTomTomWS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el API de TomTom.

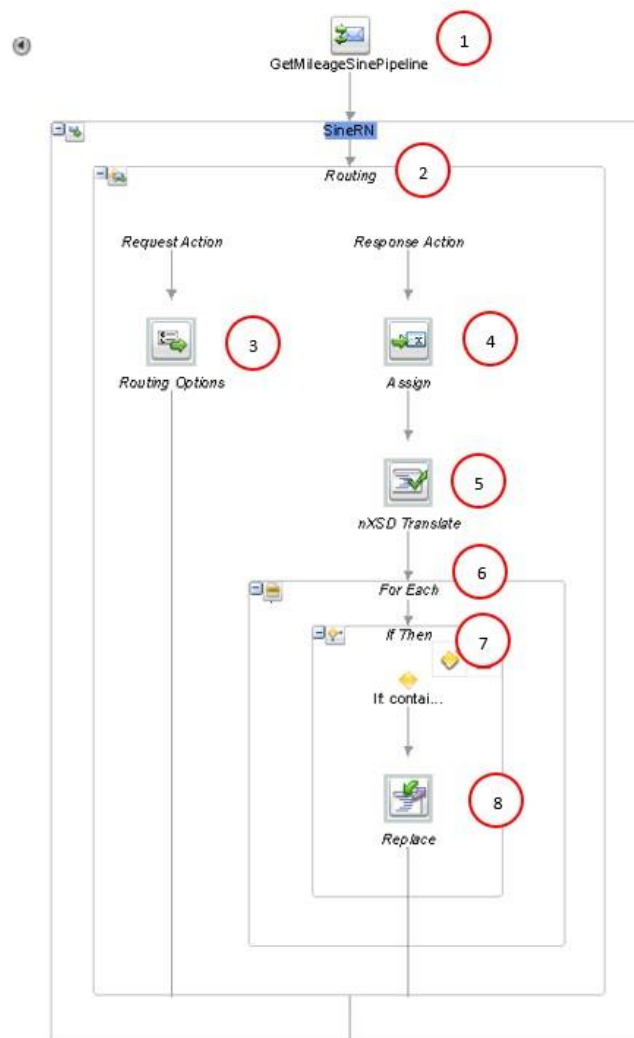
GetMileageTomTomWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación de consulta de las lecturas en TomTom.

MileageTomTomTecNXSD: Archivo que permite traducir la estructura de mensaje de un lenguaje Nativo (JSON) a XML.

El EndpointURI correspondiente a este servicio es: GetMileageTomTomPS

5. **GetMileageSine:** Este servicio de igual manera recupera el último origen por procesar, de acuerdo a los parámetros establecidos de las fechas de inicio y fin para así poder ser procesadas. Los componentes que conforman este servicio son los siguientes: Componente donde se encapsula la lógica de ruteo que permite la consulta de lecturas, así como la traducción de mensaje tipo JSON a XML el cual contiene la respuesta de la consulta ejecutada.

GetMileageSineBS: Servicio de negocio mediante el cual se virtualiza el API para la consulta de lecturas registradas en un tiempo determinado en el origen Sine. El endpoint del servicio es: <http://200.0.91.59:9090/sine/api/v1/kilometraje>



1. Componente que marca el inicio de la lógica del servicio para consultar la lectura de kilómetros en el origen Sine.
2. Elemento donde se realiza el llamado al servicio de negocio (Business Service) que representa el servicio final en Sine.
3. Componente que permite configurar las opciones de direccionamiento en el llamado al servicio de negocio final de Sine, las opciones configuradas son: la URI que contiene los parámetros de consulta y el verbo http por el cual se invocará el servicio, en este caso GET.
4. Asignación de mensaje de respuesta desde la variable body a una variable local para traducir el mensaje.
5. Traducción del mensaje JSON a formato XML mediante una estructura definida en el NXSD del servicio.
6. Inicio de ciclo foreach para iterar las lecturas obtenidas en Sine.
7. Validación de tipo IF donde se filtran los nodos que contengan la fecha de lectura.
8. Componente que permite reemplazar el mensaje de respuesta concatenando una "T" entre la fecha y hora de lectura.

Algunos componentes extras del llamado Oracle Service Bus

GetMileageSineWS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el API de Sine.

GetMileageSineWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación de consulta de las lecturas en Sine.

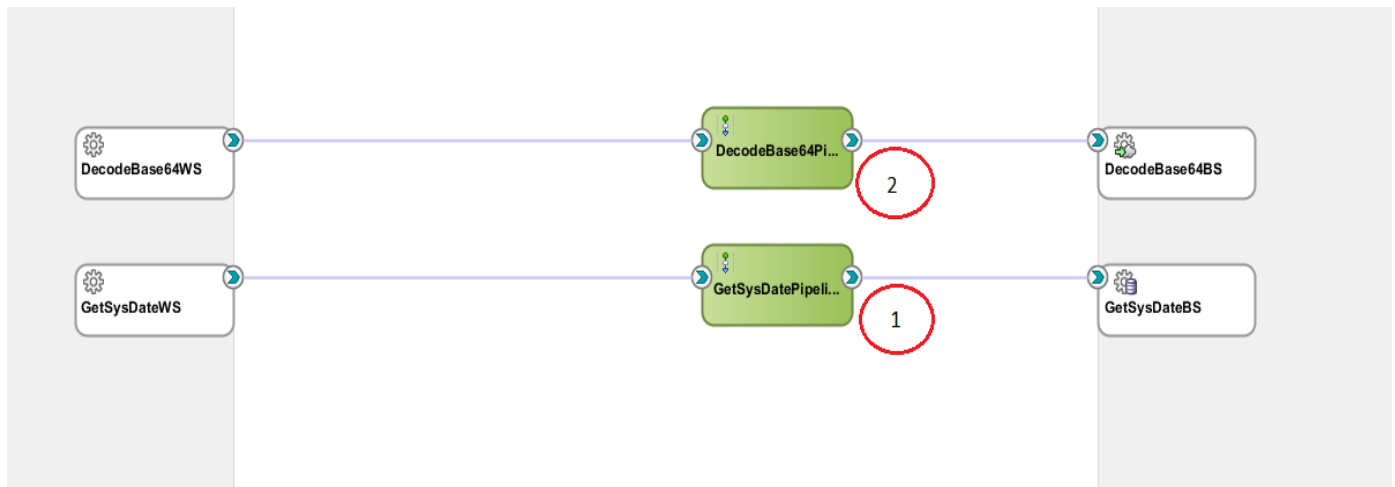
MileageSineTecNXSD: Archivo que permite traducir la estructura de mensaje de un lenguaje Nativo (JSON) a XML.

El EndpointURI correspondiente a este servicio es: GetMileageSinePS.

El EndpointURI correspondiente a este servicio es:EAMInsertWorkOrdersPS.

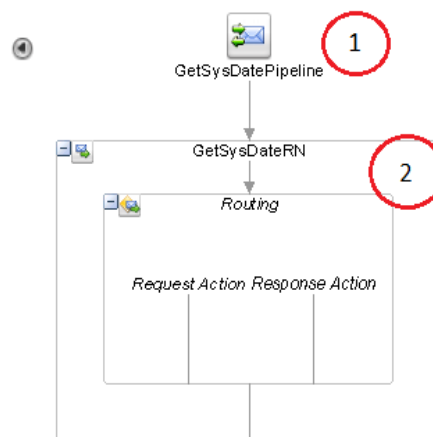
2.3 UtilityTec

Este servicio permite la consulta la fecha y base 64 bits en la cual se encuentra en Oracle Fusion para ser procesadas por el servicio UtilityTec. Los componentes que conforman este servicio son los siguientes:



1. **GetSysDatePipeline**: Servicio utilitario que permite obtener la fecha actual de sistema en el servidor.

Componente mediante el cual se realiza el ruteo de mensaje hacia el servicio de negocio para obtener la fecha de sistema mediante un store procedure en la Base de Datos Middleware.



1. Componente que marca el inicio de la lógica del servicio para obtener la fecha de sistema a través de un store procedure implementado en la base de datos Middleware.
2. Elemento donde se realiza el llamado de tipo **pass-through** al servicio de negocio (Business Service) que representa el adaptador de base de datos de tipo JCA que permite la conexión a la base de datos Middleware.

Algunos componentes extras del llamado Oracle Service Bus:

GetSysDateWS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el servicio final que permite obtener la fecha de sistema.

GetSysDateBS: Servicio de negocio mediante el cual se virtualiza el llamado a base de datos Middleware, al ser un componente utilizado mediante un DBAdapter este se accede mediante un pool de conexiones (JNDI): **jca://eis/DB/DEV/ERSEPDB1**

GetSysDateJCA: Archivo de configuración mediante el cual se obtiene la fecha de sistema desde la Base de Datos Middleware la cual se ejecuta mediante un store procedure definido en el siguiente paquete: **XXER_EAM_FS_REPORTS_PKG**

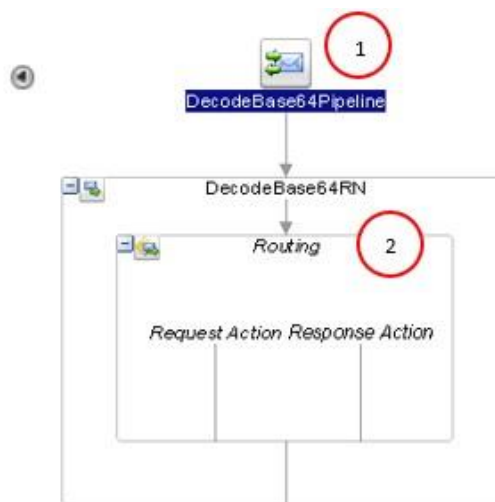
GetSysDateWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación de consulta de la fecha de sistema a la Base de Datos Middleware.

GetSysDateXSD: Esquema de servicio donde se definen los elementos, atributos y tipos de datos que conforman los mensajes de entrada y salida para el consumo del servicio.

El EndpointURI correspondiente a este servicio es: GetSysDatePS.

2. **DecodeBase64:** Servicio utilitario que permite decodificar un mensaje de base64 a string. Componente mediante el cual se realiza el ruteo de mensaje hacia el servicio de negocio para decodificar el mensaje base64 a string.

DecodeBase64BS: Servicio de negocio mediante el cual se virtualiza el BPEL para la decodificación de archivo Base64 a string. El endpoint del servicio es: <http://ersoacsprod2019-wls-1:9073/soa-infra/services/DEV/DecodeBase64/DecodeBase64Service?WSDL>



1. Componente que marca el inicio de la lógica del servicio para obtener la decodebase64 de sistema a través de un store procedure implementado en la base de datos Middleware.
2. Elemento donde se realiza el llamado de tipo **pass-through** al servicio de negocio (Business Service) que representa el adaptador de base de datos de tipo JCA que permite la conexión a la base de datos Middleware.

Algunos componentes extras del llamado Oracle Service Bus:

Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el servicio final.

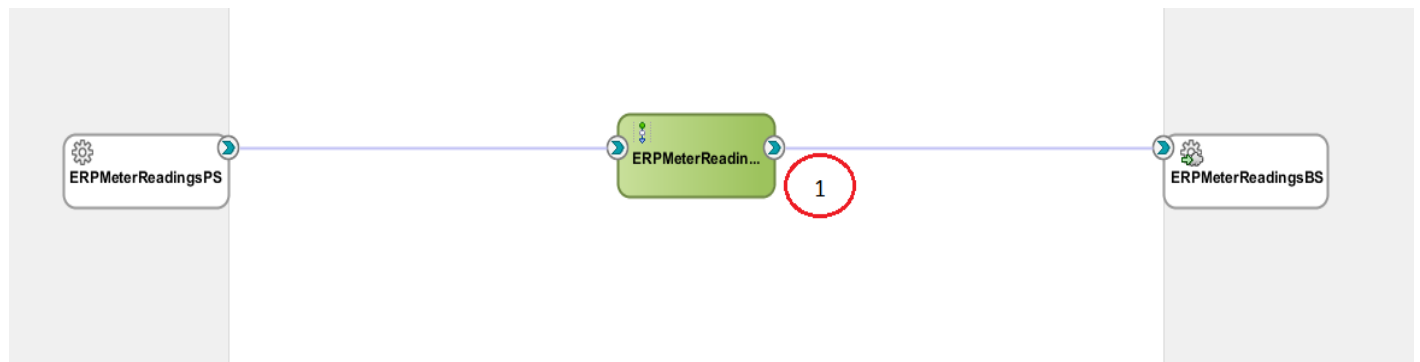
DecodeBase64WSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación decode dentro del BPEL que decodifica un mensaje de tipo Base64.

DecodeBase64XSD: Esquema de servicio donde se definen los elementos, atributos y tipos de datos que conforman los mensajes de entrada y salida para el consumo del servicio.

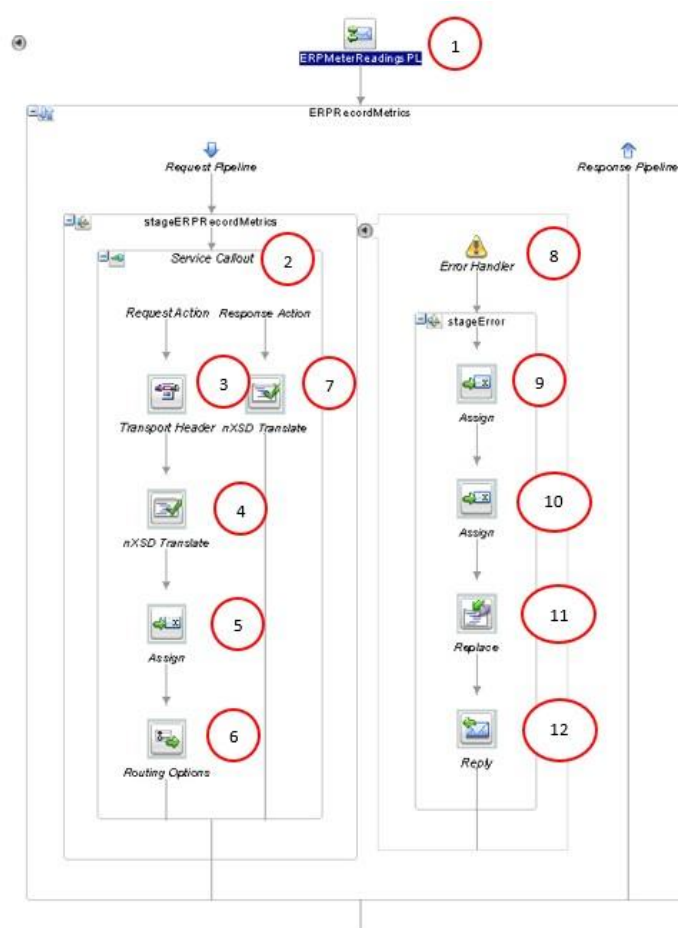
El EndpointURI correspondiente a este servicio es: DecodeBase64PS.

2.3 ERPUnitMetricsTec

Este servicio permite realizar el llamado de los rest api para insertar los registros en Oracle Fusion para ser procesadas por el servicio ERPUnitMetricsTec. Los componentes que conforman este servicio son los siguientes:



1. **ERP Meter Readings PL:** Este servicio tiene por funcionalidad mandar a escribir a Oracle Fusion los datos validados en los anteriores servicios, ingresando en cada activo un valor de lectura con fecha de lectura, la cual será la última registrada por día. Componente mediante el cual se realiza el ruteo de mensaje hacia el servicio de negocio expuesto por Oracle Fusion el cual permite registrar la lectura mayor que se procesa por activo.



1. Componente que marca el inicio de la lógica del servicio para insertar las lecturas en Oracle Fusion.
2. Actividad que permite el llamado síncrono al servicio de negocio (Business Service) para la inserción de lecturas en Fusion.
3. Componente que permite definir la cabecera que contiene el tipo de contenido, en este caso "application/json"
4. Traducción de mensaje de entrada hacia el servicio final de XML a nativo (JSON) mediante un NXSD definido.
5. Asignación de mensaje convertido de binario a texto a una variable local para la petición del servicio.
6. Componente que permite configurar las opciones de direccionamiento en el llamado al servicio de negocio final para la inserción de lecturas, la opción configurada es: el verbo http por el cual se invocará el servicio, en este caso POST.
7. Traducción de mensaje de salida del servicio final de nativo (JSON) a XML mediante un NXSD definido.
8. Elemento que permite configurar el manejo de errores del servicio final.
9. Asignación del detalle de error obtenido por parte del servicio en Oracle Fusion a una variable local.
10. Asignación de mensaje convertido de binario a texto a una variable local con el detalle de error.
11. Componente que permite reemplazar el mensaje de salida con la estructura de error definida en el XSD del servicio.
12. Componente que permite responder con fallo en caso de error.

Algunos componentes extras del llamado Oracle Service Bus

ERPMeterReadingsPS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el servicio final.

ERPMeterReadingsBS: Servicio de negocio mediante el cual se virtualiza el URI del servicio final expuesto por Oracle Fusion para el registro de lecturas. El endpoint del servicio es: <https://egte-test.fa.us2.oraclecloud.com/fscmRestApi/resources/11.13.18.05/meterReadings>

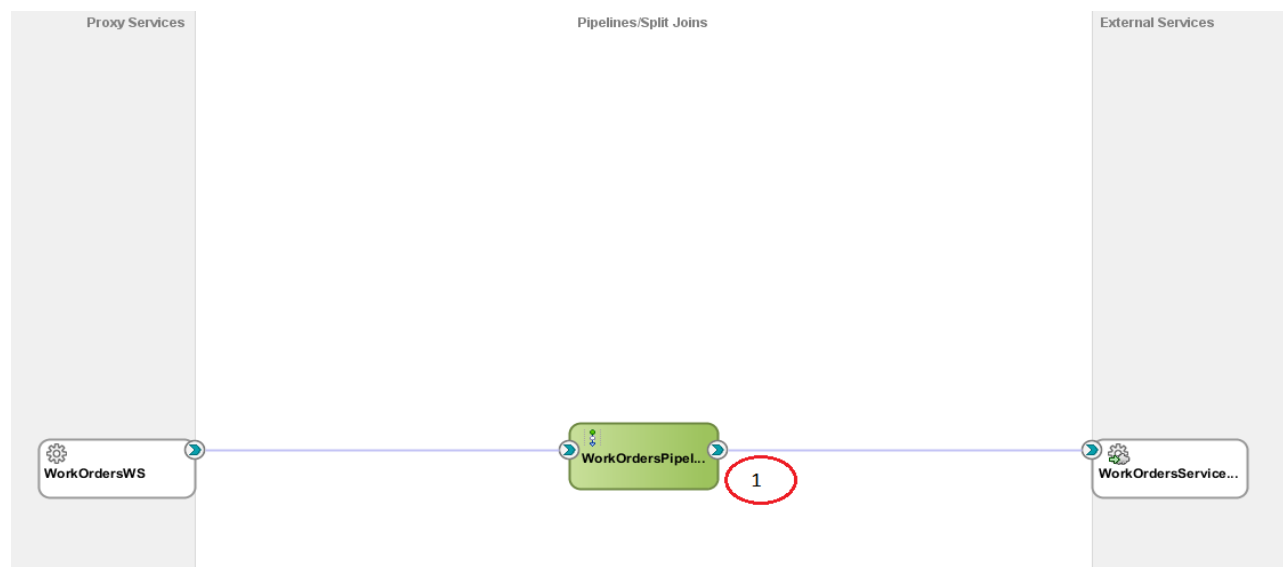
ERPMeterReadingsWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación para el registro de lecturas en Oracle Fusion.

ERPMeterReadingsTecNXSD: Archivo que permite traducir la estructura de mensaje de un lenguaje Nativo (JSON) a XML.

El EndpointURI correspondiente a este servicio es: ERPMeterReadingsPS.

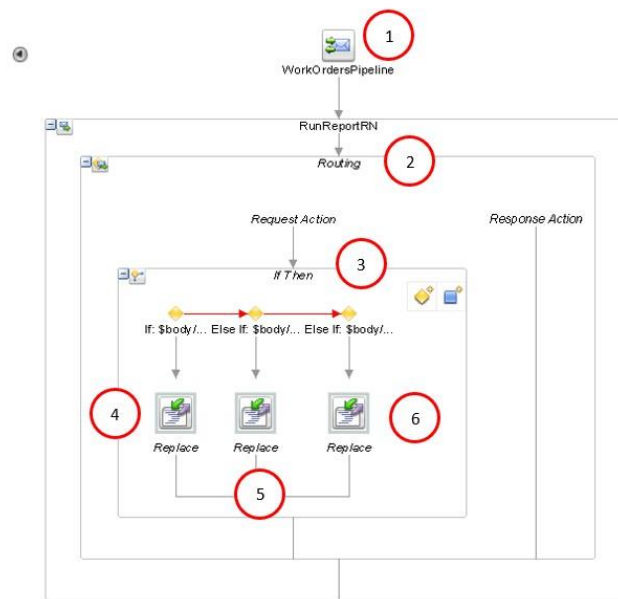
2.4 SCMWorkOrdersTec

Este servicio permite la consulta de reportes en Oracle Fusion para ser procesadas por el servicio SCMWorkOrdersTec. Los componentes que conforman este servicio son los siguientes:



1. **WorkOrders:** Este servicio permite la consulta de órdenes de trabajo registradas en Oracle Fusion para ser procesadas por el servicio SCMWorkOrdersTec. Los componentes que conforman este servicio son los siguientes:

Dentro de este componente se emplea la lógica que encapsula el manejo y ruteo del mensaje para la consulta de órdenes de trabajo en Oracle Fusión.



1. Elemento donde se realiza el llamado al servicio de negocio (Business Service) que representa el servicio final en Oracle Fusion para consulta de las work orders.
2. Validación de tipo IF para determinar que reporte se consultará en Oracle Fusión.
3. Si el valor de petición para consultar el reporte de "ASSET" se reemplazan los valores de entrada ingresando los parámetros necesarios para la consulta.
4. Si el valor de petición para consultar el reporte de "METERS" se reemplazan los valores de entrada ingresando los parámetros necesarios para la consulta.
5. Si el valor de petición para consultar el reporte de "WORKORDERS" se reemplazan los valores de entrada ingresando los parámetros necesarios para la consulta.

Algunos componentes extras del llamado Oracle Service Bus:

WorkOrdersWS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el servicio final de órdenes de trabajo.

WorkOrdersServiceBS: Servicio de negocio mediante el cual se virtualiza el URI final para la consulta de órdenes de trabajo en Oracle Fusion. El endpoint del servicio es:

<https://egte-test.fa.us2.oraclecloud.com/xmlpserver/services/ExternalReportWSSService?wsdl>

WorkOrdersServiceWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación de consulta de las órdenes de trabajo.

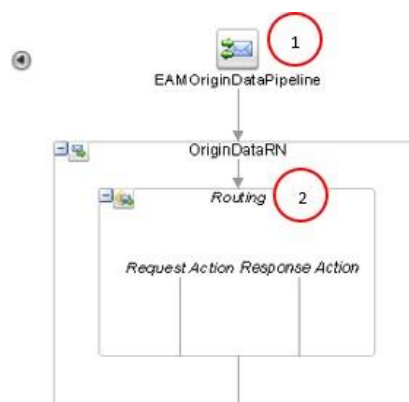
El EndpointURI correspondiente a este servicio es: WorkOrdersPS

2.5 EAMUnitMetricsTec

Este servicio permite la consulta de los reportes para la validación de la información en Oracle Fusion para ser procesadas por el servicio EAMUnitMetricsTec. Los componentes que conforman este servicio son los siguientes:



1. **EAMOriginData:** Servicio de datos encargado de realizar la consulta de los registros que se insertan en base de datos de la información del origen Excel para la lectura de kilómetros. Los componentes que conforman este servicio son los siguientes: Componente mediante el cual se realiza el ruteo de mensaje hacia el servicio de negocio para la consulta de lecturas en la Base de Datos Middleware las cuales corresponden al origen Excel.



1. Componente que marca el inicio de la lógica del servicio para consultar los registros insertados del origen Excel en la base de datos Middleware.
2. Elemento donde se realiza el llamado de tipo **pass-through** al servicio de negocio (Business Service) que representa el adaptador de base de datos de tipo JCA que permite la conexión a la base de datos Middleware.

Algunos componentes extras del llamado Oracle Service Bus:

EAMOriginDataWS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el servicio final que permite la consulta de lecturas de Excel.

EAMOriginDataBS: Servicio de negocio mediante el cual se virtualiza el llamado a base de datos Middleware, al ser un componente utilizado mediante un DBAdapter este se accede mediante un pool de conexiones (JNDI): **jca://eis/DB/DEV/ERSEPDB1**

EAMOriginDataWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación de consulta a la Base de Datos Middleware.

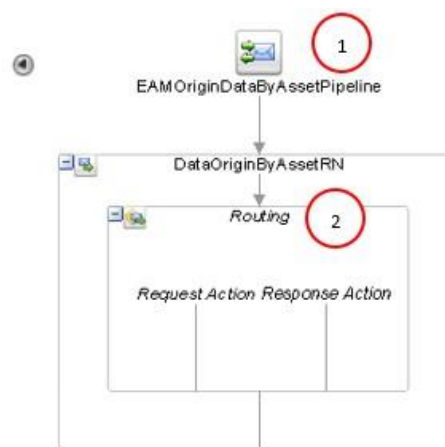
EAMOriginDataXSD: Esquema de servicio donde se definen los elementos, atributos y tipos de datos que conforman los mensajes de entrada y salida para el consumo del servicio.

EAMOriginDataJCA: Archivo de configuración mediante el cual se define la consulta de lecturas registradas en Excel mediante la configuración de un DBAdapter, la sentencia ejecutada es la siguiente: **select * from xxer_eam_data_origin_tbl where origin = #pOrigin and creation_date BETWEEN #pFechaInicio and #pFechaFin**

El EndpointURI correspondiente a este servicio es: EAMOriginDataPS

2. **EAMOriginDataByAsset:** Servicio de datos encargado de realizar la consulta de los registros por activo, que se insertan en base de datos de la información del origen Excel para la lectura de kilómetros. Los componentes que conforman este servicio son los siguientes: Componente mediante el cual se realiza el ruteo de mensaje hacia el servicio de negocio para la consulta por activo de lecturas en la Base de Datos Middleware las cuales corresponden al origen Excel.

EAMOriginDataByAssetWS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el servicio final que permite la consulta por activo de lecturas registradas en Excel.



1. Componente que marca el inicio de la lógica del servicio para consulta de por activo de las lecturas de origen Excel en la base de datos Middleware.
2. Elemento donde se realiza el llamado de tipo **pass-through** al servicio de negocio (Business Service) que representa el adaptador de base de datos de tipo JCA que permite la conexión a la base de datos Middleware.

Algunos componentes extras del llamado Oracle Service Bus:

EAMOriginDataByAssetWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación de consulta por activo a la Base de Datos Middleware.

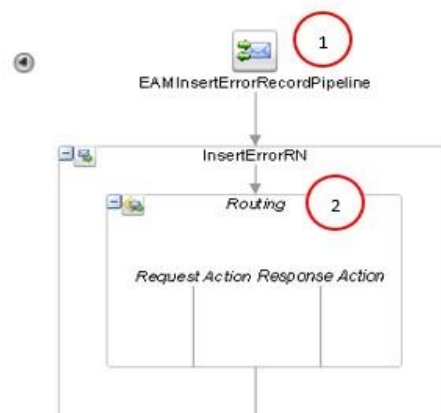
EAMOriginDataByAssetXSD: Esquema de servicio donde se definen los elementos, atributos y tipos de datos que conforman los mensajes de entrada y salida para el consumo del servicio.

EAMOriginDataByAssetJCA: Archivo de configuración mediante el cual se define la consulta de lecturas por activo registradas en Excel mediante la configuración de un DBAdapter, la sentencia ejecutada es la siguiente: **select ASSET_NUMBER, ORIGIN, to_date (READING_DATE,'DD/MM/YYYY HH24:MI:SS') AS READING_DATE, READING_VALUE AS READING_VALUE, STATUS, ERROR_MESSAGE from XXER_EAM_DATA_EXCEL_TBL where ASSET_NUMBER = #assetNumber and reading_value is not null and reading_value > 0 and reading_date is not null and to_date (reading_date,'DD/MM/YYYY HH24:MI:SS') between #fechalnicio and #fechaFin ORDER BY ASSET_NUMBER**

EAMOriginDataByAssetBS: Servicio de negocio mediante el cual se virtualiza el llamado a base de datos Middleware, al ser un componente utilizado mediante un DBAdapter este se accede mediante un pool de conexiones (JNDI): jca://eis/DB/DEV/ERSEPDB1

El EndpointURI correspondiente a este servicio es: EAMOriginDataByAssetPS.

3. **EAMInsertErrorRecord:** Servicio de datos encargado de realizar la inserción de los registros procesados en la lectura de kilómetros a fin de llevar el registro de una bitácora para el reporte de procesamiento final. Componente mediante el cual se realiza el ruteo de mensaje hacia el servicio de negocio para la inserción de lecturas en la Base de Datos Middleware.



1. Componente que marca el inicio de la lógica del servicio para la inserción de lecturas procesadas en la base de datos Middleware.
2. Elemento donde se realiza el llamado de tipo **pass-through** al servicio de negocio (Business Service) que representa el adaptador de base de datos de tipo JCA que permite la conexión a la base de datos Middleware.

Algunos componentes extras del llamado Oracle Service Bus:

EAMInsertErrorRecordWS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el servicio final que permite la inserción de lecturas procesadas.

EAMInsertErrorRecordBS: Servicio de negocio mediante el cual se virtualiza el llamado a base de datos Middleware, al ser un componente utilizado mediante un DBAdapter este se accede mediante un pool de conexiones (JNDI): **jca://eis/DB/DEV/ERSEPDB1**

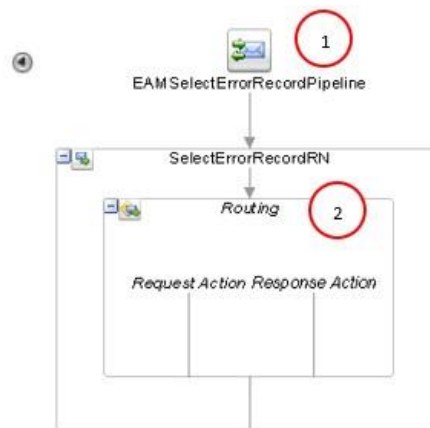
EAMInsertErrorRecordWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación de inserción de registros procesados a la Base de Datos Middleware.

EAMInsertErrorRecordXSD: Esquema de servicio donde se definen los elementos, atributos y tipos de datos que conforman los mensajes de entrada y salida para el consumo del servicio.

EAMInsertErrorRecordJCA: Archivo de configuración mediante el cual se define la inserción de registros procesados en la lectura de kilómetros mediante la configuración de un DBAdapter.

El EndpointURI correspondiente a este servicio es: **EAMInsertErrorRecordPS**

4. **EAMSelectErrorRecord:** Servicio de datos encargado de la consulta de los registros procesados en la base de datos middleware para el envío de correo electrónico con el número de lecturas procesadas. Componente mediante el cual se realiza el ruteo de mensaje hacia el servicio de negocio para la consulta de lecturas en la Base de Datos Middleware.



1. Componente que marca el inicio de la lógica del servicio para la consulta de lecturas procesadas en la base de datos Middleware.
2. Elemento donde se realiza el llamado de tipo **pass-through** al servicio de negocio (Business Service) que representa el adaptador de base de datos de tipo JCA que permite la conexión a la base de datos Middleware.

Algunos componentes extras del llamado Oracle Service Bus:

EAMSelectErrorRecordWS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el servicio final que permite la consulta de lecturas procesadas.

EAMSelectErrorRecordJCA: Archivo de configuración mediante el cual se define la consulta de registros procesados en la lectura de kilómetros mediante la configuración de un DBAdapter.

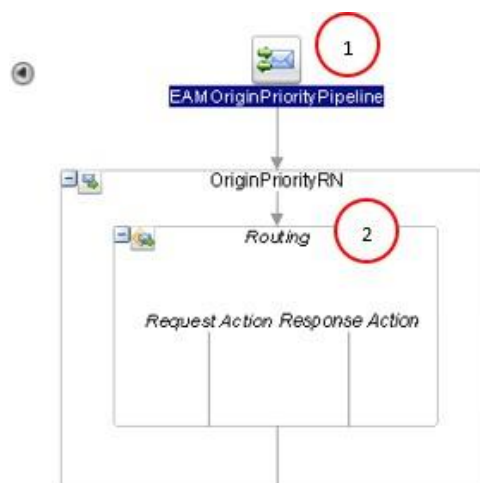
EAMSelectErrorRecordWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación de consulta de registros procesados a la Base de Datos Middleware.

EAMSelectErrorRecordXSD: Esquema de servicio donde se definen los elementos, atributos y tipos de datos que conforman los mensajes de entrada y salida para el consumo del servicio.

EAMSelectErrorRecordBS: Servicio de negocio mediante el cual se virtualiza el llamado a base de datos Middleware, al ser un componente utilizado mediante un DBAdapter este se accede mediante un pool de conexiones (JNDI): **jca://eis/DB/DEV/ERSEPDB1**

El EndpointURI correspondiente a este servicio es: EAMSelectErrorRecordPS

5. **EAMOriginPriority:** Servicio de datos que permite la consulta de la prioridad que debe ser tomada en cuanto a orígenes de la lectura de kilómetros en la base de datos. Componente mediante el cual se realiza el ruteo de mensaje hacia el servicio de negocio para la consulta de prioridades por origen en la Base de Datos Middleware.



1. Componente que marca el inicio de la lógica del servicio para la consulta de prioridades por origen en la base de datos Middleware.
2. Elemento donde se realiza el llamado de tipo **pass-through** al servicio de negocio (Business Service) que representa el adaptador de base de datos de tipo JCA que permite la conexión a la base de datos Middleware.

Algunos componentes extras del llamado Oracle Service Bus:

EAMOriginPriorityWS: Servicio virtualizado mediante el cual se permite el llamado desde un consumidor hacia el servicio final que permite la consulta de prioridad de lectura.

EAMOriginPriorityWSDL: Archivo que representa el contrato de servicio en el cual se encuentra la definición de mensajes de entrada y salida, bindings y operación de consulta de prioridades por origen en la Base de Datos Middleware.

EAMOriginPriorityXSD: Esquema de servicio donde se definen los elementos, atributos y tipos de datos que conforman los mensajes de entrada y salida para el consumo del servicio.

EAMOriginPriorityJCA: Archivo de configuración mediante el cual se define la consulta de prioridad por origen mediante la configuración de un DBAdapter, la sentencia ejecutada es la siguiente: **SELECT ID, ORIGIN, PRIORITY, ASSET_NUMBER FROM XXER_EAM_ORIGIN_PRIORITY_TBL WHERE ASSET_NUMBER = #assetNumber order by priority asc**

EAMOriginPriorityBS: Servicio de negocio mediante el cual se virtualiza el llamado a base de datos Middleware, al ser un componente utilizado mediante un DBAdapter este se accede mediante un pool de conexiones (JNDI): **jca://eis/DB/DEV/ERSEPDB1**

3 Diseño de Datos

3.1 Origen de Datos

Tabla	Columna	Comentarios
XXER_EAM_DATA_EXCEL_TBL	ASSET_NUMBER	Número de activo
	ORIGIN	Origen de lectura
	READING_DATE	Fecha de lectura
	READING_VALUE	Valor de lectura
	STATUS	Estatus de la lectura
	ERROR_MESSAGE	Mensaje de procesamiento
XXER_EAM_ERROR_LOG	ID_ERROR	Identificador de la tabla
	ASSET_NUMBER	Número de activo
	READING_DATE	Fecha de lectura
	READING_VALUE	Valor de lectura
	ERROR_MESSAGE	Mensaje de procesamiento
	ORIGIN	Origen de lectura
XXER_EAM_ORIGIN_PRIORITY_TBL	ID	Identificador de la tabla
	ORIGIN	Origen de lectura
	PRIORITY	Asignación de prioridad del origen
	ASSET_NUMBER	Número de activo

4 Reglas de Negocio

Las reglas de negocio para esta integración son las siguientes:

- Se determina el origen, fecha de inicio y fin.
- Extrae información de los diferentes orígenes de acuerdo a los parámetros anteriores.
- Se toma la última lectura registrada por día con la lectura mayor.
- Se verifica si ese activo viene de dos o más orígenes, de ser así se verifica la prioridad asignada.
- Se hacen validaciones contra la última lectura registrada en Oracle Fusion.
- De ser una fecha con valor de lectura mayor y ascendente se escribe en ese activo.
- Se envía correo electrónico y escribe en reporte de Oracle Apex con resumen de información procesada.

5 Consideraciones de Rendimiento

Este requerimiento ha sido probado con un conjunto de órdenes de trabajo en bajo volumen, tener contemplado que el compuesto puede presentar demora con una cantidad de registros muy grande a procesar.

5.1 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.

5.2 Seguridad

N/A

5.3 Personalización

- En caso de requerir modificar el proyecto, tomar la versión más reciente del controlador de versiones con el que se cuenta.
- 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

6 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_EAM_DATA_EXCEL_TBL
- ☐ XXER_EAM_ERROR_LOG
- ☐ XXER_EAM_ORIGIN_PRIORITY_TBL

En este esquema debe existir el siguiente paquete:

- ☐ XXER_EAM_FS_REPORTS_PKG
- ☐ XXER_EAM_BLK_ASSETS_PKG

En este esquema debe existir la siguiente secuencia:

- ☐ XXER_ERROR_LOG_SEQ
- ☐ XXER_EAM_ORIGIN_PRIORITY_SEQ

6.1 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:

sca_RecordUnitMetri
cs.jar

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

7 URL de Acceso y seguridad

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

Servicio	URL
RegisterAssetStatus	NA

8 Temas abiertos y cerrados

8.1 Temas Abiertos

ID	Tema	Solución	Responsabilidad	Fecha Objetivo	Fecha impacto

8.2 Temas Cerrados

ID	Tema	Solución	Responsabilidad	Fecha Objetivo	Fecha impacto