

# SOA Suite for healthcare integration Series

## Routing and Transformation using XSL Solution

michael@czapski.id.au

December 2012

### Table of Contents

Introduction .....	1
Solution Overview.....	2
Define the ADT A08 document .....	4
Add A08_v1.0 Document to Document Protocol Hierarchy.....	6
Second HL7 v2 Outbound (Sender).....	7
HL7 v2 Inbound (Receiver).....	10
Original HL7 v2 Outbound (Sender).....	11
Start Receiving “External Systems” .....	13
HL7 v2 Router and Transformer .....	13
Send ADT messages .....	24
Explore Message Tracking.....	26
Explore Dashboards .....	29
Summary .....	32

### Introduction

In this article the router, developed in “SOA Suite for healthcare integration Series – HL7 v2 Inbound CMM to Outbound Pass-through Solution”, to be found at <http://blogs.czapski.id.au/2012/12/soa-suite-for-healthcare-integration-series-hl7-v2-inbound-cmm-to-outbound-pass-through-solution>, will be extended to add a fan-out for distribution of one inbound message to two outbound receivers and a ADT A01/A03 to ADT A08 transformation using the eXtensible Style Language (XSL). SOA Suite for healthcare integration Dashboard functionality will also be explored.

This article assumes that the reader is sufficiently familiar with HL7 v2 and HL7 v2 messaging to require no elaboration on the message structures, message acknowledgement processing and the “equivalence” of HL7 v2 delimited and HL7 v2 XML message forms.

This article assumes that the reader has the SOA Suite for healthcare integration environment with all necessary components installed and ready to use. The Bill of Materials for such an environment and a discussion on where the components can be obtained is

provided in the earlier article, "SOA Suite for healthcare integration Series - Overview of the Development Environment", to be found at <http://blogs.czapski.id.au/2012/08/soa-suite-for-healthcare-integration-series-overview-of-the-development-environment>.

This article assumes that the reader completed the solution discussed in the earlier article, "SOA Suite for healthcare integration Series – HL7 v2 Inbound CMM to Outbound Pass-through Solution", to be found at <http://blogs.czapski.id.au/2012/12/soa-suite-for-healthcare-integration-series-hl7-v2-inbound-cmm-to-outbound-pass-through-solution>.

## Solution Overview

In this article the inbound HL7 v2 A01 and A03 messages will be received by the inbound endpoint and will be routed by two outbound endpoints to two separate receiver systems. One of the message streams will be passed on unchanged and the other will translate A01 and A03 messages into the equivalent A08 (Update Patient Information) messages.

The inbound SOA Suite for healthcare integration adapter will perform the casting activity while translating the message from HL7 v2 delimited to the "equivalent" XML format. The outbound SOA Suite for healthcare integration adapters will translate messages from HL7 v2 XML to the "equivalent" HL7 v2 delimited format before sending it out.

The runtime components and their relationships are presented in Figure 1.

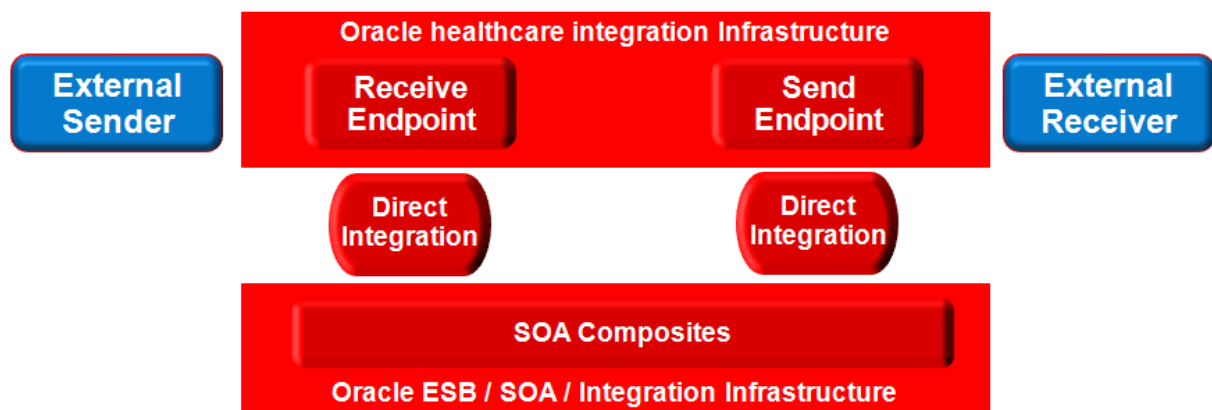


Figure 1 Runtime Components of the Solution

To summarise:

External Sender is the component which stands for a HIS or PAS – the sender of HL7 messages

The Oracle Healthcare Integration infrastructure is the part of the SOA Suite which deals with HL7 messages, acknowledgements, message tracking, message persistence, message processing KPI collection and so on, and the Receive Endpoint is the listening endpoint which receives messages

Direct Integration is the behind-the-scenes mechanism which hands messages over to an appropriate SOA Suite-based logic component for further processing or hands over messages produced by SOA Suite components to the healthcare integration infrastructure for processing (translation, sending, tracking).

ESB / SOA / Integration infrastructure hosts the SOA Composites and other logic components which process messages, whether HL7 v2 or not.

The solution components are depicted in **Error! Reference source not found..**

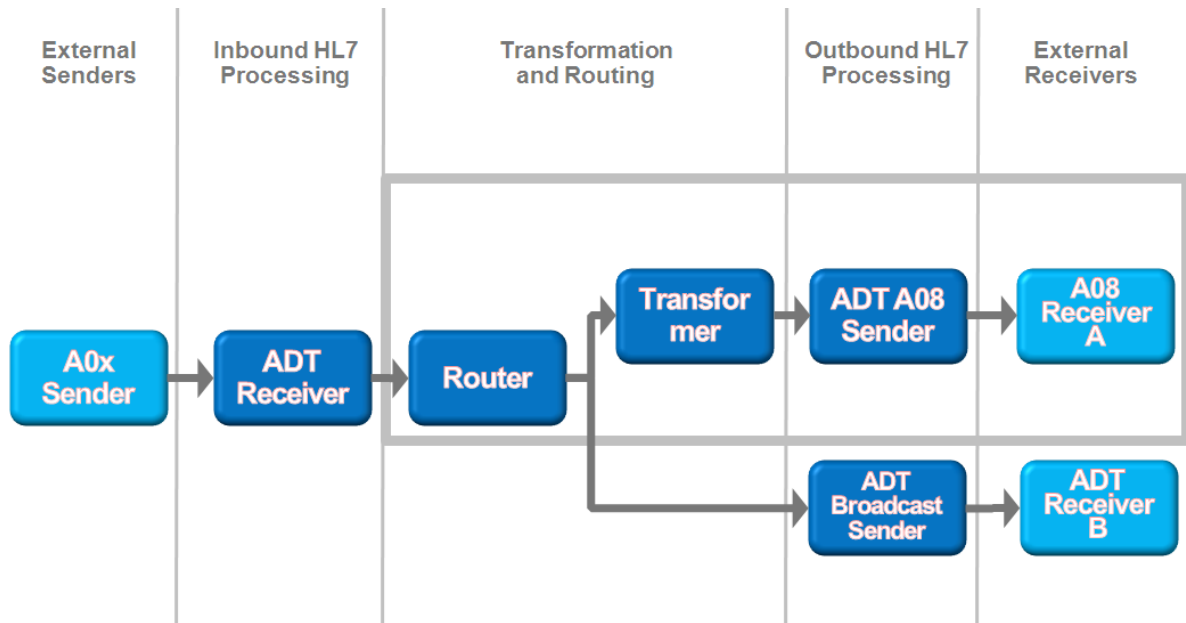


Figure 2 Solution Components

The diagram uses the convention which clearly separates the external systems, the SOA Suite for healthcare integration-specific components and generic SOA Suite components using the "swim-line" analogy.

A0x Sender is the CMDHL7 sender tool, or another tool capable of sending HL7 v2 Delimited messages over TCP/IP using the MLLP protocol. It will send A01 and A03 messages.

ADT Receiver is the SOA Suite for healthcare integration HL7 v2 listener.

Router is a SOA Composite which receives the message from the HL7 listener and "routes" it, as a fan out device, a copy to the HL7 sender ADT Broadcast Sender and a copy to the Transformer Mediator component.

Transformer is a SOA Composite which receives the message from the HL7 listener, transforms the ADT A01 or the A03 message into the ADT A08 message, and "routes" it to the HL7 sender ADT A08 Sender.

ADT A08 Sender is the SOA Suite for healthcare integration HL7 v2 sender.

ADT Broadcast Sender is the SOA Suite for healthcare integration HL7 v2 sender.

A08 Receiver A is the CMDHL7 listener tool, or another tool capable of receiving HL7 v2 Delimited messages over TCP/IP using the MLLP protocol and acknowledging them. It will receive A08 messages.

ADT Receiver B is the CMDHL7 listener tool, or another tool capable of receiving HL7 v2 Delimited messages over TCP/IP using the MLLP protocol and acknowledging them. It will receive A01 and A03 messages.

The solution will receive a HL7 v2 message, acknowledge it with an immediate acknowledgement and pass it onto the SOA Composite. The immediate acknowledgement will be sent as soon as the message is received and persisted, before it is processed in any way. The acknowledgement received from the HL7 listener will be discarded.

We need to add a couple of components to the earlier solution – an ADT A08 message definition and another outbound endpoint.

## Define the ADT A08 document

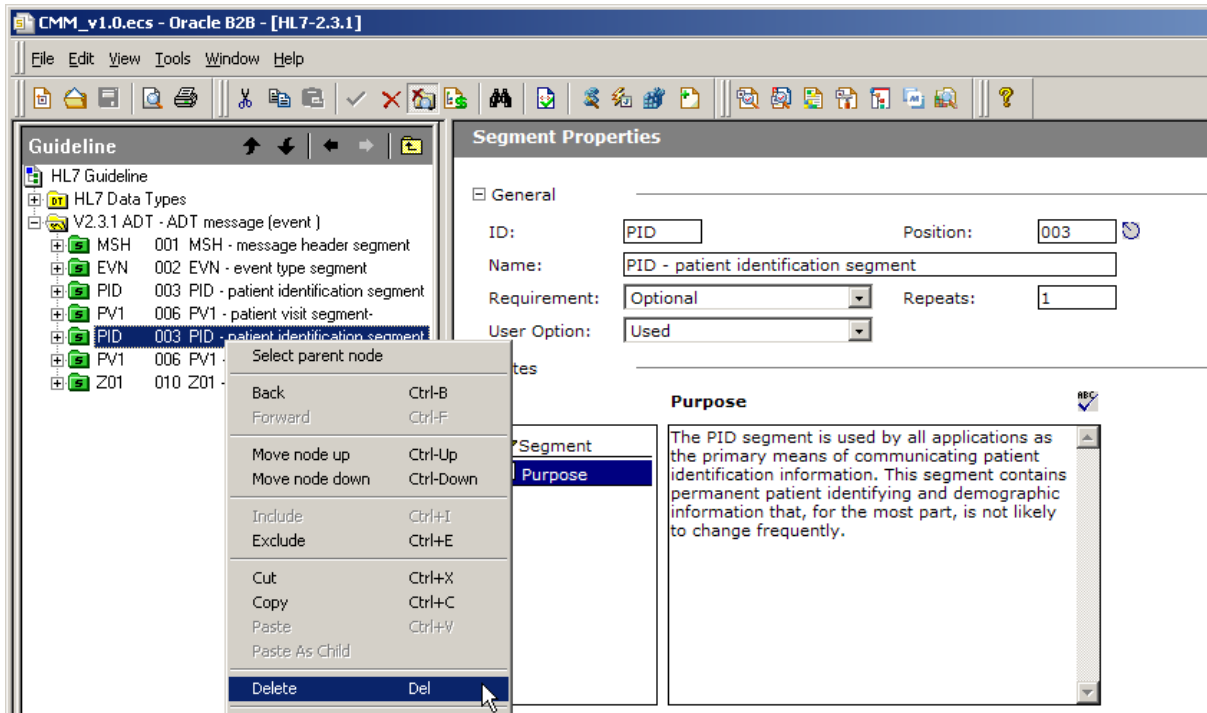
In this section we will define the AFDT A08 document using the canonical message as the basis. The canonical message, developed in the article "SOA Suite for healthcare integration Series – Creating a Canonical HL7 v2 Message Model", at <http://blogs.czapski.id.au/2012/09/soa-suite-for-healthcare-integration-series-creating-a-canonical-hl7-v2-message-model>, already contains all the segments we need, see pages 5 and 6 for tables representing segments to be included and the discussion which follows them. We must merely remove repeating optional PID and PV1 segments and set the trigger event to A08.

The final message will have the following segments:

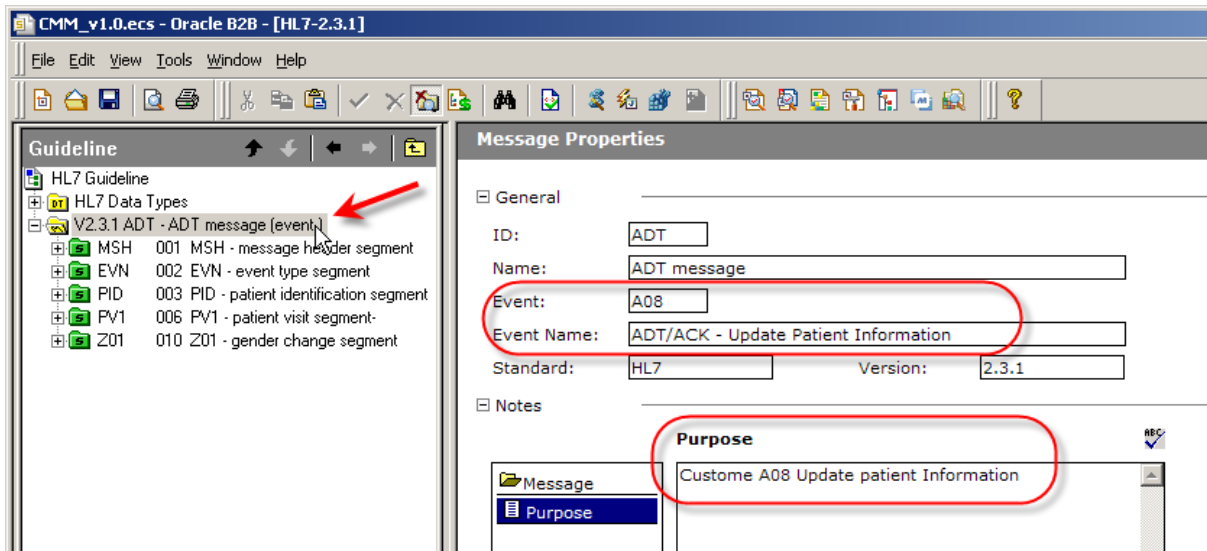
A08
MSH
EVN
PID
PV1
Z01?

Figure 3 ADT A08 Document segments

- Start the Oracle B2B Document Editor.
- Click Open Document
- Navigate to the location of the "CMM\_v1.0.ecs" file and double-click it to open it on the Oracle B2B Document Editor
- Select and delete the second PID segment



- Repeat the process to delete the second PV1 segment
- Click on the "V2.3.1 ADT – ADT message (event )" parent node then enter information into appropriate fields – Event: "A08", Event Name: "ADT/ACK – Update Patient Information", Purpose: "Custom A08 Update patient information".



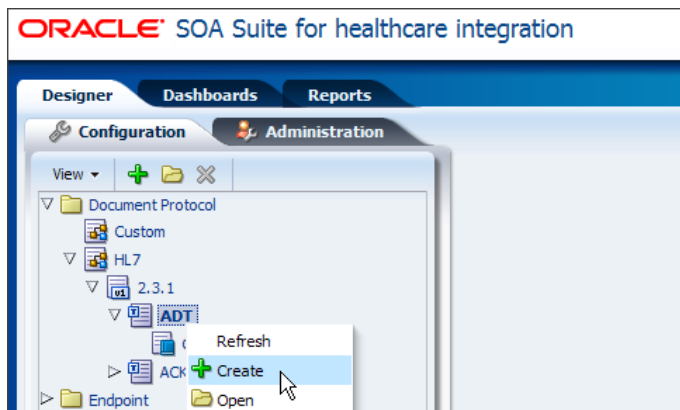
- Pull down the "File" menu, choose "Save As" and save the ADT A08 as "A08\_v1.0.ecs"
- Pull down the "File" menu, choose "Export", choose "Oracle B2B" and export the XDS version of the document as "A08\_v1.0.xsd"
- Exit the Oracle B2B Document Editor.

We are done. The document definition is ready to be used. Let's "introduce" it to the SOA Suite for healthcare integration.

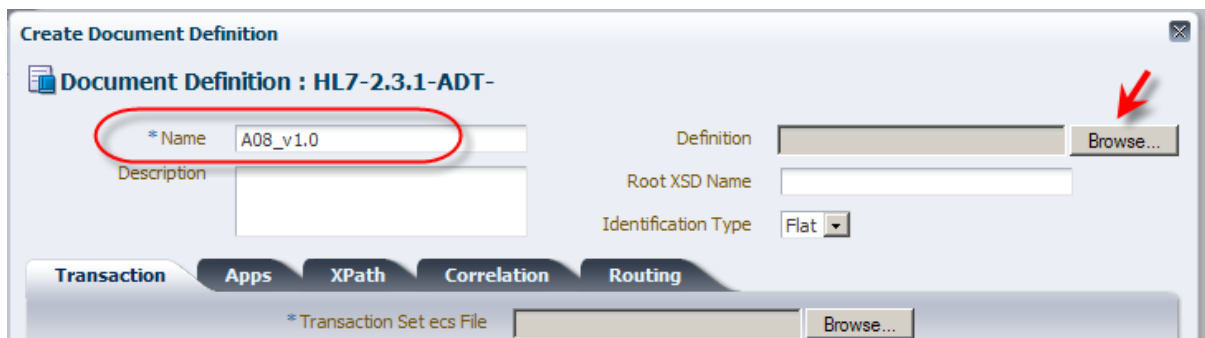
## Add A08\_v1.0 Document to Document Protocol Hierarchy

In this section we will introduce the ADT A08 document to the SOA Suite for healthcare integration document hierarch as HL7, v2.3.1, ADT\_A08 document.

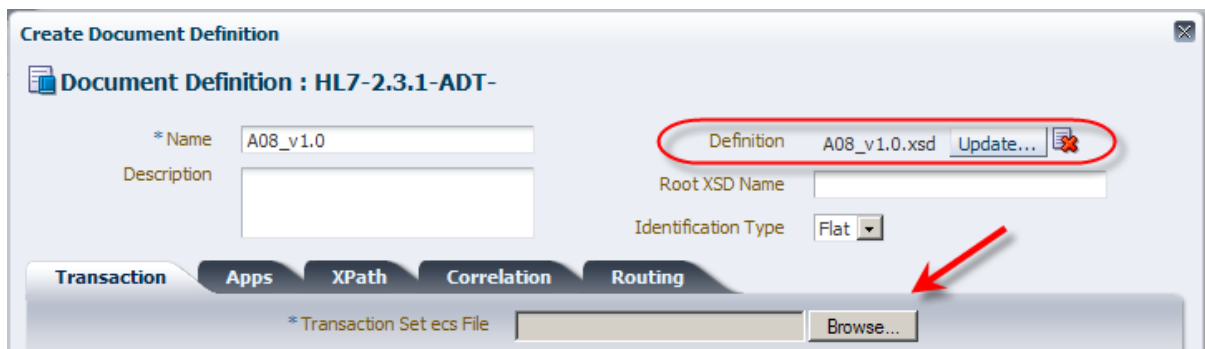
- Start the Healthcare Integration Console application in your favorite web browser – <http://localhost:7001/healthcare>.
- Log in with administrative credentials, for example weblogic/welcome1.
- Expand "Configuration" → "Document Protocol" → "HL7" → "2.3.1", right-click "ADT" and choose "Create"



- Enter "A08\_v1.0" into the Name field of the "Create Document" dialog box, click the "Browse" button alongside the "Definition" label, locate the XML Schema Definition file, A08\_v1.0.xsd, and select it

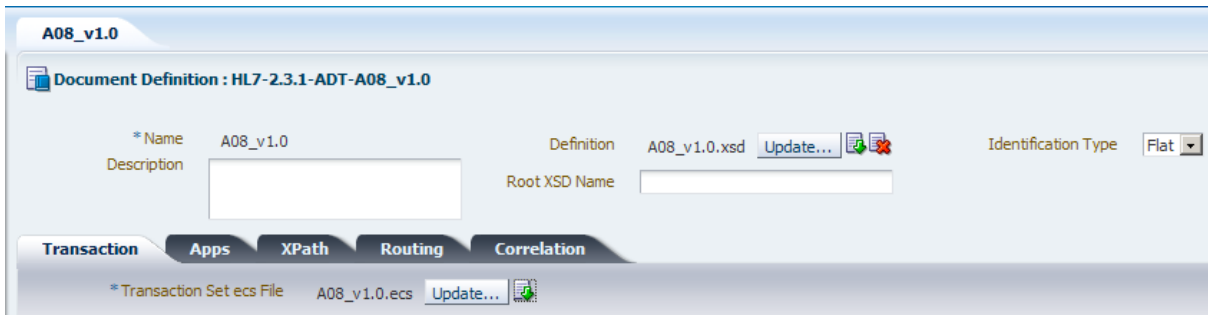


- Click the "Browse" button alongside the "Transaction Set ecs File" label, locate the ECS file, A08\_v1.0.ecs, and select it



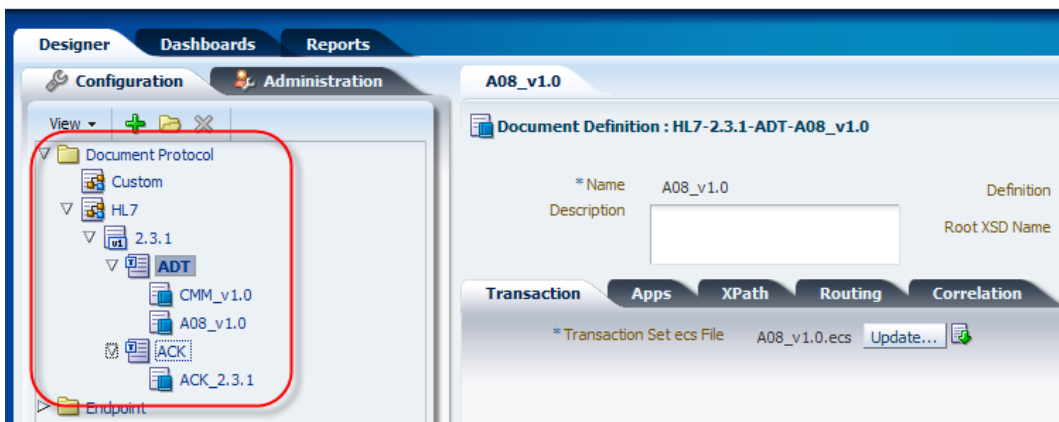
- Click "OK" to complete the dialogue

- Use the “Close” “Button” to close the “ADT”, “2.3.1” and “A08\_v1.0” Tabs – you will find the Healthcare Integration Console more responsive with fewer open tabs



Our document hierarchy should now look like that shown in the illustration.

### ORACLE SOA Suite for healthcare integration



## Second HL7 v2 Outbound (Sender)

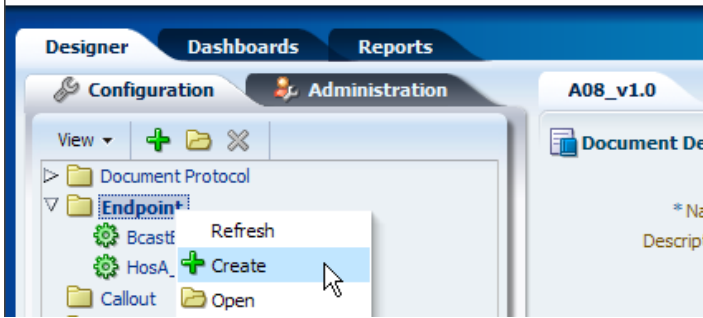
In this section we will configure the second sender endpoint, which will send HL7 v2 ADT A08 messages to the external receiver.

The sender endpoint will send messages of the A08\_v1.0 type to the external system.

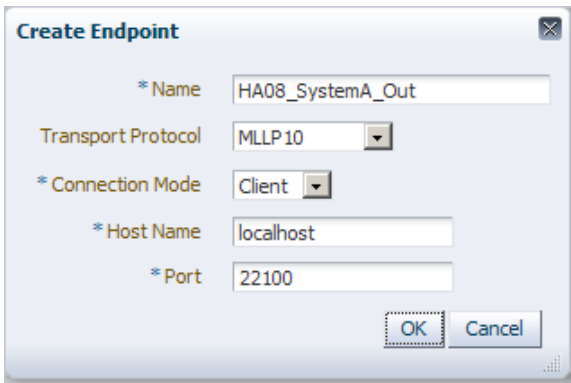
The sender will be a TCP/IP sender supporting the MLLP 1.0 encapsulation protocol. Such a sender establishes a connection to a listener which listens on a specific TCP/IP port, sends messages to it, receives acknowledgements back over the same connection, and processes them until either party disconnects.

In the sending mode the endpoint is configured as a “client”.

- In the “Healthcare Integration Console” right-click the “Endpoint” node in the “Configuration” tab and choose “Create”

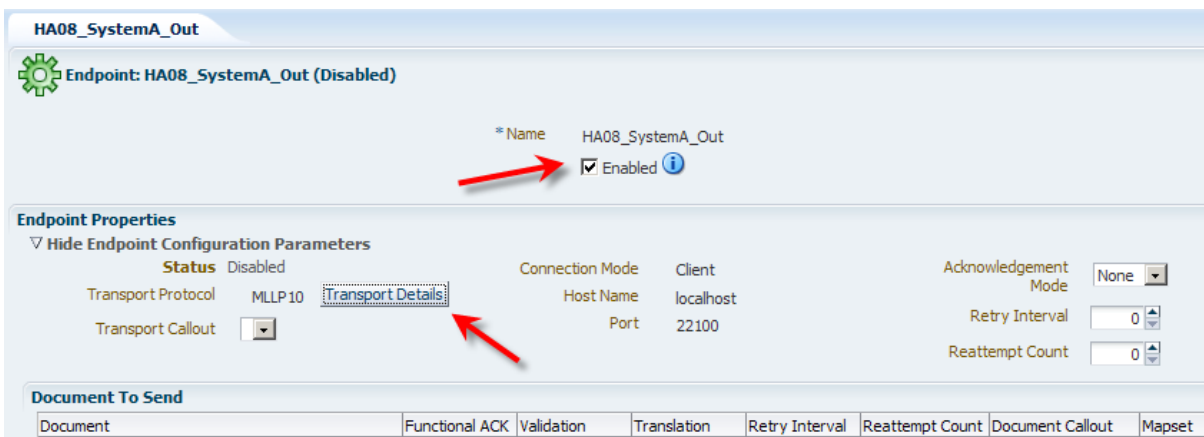


- Enter the following in the "Configure Endpoint" dialogue box then click "OK"
  - Name: HA08\_SystemA\_Out
  - Transport Protocol: MLLP10
  - Connection Mode: client
  - Host Name: localhost (or the name of whatever host you are using)
  - Port: 22100



The endpoint is not quite configured as we want it. We will change the non-default values to suit our requirement in the following steps.

- Check the "Enabled" checkbox. When we "Apply" this configuration later the endpoint will be started.
- Click the "Transport Details" button

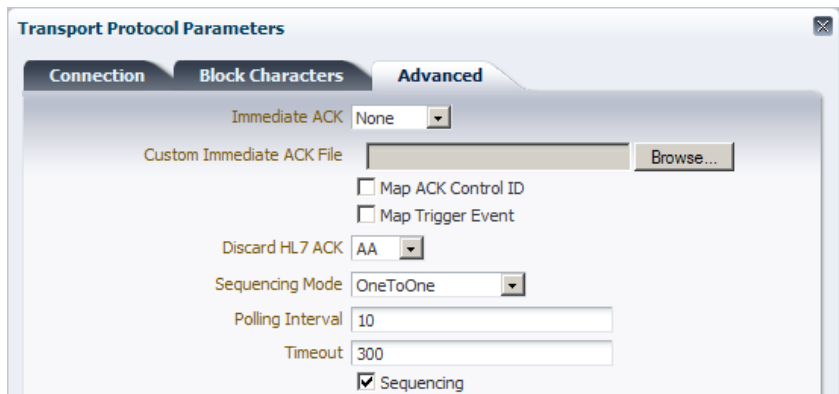


- Click the "Advanced" tab in the "Transport Protocol Parameters" dialogue box, set the

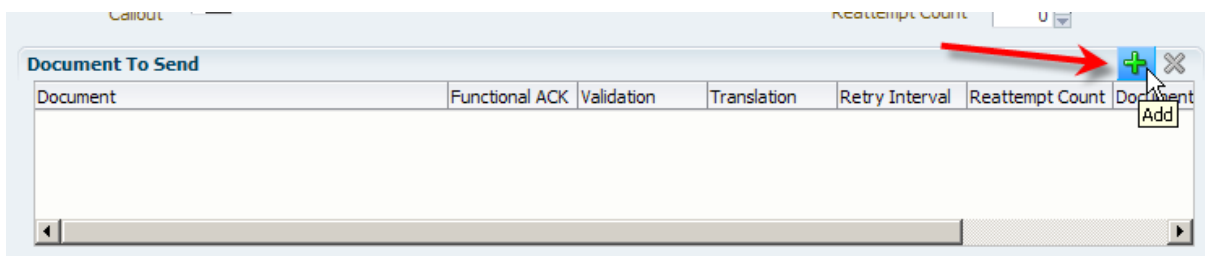


following properties, and click "OK":

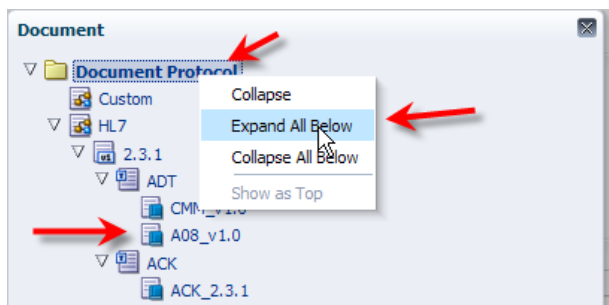
- Discard ACK: AA



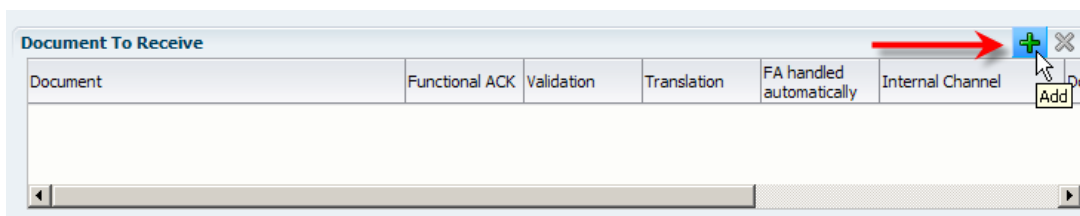
- Click the "Add" "button" (a plus sign) in the "Documents to Send" section



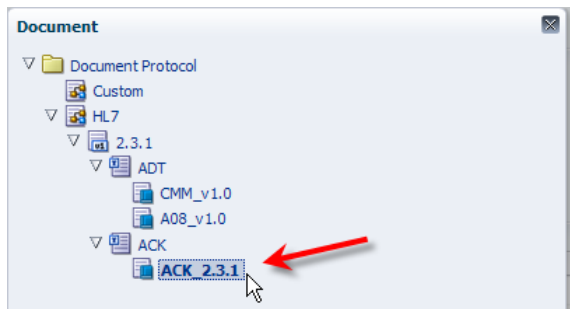
- Right-click the "Document Protocol" node in the "Document" dialog box and choose "Expand All Below"
- Select the "A08\_v1.0" document in the HL7→2.3.1→ADT hierarch and click "OK"



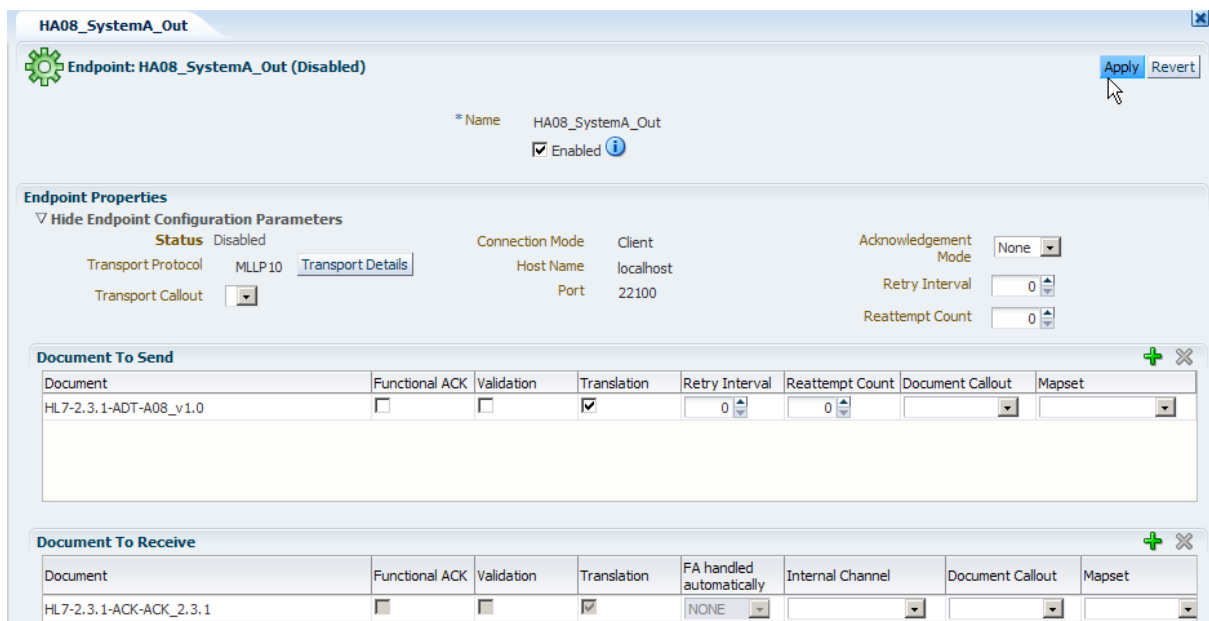
- Click the "Add" "button" (a plus sign) in the "Documents to Receive" section



- Right-click the "Document Protocol" node in the "Document" dialog box and choose "Expand All Below"
- Select the "ADT\_2.3.1" document in the HL7→2.3.1→ACK hierarch and click "OK"



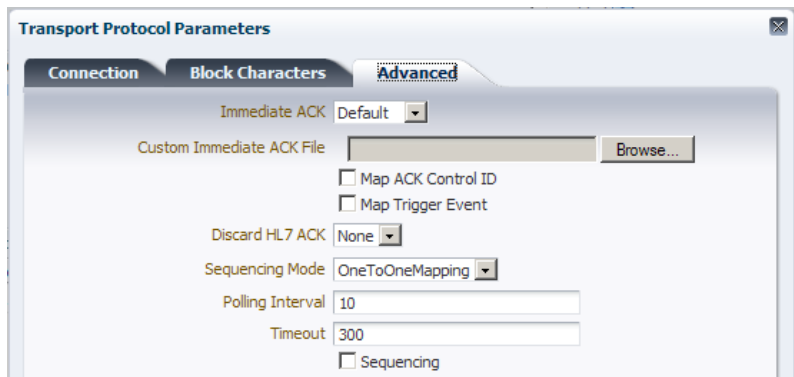
- Review the configuration to make sure it is correct and click the “Apply” button, remembering that with the “Enabled” checkbox checked this action will cause the SOA Suite for healthcare integration to attempt to start the endpoint



## HL7 v2 Inbound (Receiver)

In this section we will confirm the existence and configuration of the HosA\_SystemA\_In endpoint.

- Expand “Designer”→“Configuration”→“Endpoints” hierarchy, making sure that the endpoint “HosA\_SystemA\_In” exists and is enabled
- Click the “Transport Details” button, click the “Advanced” Tab, choose “Default” for the “Immediate ACK” setting then click “OK” to complete the dialogue



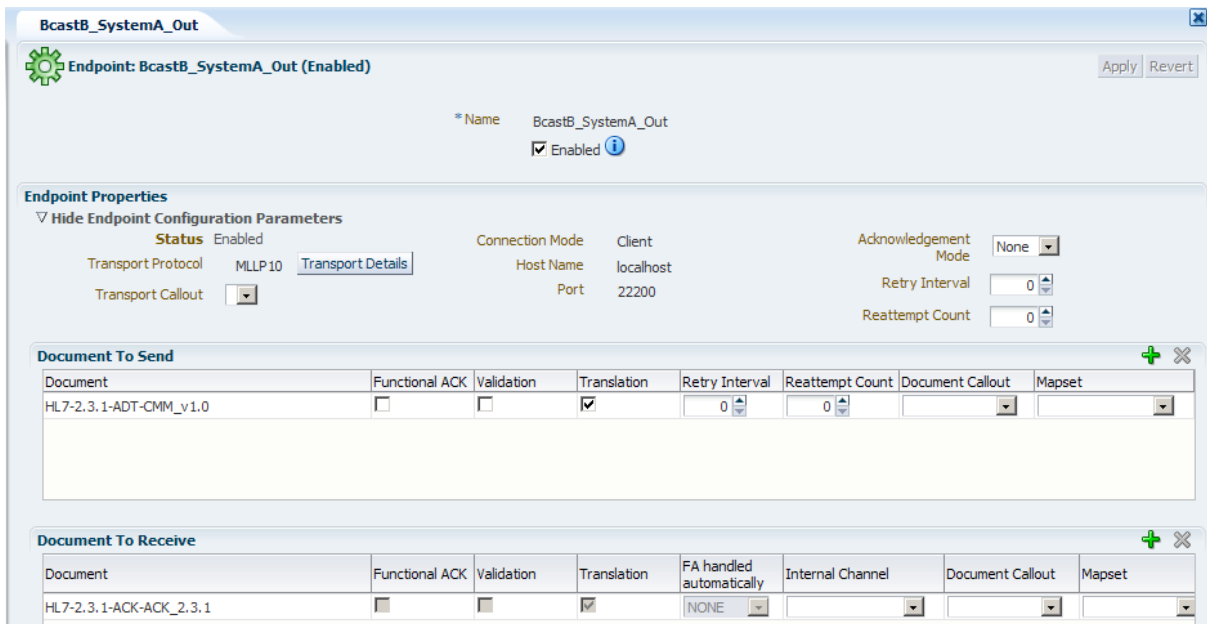
- Uncheck the "Translation" checkbox, uncheck the "Functional ACK" checkbox and change the "FA handled automatically" to NONE in "Document To Receive" part of the configuration panel and click "Apply" to apply changes

Document To Receive				
Document	Functional ACK	Validation	Translation	FA handled automatically
HL7-2.3.1-ADT-CMM_v1.0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NONE

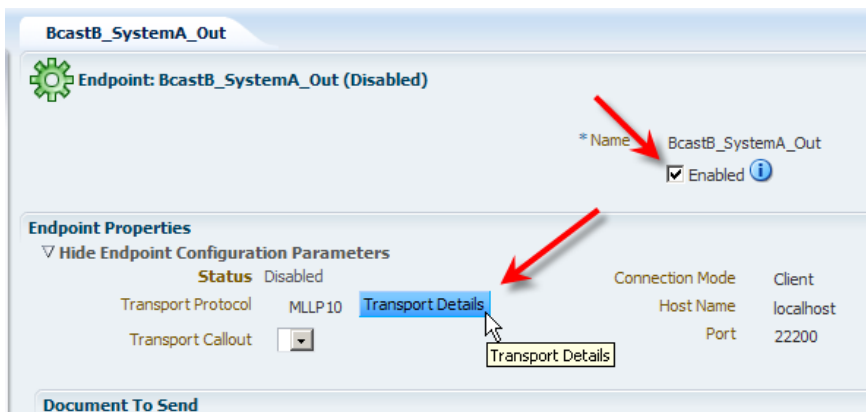
## Original HL7 v2 Outbound (Sender)

In this section we review the configuration of the original sender endpoint, which was configured in the previous blog article and used to pass through ATD messages to an external system.

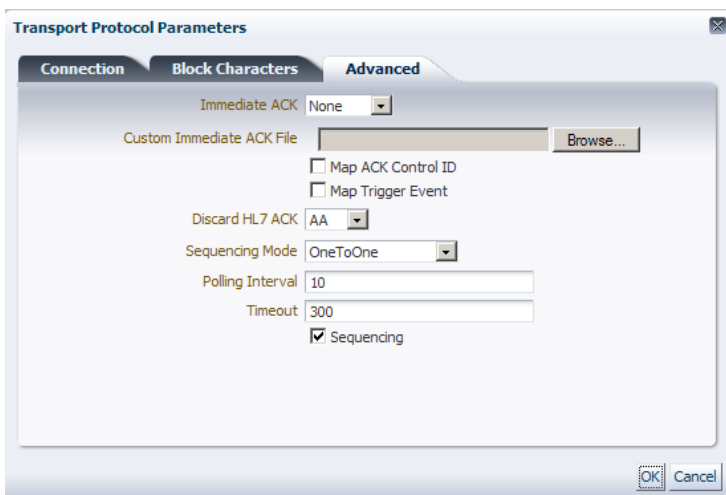
- Open the BcastB\_SystemA\_Out configuration panel by clicking the name of the endpoint under the "Configuratoon" Tab → "Endpoints" node. Verify that the following configuration is set
  - Transport Protocol: MLLP10
  - Connection Mode: client
  - Host Name: localhost (or the name of whatever host you are using)
  - Port: 22200



- Click the "Transport Details" button



- Click the "Advanced" tab in the "Transport Protocol Parameters" dialogue box, check the following properties, and click "OK":
  - Discard ACK: AA



- Verify that document "HL7-2.3.1-ADT-CMM\_v1.0" is configured in the "Document to Send" and that "Functional ACK" and "Validation" checkboxes are not checked, and that "Translation" checkbox is checked.
- Verify that document "HL7-2.3.1-ACK-ACK\_2.3.1" is configured in the "Document to Receive".

## Start Receiving "External Systems"

We will use the CMDHL7Listener command line client to receive HL7 ADT messages look at the output in the output directory specified on the listener's command line – for me c:\hl7\received. The CMDHL7Listener will display trace of message exchange in the console window. The SOA Suite for healthcare integration will record message tracking information which we will look at a later stage.

Please note that in this solution the CMDHL7Listener returns an ACK as soon as it gets the message.

- Check that your configured output directory is empty
- To start the listener for the "BcastB\_SystemA\_Out" sender endpoint, in a command / terminal window execute the following command

```
java -jar c:\tools\CMDHL7\CMDHL7Listener_v0.7.jar -c ID_ -p 22200 -s c:\hl7\received
```

- Inspect the CMDHL7Listener console output making sure the listener started and is listening on the appropriate port

```
08/12/2012 10:45:36 AM au.id.czapski.hl7.CMDHL7Listener main
INFO: Port: 22200
08/12/2012 10:45:36 AM au.id.czapski.hl7.CMDHL7Listener main
INFO: Store in: c:\hl7\received
08/12/2012 10:45:36 AM ca.uhn.log.HapiLogImpl info
INFO: au.id.czapski.hl7.SimpleACKApplication registered to handle **^*
messages
08/12/2012 10:45:36 AM ca.uhn.log.HapiLogImpl info
INFO: SimpleServer running on port 22200
```

- To start the listener for the "HA08\_SystemA\_Out" sender endpoint, in a command / terminal window execute the following command

```
java -jar c:\tools\CMDHL7\CMDHL7Listener_v0.7.jar -c ID_ -p 22100 -s c:\hl7\received
```

## HL7 v2 Router and Transformer

In this segment of the article, the SOA Composite called the "Router" will "route" each message it receives to an external HL7 receiver via a HL7 Sender endpoint, without changing it, and will also pass copies to a "Transformer" component to be transformed into corresponding ADT A08 messages and sent to another external HL7 receiver.

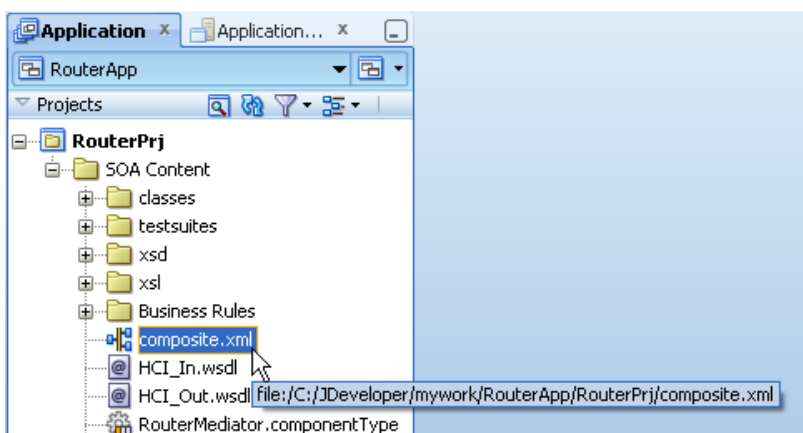
We will extend the basic "Router" composite, which was developed I the previous article, to provide additional functionality.

Tying together adapters in SOA Suite requires a composite application which implements business logic - BPEL, Mediator, Business Rules, ... . The role of this application is to receive a message from the inbound adapter and pass it to one or more outbound adapters,

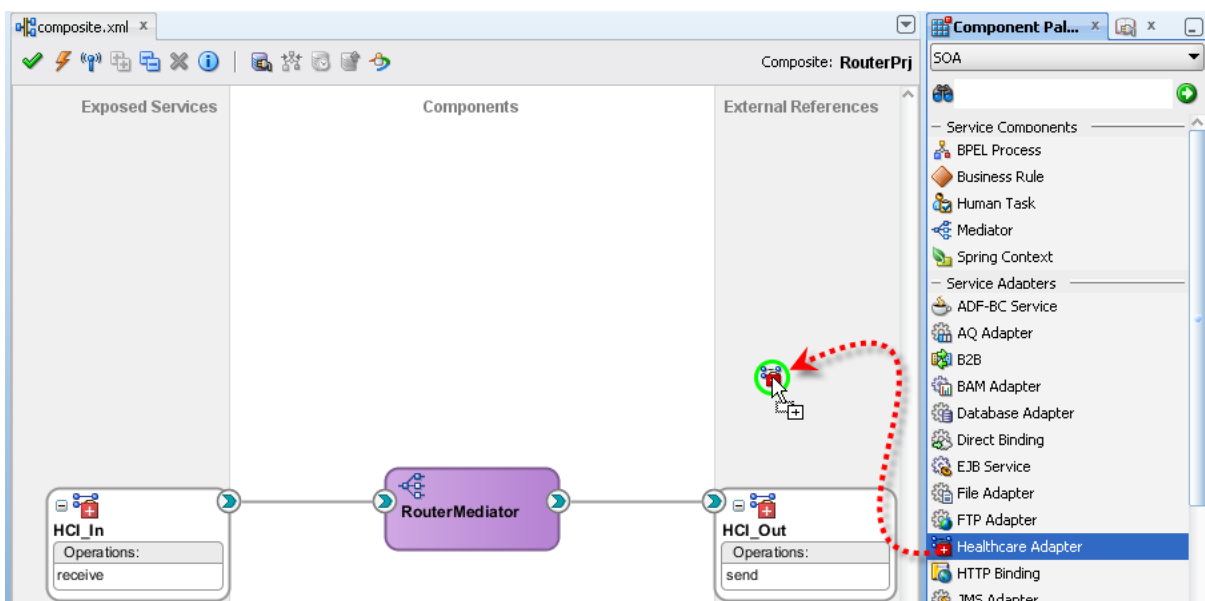
potentially transforming it in the process. We already have this application from the previous work.

Before proceeding with configuration make sure that WebLogic Server is running. JDeveloper will connect to the backend repository to obtain information about configured documents and endpoints.

- Start the JDeveloper Studio IDE
- Pull down the "Application" menu and choose "Open..."
- Locate the file "RouterApp.jws", select it and click "Open" to open it in the JDeveloper Studio
- Expand the "Projects" --? "RouterPrj" → "SOA Content" hierarch and double-click the "composite.xml" object to open the composite application designer



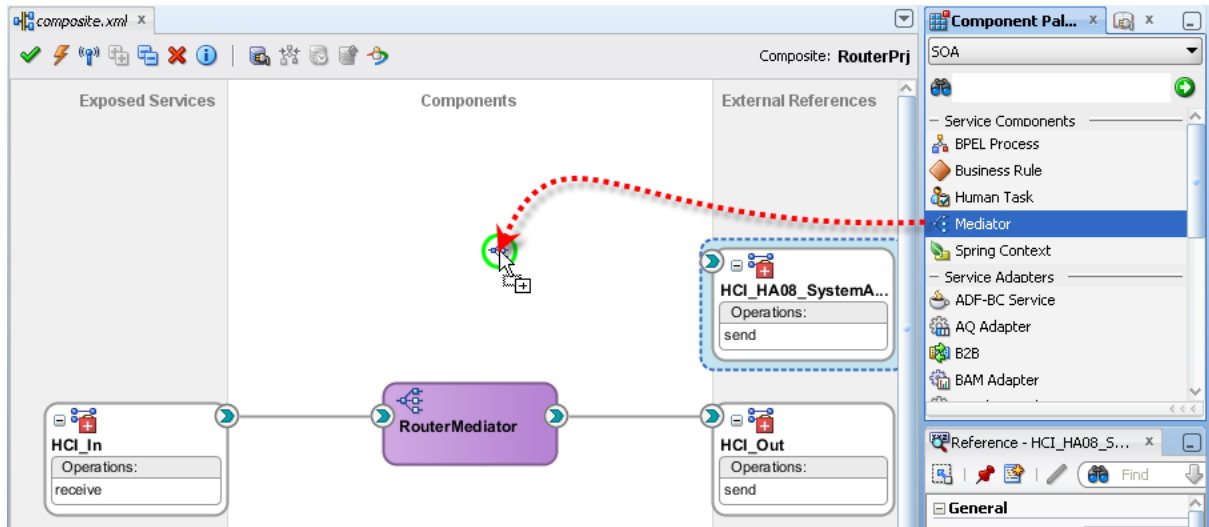
- Drag the "healthcare adapter" Service Adapter to the canvas in the "External References" swim line to provide a sender for the new receiver



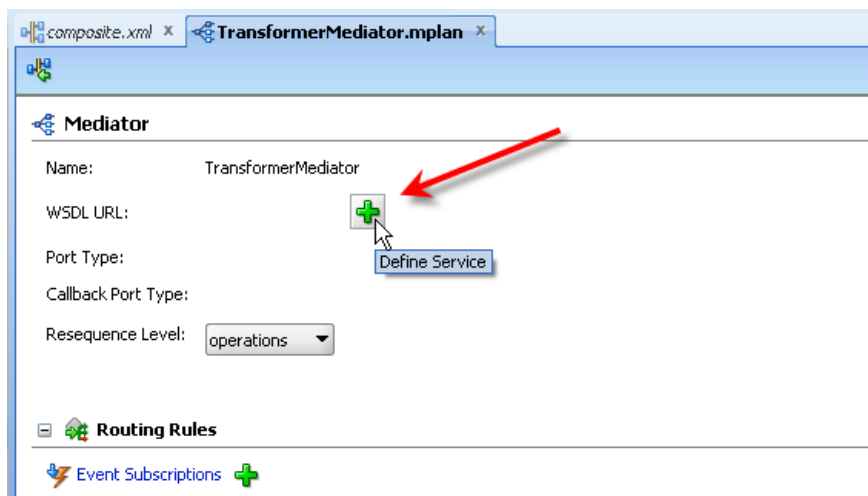
- Configure the endpoint as follows:
  - Service Name: HCI\_HA08\_SystemA\_Out
  - Operation: Send

- Document Definition: HL7 → 2.3.1 → ADT → A08\_v1.0

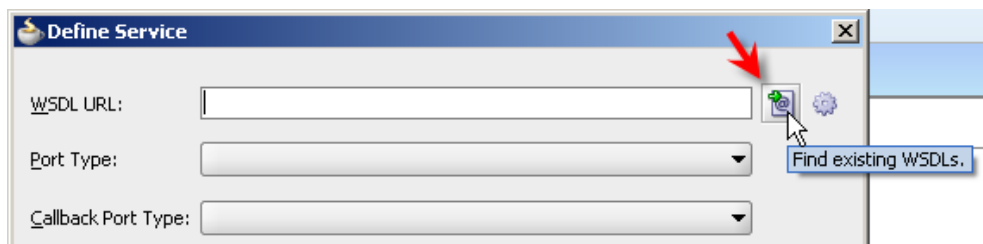
- Drag the Mediator Service Component to the canvas and drop it into the “Components” swim line



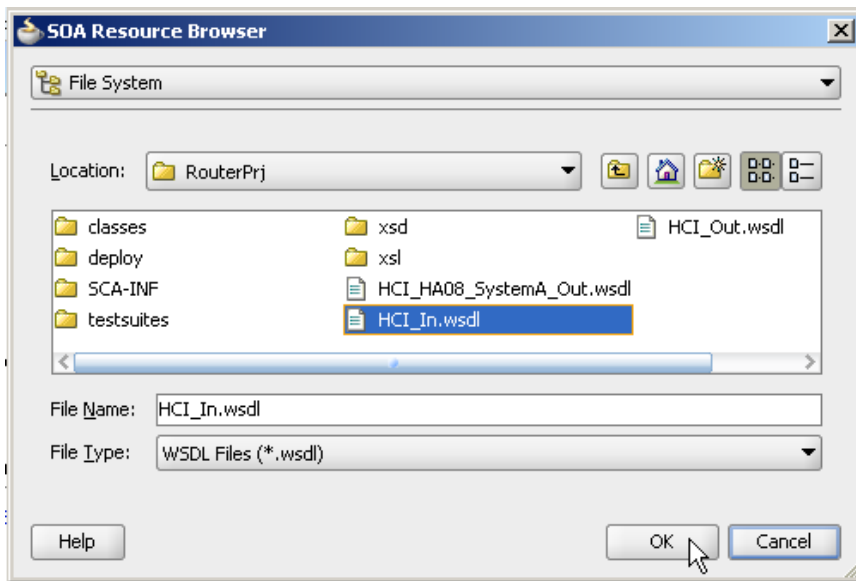
- Name this component “TransformerMediator” and click “OK”
- Right-click on the new “TransformerMediator” component and choose “Edit”
- Start the “Define Service” wizard by clicking on the “Define Service” icon, the plus sign



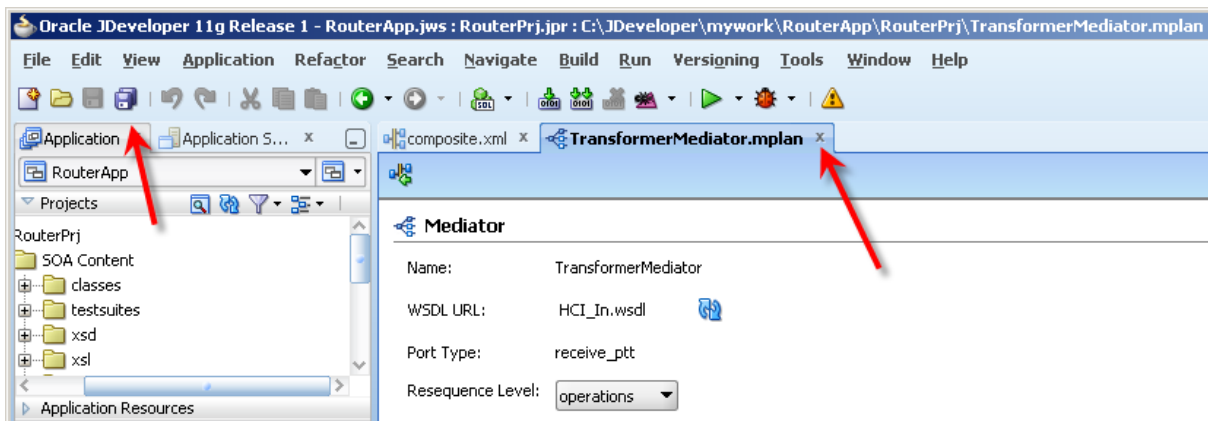
- Click the “Find existing WSDLs.” Button



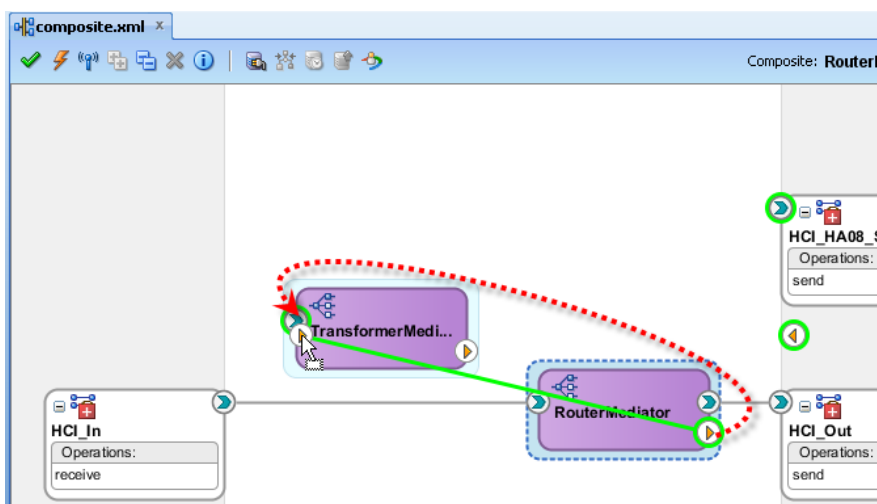
- Choose “HCI\_In.wsdl” and click “OK” then “OK” again



- Click "Save All" then "Close" on the "TransformerMedisator.mplan"

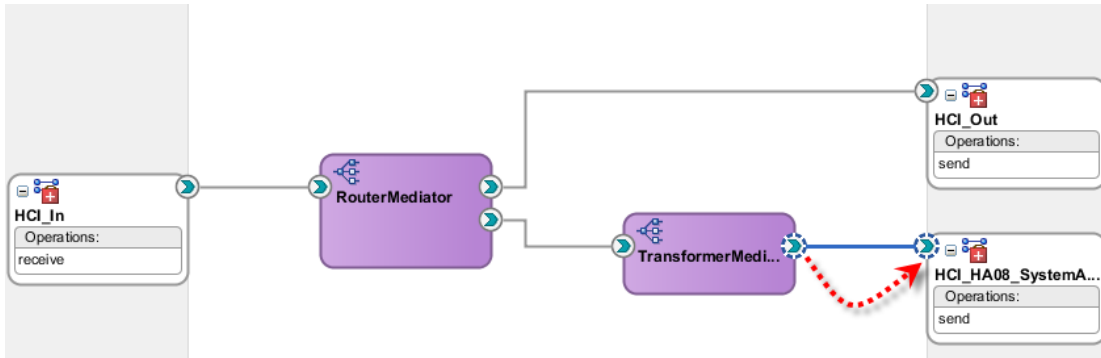


- Connect "RouterMediator" to "TransformerMediator" by dragging from the one to the other connecting the output triangle to the input triangle



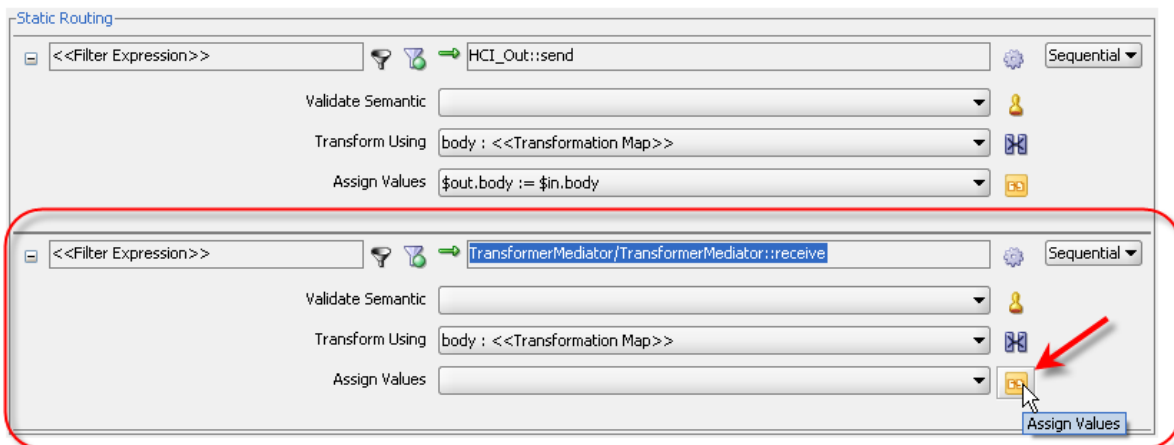
- Connect output of the "TransformerMediator" to the input on the "HCI\_HA08\_SystemA\_Out" adapter – the canvas will be redrawn with a better layout



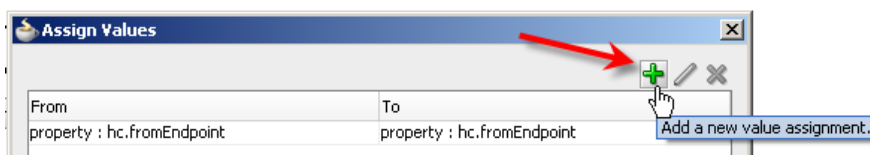


We need to pass certain properties, to which the "RouterMediator" has access and which it can use in its message processing, on to the "TransformerMediator" so it too can use the values of these properties if needed. There are several properties which the inbound healthcare adapter makes available. We will only use a few. The method will hold for all the properties.

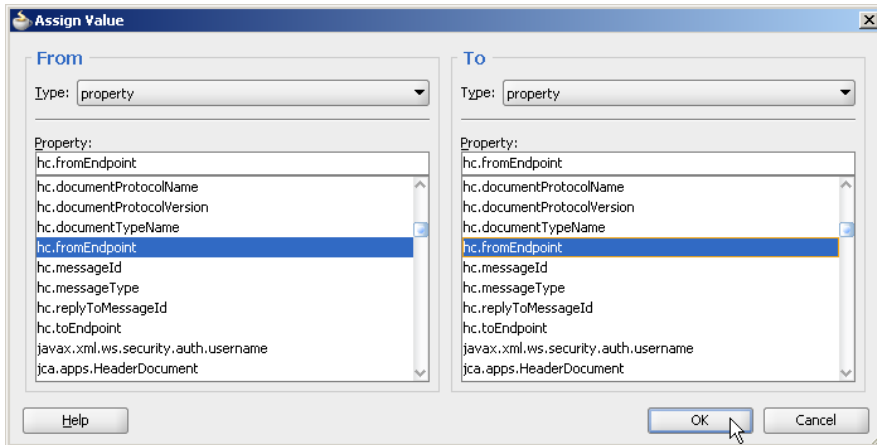
- Double-click on the "RouterMediator" to edit its properties
- Click the "Assign Values" button in the "TransformerMediator/TransformerMediator::receive" route to add properties which need to be forwarded to the "TransformerMediator" for it to use in routing to its endpoint



- Add three value assignments, assigning the following properties:



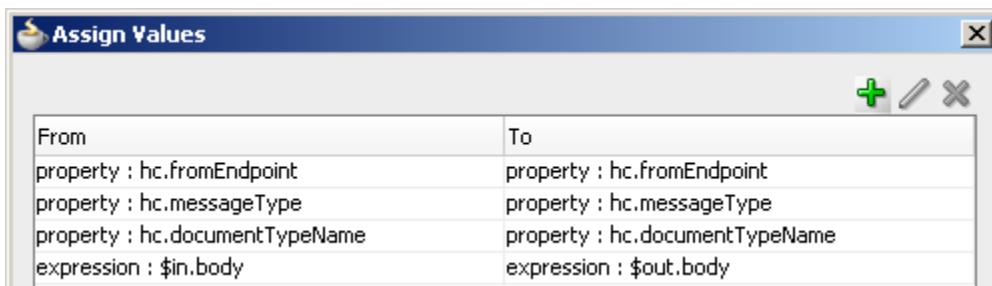
- o hc.fromEndpoint → hc.fromEndpoint



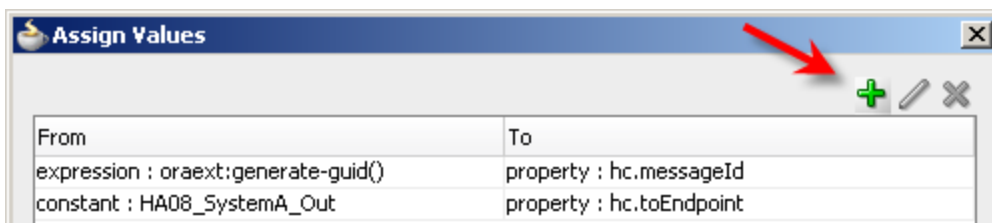
- hc.messageType → hc.messageType
- hc.documentTypeName → hc.documentTypeName

Add a value assignment, assigning the following expression:

- \$in.body → \$out.body



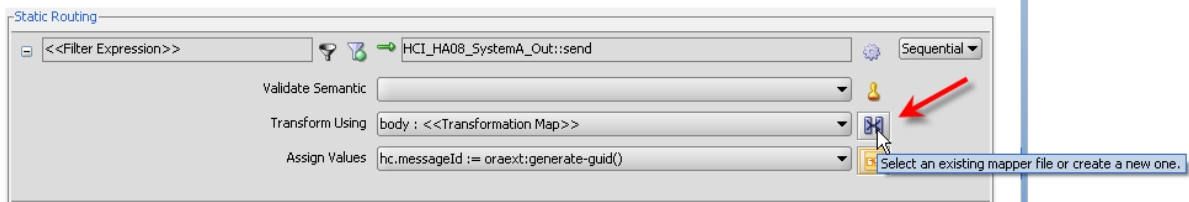
- Click the "Save All" button and close the "RouterMediator.mplan"
- Double-click the "TransformerMediator" in the composite editor to open its mplan, then click the "Assign Values" button to assign an expression and a constant which the "TransformerMediator" needs to pass to the healthcare adapter to allow it to find the correct route for sending messages and to make the identifier of the copy of the message unique
- Assign expression "generate-guid()" to property hc.messageId, and the constant "HA08\_SystemA\_Out" to the property hc.toEndpoint



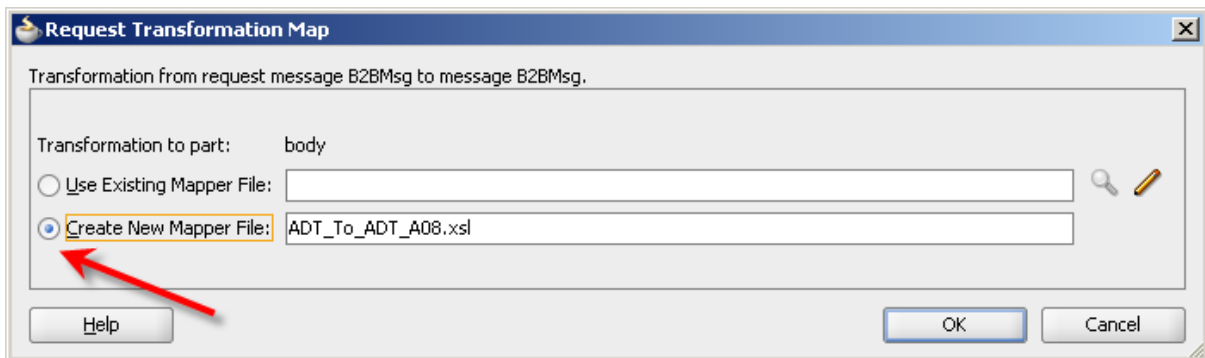
We need to transform the ADT A01 or ADT A03 to ADT A08. Since our A01, A03 and A08 have the same structure in terms of segments, fields, components and subcomponents we can simply copy entire segments and modify specific fields and components which are different. The fields and components which differ are: MSH.9.MSG.2 (which needs to have a literal value A08), MSH.10 (which needs to have a new, unique message id value) and EVN.1 (which needs to have a literal A08 as well). All other content will be the same as the

original message from which it came. While there are at least 4 different ways in which this kind of mapping can be implemented in SOA Suite we will use the XSL transformation for the purpose of this exercise. I may or may not provide worked through examples of other mapping technologies in subsequent articles.

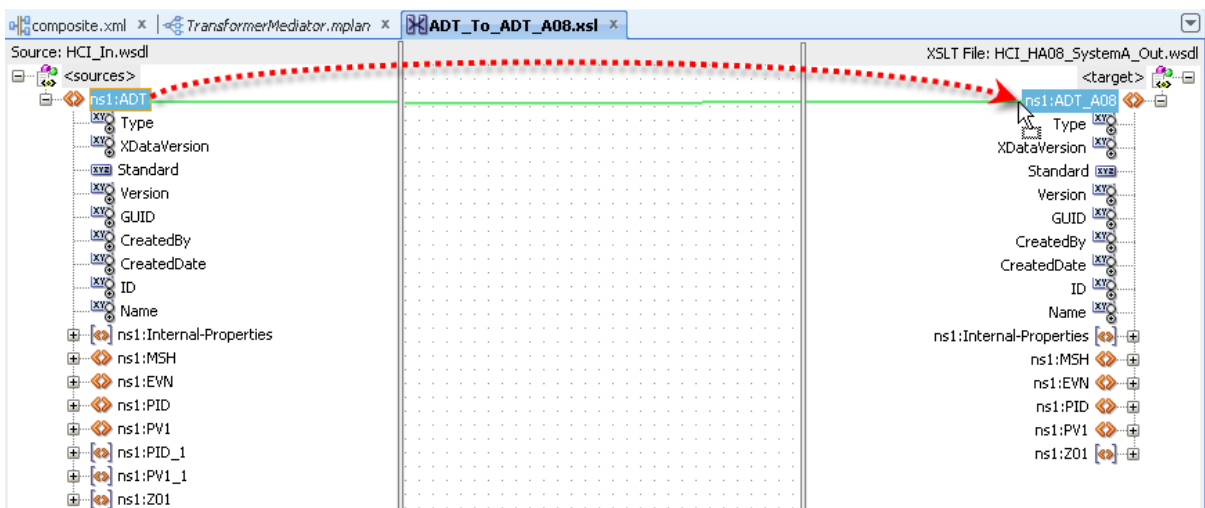
- ❑ Still in the "TransformerMediator.mplan" click the "Mapper" button to create a new mapper file



- ❑ Select the "Create New Mapper File" radio button and click "OK", accepting the name which JDeveloper generated

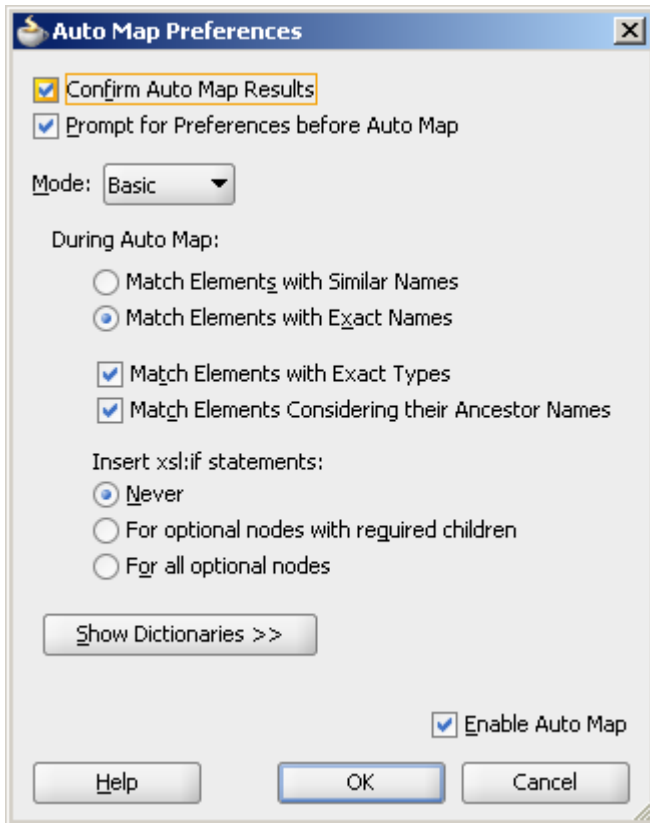


- ❑ Drag from the node "ns1:ADT" in the left hand side element tree to the node "ns1:ADT\_A08" in the right hand side node tree

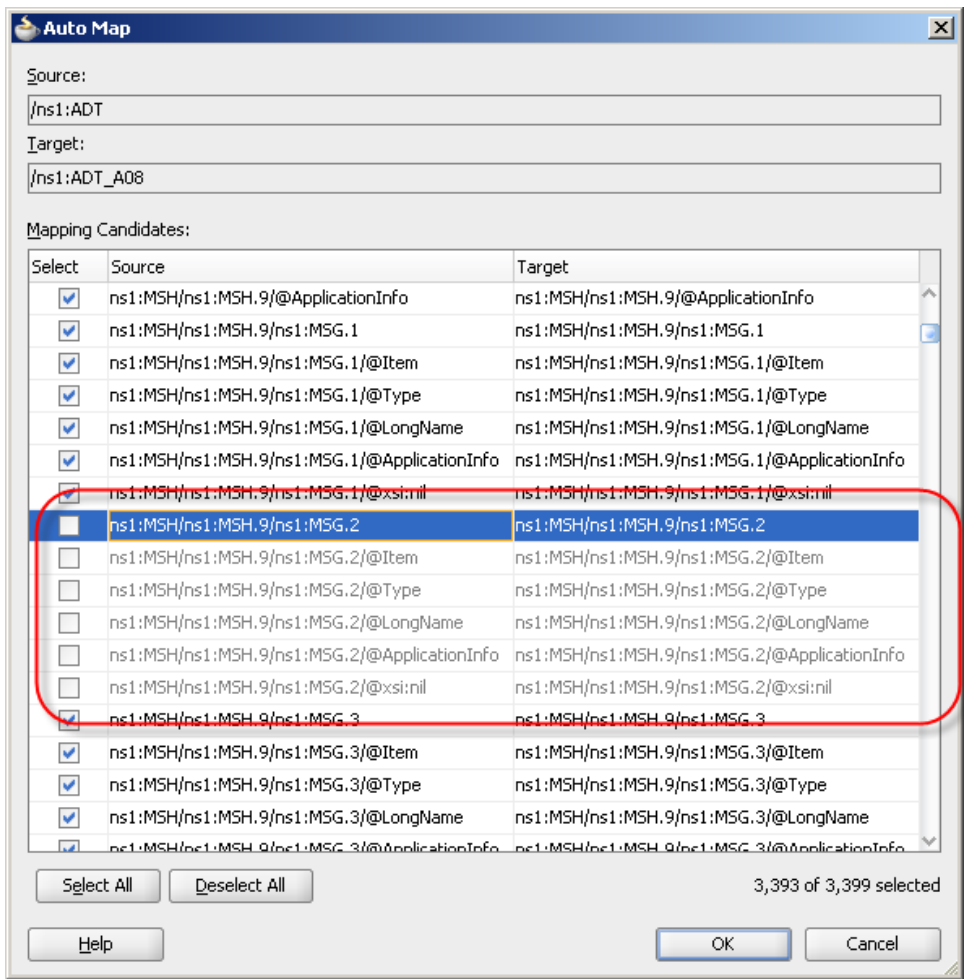


- ❑ Make sure the following checkboxes and radio buttons are checked then click "OK" to begin auto mapping process
  - Confirm Auto Map Results
  - Prompt for Preferences before Auto Map

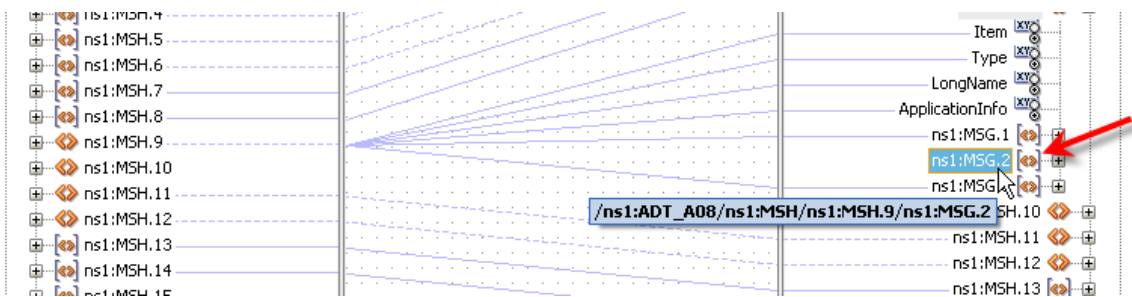
- Match Elements with Exact Names
- Match Elements with Exact Types
- Match Elements Considering their Ancestor Names
- Never
- Enable Auto Map



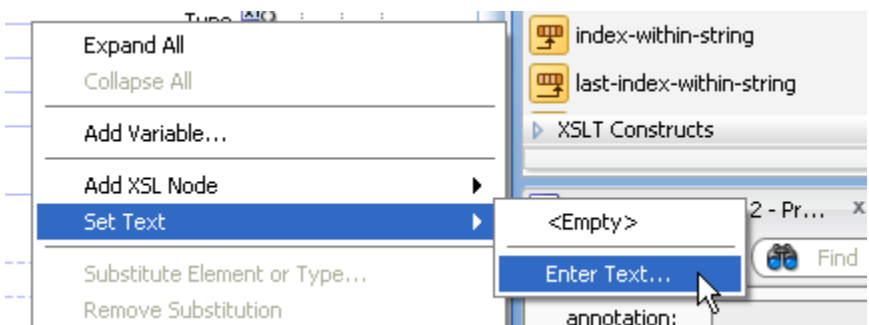
- Dismiss the dialogue box about element recursion by clicking “OK”
- Scroll through the list of nodes being auto mapped, uncheck the following nodes and their sub-nodes, if any, and click “OK” to accept the results
  - MSH.9.MSG.2
  - MSH.10
  - EVN.1



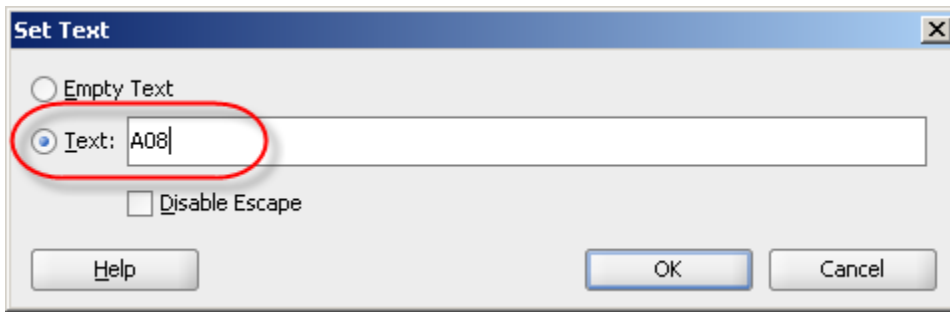
- To set the literal "A08" as the value of the node MSH.9.MSG.2 expand the right-hand node tree through MSH→MSH.9→MSH.9.MSG.2



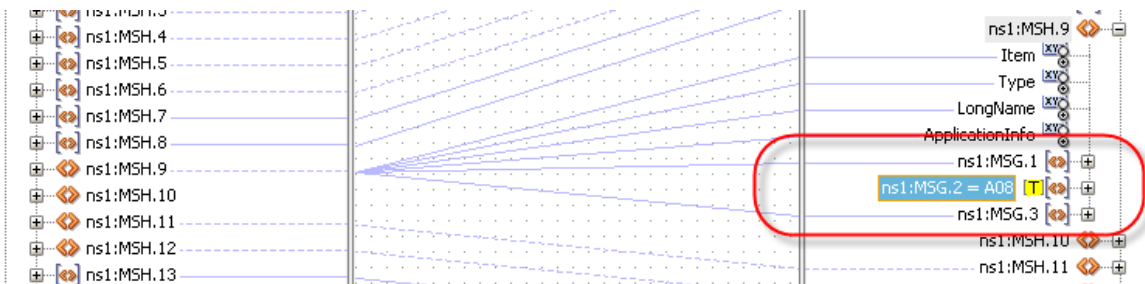
- Right-click ns1:MSG.2 and choose "Set Text" → "Enter Text..."



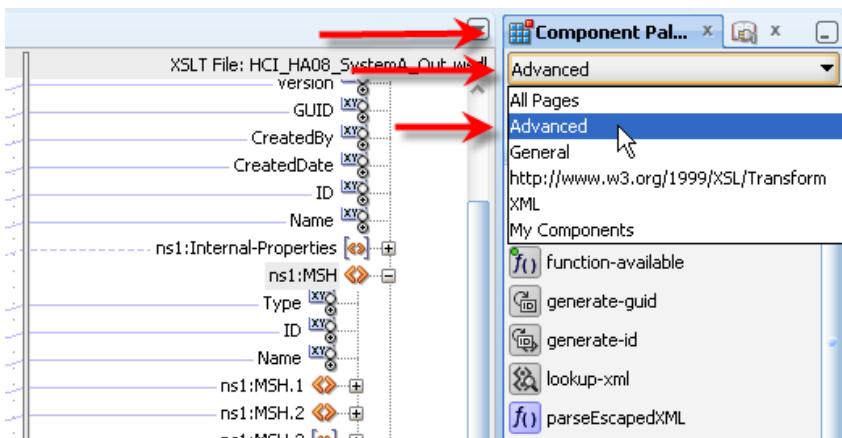
- Type literal "A08" and click "OK"



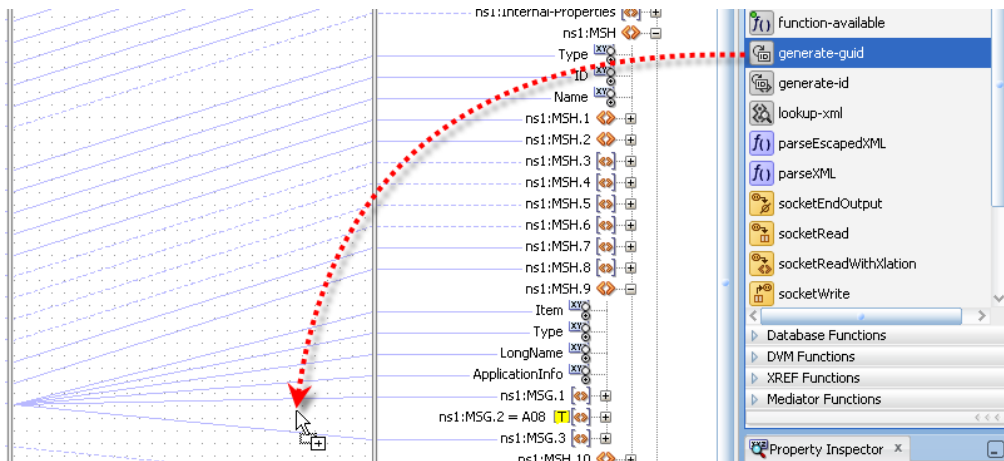
- Ensure the value is set as shown



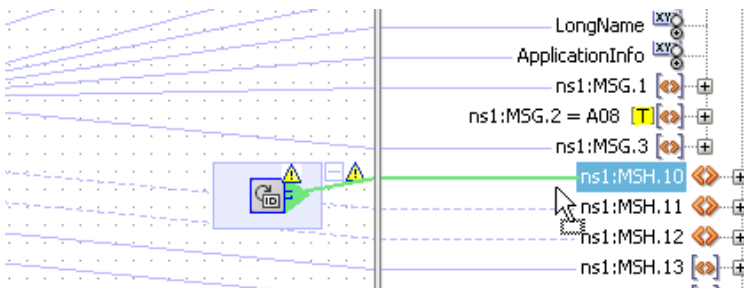
- To assign the unique message Id expand the right-hand MSH node tree until the ns1:MSH.10 field is visible
- In the "Component Palette" dropdown choose "Advanced" then expand "Advanced Functions" node



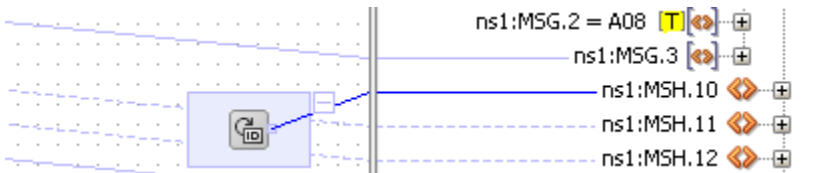
- Drag the "generate-guid" function to the central swim line opposite the ns1:MSH.10 node



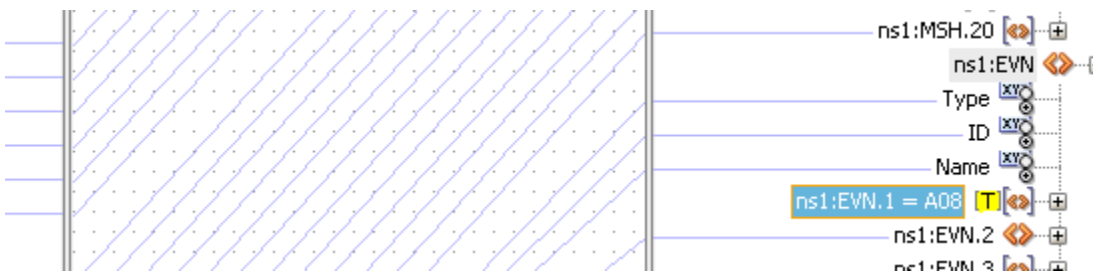
- Drag from the tiny square at the right of the function to the ns1:MSH.10 node to connect the two



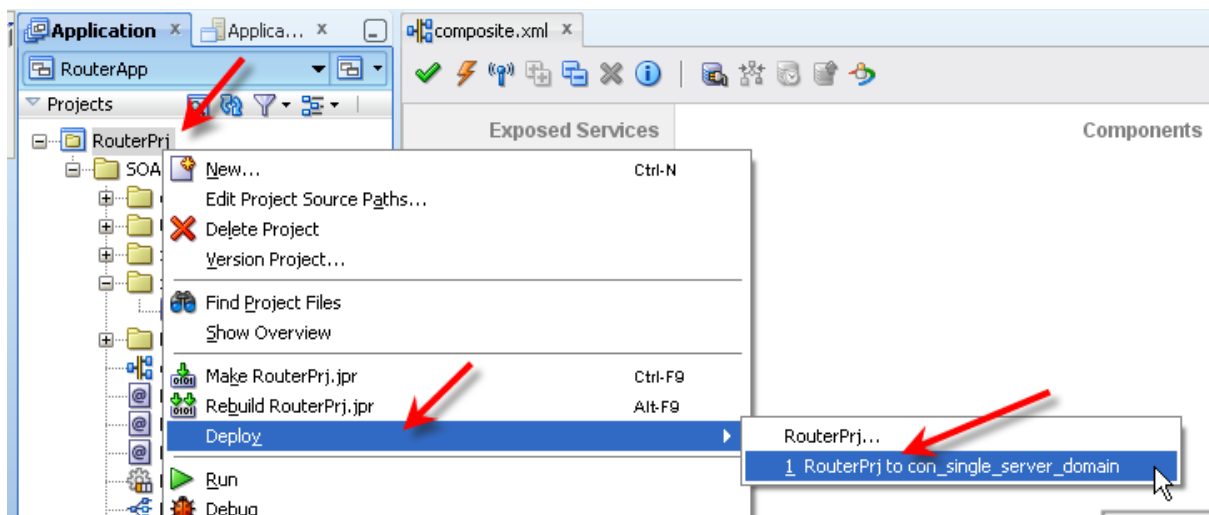
- Confirm that the function value is being assigned to the correct node



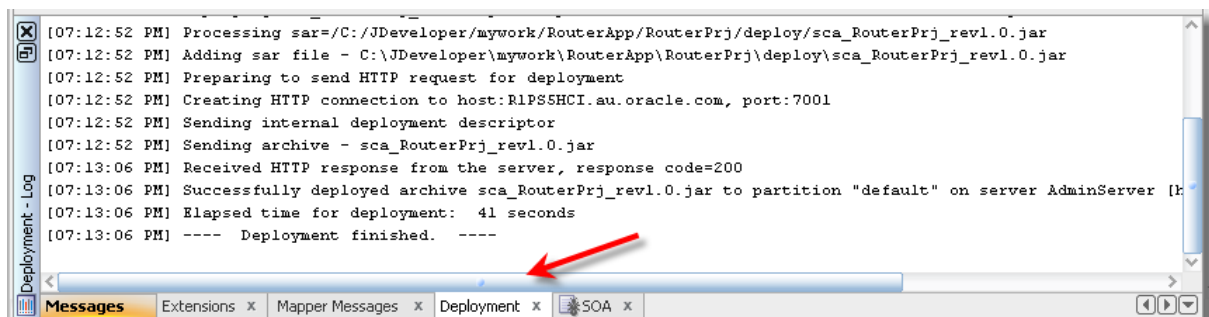
- Set the literal value "A08" in the field EVN.1



- Click the "Save All" button to save all work then close the XSL designer and the mplan designer
- Right-click "RouterPrj" node in the project tree, choose "Deploy" and deploy the project



- Verify that the project deployed successfully



The application is deployed and ready to accept and write messages.

At this point you can close the JDeveloper Studio IDE. We don't need it for the moment.

## Send ADT messages

As before, we will use the CMDHL7Sender command line client to read files containing a single HL7 ADT message and submit them to the ADT Receiver endpoint. We will then look at the console output produced by the CMDHL7Listener which we started earlier, then look at the output in our configured output directory – for me c:\hl7\received, and finally review message tracking information in the Healthcare Integration Console.

Please note that in this solution the receiver endpoint returns immediate ACK as soon as it gets the message. There may be a delay, most noticeable the first time one executes the processing flow after application server restart, between the receipt of the ACK and the time the message is written to a file in the file system.

- Check that your configured output directory is empty
- Locate the input file containing a single HL7 message - for me this will be C:\hl7\adt\sources\ADT\_A01\_output\_1.hl7

The content of my file, where each segment starting with the 3 character segment ID in bold text is a single line up to the next 3 character segment ID, looks like this:



```
MSH|^~\&|SystemA|HosA|PI|MDM|2008090801529||ADT^A01|000000_CTLID_2008
090801529|P|2.3.1||AL|NE
EVN|A01|2008090801529||JavaCAPS6^^^^^^^USERS
PID|1||A000010^^^HosA^MR^HosA||Kessel^Abigail||19460101123045|M|||7
South 3rd Circle^^Downham Market^England -
Norfolk^30828^UK|||||||A2008090801529
PV1|1|I||I||FUL^Fulde^Gordian^^^^^^^^^^MAIN||EMR|||||||V200809080
1529^^^^VISIT|||||||2008090801529
```

- In a command / terminal window execute the following command

```
java -jar c:\tools\CMDHL7\CMDHL7Sender_v0.7.jar -a SystemA -b HosA -c ID_
-n 1 -d \r\r\n -p 22222 -h localhost -t 30000 -f
c:\hl7\adt\sources\ADT_A01_output_1.hl7
```

- Locate the output files in the received directory and inspect them to confirm that a) A01 or A03 and A08 have been written and b) that A01/A03 has the same content as the input file and the A08 has the same content as the input file except for the tree changes we made in the "TransformerMediator" component.

The content of the A01 output file, where each segment starting with the 3 character segment ID in bold text is a single line up to the next 3 character segment ID, looks like this:

```
MSH|^~\&|SystemA|HosA|PI|MDM|2008090801529||ADT^A01|ID__000000|P|2.3
.1||AL|NE
EVN|A01|2008090801529||JavaCAPS6^^^^^^^USERS
PID|1||A000010^^^HosA^MR^HosA||Kessel^Abigail||19460101123045|M|||7
South 3rd Circle^^Downham Market^England -
Norfolk^30828^UK|||||||A2008090801529
PV1|1|I||I||FUL^Fulde^Gordian^^^^^^^^^^MAIN||EMR|||||||V200809080
1529^^^^VISIT|||||||2008090801529
```

The content of the file named "ID\_\_000000\_ADT\_A01\_1356512788894.hl7" is the same as the message which was sent. The only difference is the message control id, which the send command explicitly changed with the -c switch to a serial number prefixed by "ID\_\_".

The content of the A08 output file looks like this:

```
MSH|^~\&|SystemA|HosA|PI|MDM|2008090801529||ADT^A08|38313332333630313
831353734383532|P|2.3.1||AL|NE
EVN|A08|2008090801529||JavaCAPS6^^^^^^^USERS
PID|1||A000010^^^HosA^MR^HosA||Kessel^Abigail||19460101123045|M|||7
South 3rd Circle^^Downham Market^England -
Norfolk^30828^UK|||||||A2008090801529
PV1|1|I||I||FUL^Fulde^Gordian^^^^^^^^^^MAIN||EMR|||||||V200809080
1529^^^^VISIT|||||||2008090801529
```

The content of the file named "38313332333630313831353734383532\_ADT\_A08\_1356512791863.hl7" is the same as the message which was sent except for the fields and components which we explicitly modified and which are marked as bold italic text.

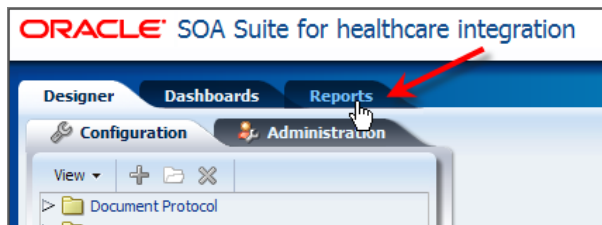
- Submit the ADT A03 file, ADT\_A03\_output\_1.hl7, and inspect the output.

Our solution works to the extent of receiving HL7 v2.3.1 messages, and acknowledging them, transforming ADT messages to A08 for one stream of processing and writing them to files in the file system.

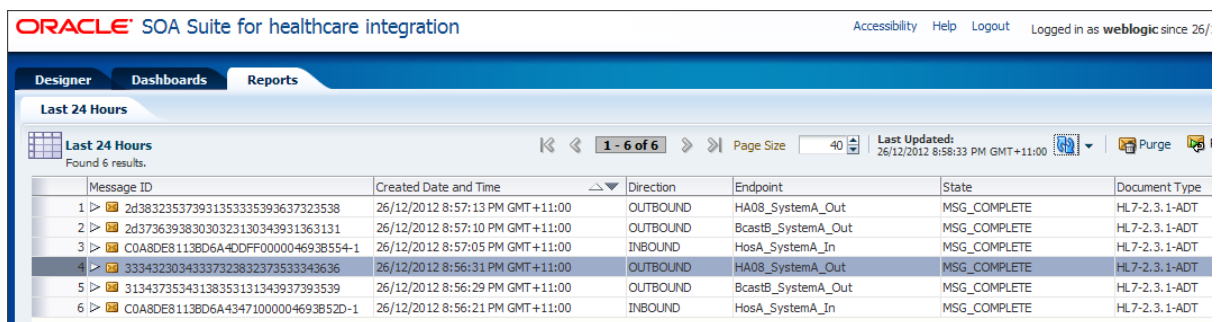
## Explore Message Tracking

Explore message tracking.

- Start the Healthcare Integration Console – <http://localhost:7001/healthcare>
- Log in with your credentials – mine are weblogic/welcome1
- Click the “Reports” Tab



- Note two set of three messages for ADT A01 and A03.

A screenshot of the Oracle SOA Suite for healthcare integration web interface showing a message tracking report. The report is titled 'Last 24 Hours' and shows 6 results. The report includes a table with columns for Message ID, Created Date and Time, Direction, Endpoint, State, and Document Type. The first message is highlighted in blue.

Message ID	Created Date and Time	Direction	Endpoint	State	Document Type	
1 > [icon]	2d383235373931353335393637323538	26/12/2012 8:57:13 PM GMT+11:00	OUTBOUND	HA08_SystemA_Out	MSG_COMPLETE	HL7-2.3.1-ADT
2 > [icon]	2d373639383030323130343931363131	26/12/2012 8:57:10 PM GMT+11:00	OUTBOUND	BcastB_SystemA_Out	MSG_COMPLETE	HL7-2.3.1-ADT
3 > [icon]	C0A8DE8113BD6A4DDFF0000046938554-1	26/12/2012 8:57:05 PM GMT+11:00	INBOUND	HosA_SystemA_In	MSG_COMPLETE	HL7-2.3.1-ADT
4 > [icon]	33343230343337323832373533343636	26/12/2012 8:56:31 PM GMT+11:00	OUTBOUND	HA08_SystemA_Out	MSG_COMPLETE	HL7-2.3.1-ADT
5 > [icon]	31343735343138353131343937393539	26/12/2012 8:56:29 PM GMT+11:00	OUTBOUND	BcastB_SystemA_Out	MSG_COMPLETE	HL7-2.3.1-ADT
6 > [icon]	C0A8DE8113BD6A43471000004693852D-1	26/12/2012 8:56:21 PM GMT+11:00	INBOUND	HosA_SystemA_In	MSG_COMPLETE	HL7-2.3.1-ADT

- Select the first OUTBOUND message in the list (it ought to be the outbound ADT A08 message)

Designer Dashboards Reports

Last 24 Hours

Last 24 Hours Found 6 results. Page Size 40 Last Updated: 26/12/2012 8:58:33 PM GMT+11:00

Message ID	Created Date and Time	Direction	Endpoint	State	Docur
1 > 2d383235373931353335393637323538	26/12/2012 8:57:13 PM GMT+11:00	OUTBOUND	HA08_SystemA_Out	MSG_COMPLETE	HL7-2
2 > 2d373639383030323130343931363131	26/12/2012 8:57:10 PM GMT+11:00	OUTBOUND	BcastB_SystemA_Out	MSG_COMPLETE	HL7-2
3 > C0A8DE8113BD6A4DFF000004693B554-1	26/12/2012 8:57:05 PM GMT+11:00	INBOUND	HosA_SystemA_In	MSG_COMPLETE	HL7-2
4 > 33343230343337323832373533343636	26/12/2012 8:56:31 PM GMT+11:00	OUTBOUND	HA08_SystemA_Out	MSG_COMPLETE	HL7-2
5 > 31343735343138353131343937393539	26/12/2012 8:56:29 PM GMT+11:00	OUTBOUND	BcastB_SystemA_Out	MSG_COMPLETE	HL7-2
6 > C0A8DE8113BD6A43471000004693B52D-1	26/12/2012 8:56:21 PM GMT+11:00	INBOUND	HosA_SystemA_In	MSG_COMPLETE	HL7-2

Business Message Details

Identification

Endpoint HA08\_SystemA\_Out Message ID 2d383235373931353335393637323538  
 State MSG\_COMPLETE Label soa\_b2b\_ - Wed Dec 26 16:25:56 EST 2012 - 1  
 Direction OUTBOUND Interchange Control Number  
 Message Type REQ Group Control Number  
 ID C0A8DE8113BD6A4F929000004693B56B Transaction Set Control Number 2d353638313531353232393136313338

Communication and Protocol

Document Protocol HL7 Sent Date 26/12/2012 8:57:15 PM GMT+11:00

- Note the details
- Review the State, Endpoint and other attributes, and review "Identification" and "Communication and Protocol" attribute sections – you should recall most of these from the endpoint configuration steps – note the "Native Message Size" and "Translated Message Size" information

ORACLE SOA Suite for healthcare integration Accessibility Help Logout Logged in as weblo

Designer Dashboards Reports

Last 24 Hours

Last 24 Hours Found 6 results. Page Size 40 Last Updated: 26/12/2012 8:58:33 PM GMT+11:00

Message ID	Created Date and Time	Direction	Endpoint	State	Doc
1 > 2d383235373931353335393637323538	26/12/2012 8:57:13 PM GMT+11:00	OUTBOUND	HA08_SystemA_Out	MSG_COMPLETE	HL7
2 > 2d373639383030323130343931363131	26/12/2012 8:57:10 PM GMT+11:00	OUTBOUND	BcastB_SystemA_Out	MSG_COMPLETE	HL7
3 > COA8DE8113BD6A4DDFF000004693B554-1	26/12/2012 8:57:05 PM GMT+11:00	INBOUND	HosA_SystemA_In	MSG_COMPLETE	HL7

HA08\_SystemA\_Out ← Wire Message Business Message Application Message Composite

Business Message Details

Identification

Endpoint: HA08\_SystemA\_Out Message ID: 2d383235373931353335393637323538  
 State: MSG\_COMPLETE Label: soa\_b2b\_ - Wed Dec 26 16:25:56 EST 2012 - 1  
 Direction: OUTBOUND Interchange Control Number  
 Message Type: REQ Group Control Number  
 ID: COA8DE8113BD6A4F929000004693B56B Transaction Set Control Number: 2d353638313531353232393136313338

Communication and Protocol

Document Protocol: HL7 Sent Date: 26/12/2012 8:57:15 PM GMT+11:00  
 Protocol Version: 2.3.1 Received Date: 26/12/2012 8:57:13 PM GMT+11:00  
 Document Type: ADT Acknowledgement Mode: NONE  
 Document Definition: A08\_v1.0 Response Mode: NONE  
 Document Retry Interval: 0 Transport Protocol Name: MLLP  
 Document Remaining Retry: 0 Transport Protocol Version: 1.0

Content

Native Message Size: 781 Translated Message Size: 52278

Payload Key Fields

- Click the "Composite" link, log into the Enterprise Manager Console with your credential (my credentials are weblogic/welcome1) and review the message processing "Trace", noting component hierarchy, component names, types, state and so on

Trace

Click a component instance to see its detailed audit trail.  
 Show Instance IDs

Instance	Type	Usage	State
HCI_In	Healthcare Binding	Service	Completed
RouterMediator	Mediator Component		Completed
HCI_Out	Healthcare Binding	Reference	Completed
TransformerMediator	Mediator Component		Completed
HCI_HA08_SystemA_Out	Healthcare Binding	Reference	Completed

- Click the "RouterMediator" link and inspect the instance details, expanding nodes as you go along to see what can be seen – this display shows the SOA Composite and the message and message properties as they are at different stages of processing
- Click the "TransformerMediator" link and inspect the instance details, expanding nodes as you go along to see what can be seen – this display shows the SOA Composite and the message and message properties as they are at different stages of processing

Flow Trace > Instance of TransformerMediator

### Instance Details of TransformerMediator

This page shows Mediator component instance details.

**Audit Trail** | **Faults**

Expand the payload nodes to view the details of the instance audit trail.

- onMessage
  - 26/12/2012 8:57:08 PM Input payload received
    - <payload>
  - 26/12/2012 8:57:08 PM onCase "HCI\_HA08\_SystemA\_Out.send"
    - 26/12/2012 8:57:08 PM Evaluation of xpath condition "No Filter" resulted true
      - <payload>
    - 26/12/2012 8:57:09 PM Transformed message part "body" using "xsl/ADT\_To\_ADT\_A08.xsl"
      - <payload>
    - 26/12/2012 8:57:09 PM Assigned "oraext:generate-guid()" to "\$out.property.hc.messageId"
      - <payload>
    - 26/12/2012 8:57:09 PM Assigned "HA08\_SystemA\_Out" to "\$out.property.hc.toEndpoint"
      - <payload>
    - 26/12/2012 8:57:09 PM Invoked 1-way operation "send" on target service "HCI\_HA08\_SystemA\_Out"
      - <payload>

## Explore Dashboards

The SOA Suite for healthcare integration Console includes a capability which allows an operator to define and use dashboards for quick access to operational state of the system, message counts and gauges, and other kinds of information artifacts which are useful for monitoring the system.

- In Healthcare Integration Console click on the Dashboards tab
- Pull down the "+" menu in the top right hand corner and chose "New Dashboard..."

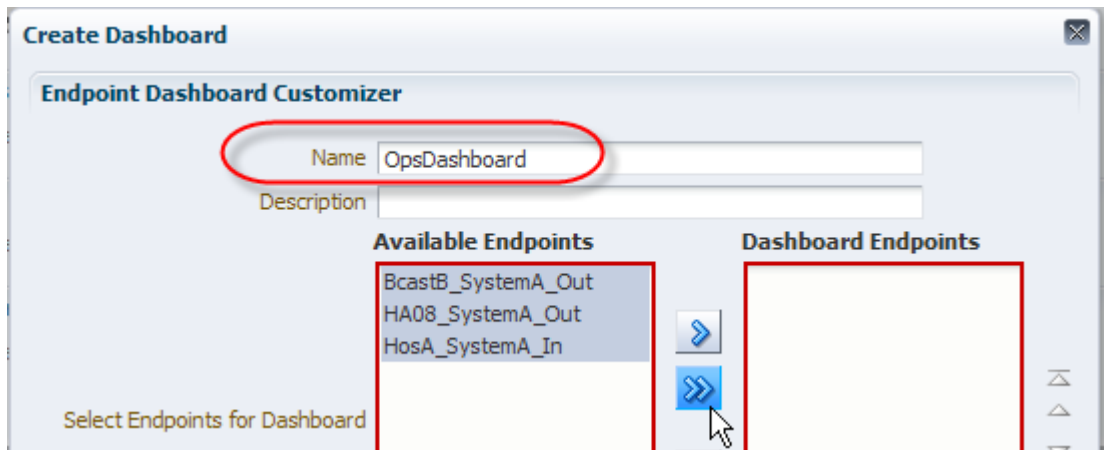
ORACLE SOA Suite for healthcare integration Accessibility Help Logout Logged in as weblogic since 26/12/2012 8:17:10 PM GMT+11

Designer | **Dashboards** | Reports

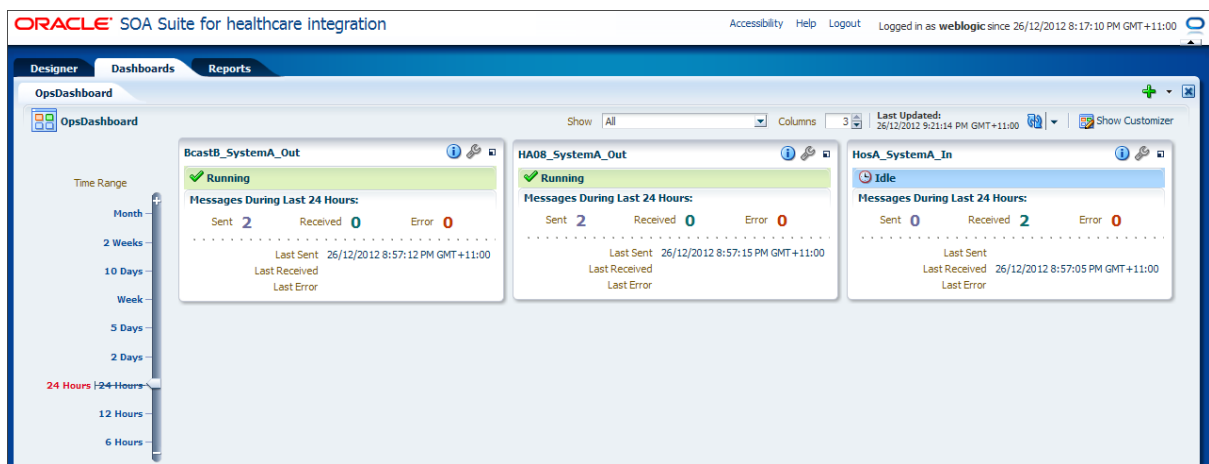
Welcome to Dashboards. Here are some possible next steps:

- Set Up Default Dashboard to Show Whenever Tab Is Selected
  - Selected Dashboard: None
  - Apply
- Open or Edit Existing Dashboards
  - Click on green plus sign near upper right corner and select desired Dashboard to open.
- Create Dashboard
  - Click on green plus sign near upper right corner and select **New Dashboard...**, or just click [here](#)
- Delete Existing Dashboards
  - Click on green plus sign near upper right corner and select **Delete Dashboards...**, or just click [here](#)

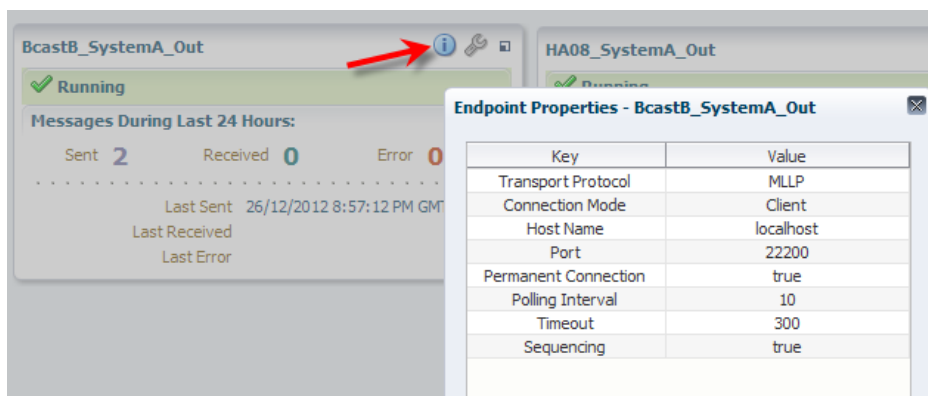
- Name the new dashboard "OpsDashboards", select all endpoints, click the "move all" button and click the "Create" button



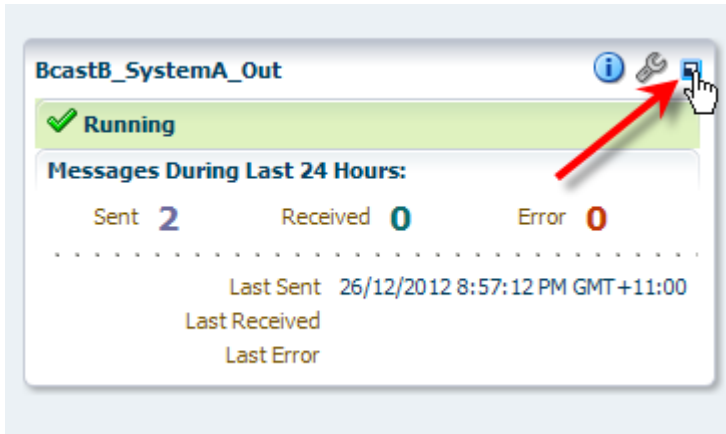
- Inspect the display, noting message counts, last message received/sent timestamps, time period slider and other features



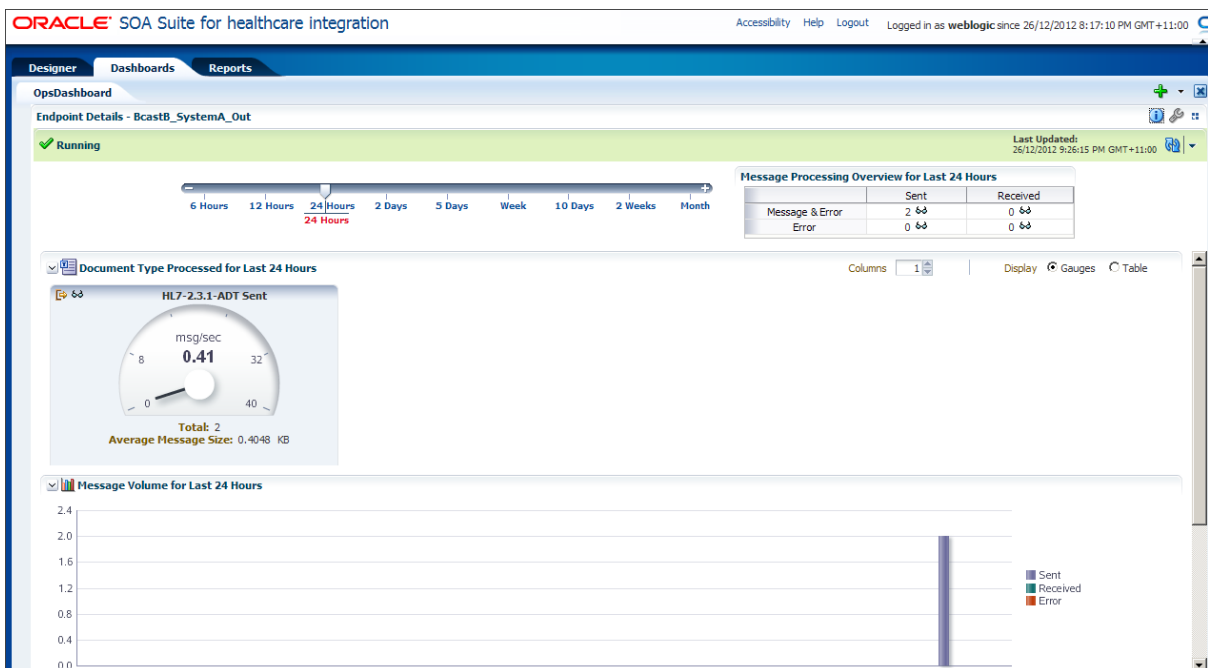
- Click the "i" in circle button to see the endpoint configuraiton details then close the dialog



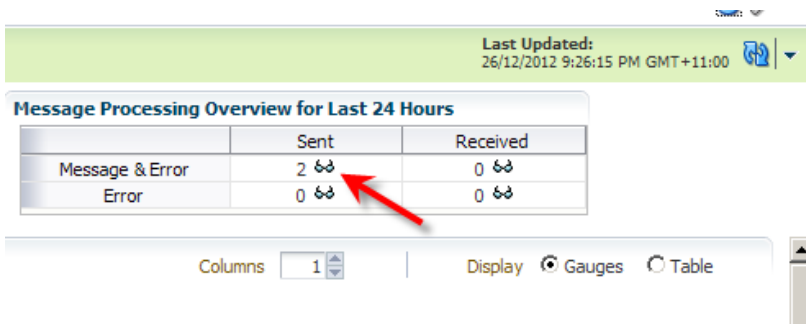
- Click the "Maximize" button on the "BcastB\_SystemA\_Out" control



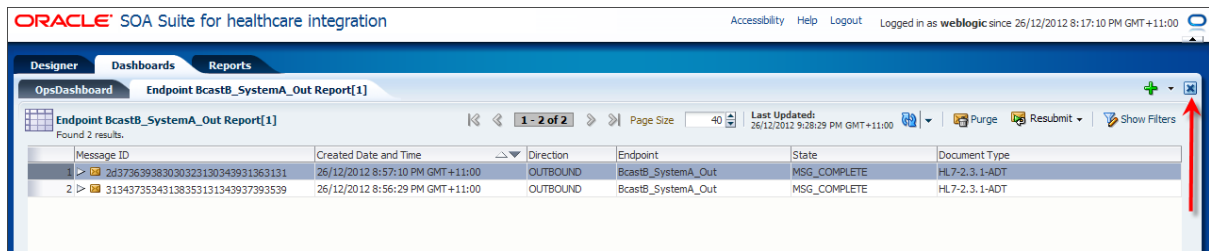
- Inspect the display and explore various controls



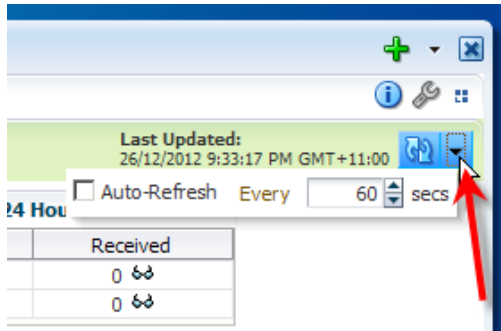
- Click the tiny eye glasses in the "Message Processing Overview" table in the "Sent" column of the "Message & Error" row



- Note the two outbound messages that this endpoint sent in the period covered by the display, explore the messages, then click the "Close" button to close the panel and return to the "OpsDashboard"



- Click the downward pointing triangle at the far right next to the "Refresh" icon, then configure auto refresh for a 30 second interval



- Submit a file of message containing at least 10 messages and watch the gauge on the display
- Explore other Dashboard features and facilities

## Summary

In this article the router developed in the previous article in the series, the "SOA Suite for healthcare integration Series – HL7 v2 Inbound CMM to Outbound Pass-through Solution", was extended to add a fan-out for distribution of one inbound message to two outbound receivers and a ADT A01/A03 to ADT A08 transformation using the eXtensible Style Language (XSL). Additionally, SOA Suite for healthcare integration Dashboard functionality was explored.