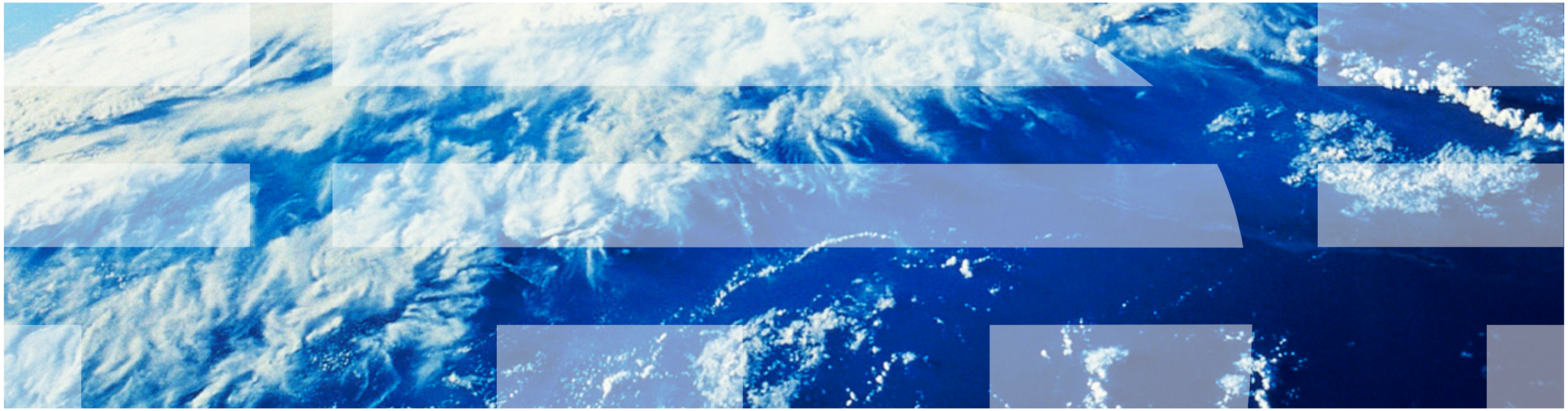
IBM WebSphere User Group

- Click to add text

An introduction to service mapping

Integrating evolving web services in WebSphere Application Server V8.5.5

Andrew Borley



Agenda

- Introducing Service Mapping
- Service Mapping Concepts
- Developing a Service Map
- Administration: Mapping Services
- Administration: Deployment and administration of Service Maps
- Demonstration

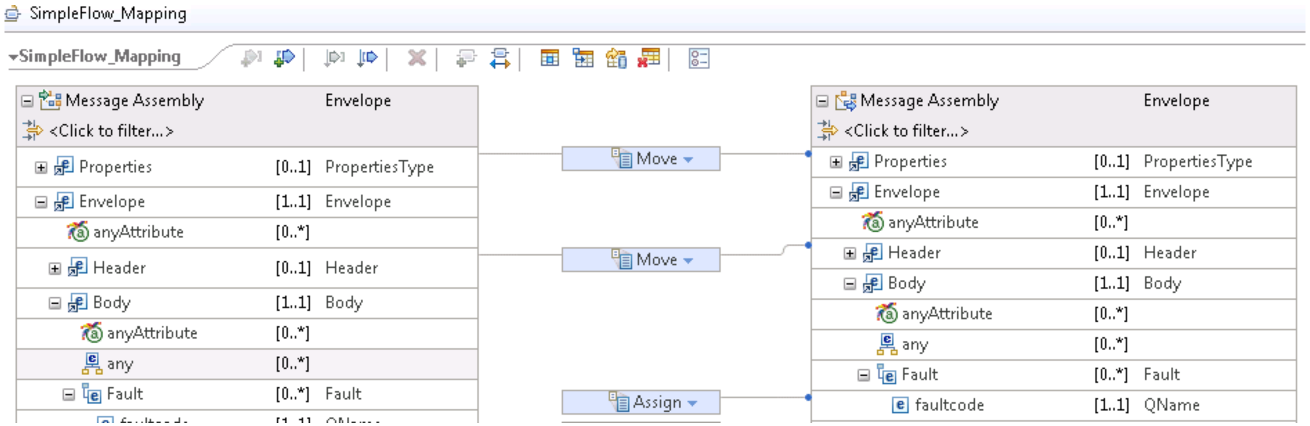
Why Service Mapping?

- Main purpose of service mapping
  - Service Mapping provides a way of insulating an application that consumes a service from the details of that service provider's interface or location
  - This is achieved by providing a simple way of performing content-based routing and message transformation

- Why is this useful?
  - This helps in many scenarios – for example,
    - **Service versions:** When a service is modified to cope with additional requirements, its interface and location are likely to go through modifications that, although small, are likely to break existing service consumers
    - **'Meet in the middle':** When a business unit defines the service interface that it expects to use perform some task, this may well not match the details of an existing 'enterprise service' owned and offered by the IT organisation
    - ...and many more

So – what is a service map?

When restructuring a message from one structure to another, is it common for graphical tooling to exploit the idea of a graphical message map which allows the developer to decide how a field in the input **message maps** to a field in the output message

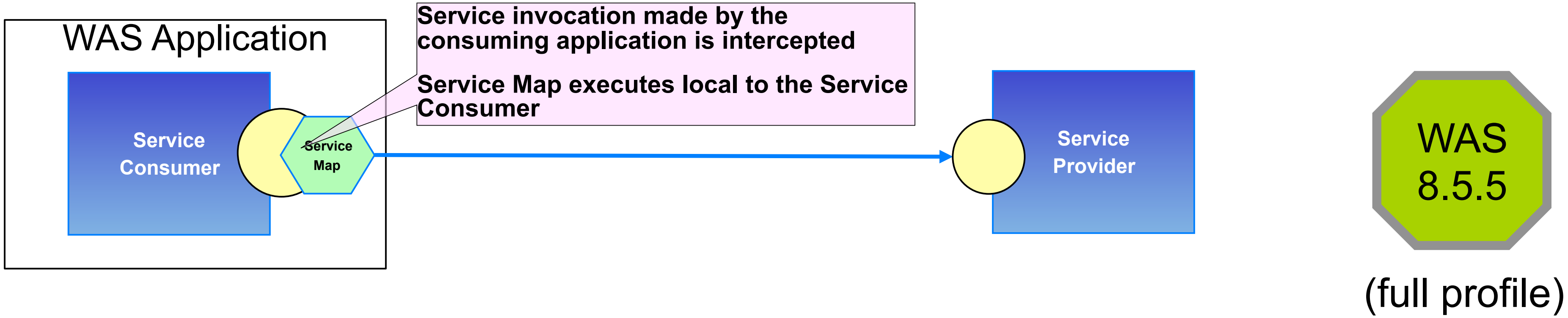


Message Assembly		Envelope
<Click to filter...>		
Properties	[0..1]	PropertiesType
Envelope	[1..1]	Envelope
anyAttribute	[0..*]	
Header	[0..1]	Header
Body	[1..1]	Body
anyAttribute	[0..*]	
any	[0..*]	
Fault	[0..*]	Fault
faultcode	[1..1]	QName

Message Assembly		Envelope
<Click to filter...>		
Properties	[0..1]	PropertiesType
Envelope	[1..1]	Envelope
anyAttribute	[0..*]	
Header	[0..1]	Header
Body	[1..1]	Body
anyAttribute	[0..*]	
any	[0..*]	
Fault	[0..*]	Fault
faultcode	[1..1]	QName

- When a service consumer interacts with a service provider that may change its location, or may have a different interface, there are three additional considerations
  - Which **service location** should we route a message to?
  - Which **operation** on the service provider should be invoked when dealing with a request from a service consumer?
  - For any particular operation, how must each **input** and **output/fault** message be transformed?
- **Service Mapping** extends the idea of mapping to cover each of these additional considerations

WAS and Service Maps topology



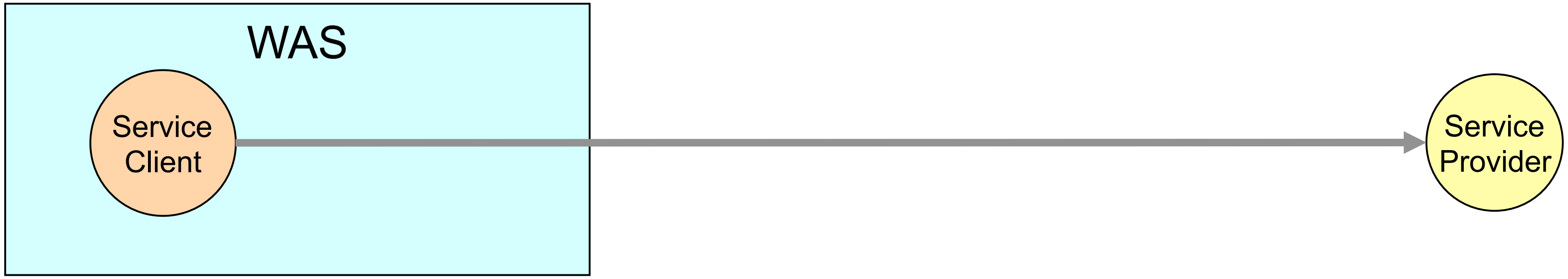
Service invocation made by the consuming application is intercepted

Service Map executes local to the Service Consumer

Service Mapping Concepts

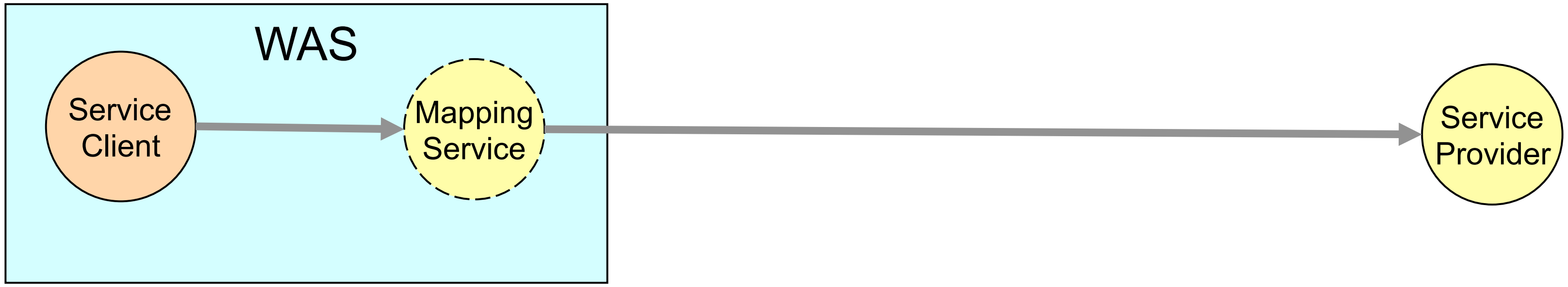
The Service Mapping Conceptual Model: Before we start…

- Starting point
  - Service Clients are configured to interact with a particular Web Service



Service Mapping Conceptual Model: Mapping Services

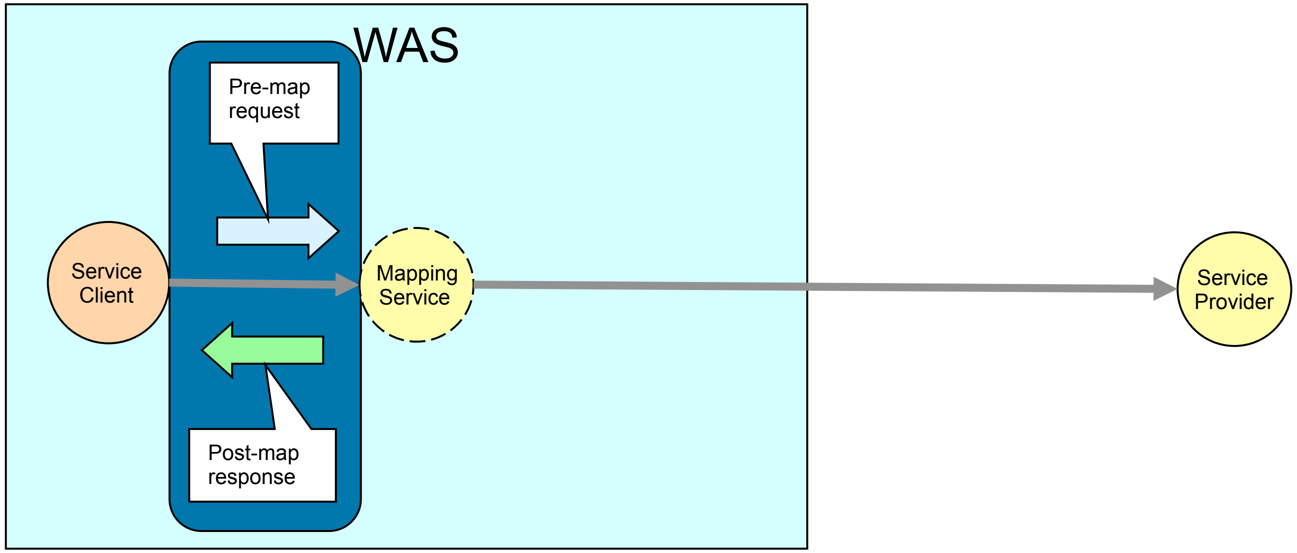
- Mapping Service
  - A local mapping service can be defined via admin commands or the admin console to intercept all invocations bound for a particular service
  - [Interception happens on the thread of execution of the Service client invocation]



- In this state, the Mapping Service adds the following behaviour
  - The mapping service can be started or stopped
  - The mapping service can be configured by the administrator to publish service interaction events to a JMS topic

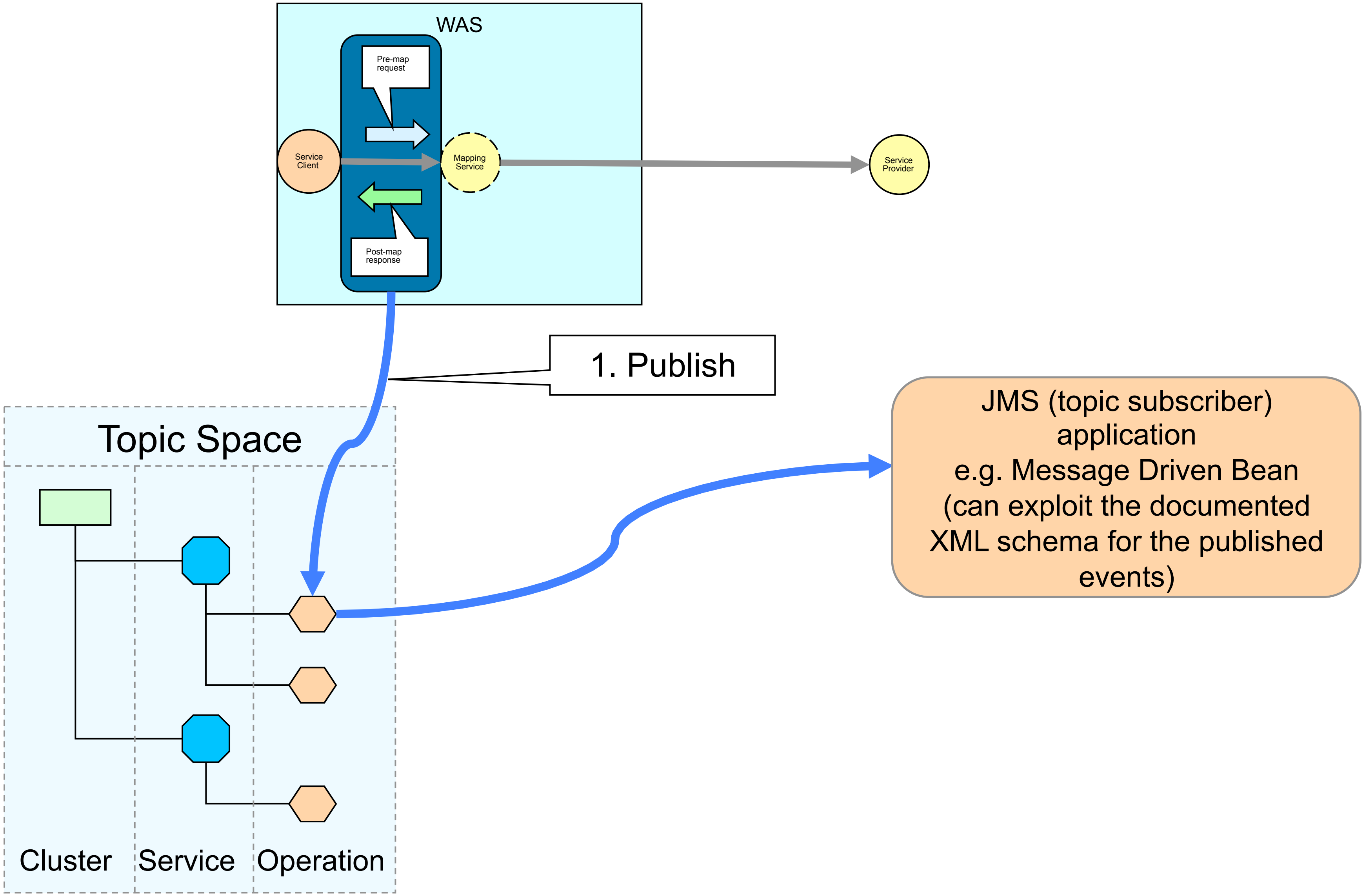
Mapping Service Event Points

- Event emission can be configured by the WAS administrator



- Events are published using JMS Pub/Sub to a structured topic space
  - <cell>/<cluster || node.server>/<mapping service>/<operation>
  - This can optionally be overridden via a specified JNDI reference
- Event messages contain;
  - Information about where they are emitted from
  - Request/Response/Fault differentiation
  - (optionally) message content

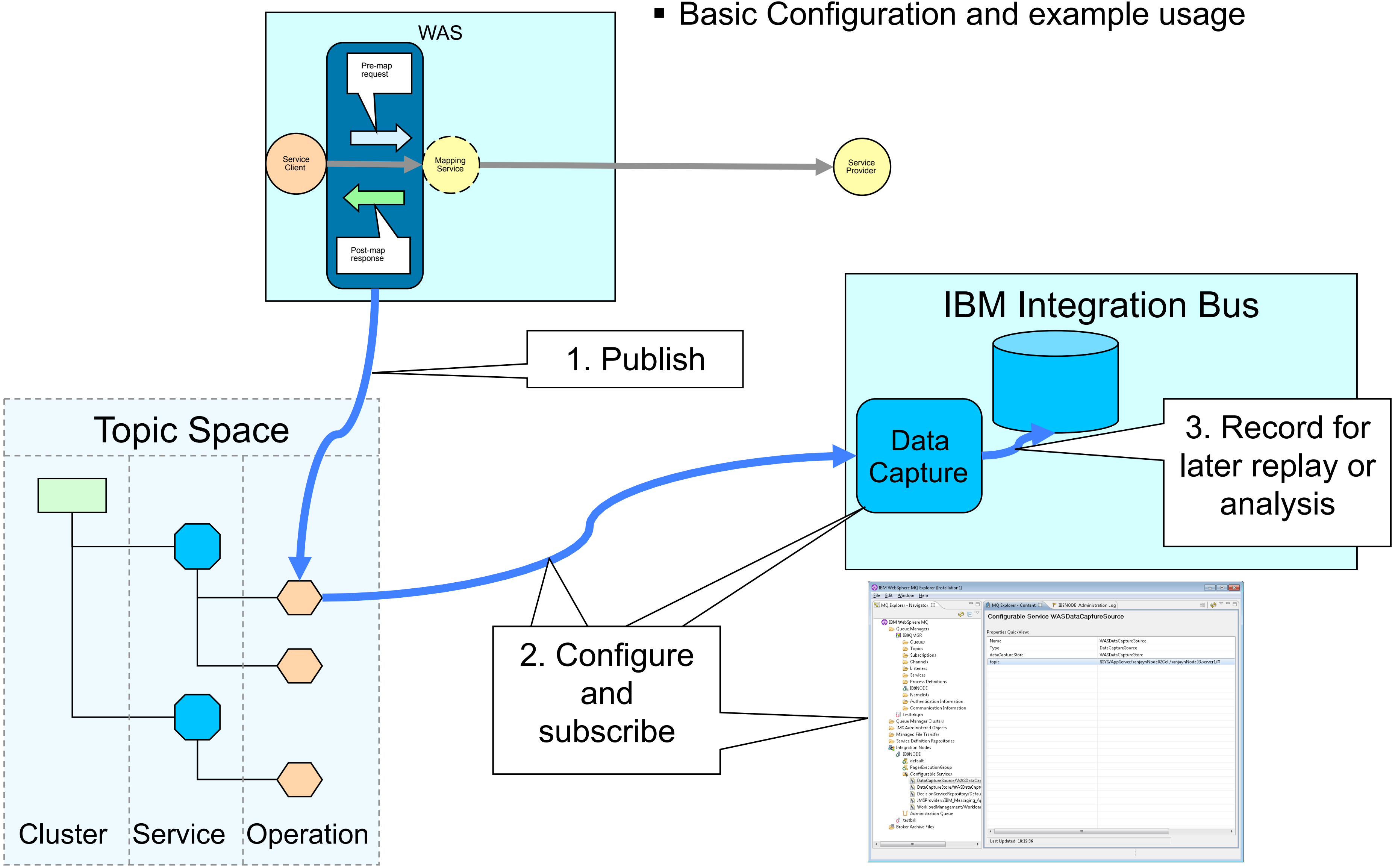
Mapping Service Event Emission: General use



JMS (topic subscriber) application
e.g. Message Driven Bean
(can exploit the documented XML schema for the published events)

Mapping Service Event Emission and IBM Integration Bus

- Basic Configuration and example usage



Configurable Service WASDataCaptureSource

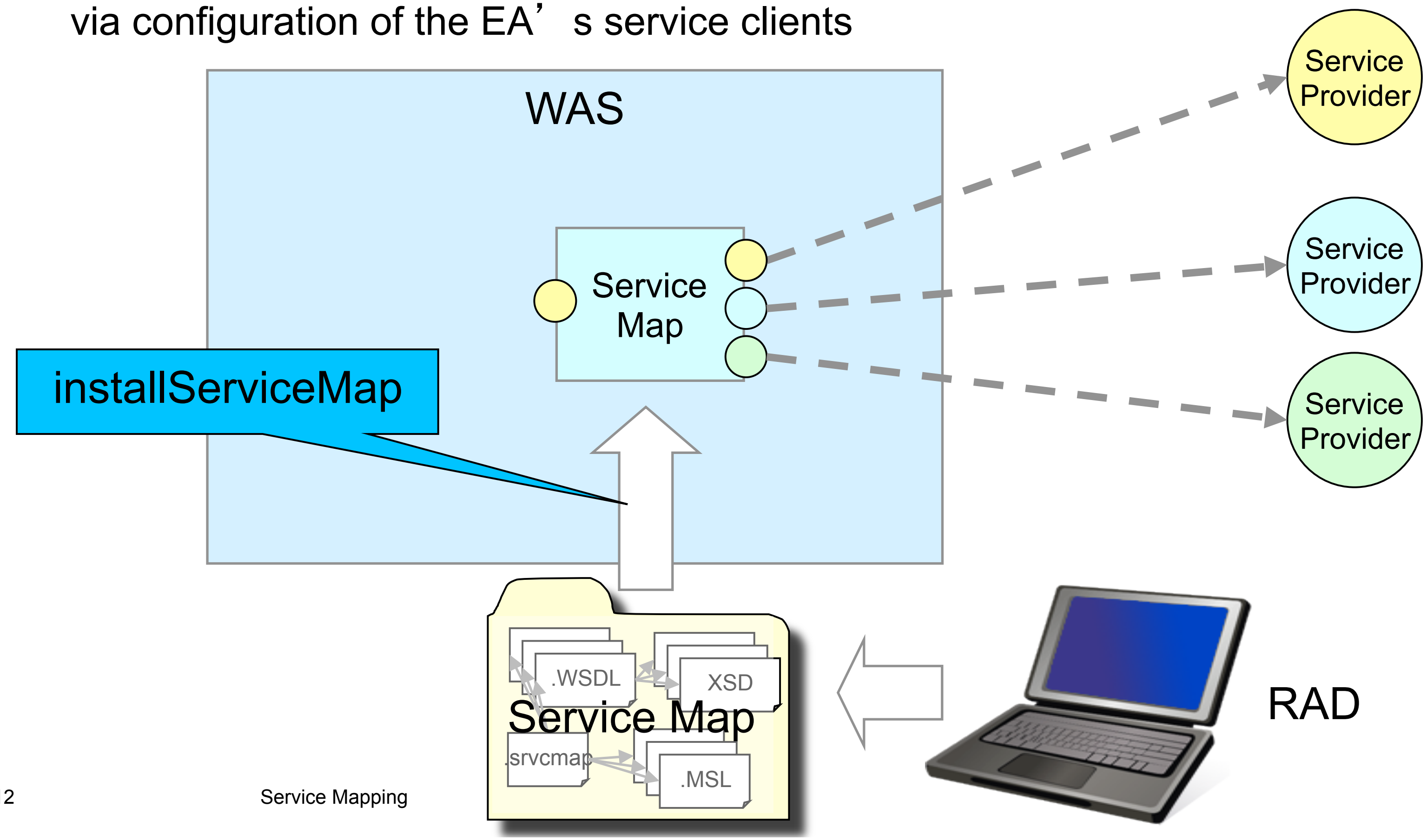
Properties QuickView:

Name	WASDataCaptureSource
Type	DataCaptureSource
dataCaptureStore	WASDataCaptureStore
topic	$SYS/AppServer/sanjaynNode02Cell/sanjaynNode03.server1/#

Last Updated: 18:19:36

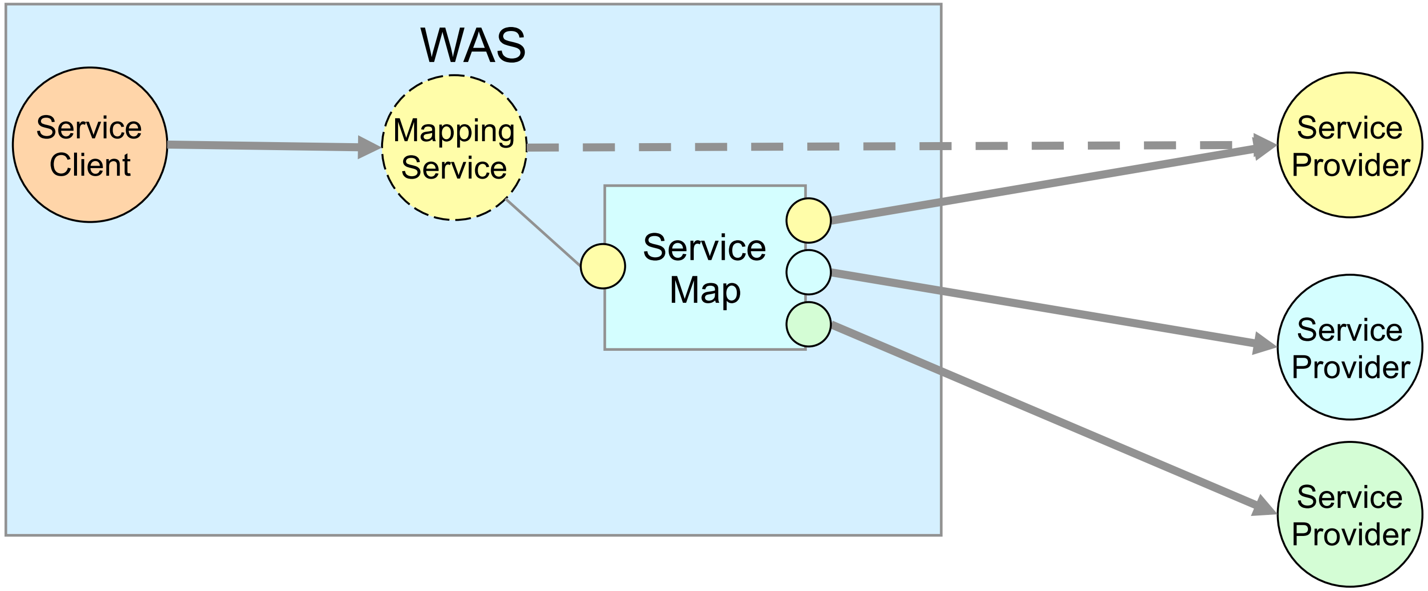
Service Mapping Conceptual Model: Service Maps

- Service Maps are developed in Rational Application Developer and define the re-routing and transformation
- When a Service Map is installed to the WAS runtime, a Service Map Enterprise Application is created in the WAS runtime
- The endpoints that it can route service requests to can be overridden at install time, or via configuration of the EA’ s service clients



The Mapping Service Conceptual Model: 3

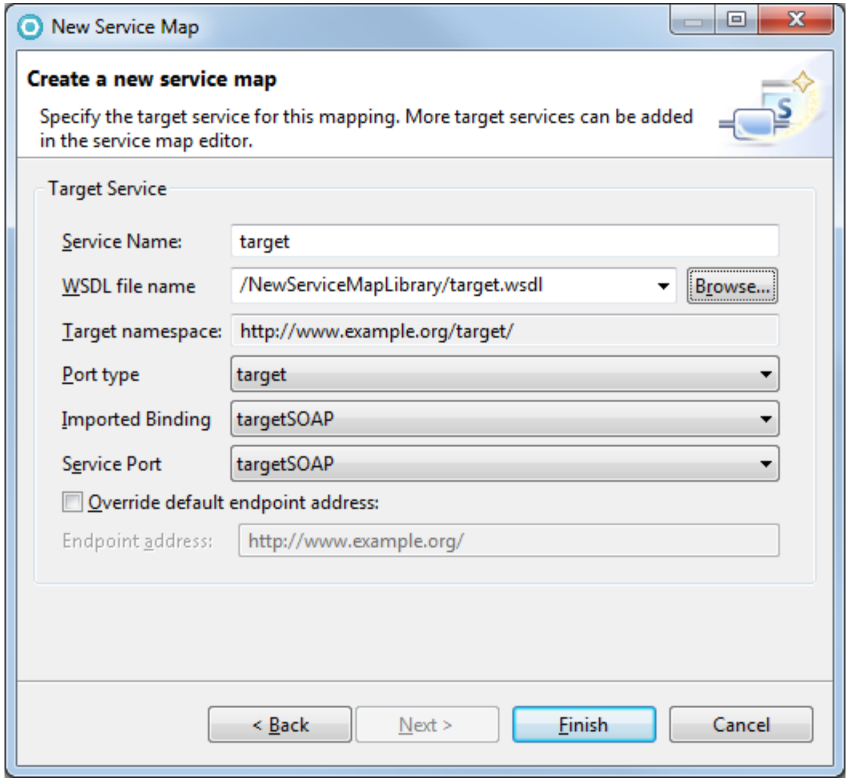
- A Service Map can be attached to one or more mapping services, in order to transform and route the intercepted messages
- Note that – if a Service Map is not attached to any mapping services, then it does not receive any messages to process



Service Map Development in RAD

Create a new Service Map: 1

- Answers the question "What is the interface of the service invocation that we want to intercept and map?"



New Service Map

Create a new service map

Specify the target service for this mapping. More target services can be added in the service map editor.

Target Service

Service Name: target

WSDL file name /NewServiceMapLibrary/target.wsdl

Target namespace: http://www.example.org/target/

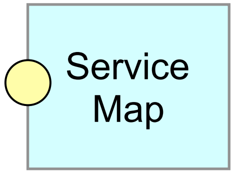
Port type target

Imported Binding targetSOAP

Service Port targetSOAP

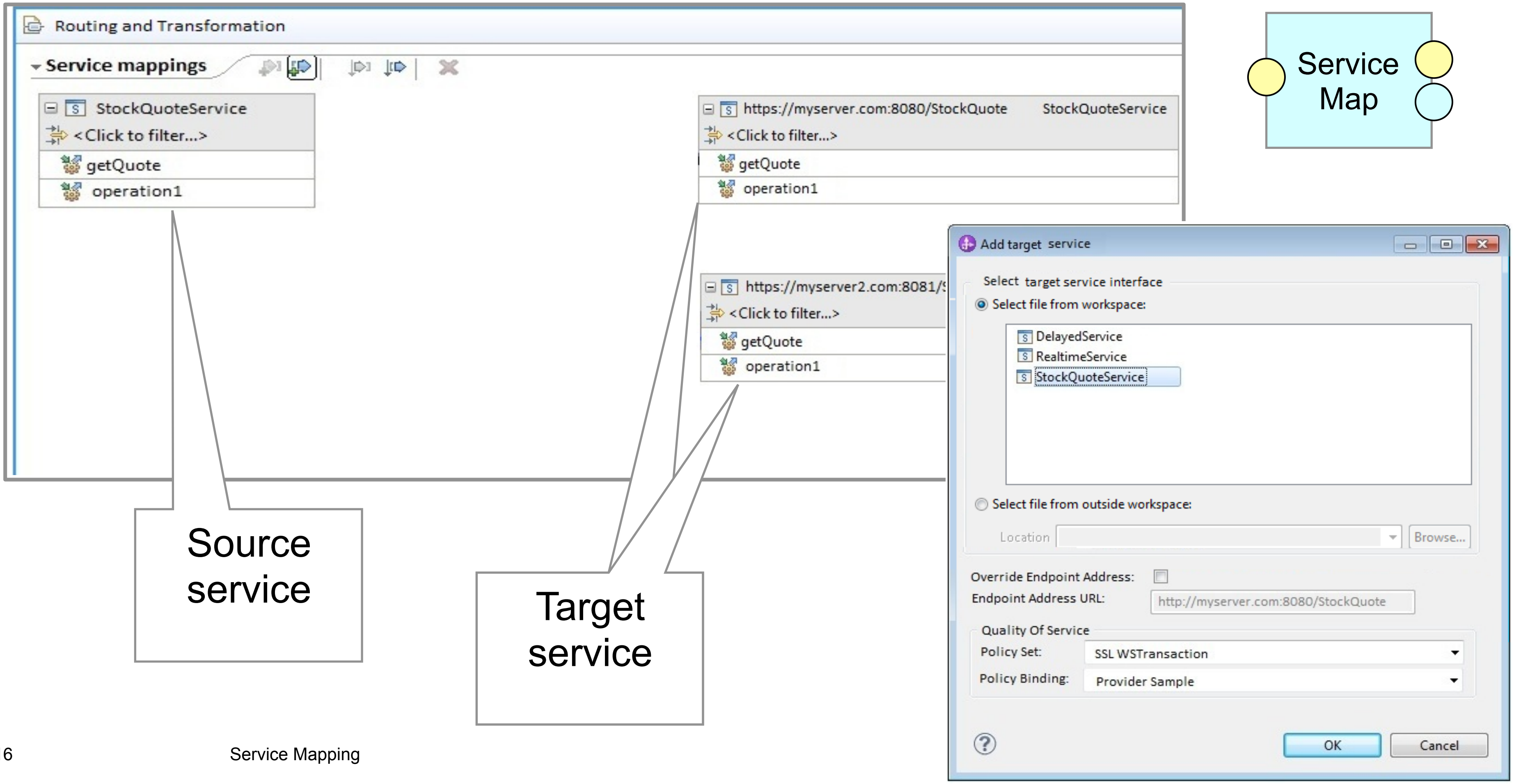
□ Override default endpoint address:

Endpoint address: http://www.example.org/



Service Map Development 3 (Select service)

- Answers the question “what are the interfaces/addresses of the services that we may need to route to?”
    - Addresses can be overridden at install/admin



Add target service

Select target service interface

● Select file from workspace:

- DelayedService
- RealtimeService
- StockQuoteService

○ Select file from outside workspace:

Location: Browse...

Override Endpoint Address: □

Endpoint Address URL: http://myserver.com:8080/StockQuote

Quality Of Service

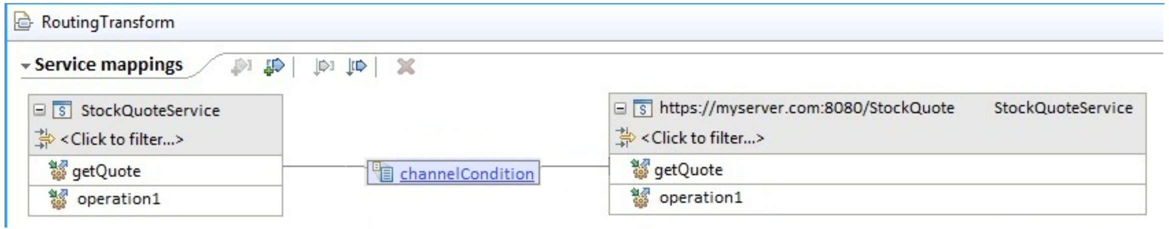
Policy Set: SSL WSTransaction

Policy Binding: Provider Sample

OK Cancel

Service Map Development – 4: Selecting a target operation to invoke

- Source and target services are wired together and XPath conditional expressions provided.
  - The XPath expressions are used at run-time to select one of the target services



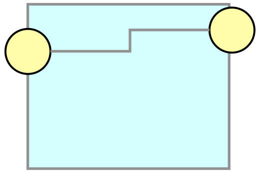
RoutingTransform

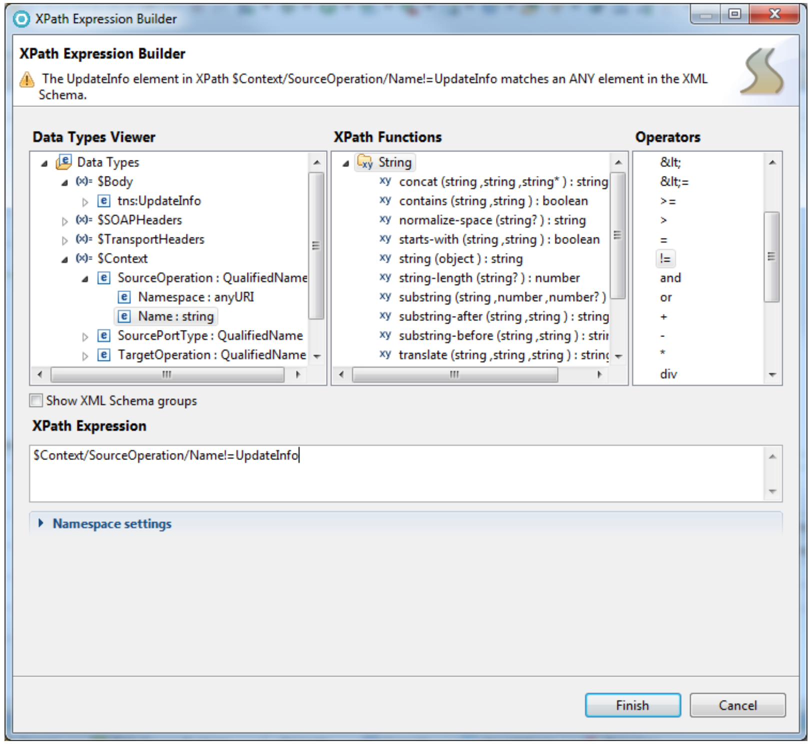
Service mappings

StockQuoteService
<Click to filter...>
getQuote
operation1

channelCondition

https://myserver.com:8080/StockQuote StockQuoteService
<Click to filter...>
getQuote
operation1





XPath Expression Builder

XPath Expression Builder

⚠ The UpdateInfo element in XPath $Context/SourceOperation/Name!=UpdateInfo matches an ANY element in the XML Schema.

Data Types Viewer
Data Types
$Body
tns:UpdateInfo
$SOAPHeaders
$TransportHeaders
$Context
SourceOperation : QualifiedName
Namespace : anyURI
Name : string
SourcePortType : QualifiedName
TargetOperation : QualifiedName
XPath Functions
String
concat (string ,string ,string*) : string
contains (string ,string) : boolean
normalize-space (string?) : string
starts-with (string ,string) : boolean
string (object) : string
string-length (string?) : number
substring (string ,number ,number?)
substring-after (string ,string) : string
substring-before (string ,string) : strir
translate (string ,string ,string) : string
Operators
&lt;
&lt;=
>=
>
=
!=
and
or
+
-
*
div
□ Show XML Schema groups

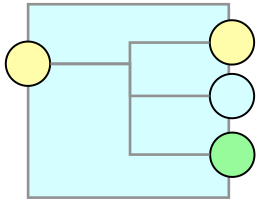
XPath Expression

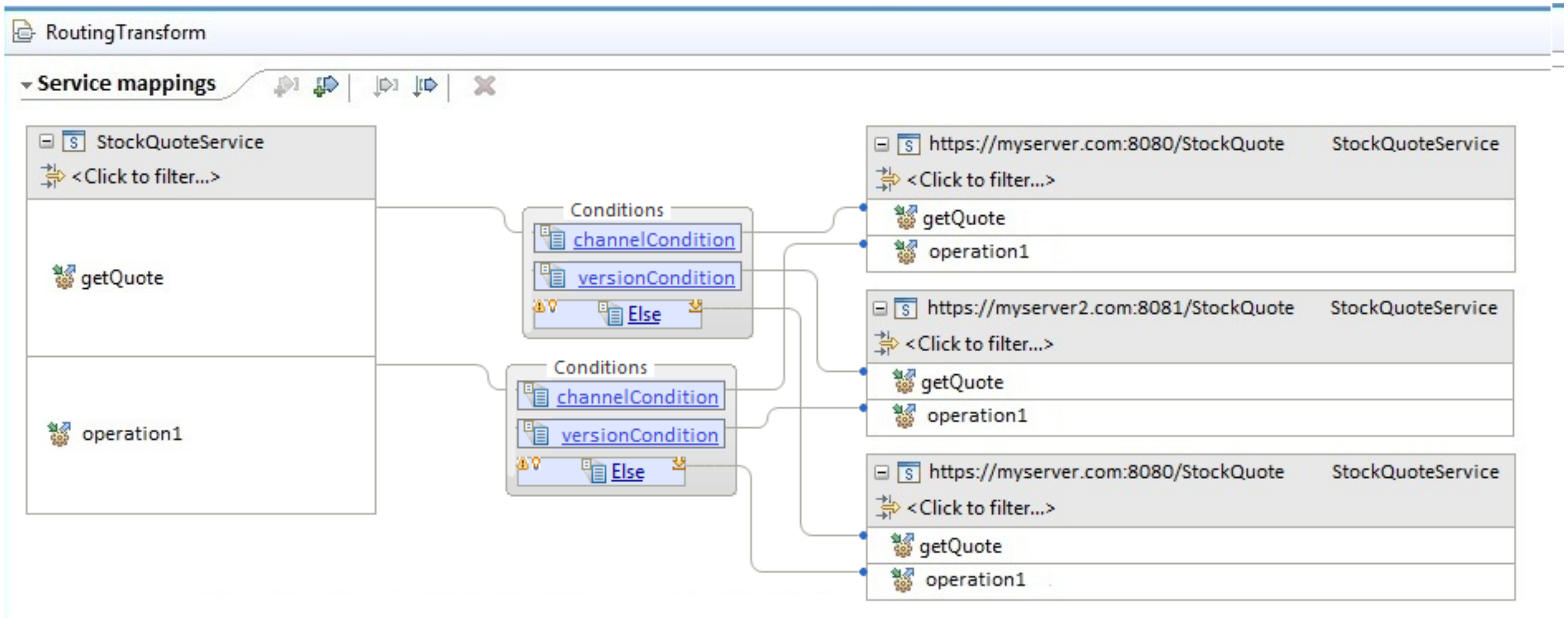
```
$Context/SourceOperation/Name!=UpdateInfo
```

Namespace settings
Finish
Cancel

Service Map Development – 5: …more conditions on what operation to invoke

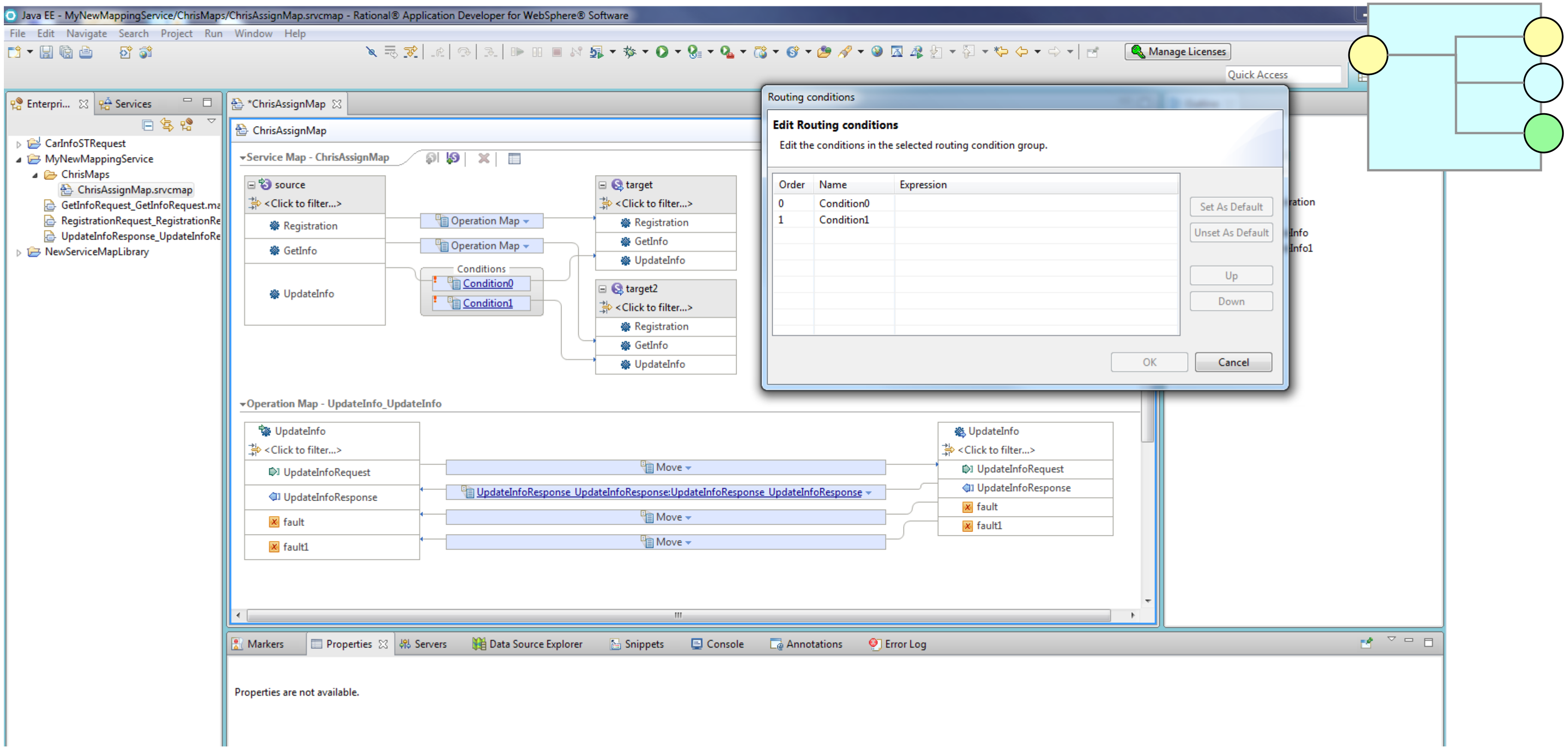
- More wires and conditions are added





Filling in the details for a particular (source, target) pair of operations

- Selecting a single wire allows developers to edit the Operation-level mapping



Java EE - MyNewMappingService/ChrisMaps/ChrisAssignMap.srvcmap - Rational® Application Developer for WebSphere® Software

Routing conditions

Edit Routing conditions

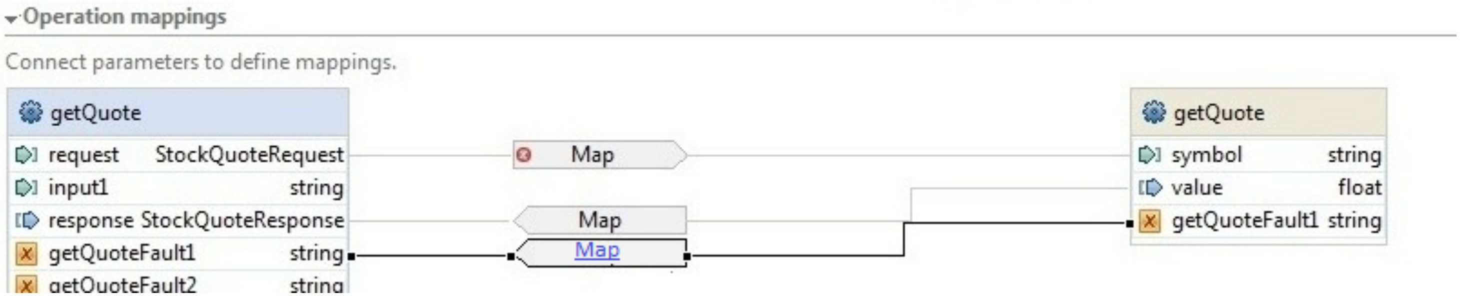
Edit the conditions in the selected routing condition group.

Order	Name	Expression
0	Condition0	
1	Condition1	

Properties are not available.

Service Map Development – 7 (message-level mapping)

- Each wire in the Operation-level mapping can have a Message-level data mapping (if required)

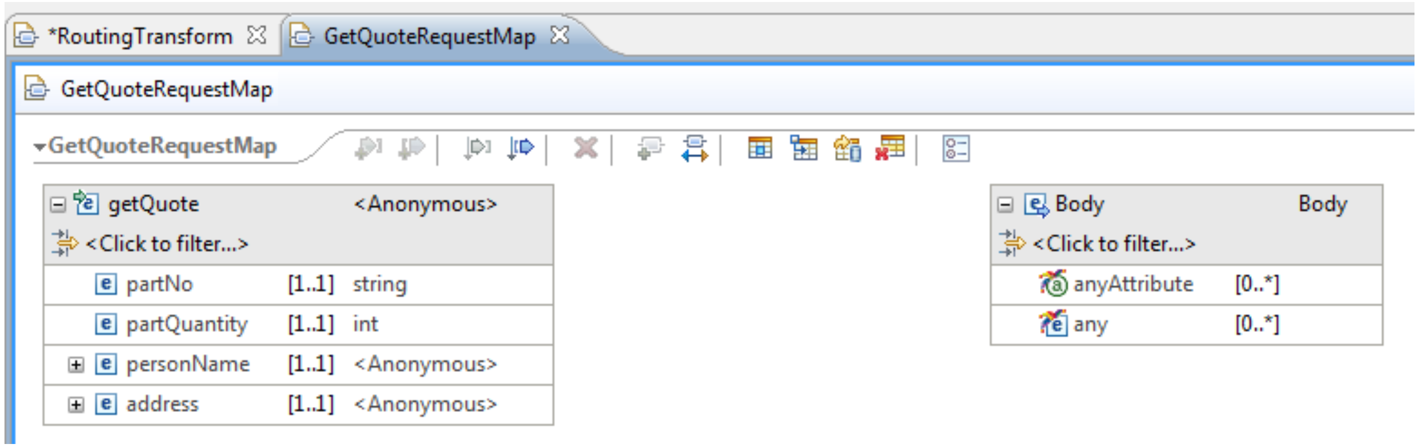


Operation mappings

Connect parameters to define mappings.

getQuote	
request	StockQuoteRequest
input1	string
response	StockQuoteResponse
getQuoteFault1	string
getQuoteFault2	string

getQuote	
symbol	string
value	float
getQuoteFault1	string



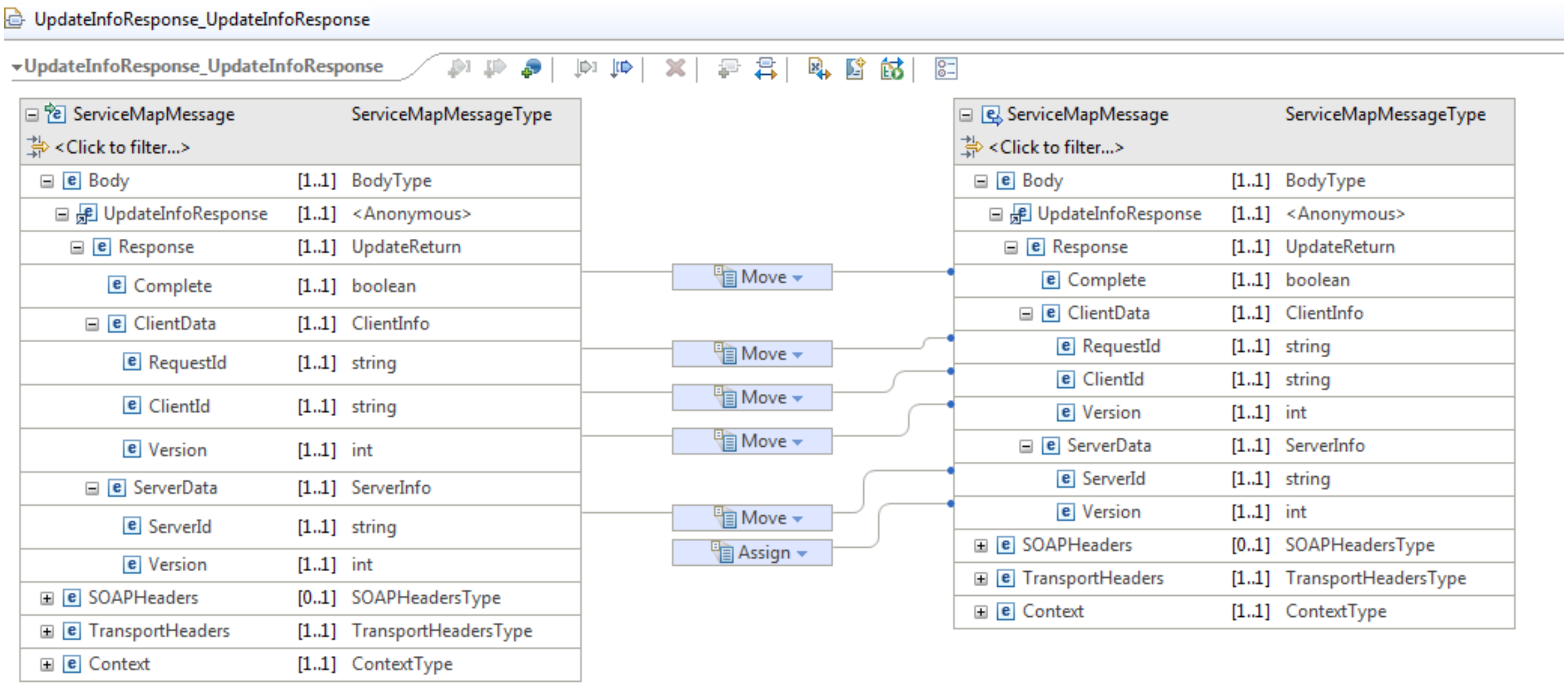
GetQuoteRequestMap

GetQuoteRequestMap

getQuote		<Anonymous>
<Click to filter...>		
partNo	[1..1]	string
partQuantity	[1..1]	int
personName	[1..1]	<Anonymous>
address	[1..1]	<Anonymous>

Body	Body
<Click to filter...>	
anyAttribute	[0..*]
any	[0..*]

Data Map: Maps from one particular message to another



UpdateInfoResponse_UpdateInfoResponse

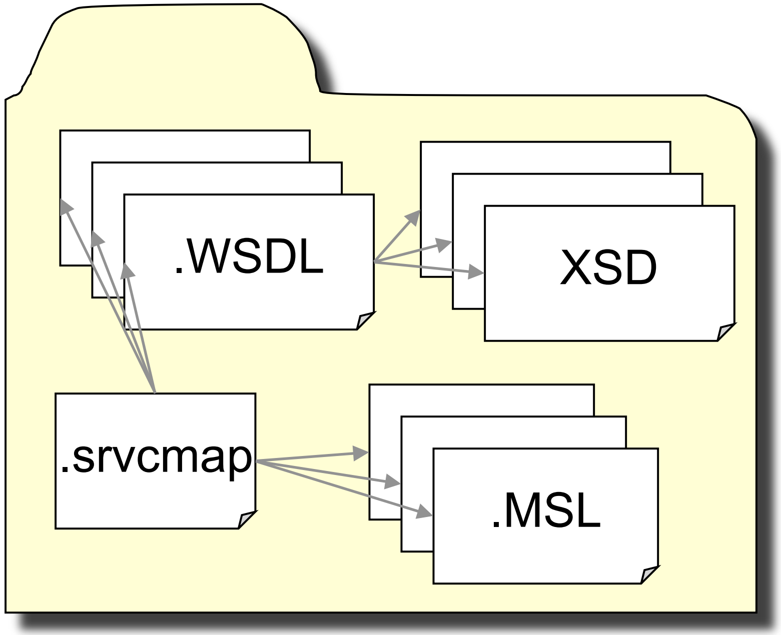
UpdateInfoResponse_UpdateInfoResponse

ServiceMapMessage		ServiceMapMessageType
<Click to filter...>		
Body	[1..1]	BodyType
UpdateInfoResponse	[1..1]	<Anonymous>
Response	[1..1]	UpdateReturn
Complete	[1..1]	boolean
ClientData	[1..1]	ClientInfo
RequestId	[1..1]	string
ClientId	[1..1]	string
Version	[1..1]	int
ServerData	[1..1]	ServerInfo
ServerId	[1..1]	string
Version	[1..1]	int
SOAPHeaders	[0..1]	SOAPHeadersType
TransportHeaders	[1..1]	TransportHeadersType
Context	[1..1]	ContextType

ServiceMapMessage		ServiceMapMessageType
<Click to filter...>		
Body	[1..1]	BodyType
UpdateInfoResponse	[1..1]	<Anonymous>
Response	[1..1]	UpdateReturn
Complete	[1..1]	boolean
ClientData	[1..1]	ClientInfo
RequestId	[1..1]	string
ClientId	[1..1]	string
Version	[1..1]	int
ServerData	[1..1]	ServerInfo
ServerId	[1..1]	string
Version	[1..1]	int
SOAPHeaders	[0..1]	SOAPHeadersType
TransportHeaders	[1..1]	TransportHeadersType
Context	[1..1]	ContextType

Service map file format

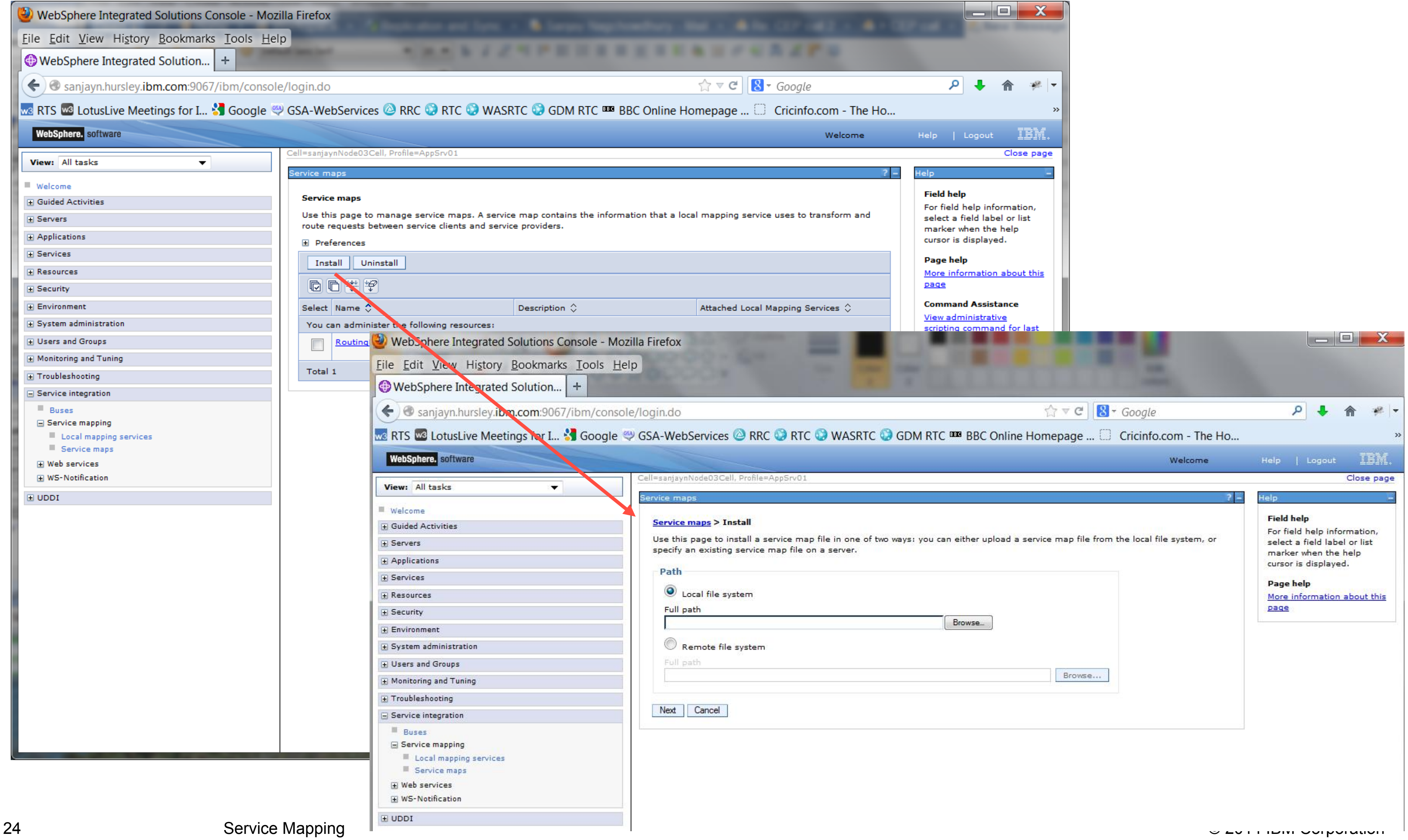
- The Service Map library exported from RAD contains
  - WSDL and XSD documents required to define the interfaces involved
  - The service map itself, which is an XML Document
  - For each **message** map, an **MSL** (Mapping Specification Language) document
    - MSL: Is a dedicated XML syntax for expressing Data Maps – which express a message restructuring from one data structure to another – already supported by RAD Graphical Data Mapper



WAS Administration

WAS Deployment & Administration - 1

- Developer exports Service Map from RAD
- Administrator installs Service Map into WAS



Cell=sanjaynNode03Cell, Profile=AppSrv01

Service maps

Use this page to manage service maps. A service map contains the information that a local mapping service uses to transform and route requests between service clients and service providers.

Preferences

You can administer the following resources:

Help

Field help

For field help information, select a field label or list marker when the help cursor is displayed.

Page help

More information about this page

Command Assistance

View administrative scripting command for last

Service maps > Install

Use this page to install a service map file in one of two ways: you can either upload a service map file from the local file system, or specify an existing service map file on a server.

Path

● Local file system

Full path

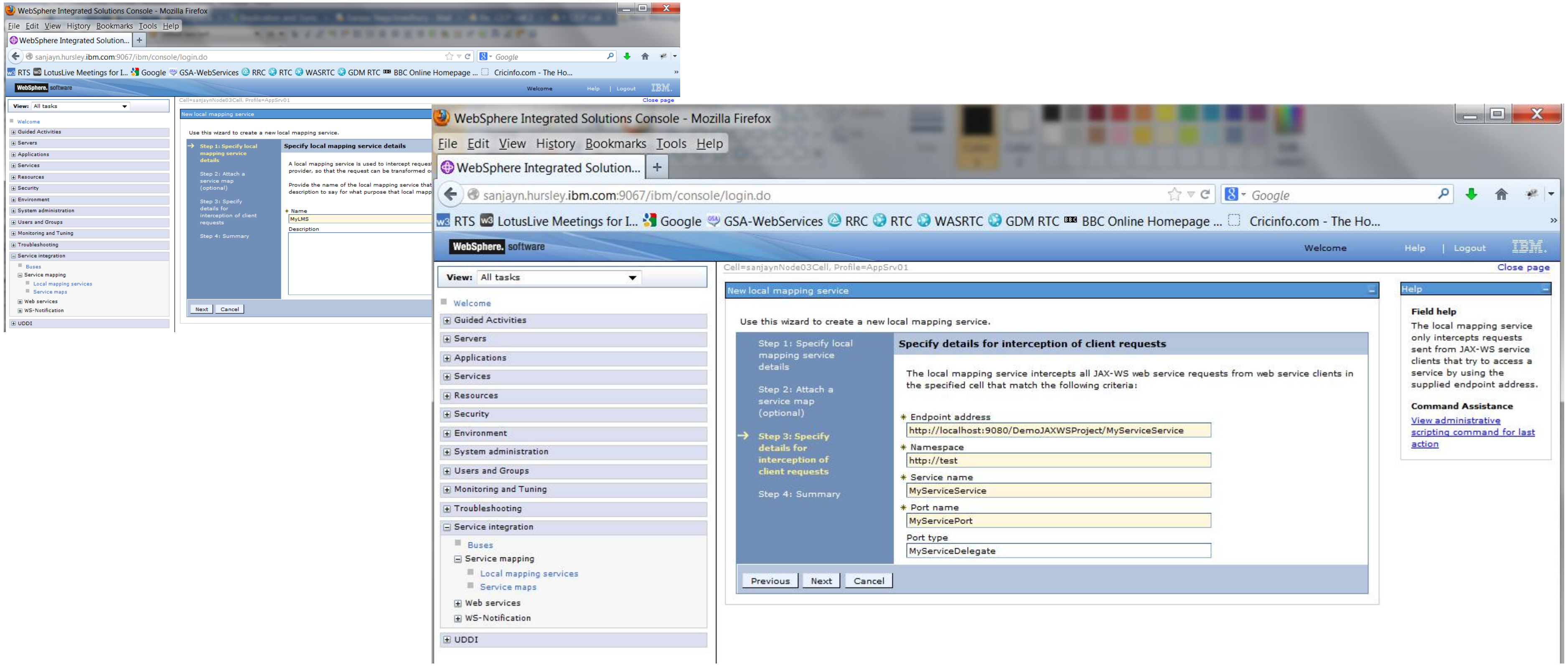
○ Remote file system

Full path

WAS Deployment & Administration - 2

- Administrator defines a Local Mapping Service
  - Specify a Service Map to attach to (optional)
  - Specify the endpoint address of a service provider - all calls to that service provider are intercepted, but are only routed/transformed if a service map is attached

- Web Service messages from the Service Client (or messages to the Service Provider) are intercepted by the Mapping Service and re-routed and/or transformed according to the Service Map



Cell=sanjaynNode03Cell, Profile=AppSrv01

New local mapping service

Use this wizard to create a new local mapping service.

→ Step 1: Specify local mapping service details

Step 2: Attach a service map (optional)

Step 3: Specify details for interception of client requests

Step 4: Summary

Specify local mapping service details

A local mapping service is used to intercept request provider, so that the request can be transformed o

Provide the name of the local mapping service that description to say for what purpose that local mapp

* Name: MyLMS

Description:

Cell=sanjaynNode03Cell, Profile=AppSrv01

New local mapping service

Use this wizard to create a new local mapping service.

Step 1: Specify local mapping service details

Step 2: Attach a service map (optional)

→ Step 3: Specify details for interception of client requests

Step 4: Summary

Specify details for interception of client requests

The local mapping service intercepts all JAX-WS web service requests from web service clients in the specified cell that match the following criteria:

* Endpoint address: http://localhost:9080/DemoJAXWSProject/MyServiceService

* Namespace: http://test

* Service name: MyServiceService

* Port name: MyServicePort

Port type: MyServiceDelegate

Help

Field help

The local mapping service only intercepts requests sent from JAX-WS service clients that try to access a service by using the supplied endpoint address.

Command Assistance

View administrative scripting command for last action

Demonstration scenario

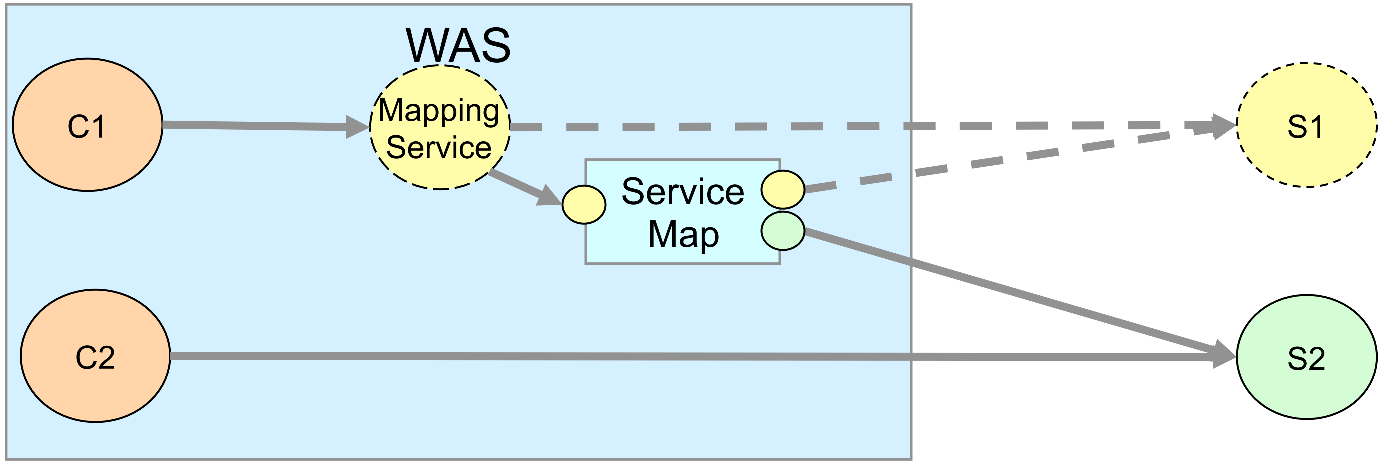
A simple story of Service Interface Evolution…

Timeline/ Stage	Service Provider versions	Service Consumer versions	Description	Mapping Needed
1	S1	C1	In the beginning – consumers and providers are created compatible	None required. C1 is configured to use S1
2	S2	C1, C2	Service provider is updated in a way that it not backwards compatible – e.g. An operation is updated to add a new response parameter	Interactions from C1 and need to have a transformation applied to cope with the new signature

User actions (1 of 2)

- For Version 1 Service Clients (C1)
  - Administrator
    - configures a Mapping Service for Version 1 Service Clients to intercept their existing interactions with the endpoint of the Version 1 Service (S1)
  - RAD Developer
    - Creates a service map designed to resolve the simple mismatch between Version 1 Clients (C1) and the version 2 web service provider (S2)
    - If (for some reason) there is a desire to continue to route some interactions to the Version 1 service, then this can be done by exploiting dynamic routing.

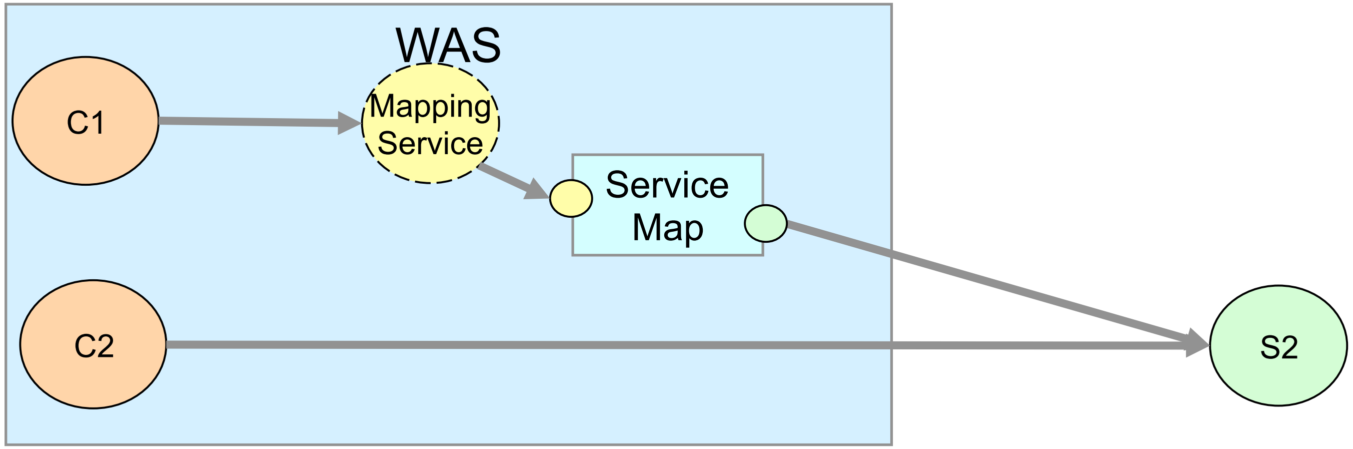
- For Version 2 Service Clients (C2)
  - No service map is required
  - Administrator
    - Does not need to define a Mapping Service: simply configures the Version 2 clients as usual



User actions (2 of 2)

- For Version 1 Service Clients (C1)
  - **Administrator**
    - configures a Mapping Service for Version 1 Service Clients to intercept their existing interactions with the endpoint of the Version 1 Service (S1)
  - **RAD Developer**
    - Creates a service map designed to resolve the simple mismatch between Version 1 Clients (C1) and the version 2 web service provider (S2)
    - If (for some reason) there is a desire to continue to route **some** interactions to the Version 1 service, then this can be done by exploiting dynamic routing.
    - **If not, then S1 can be taken out of commission, and all interactions are routed to S2 (with transformation as required)**

- For Version 2 Service Clients (C2)
  - No service map is required
  - Administrator
    - Does not need to define a Mapping Service: simply configures the Version 2 clients as usual



Demonstration

Demonstration summary

- The demonstration provides a brief run-though of both the Developer experience (in RAD) and the Administrator experience (using both commands and the WAS Admin console)

Summary

Summary

- **Service Mapping** provides a way of insulating an application that consumes a service from the details of that service provider's interface or location
  - This is achieved by providing a simple way of performing content-based routing and message transformation
- The **Service Map** extends the idea of a data map to cover:
  - Service-level mapping – selecting a target service
  - Interface-level mapping – mapping from source service operations to target service operations
  - Operation-level mapping – mapping the request, response and fault messages
- The **Mapping Service** intercepts web service requests from existing applications.
- **Mapping Service Event Points**
  - Can be configured for any Mapping Service
  - Publish events to a JMS Topic Space
  - Events can be consumed by any JMS topic subscriber application
  - Events consumption is explicitly supported by IBM Integration Bus
- **Service Mapping** is supported in WebSphere Application Server v8.5.5 (full profile – not Liberty) via tooling in RAD v9.0
  - Standard RAD/WAS development and administration tasks

References

- IBM WebSphere Application Server V8.5.5 information center Service Mapping topics
  - http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/topic/com.ibm.websphere.sm.nd.doc/ae/csmwas_servicemappingintro.html
- IBM Rational Application Developer V9.0 information center Service Mapping topics
  - http://pic.dhe.ibm.com/infocenter/radhelp/v9/topic/com.ibm.websphere.sm.rad.doc/ae/csmrad_servicemappingintro.html
- Service Mapping DeveloperWorks tutorial
  - http://www.ibm.com/developerworks/websphere/techjournal/1312_borley/1312_borley.html
- Further questions?
  - Andrew Borley - borley@uk.ibm.com

Questions?

The information on the new product is intended to outline our general product direction and it should not be relied on in making a purchasing decision. The information on the new product is for informational purposes only and may not be incorporated into any contract. The information on the new product is not a commitment, promise, or legal obligation to deliver any material, code or functionality. The development, release, and timing of any features or functionality described for our products remains at our sole discretion.