10

Adapters and Transports

Objectives

After completing this lesson, you should be able to:

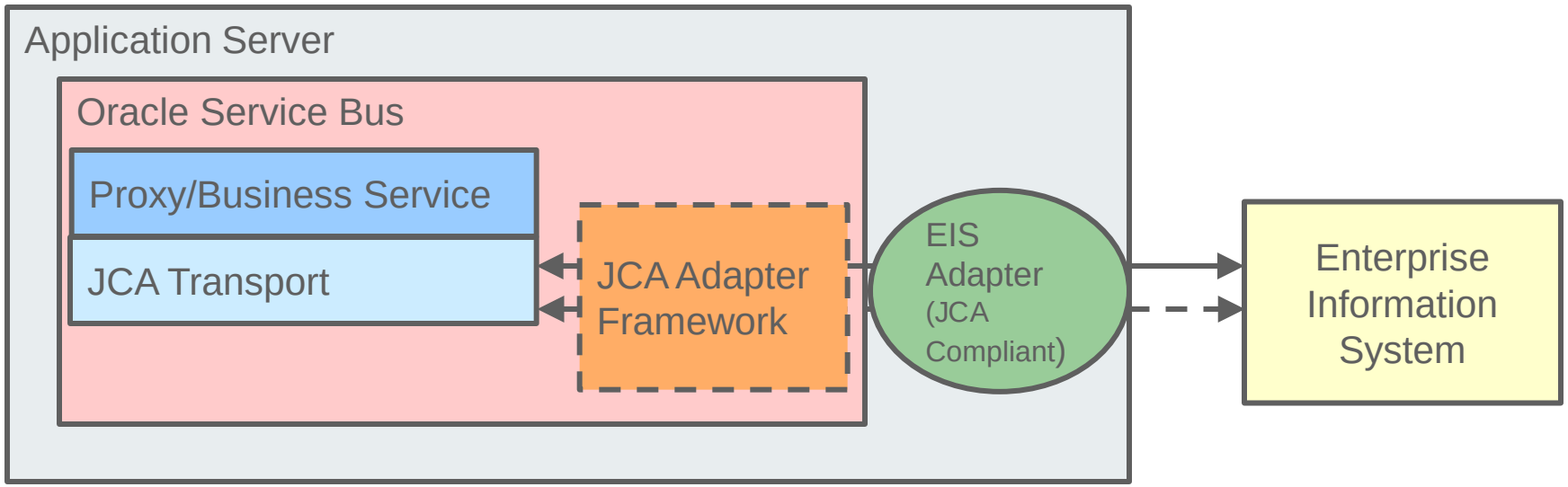
- Describe JCA transport and adapters
- Describe SOA-DIRECT transport
- Call SOA services using SOA-DIRECT protocol
- Expose existing SOAP services as REST services using the REST binding

Agenda

- JCA transport and adapters
- SOA-DIRECT transport
- REST binding

JCA Transport

- Provides native connectivity between Service Bus and external systems
- Uses JCA adapter framework to interact with JCA-compliant adapters that, in turn, provide connectivity to external EIS services

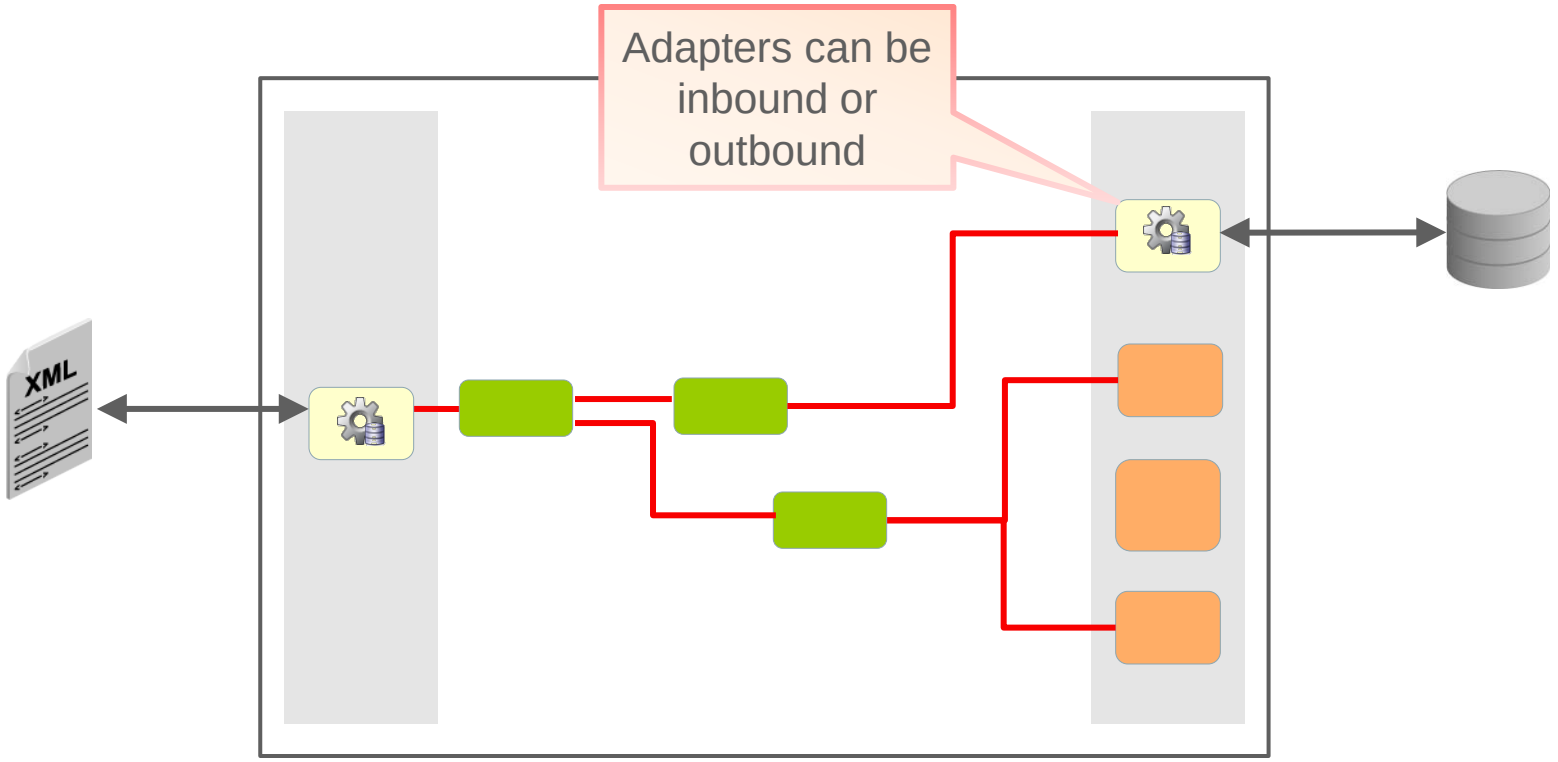


JCA Adapter Framework

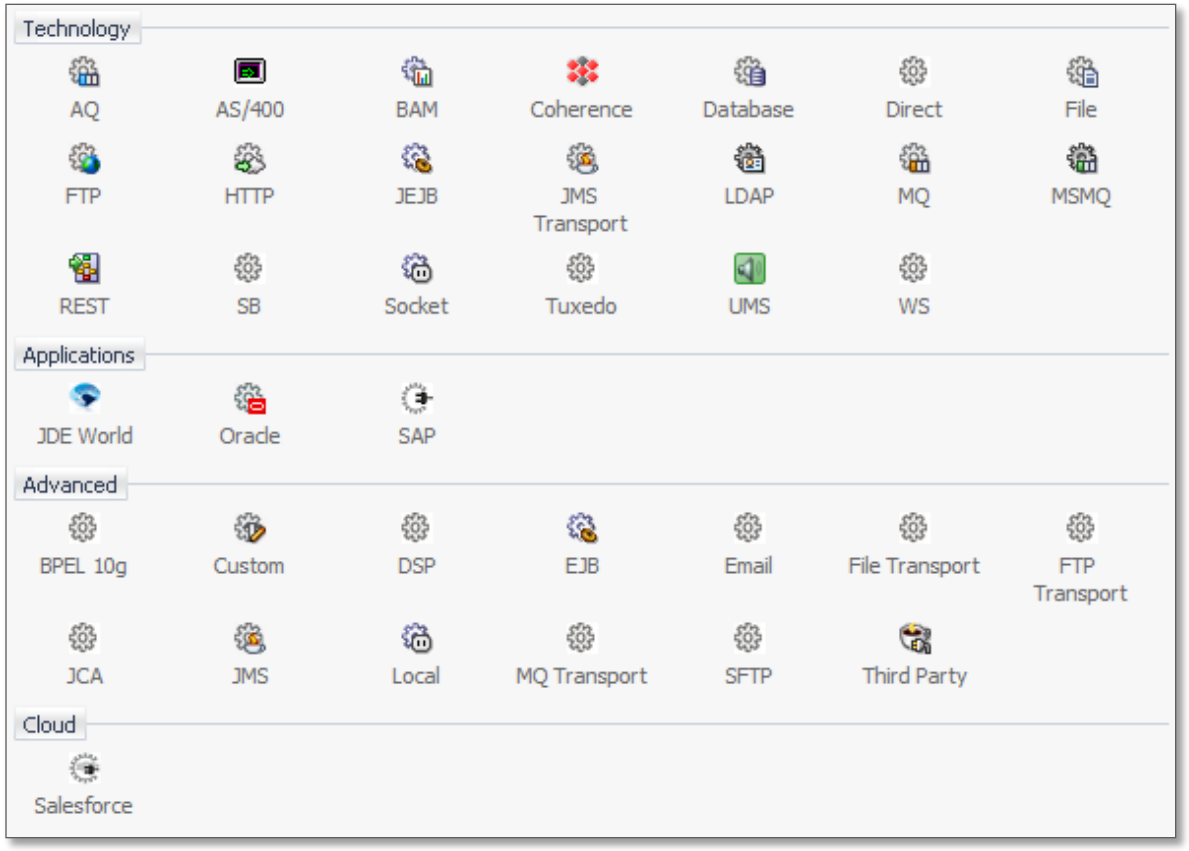
- The JCA transport uses the Oracle Service Bus JCA adapter framework to interact with JCA-compliant adapters that, in turn, provide connectivity to external EIS services.
- The JCA adapter framework abstracts the complexity of interacting with those adapters, enabling you to focus on proxy and business service development in Oracle Service Bus.

JCA Adapters

Adapters integrate with existing back-end applications through Java EE Connector Architecture standards.

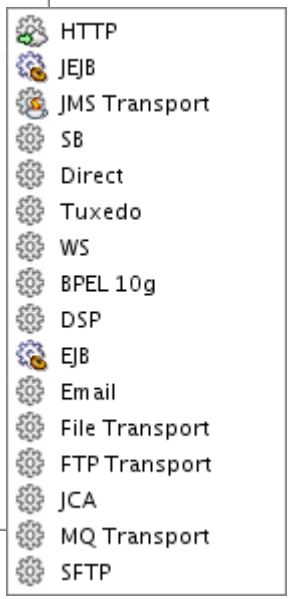


Supported Adapters and Transports

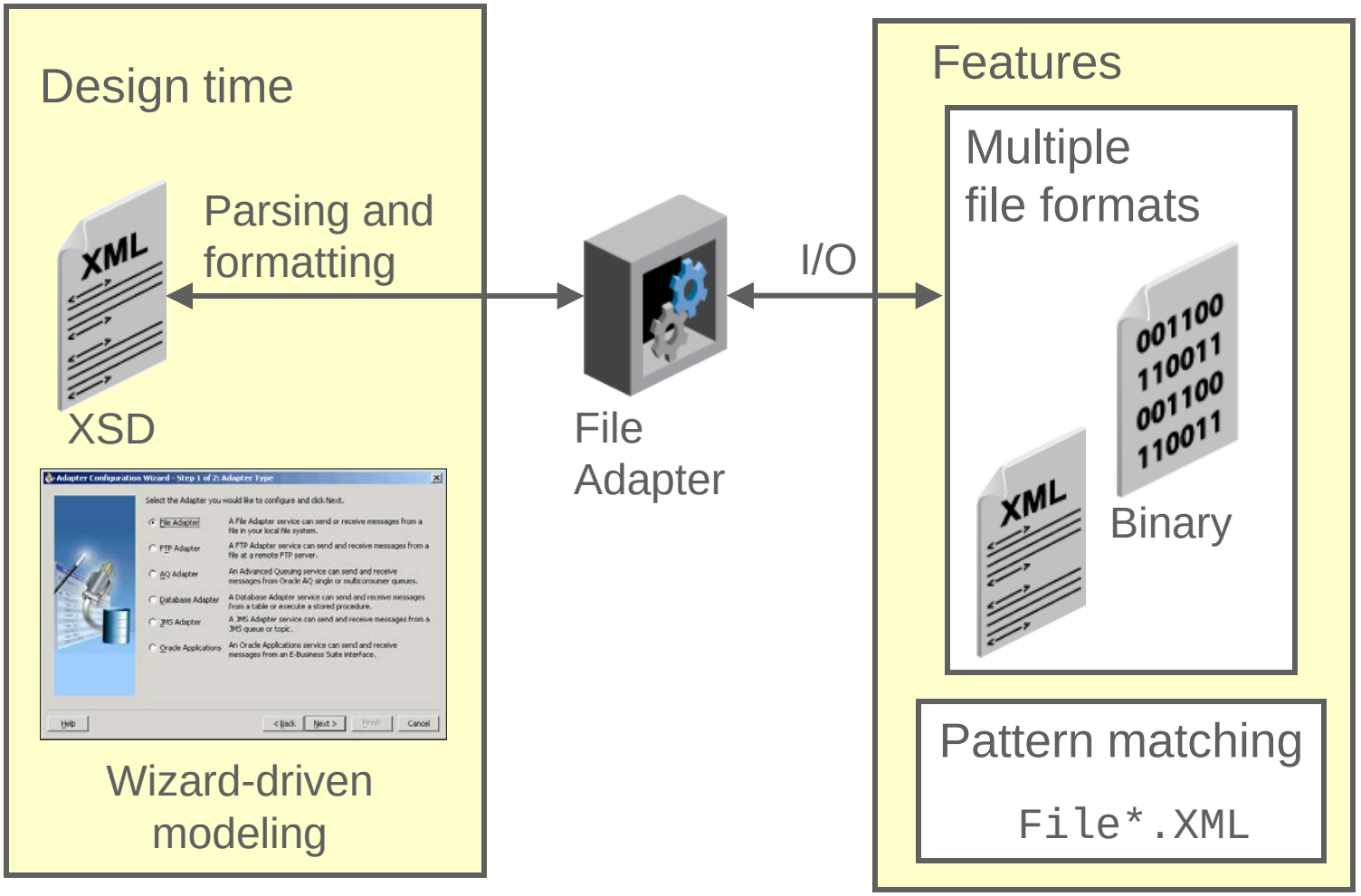


Adapters

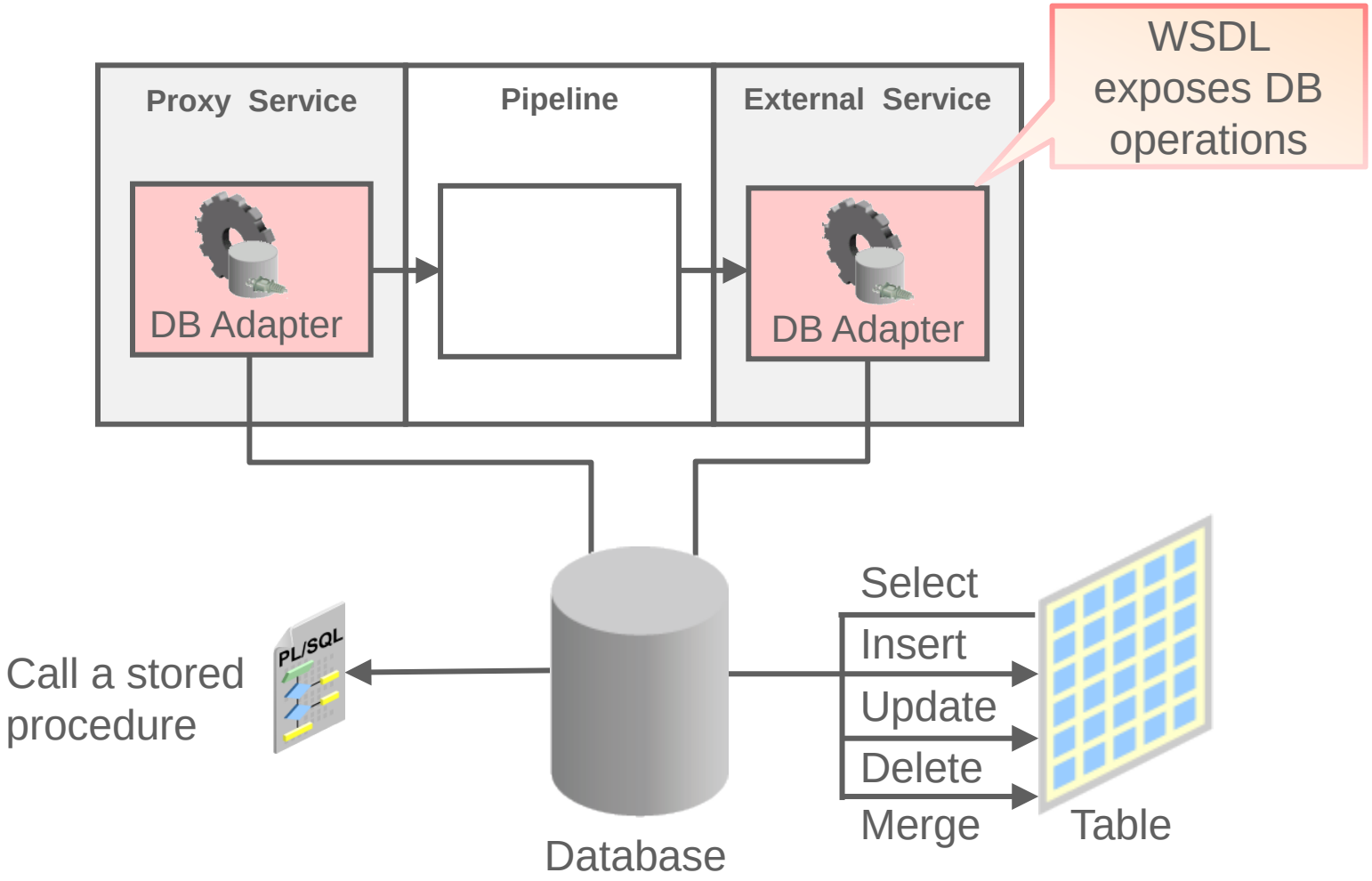
Transport



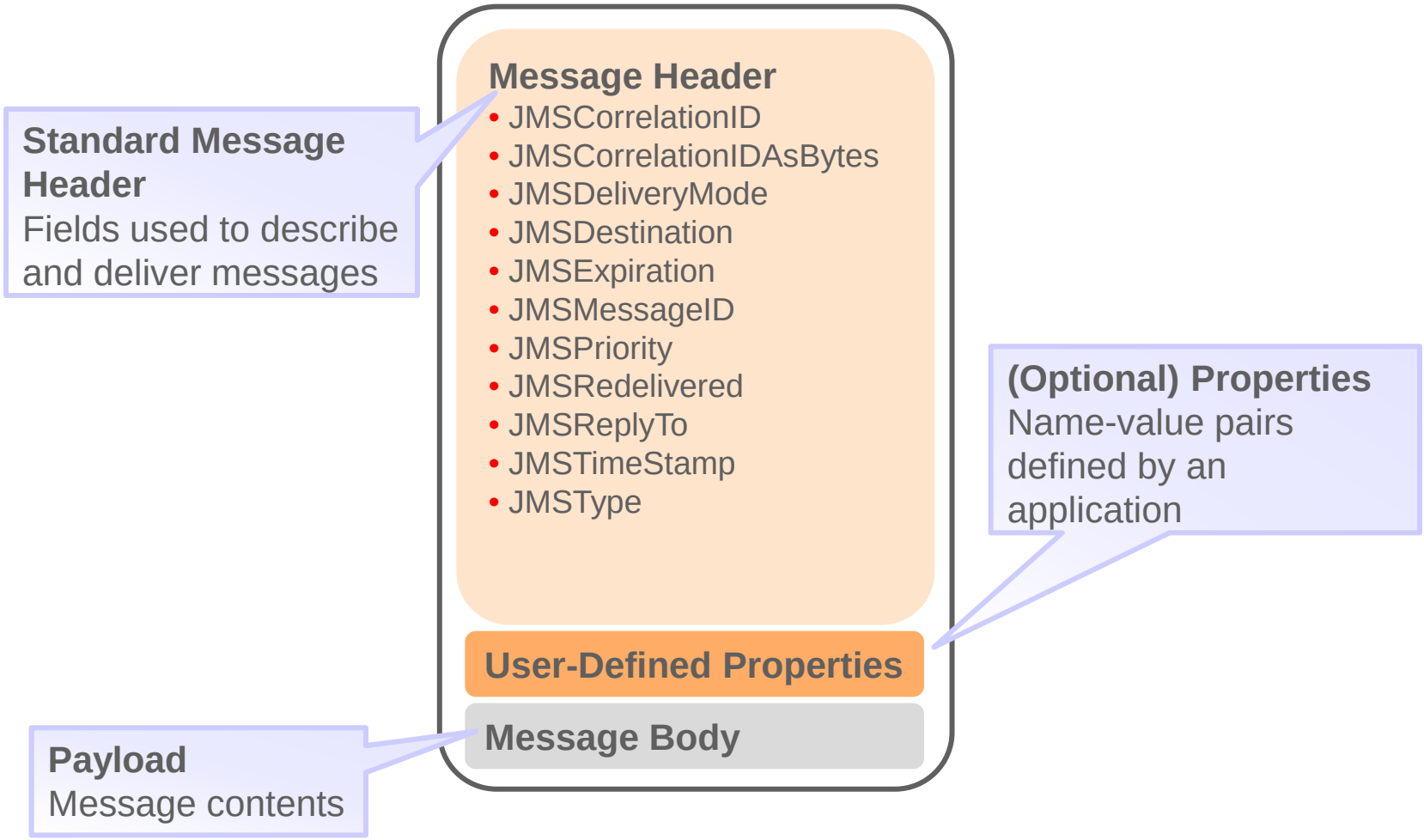
File Adapter



Database Adapter

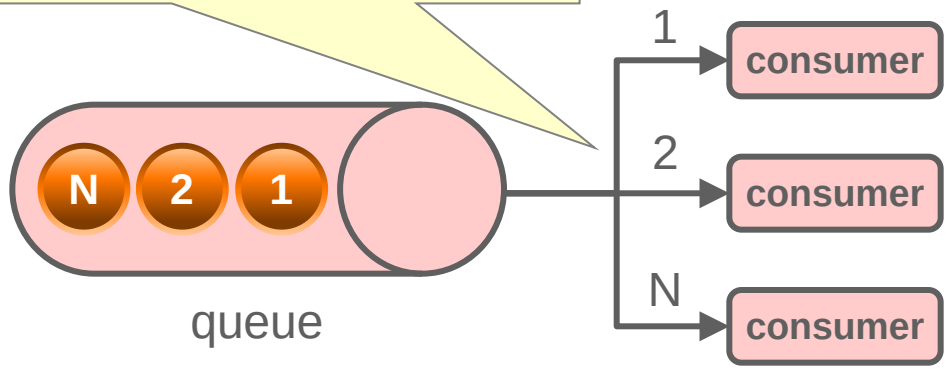


JMS Message

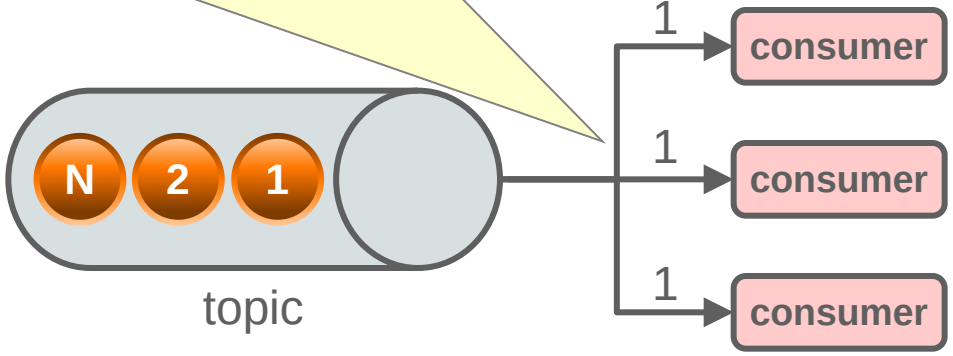


JMS Messaging

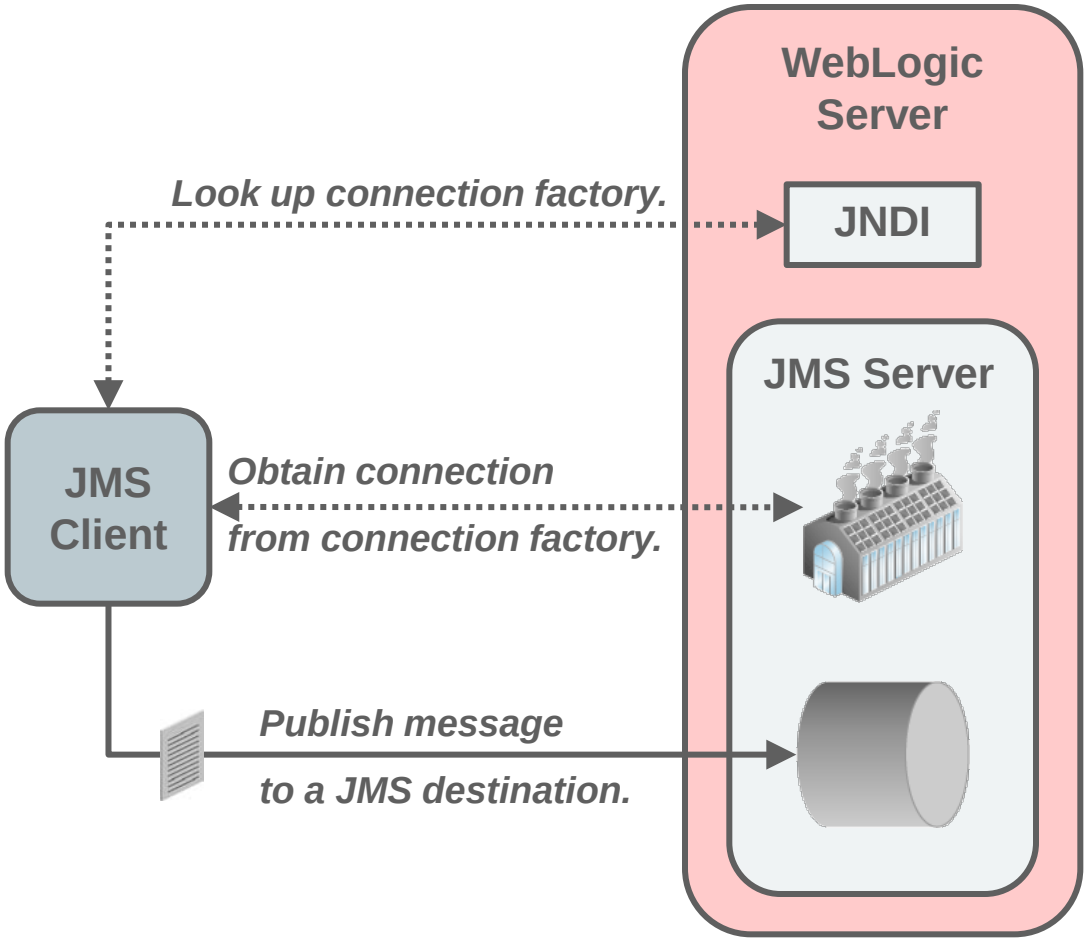
Point-to-point
Each message is delivered to only one consumer.



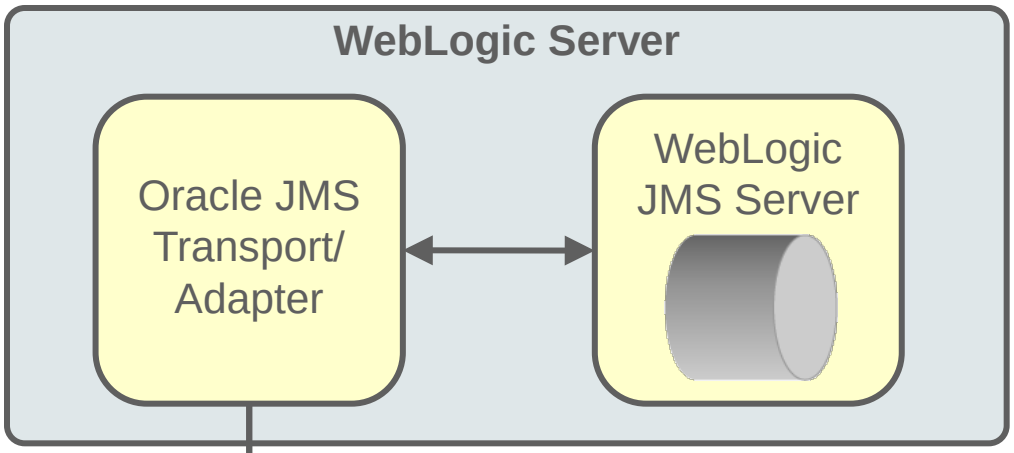
Publish/subscribe
Each message may be delivered to multiple subscribers.



JMS Resources



Oracle JMS Transport



Supported features include:
- JMS topics and queues
- Byte, text, and map message types
- JMS headers and properties
- JMS message selector
- Durable JMS subscriber
- Persistent and nonpersistent publication

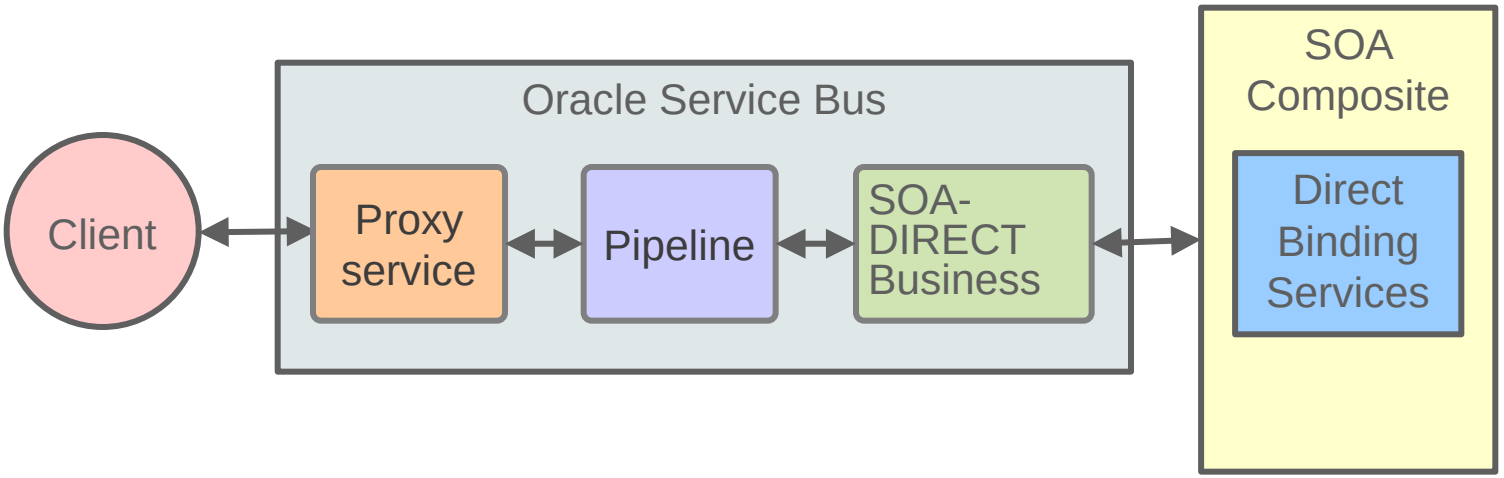
Supported JMS servers include:
- Active MQ
- Apache
- AQ JMS
- IBM WebSphere MQSeries
- TIBCO JMS

Agenda

- JCA transport and adapters
- SOA-DIRECT transport
- REST binding

SOA-DIRECT Transport

The SOA-DIRECT transport provides native connectivity between Oracle Service Bus and Oracle SOA Suite service components.



Features of SOA-DIRECT Transport

The SOA-DIRECT transport supports the following features:

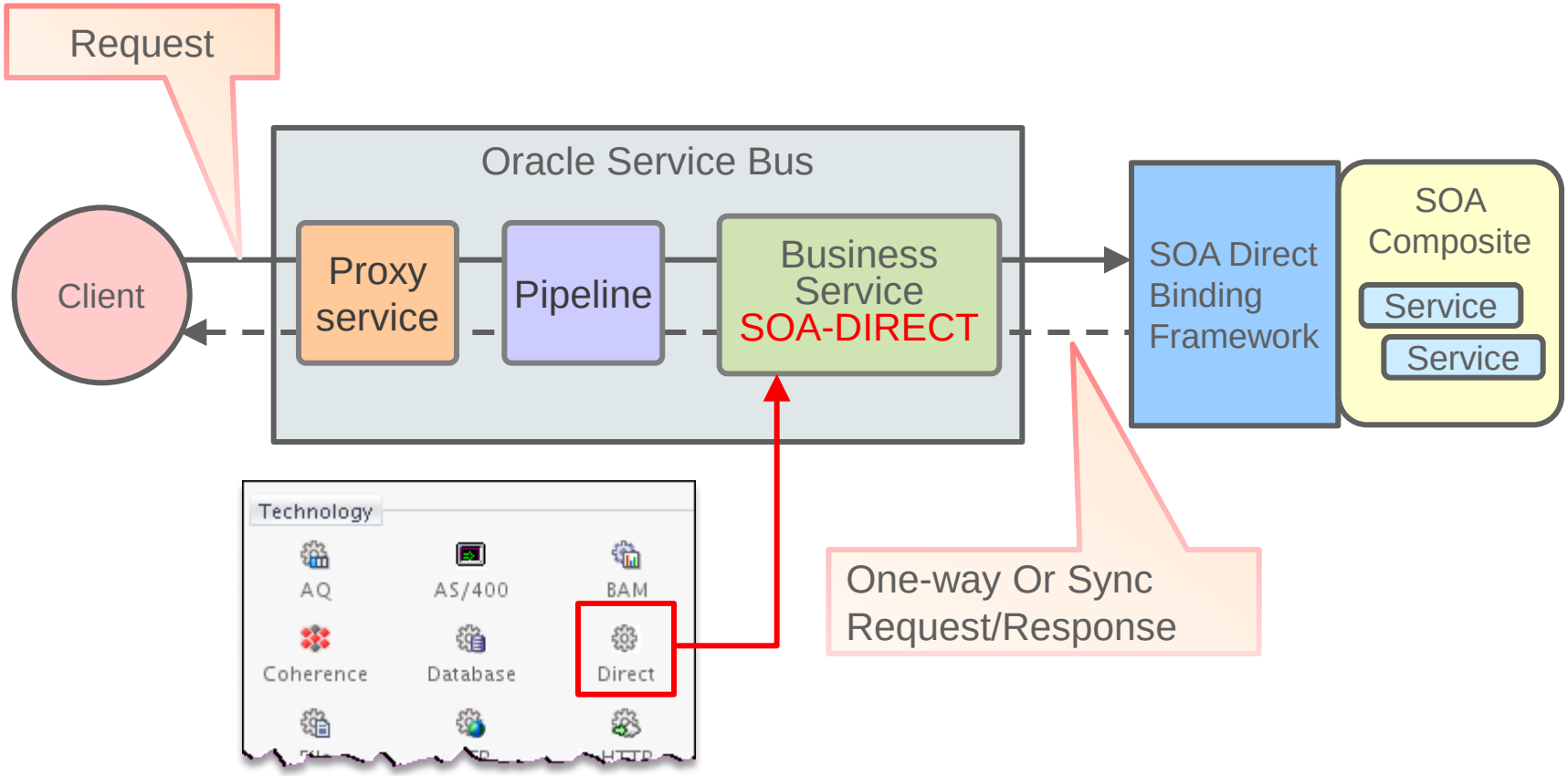
- Invocation of any SOA binding component services through Java remote method invocation (RMI)
- Optimized RMI transport for invoking SOA services
- Transaction propagation
- WS-Addressing
- Identity propagation
- Attachments
- High-availability and clustering support
- Failover and load balancing
- Connection and application retries on errors

Transactional Boundaries

- When synchronous BPEL components use the direct binding to interact with proxy services, Service Bus and BPEL components share the same transactional context.
- Transaction must be rolled back if something fails.
- Service Bus direct binding failures are thrown back to the BPEL component as system faults.

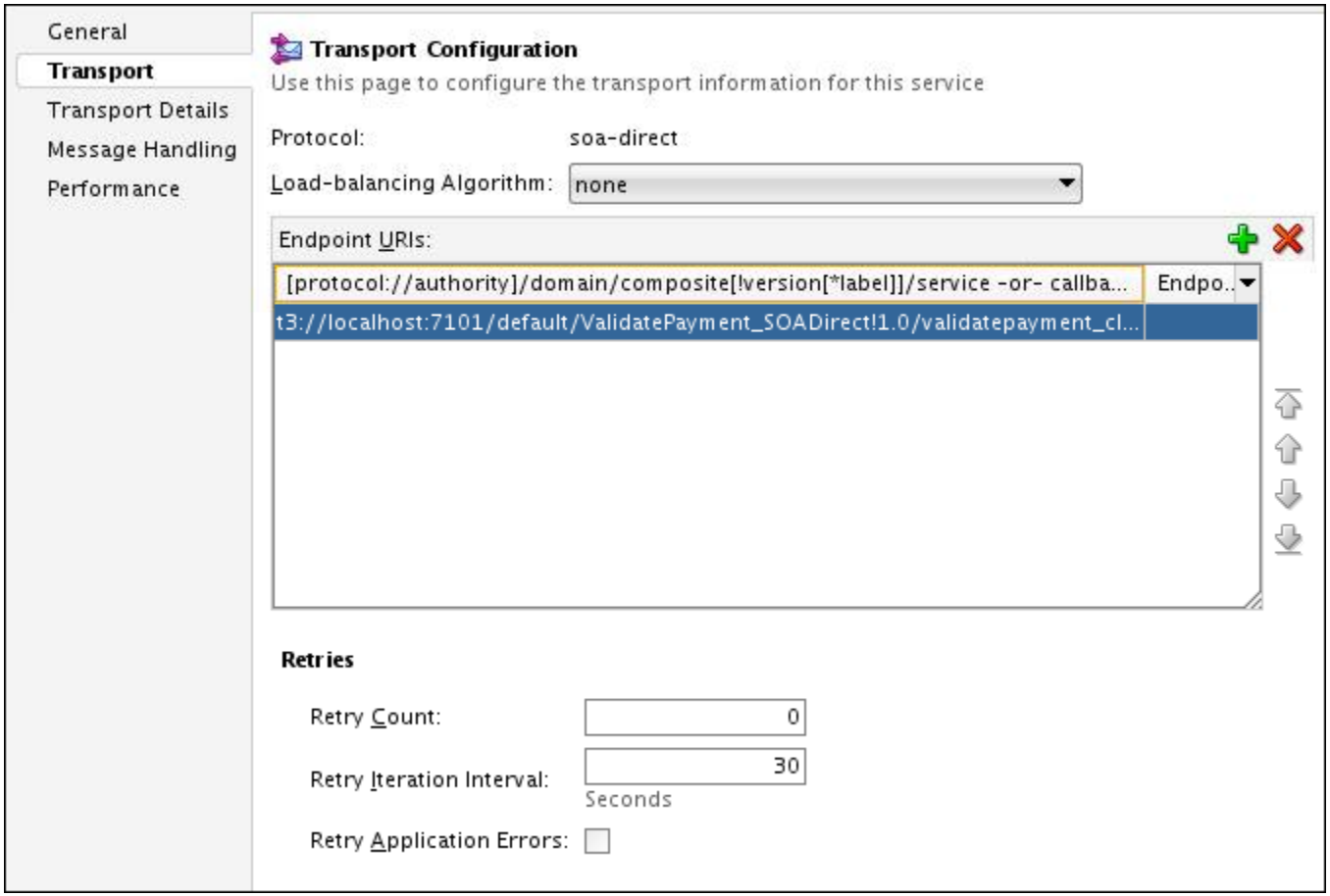
Synchronous Invocation of a SOA Composite

The SOA-DIRECT transport can invoke any component in a SOA composite that is exposed as a direct binding service.



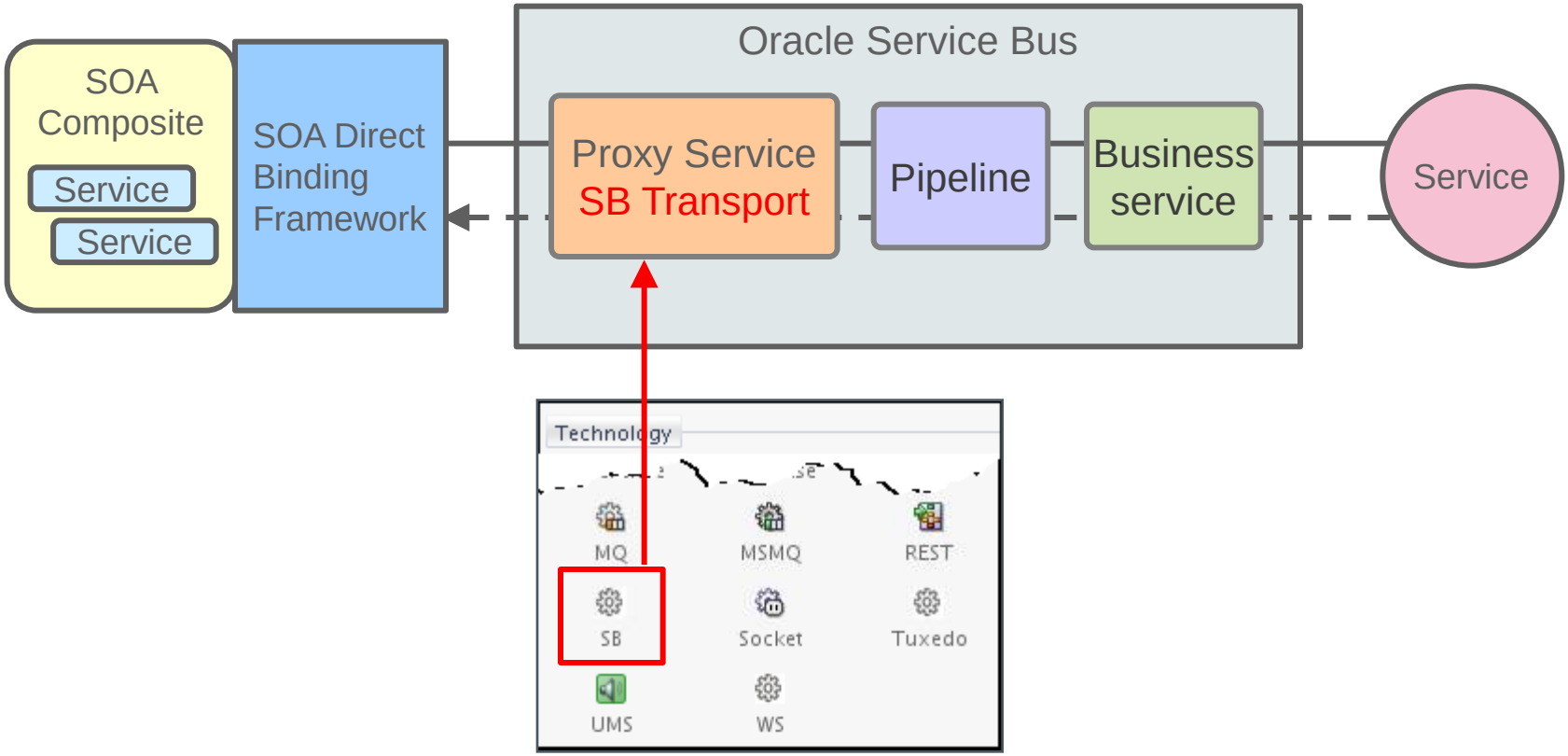
Endpoint URI Linking to a SOA Composite

Get the URI from `http://hostname:port/soa-infra`

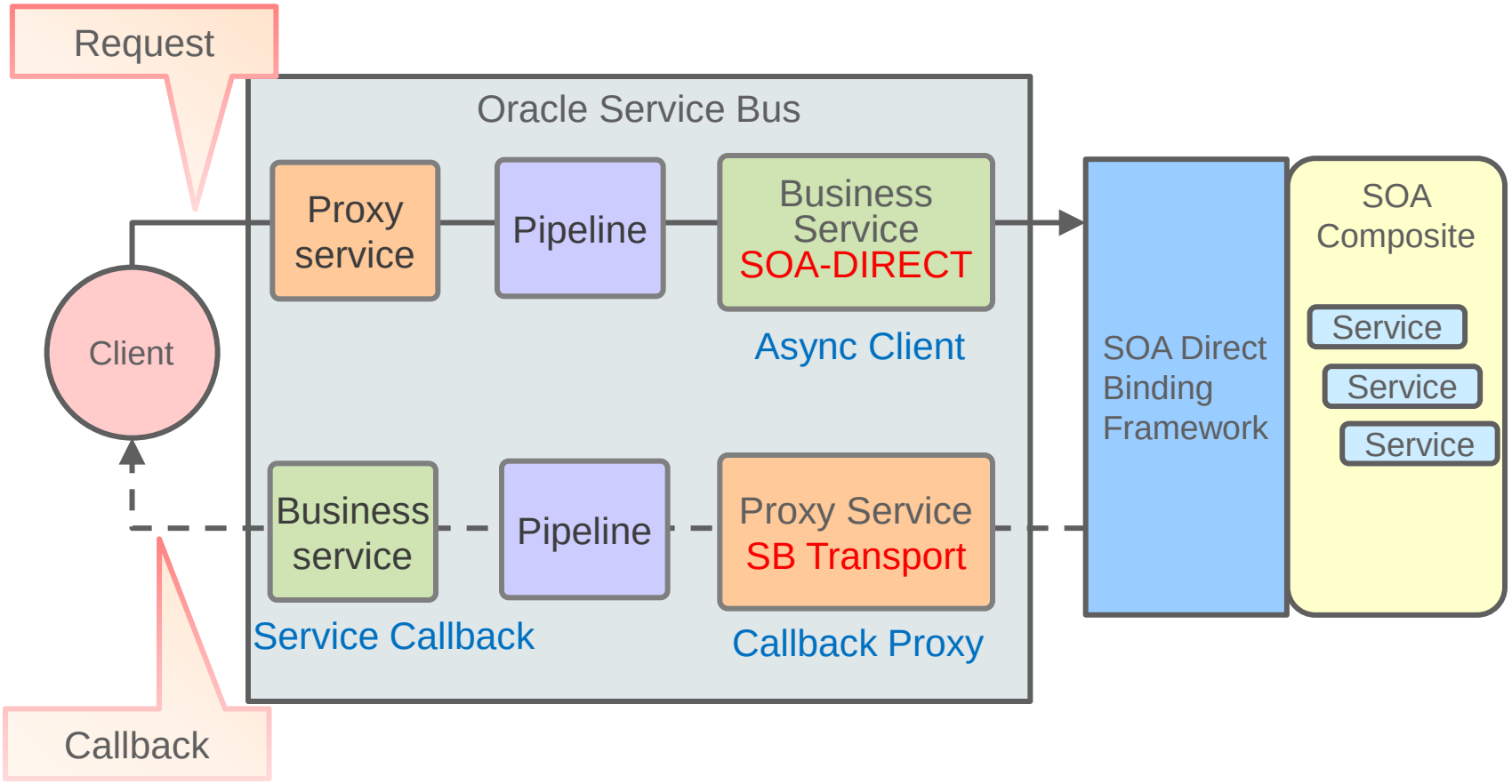


Synchronous Invocation from a SOA Composite

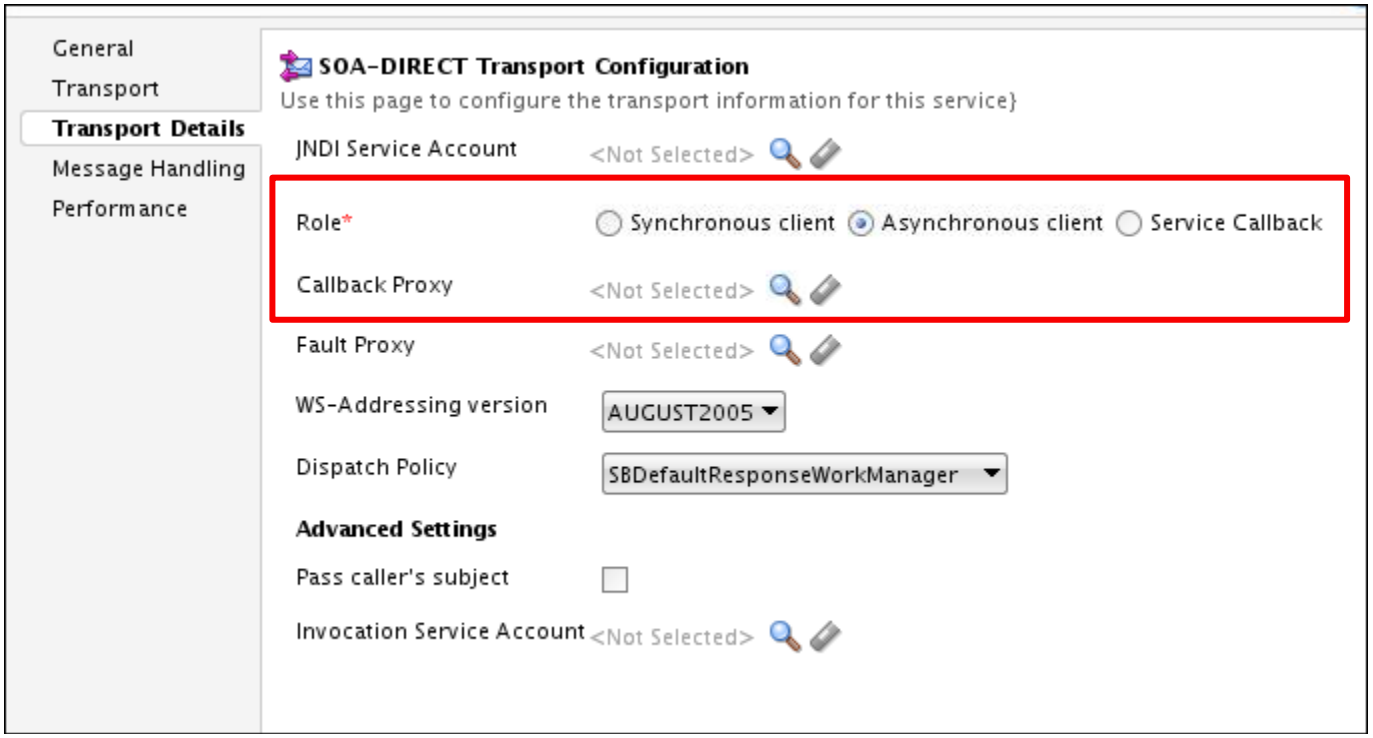
A SOA composite can invoke any Oracle Service Bus SB WSDL-based proxy service.



Asynchronous Invocation of a SOA Composite



Service Roles in Asynchronous Invocation

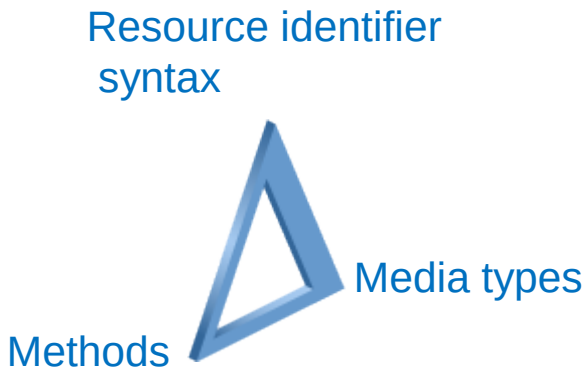


Agenda

- JCA transport and adapters
- SOA-DIRECT transport
- REST binding

REST: Overview

- REST is a model based on resources.
- RESTful services rely on the HTTP protocol.
- REST is based on three fundamental elements:
  - Resource identifier syntax – Represent the actual resources that a service exposes: URI
  - Methods – Protocol mechanisms used to transfer the data: GET, PUT, POST, DELETE
  - Media types – Type of data being transferred: XML, JSON



REST Queries

Action	Method	URI
Get all the items	GET	/items
Get a single item	GET	/items/id
Create a new item	POST	/items
Edit an item	PUT	/items/id
Delete an item	DELETE	/items/id

Comparing REST and SOAP

Action (REST)	Method (SOAP)
Is an architectural style that leverages web standards	Is a formal standard for message exchange
Uses HTTP	Is protocol independent
Permits many data formats	Uses XML
Uses URI and HTTP verbs to access *resources* (data)	Uses a WSDL document to access *operations* (business logic)
Is stateless	WS* standards provide support for stateful transactions.

Example: Use Cases for REST and SOAP

REST

- Operations with limited bandwidth and resources
  - Mobile
  - Series of short, chatty conversations
- Stateless operations

SOAP

- Operations that require contextual information and conversational state management
- Asynchronous operations
- Operations that require high levels of security and reliability

Supported Features in OSB

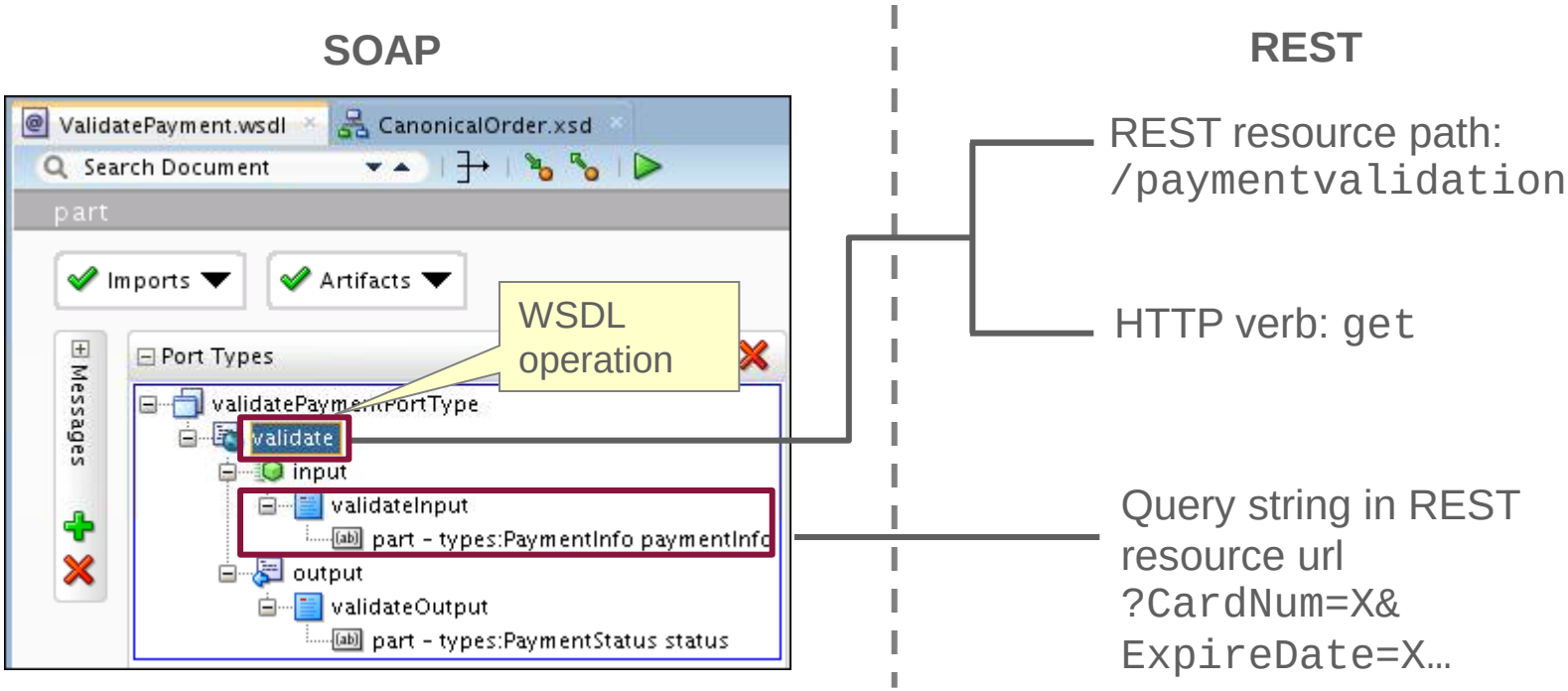
REST support in proxy and business services include:

- Integration with external REST APIs
- XML, JavaScript Object Notation (JSON) with translation to and from XML, and URL-encoded data
- Automatic creation of the required WADL file
- OWSM policy support for REST security
- Setting the HTTP link header in a REST proxy service response
- Reading the value of the HTTP link header in a REST business service response
- Overriding the endpoint URI for a REST business service request
- Ability to browse and consume Oracle REST endpoints from within JDeveloper

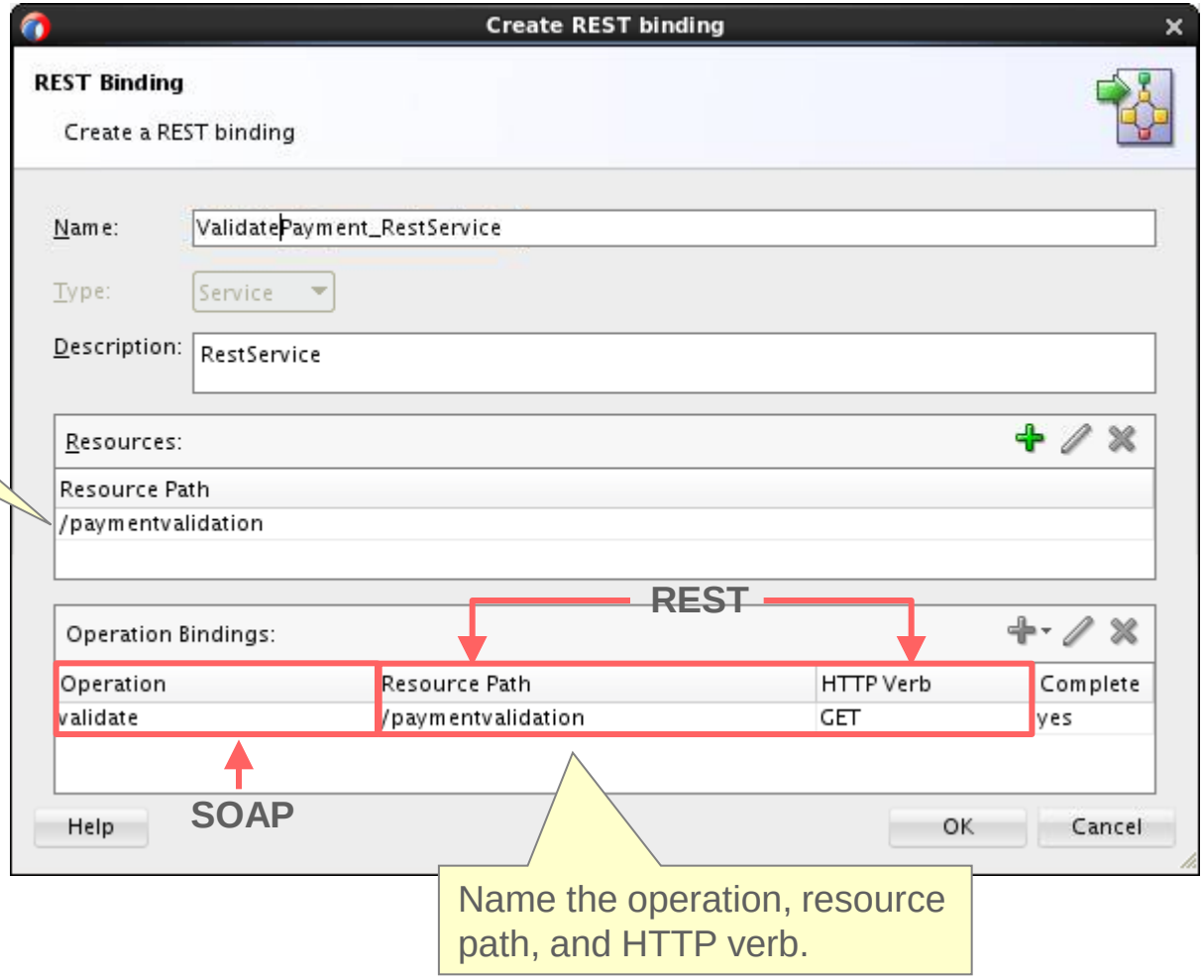
Creating a REST Binding

To create REST proxy services, you need to:

- Specify a RESTful resource path
- Map the HTTP verb of the resource path to a WSDL operation



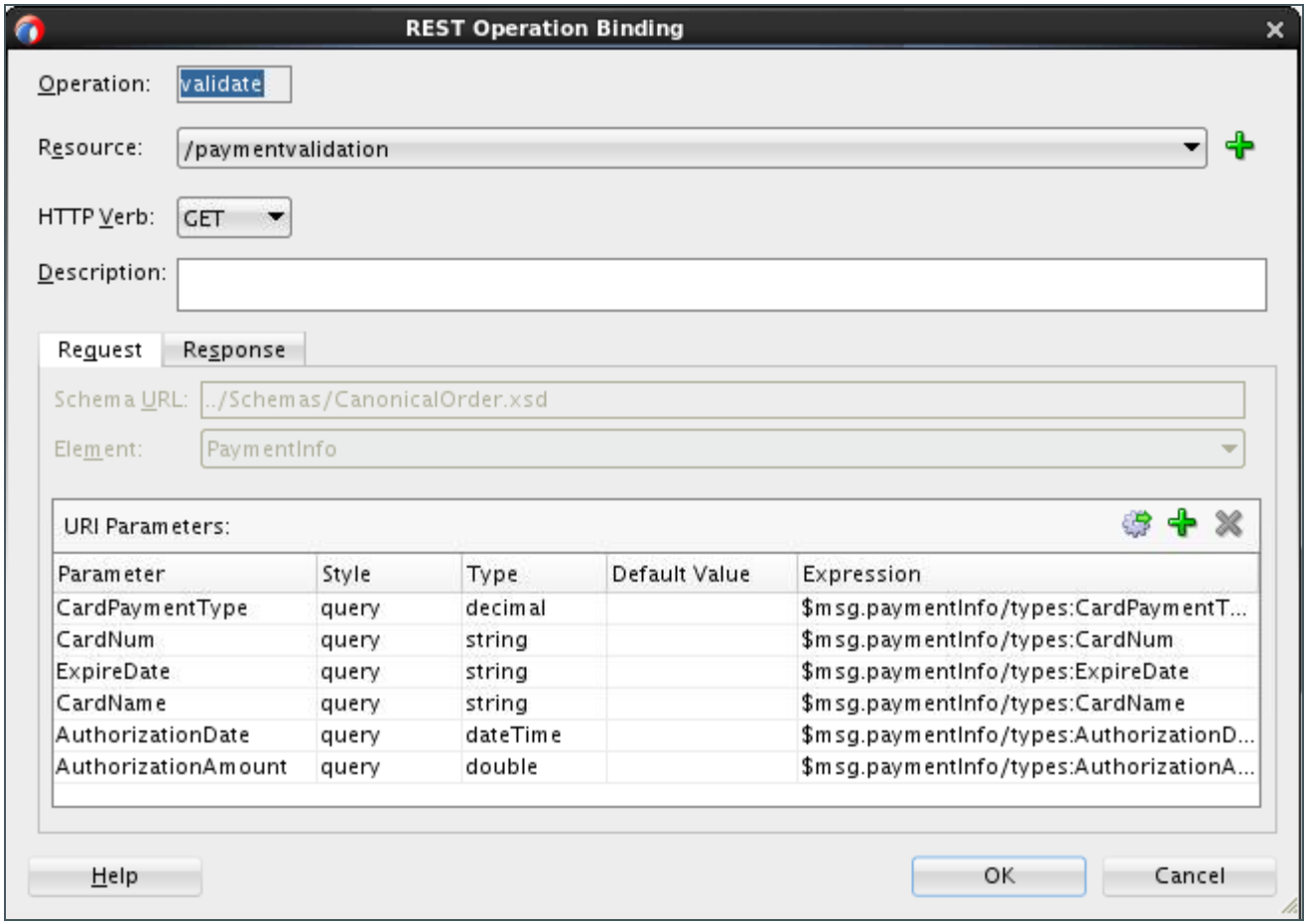
Configuring a REST Binding



Provide a relative path for the resources.

Name the operation, resource path, and HTTP verb.

Configuring a REST Binding



Summary

In this lesson, you should have learned how to:

- Describe JCA transport and adapters
- Describe SOA-DIRECT transport
- Call SOA services using SOA-DIRECT protocol
- Expose existing SOAP services as REST services using the REST binding

Practice 10: Overview

- 10-1: Calling a SOA Service Using SOA-Direct Protocol
- 10-2: Exposing a SOAP Service as a REST Service