

Enriching Messages

Objectives

After completing this lesson, you should be able to:

- Describe the message enrichment options
- Use service callout to enrich a request message before routing it to the target service
- Describe the usage of Java callout in OSB

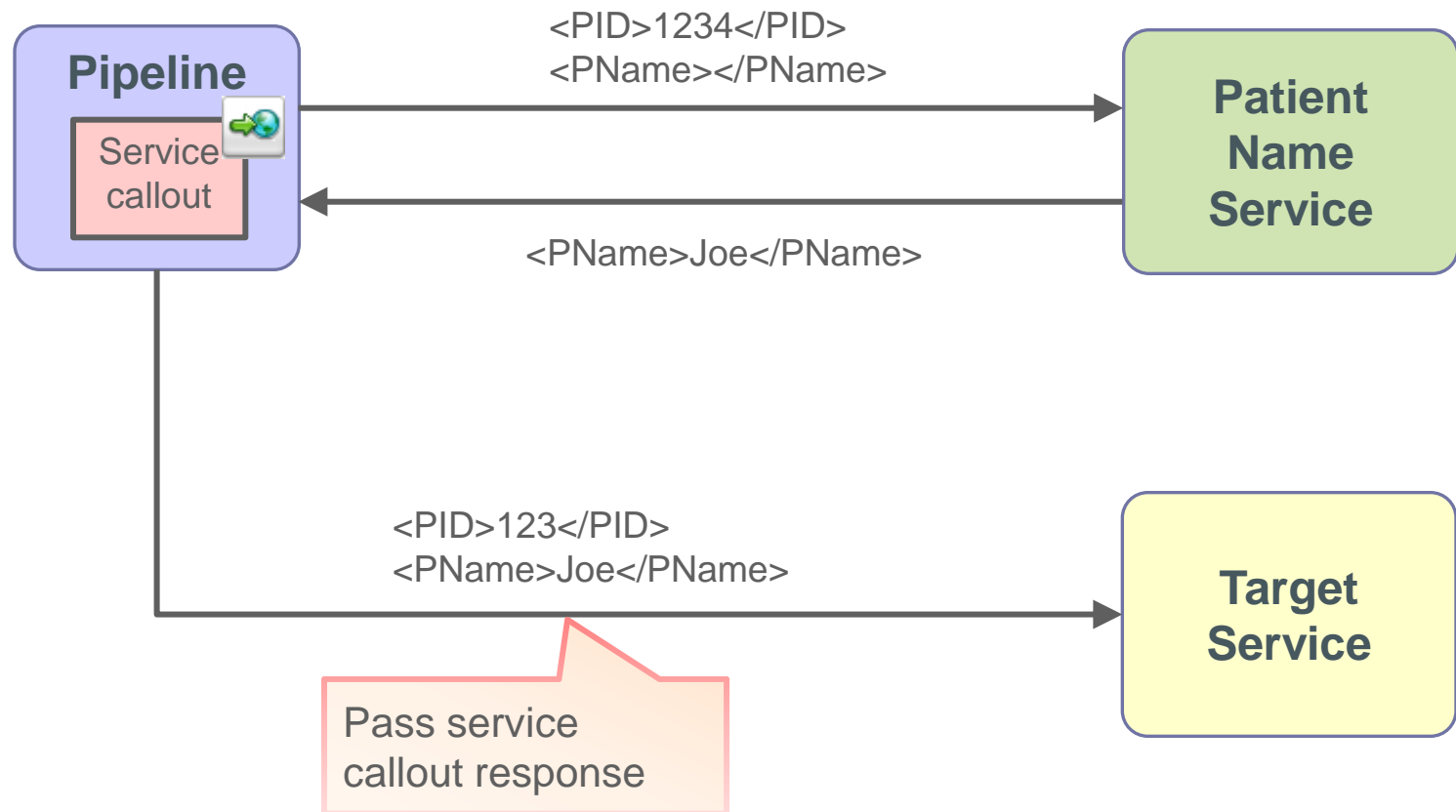


Message Enrichment Options

- Service Callout
- Java Callout

Service Callout

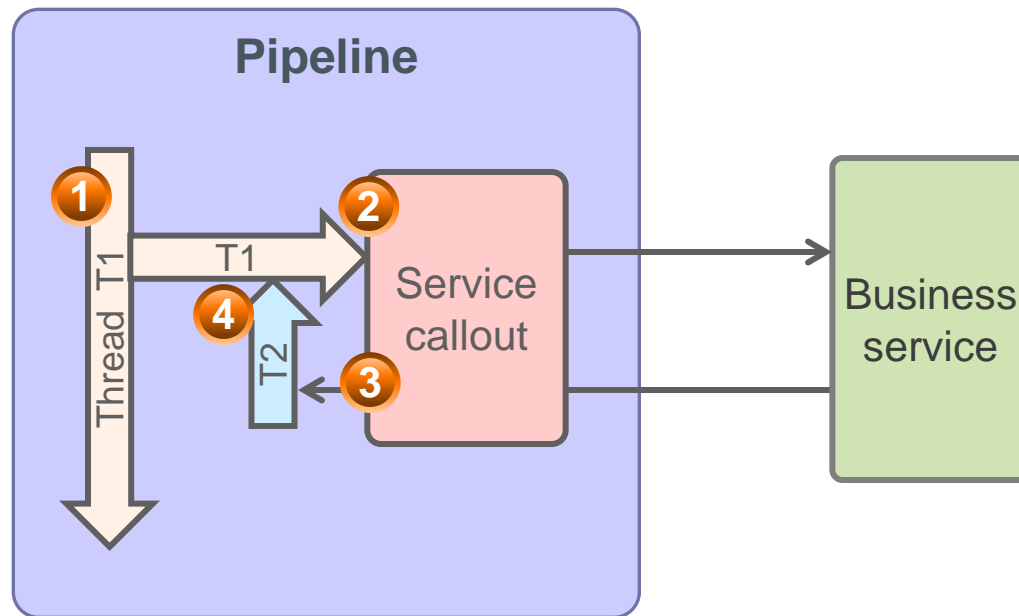
Enables you to query another service for a value that can then be used in the message flow



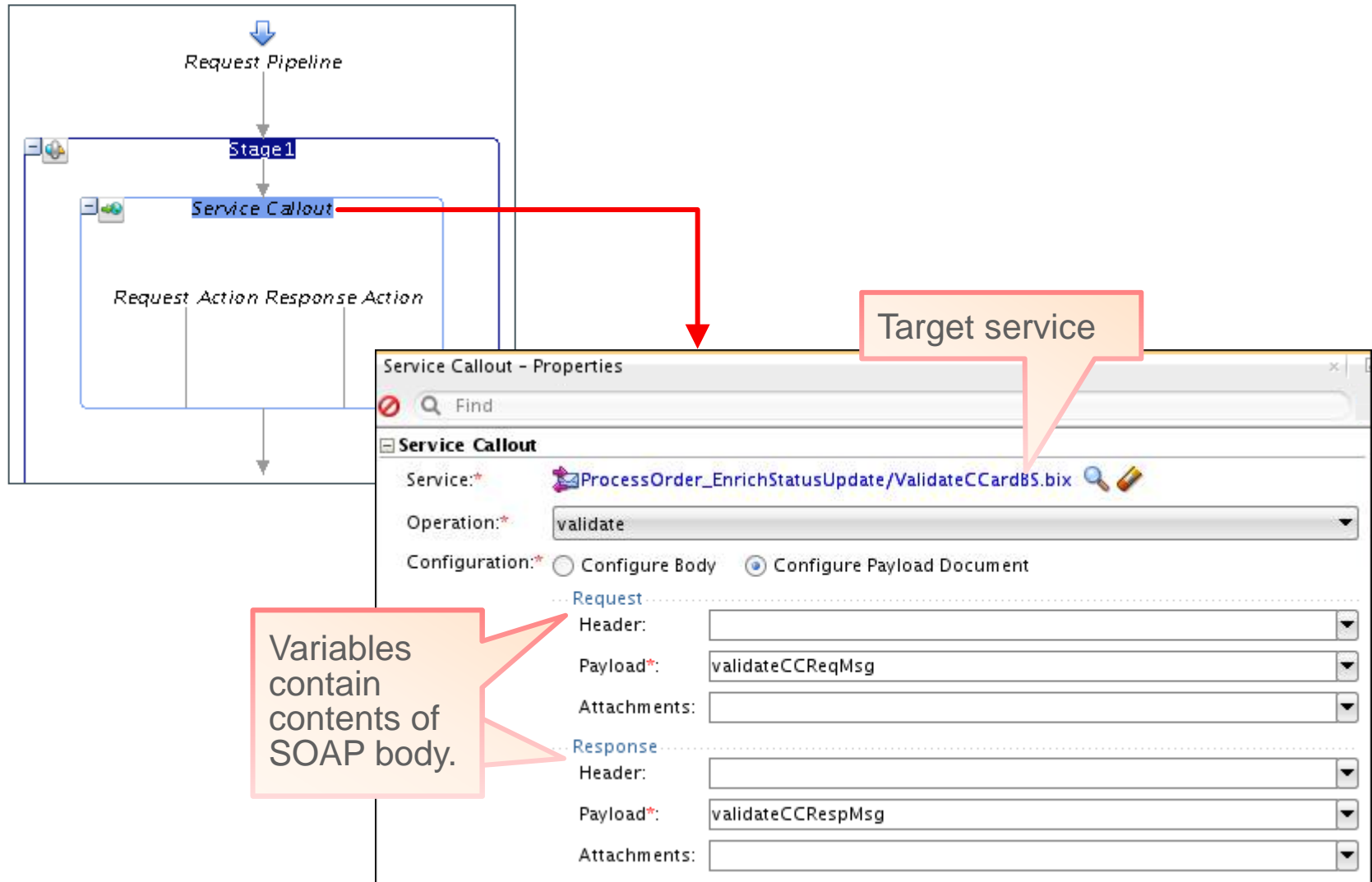
Using Service Callout

- Used for real-time request/response calls (Synchronous calls)
- Multiple service callouts can be executed in any order
- A response is needed from the service execution.
 - When responses are not needed, use Publish action.

Service Callout Action



Configuring Service Callout Action



Constructing Service Callout Messages

- The service options that can be selected for a service callout action are:
 - SOAP Document Style Service Callout
 - SOAP RPC Style Service Callout
 - XML Service Callouts
 - Messaging Service Callouts
- You can use attachments with service callout messages.

Java Callout

- Java Callout activity is used to invoke Java custom code when the OSB predefined functions are not sufficient to complete the requirement.
- The Java code must be packaged as jar.



Use Cases

- Performing custom validations, such as validating against a DTD, or doing cross-field semantic validation in Java
- Performing custom transformations, such as converting a binary document to base64Binary (or vice versa) or using a custom Java transformation class
- Performing custom authentication and authorizations
- Performing lookups for message enrichment. For example, a file or Java table can be used to look up any piece of data that can enrich a message.

Java Callout Guidelines

- The method written for Java callout must be static method.
- You can specify arguments for the method.
- Java callout is not recommended to call big methods with lots of business functionality behind it.

Java Callout Action

Messages - Log | Live Issues: EmployeeLoanProcessPSPipeline.pipeline - Issues | Java Callout - Properties

Find

Java Callout

Method: * `in.bench.resources.servicebus.javacallout.demo.EmployeeValidationJavaCallout.validateEmployeeDetail(java.lang.String, java.lang.String, java.lang.String)`
Jar: `Resources/jars/EmployeeValidationService.jar`

Arguments: *

Type	Value
java.lang.String	<code>\$body/loan:ProcessLoanRequestType/loan:employeeId/text()</code>
java.lang.String	<code>\$body/loan:ProcessLoanRequestType/loan:employeeName/text()</code>
java.lang.String	<code>\$body/loan:ProcessLoanRequestType/loan:employeeGrade/text()</code>

Return: *

Type: `java.lang.String`
Variable: `employeeValidated`

☐ As Reference

Service Account: `<Service Account>`

Optionally set Service Account for security purposes.

General

Summary

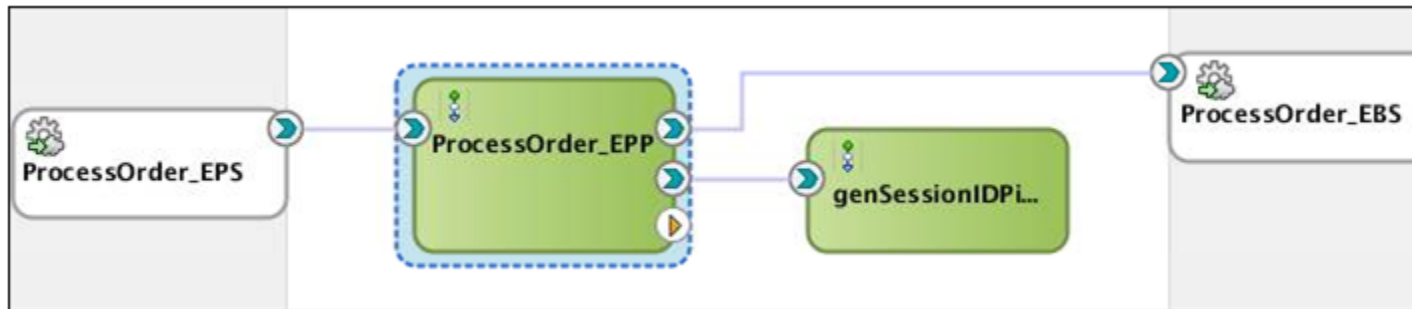
In this lesson, you should have learned how to:

- Describe the message enrichment options
- Use service callout to enrich a request message before routing it to the target service
- Describe the usage of Java callout in OSB



Practice 8: Overview

- 8-1: Enriching the Message by Calling a Pipeline Service (Session ID Generation)



- 8-2: Enriching the Message by Calling a Business Service (ValidateCreditCard)

