Introduction

Integration Framework Service is used by to perform event delivery to a middleware and to perform transformations

Overview

Integration Framework Service handles the following the high-level operations:

- Transformations
- Event Delivery through middleware

Invoking Framework Service Operations

Framework Service is available as a T24 component and can be invoked using the following interfaces:

- JBC API will be available to use by other T24 components
- C++ API will be available to use by 3rd party systems with C++ supported technologies.
- Java API will be available to use by 3rd party systems with Java supported technologies.

Note: Refer "Deploying a component Service" User Guide, to know about the various ways in which a service could be deployed. This user guide deals with the configuration of Framework Service as Web Service.

Framework Service Methods

The following table lists the various Framework Service Methods:

findEventsByFlow	This method is used find the events based on the flow name.
pollEventsByFlow	This method is used poll events from based on the flow name
pollEventsByCriteria	This method is used poll events from based on the provided criteria
pollLatestEventsByCriteria	This method is used poll events from based on the provided criteria that are generated after the timestamp provided as input
getTransactionId	This method is get the transaction ID for which the event will be genetated for a TSA Service exit point.
setEventsDelivered	This method marks the events that are delivered as delivered
transform	This method constructs the event XML from the data

Framework Service Method Explained:

findEventsByFlow

This method is used find the events based on the flow name.

Parameter Name	Type	Description	Direction	Format
flowName	String	Holds the name of the flow that is to be searched	IN	
eventMessages	EventMessages	Holds the list of event that belongs to the flowName	OUT	

pollEventsByFlow

This method is used poll events from based on the flow name

Parameter Name	Type	Description	Direction	Format
flowName	String	Holds the name of the flow that is to be searched	IN	
eventMessages	EventMessages	Holds the list of event that belongs to the flowName	OUT	

pollEventsByCriteria

This method is used poll events from based on the provided criteria

Parameter Name	Type	Description	Direction	Format
flowName	String	Holds the name of the flow that is to be searched	IN	
eventMessages	EventMessages	Holds the list of event that belongs to the flowName	OUT	

pollLatestEventsByCriteria

This method is used poll events from based on the provided criteria that are generated after the timestamp provided as input

Parameter Name	Type	Description	Direction	Format
flowName	String	Holds the name of the flow that is to be searched	IN	
latestByTimestamp	String	Holds the Timestamp	IN	
eventMessages	EventMessages	Holds the list of event that belongs to the flowName and created after the latestByTimestamp	OUT	

getTransactionId

This is a default implementation to the Integration Framework call-back which provides an option for implementations to override the key to use in the event generation during the TSA service exit points. This implementation is to return the incoming key as it is - not providing any additional functionality.

Parameter Name	Type	Description	Direction	Format
flowBaseData	FlowBaseData	Holds the flow name and flow source	IN	
serviceContext	ServiceContext	Holds the details about the TSA job, the processing ID, Date and the Company ID	IN	
transactionId	TransactionId	New key based on which IF generated events	OUT	

setEventsDelivered

This method marks the events that are delivered as delivered

Parameter Name	Type	Description	Direction	Format
eventMessageIds	String	List of message ID's that are to be marked as delivered	IN	

Transform

This is a default implementation to the Integration Framework call-back which provides an option for implementations to apply any transformation to have the event building further customized. It is expected that the implementations would make use of the incoming data to 1) filter any unwanted events 2) modify the event data to get a customized event XML built or 3) provide an event XML document completely built outside which integration framework would simply take it for delivery.

This default implementation would return the "TransformResult>transformStatus" to be "NO_TRANSFORM" to indicate that there is no transform activity performed so that integration framework would go ahead to build the event XML.

Parameter Name	Type	Description	Direction	Format
flowMetadata	flowMetadata	Holds the details about the flow	IN	
transactionContext	TransactionContext	Holds the transaction data and the event common data	INOUT	
transformResult	TransformResult	Holds the transform status and the transformed event	OUT	