



Expect a Trusted Partner
Expect a Trusted Partner

Infrastructure Series

TechDoc

WebSphere Message Broker / IBM Integration Bus

*Subflows: Usage & Nodes
(Message Flow Development)*





Table of Contents

Introduction	3
Document Version	3
Product Naming History	3
Product Component Terminology	3
Subflows	5
Overview	5
Subflow Creation	5
File Type: “.subflow”	5
File Type: “.msgflow”	5
Promoting Properties	5
Subflow Versioning	6
Deployment Time	6
Runtime	6
Best Practices	6
References	6



Introduction

Document Version

This document describes how to use Subflows to create reusable Message Broker/Integration Node solutions. Subflows make use of the following two specialized nodes:

- Input Node 
- Output Node 
- Passthrough Node 

The contents of this document have been specifically verified on the following production versions:

- WebSphere Message Broker v7.0.0.2
- IBM Integration Bus v9.0.0.0

This documentation has been created and maintained by:

- Glen Brumbaugh

This documentation has been reviewed by:

- Glen Brumbaugh

This documentation was last updated:

- Version 1.0 June 2015

Product Naming History

The product currently known as IBM Integration Bus has been through a number of different product names during its several decade long evolution. The product was originally developed by the New Era of Networks (NEON) Corporation and was marketed and resold by IBM. IBM completely redesigned and rebuilt the product and released their own in-house developed product beginning with version 2.0. The product has had the following names and version numbers:

- | | |
|---|--------------------|
| • MQSeries Integrator (MQSI) | Version 1.0 – 2.0 |
| • WebSphere MQSeries Integrator | Version 2.1 |
| • WebSphere Business Integration Message Broker (WBIMB) | V5.0 |
| • WebSphere Message Broker (WMB) | Version 6.0 - 8.x |
| • IBM Integration Bus (IIB) | Version 9.0 - 10.0 |

For the remainder of this document, the product will be referred to as “*Message Broker*”. This is both for historical reasons and to signify that this documentation applies to both the WMB and IIB product versions.

Product Component Terminology

With the Version 9.0 product rename (to IBM Integration Bus), several key product architectural components were given new names; while continuing to fill virtually the same role they had previously filled. This documentation will continue to refer to the “old” names because the steps documented here refer to both old and new product versions.



The old and corresponding new names are as follows:

- Message Broker → Now called “Integration Node” (Beginning with v9.0)
- Execution Group → Now called “Integration Server” (Beginning with v9.0)
- Message Flow → Still called “Message Flow”



Subflows

Overview

A Subflow is a group of Message Broker nodes that can be reused in multiple different Message Flows. A common Error Handling framework is a widely used example. In order for Subflows to be used in multiple different Message Flows, the Subflows must be decoupled from the specific Input and Output nodes of the parent Message Flows. This is done by using the logical *Input* and *Output* Nodes.

Subflow Creation

A Subflow creates a common flow of Message Broker nodes that can then be used in multiple other Message Flows. A Subflow is constructed in an identical manner to a normal Message Flow, but a Subflow begins with a logical input Node (Input) and may optionally have a logical output Node (Output). A Subflow can be saved into either a “.subflow” file or a “.msgflow” file type. This choice impacts how the Subflow and the Message Flows that use it are deployed.

File Type: “.subflow”

The “.subflow” file type was introduced in Message Broker v8.0. Subflows saved as “.subflow” files are deployed independently from any of the Message Flows that invoke them. That is to say that the “.subflow” file is deployed to the appropriate Execution Group(s). The Subflow can be explicitly deployed or can be contained within a Library and deployed with the Library. This means that changes to the Subflow can be deployed independently from any of the Message Flows that use the Subflow. Updates to the newly deployed Subflow will thus be effectively immediately upon its deployment.

Note: Subflows deployed in this manner may not contain either static Subflows (“.msgflow”) or User Defined Nodes that are saved in “.msgflow” files.

File Type: “.msgflow”

Subflows saved as “.msgflow” files are embedded inside any Message Flows that use them. They are therefore included in the “.msgflow” file of the parent Message Flow. To deploy any updates to the Subflow, the BAR file of the parent Message Flow must be regenerated and then deployed. This must be done for each Message Flow that uses the Subflow. This type of Subflow can be packaged as a User-Defined Node.

Promoting Properties

Promoted Properties are individual Node properties that are “promoted” to the Message Flow level. This allows the Message Flow to have a common property that may be used for multiple nodes. This is especially useful, for example, to define a Data Source for the nodes within the Message Flow. This is a common need in a Subflow, which may be deployed into many other Message Flows and environments.

Properties are promoted within the Toolkit. A right-click on an individual Node brings up the “Promote Property” dialog. Once a property has been promoted, it may no longer be changed within the Toolkit at the Node level. Promoted properties may be listed individually under the Message Flow or collected into “groups”. Groups and promoted properties can be renamed to simplify the management of the properties.



Subflow Versioning

Deployment Time

The *Passthrough* Node can be used to enable version identification of a Subflow. The use of the *Passthrough* Node allows a label to be added to the Subflow. This label can then be used to describe the Subflow version. The value for the label can be obtained in the following ways:

- By using the *mqsireadbar* command to display BAR file properties.
- By using the Message Broker Toolkit and examining the Message Flow properties.
- In runtime User Traces.

This label can be used within an automated software deployment system to determine which version of a Subflow should be deployed. Note that this method can be used with both “.subflow” and “.msgflow” files.

Runtime

It is possible for multiple versions of a Subflow to exist within a Message Broker at the same time. This is because the deployment scope of the Subflow is limited by:

- The BAR file that contains the statically linked (“.msgflow”) Subflow.
- The Execution Groups to which the BAR file is deployed.
- The Execution Groups to which dynamically linked (“.subflow”) Subflows are deployed.

If the *Passthrough* Node is used, the label produced by this node is visible in User Traces. This provides the ability to see in a trace which “version” of the Subflow was in use. Note that this assumes that the Label of the *Passthrough* Node is updated for each version!

Best Practices

- **Subflow File Types:** Save new Subflows as “.subflow” files. This is the IBM recommendation and allows Subflows to be updated and deployed independently of the Message Flows using them.
- **Use Libraries:** Save “.subflow” files in libraries. This facilitates their reuse and makes them easier to manage. All a parent Subflow needs to do is to reference the library.

References

- IBM – Integration Bus – Knowledge center (v9.0) - Subflows
[http://www-01.ibm.com/support/knowledgecenter/#!/SSMKHH_9.0.0/com.ibm.etools.mft.doc/ac00370 .htm](http://www-01.ibm.com/support/knowledgecenter/#!/SSMKHH_9.0.0/com.ibm.etools.mft.doc/ac00370.htm)
- IBM – Integration Bus – Knowledge center (v9.0) – Reusing Common Application Logic
http://www-01.ibm.com/support/knowledgecenter/#!/SSMKHH_9.0.0/com.ibm.scenarios.doc/subflows/topics/scnsubflows_01_11 .htm



- IBM – Integration Bus – Knowledge center (v9.0) – Promoted Properties
http://www-01.ibm.com/support/knowledgecenter/#!/SSMKHH_9.0.0/com.ibm.etools.mft.doc/ac00640.htm%23ac00640