

Building a parallel processing Web service with the service facade pattern and WebSphere Message Broker aggregation nodes

Fangtai Li
Zhang Hua

December 07, 2011

This article shows you how to build a parallel processing Web service using the service facade pattern and the aggregation mechanisms in WebSphere Message Broker. The aggregation nodes fan out the initial input message to a number of MQ requests that can be processed asynchronously, then fan in the multiple responses to a single consolidated response. The service facade pattern provides a single Web service interface decoupled from MQ services, enabling you to build a parallel synchronous Web service.

Introduction

This article describes the IBM® WebSphere® Message Broker aggregation nodes and the MQ request-response pattern, and then shows you how to use them to create message flows. To benefit from this article, you should be able to design a parallel processing Web service to improve the performance.

Aggregation nodes

Aggregation is the generation and fan-out of related requests that are derived from a single input message, and the fan-in of the corresponding replies to produce a single aggregated reply message. WebSphere Message Broker provides three message flow nodes that support aggregation:

- **AggregateControl node**
Marks the beginning of a fan-out of requests that are part of an aggregation and sends a control message that is used by the AggregateReply node to match the different requests that have been made.
- **AggregateRequest node**
Records the fact that the request messages have been sent, and collects information that helps the AggregateReply node construct the aggregated reply message.
- **AggregateReply node**

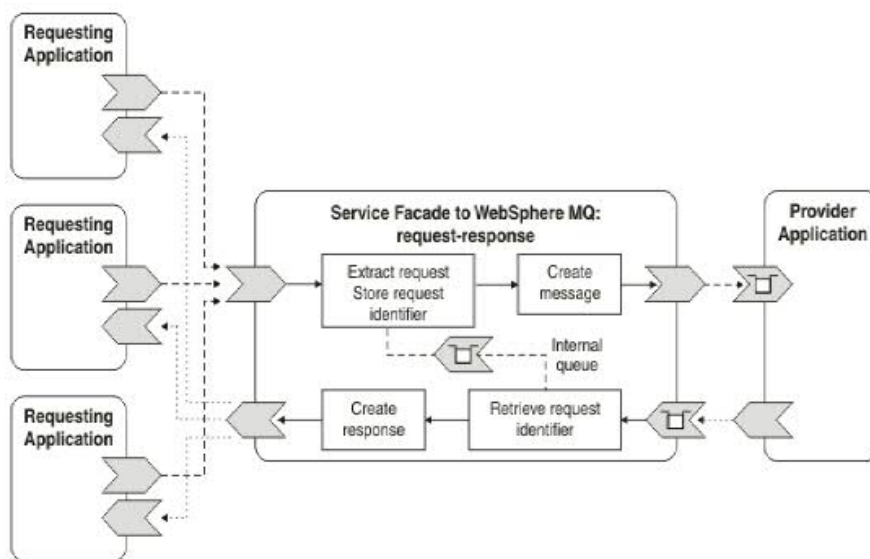
Marks the end of an aggregation fan-in, collects replies, and combines them into a single aggregated reply message.

MQ request-response pattern

You can use the service facade to the WebSphere MQ request-response pattern to provide a Web service facade to functions that are accessible only through WebSphere MQ. This pattern creates a bridge between the synchronous HTTP protocol, which is typically used with Web services, and existing applications with WebSphere MQ interfaces that cannot easily be upgraded.

The solution involves implementing a message flow that provides a service entry point. When a service request is received, the reply identifier is stored on an internal queue and the request is forwarded to the provider application. When the response is received from the provider application, the reply identifier is recovered and a Web service reply containing the provider response is returned to the requesting application:

MQ request-response pattern



Developing the message flows

This section shows you how to create a parallel synchronous Web service.

Creating the service facade flows

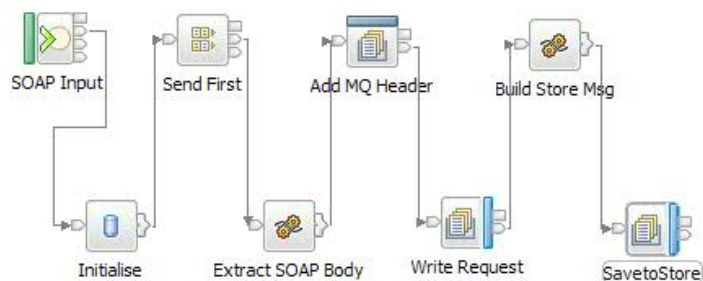
The service facade flows can be generated from the MQ request-reply pattern:

- **Facade Request flow**

When a SOAP request is received, the SOAP and HTTP headers are removed to leave the basic XML message, and then an MQMD header is added before the message is put into the aggregation fan-out input queue. Finally, the MQOutput node SaveToStore is used to store

the SOAP reply identifier and the related MQ message identifier. In the Facade Response flow, the related MQ message identifier is used to recover the SOAP reply identifier, so in subsequent flows, the MQ message identifier must not be lost. To track the identifier conveniently, hereafter this MQ message identifier is called the original message identifier.

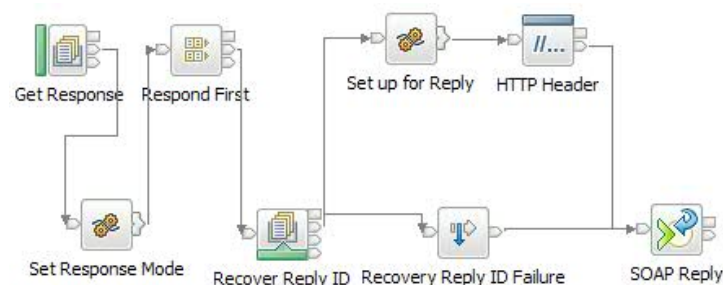
MQ request-response pattern SOAP request flow



• Facade Response flow

This flow reads the response message from the aggregate fan-in output queue, and then the SOAP reply identifier is recovered from MQ by the original message identifier. Next, an HTTP reply header is created and the MQMD header is removed, and finally, the response message is sent to the SOAP Reply node.

MQ request-response pattern SOAP response flow



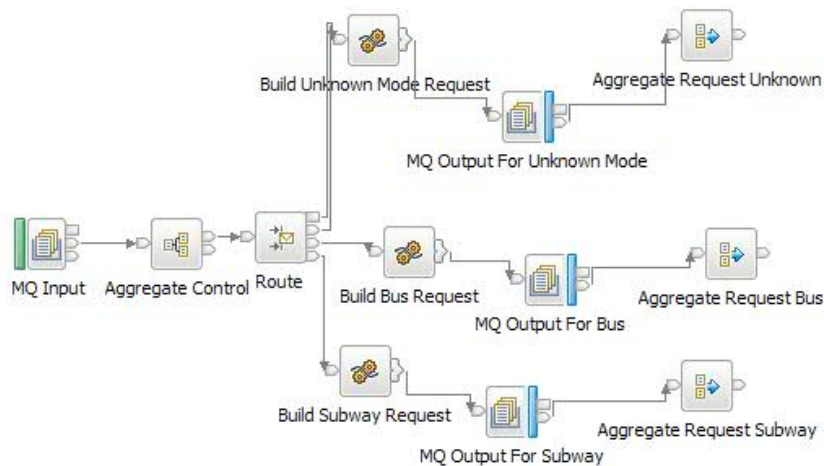
Creating the aggregation flows

To save time, start from the aggregation sample in the WebSphere Message Broker samples gallery: Select **WebSphere Message Broker Toolkit => Help => Samples and Tutorials**.

• Aggregate Fan-Out flow (serial mode)

This flow takes the incoming request message, generates two different request messages (bus and subway requests), sends them out to MQ for the processor to consume, and starts tracking the aggregation operation. The sub-request branches are processed in sequence: the bus request is put into MQ first, followed by the subway request. In this flow, the MQ messages (bus request and subway request) have the same MQMD message identifier as the original message identifier. In addition, the MQMD correlIds field is set to the original message identifier, because the new message identifiers will be generated after the request messages are put into the aggregate request queue.

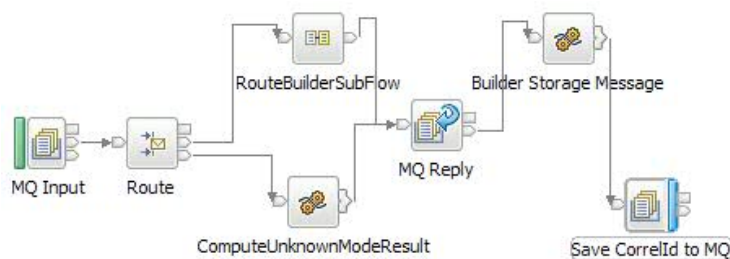
Aggregate fan-out flow



- **Aggregate Request-Reply flow (parallel mode)**

This flow is used to process the request messages from the aggregation operation. The messages from MQ Input nodes are processed in parallel, and therefore sufficient additional instance numbers should be configured for this flow. After the fan-out flow, a new MQMD MsgId is generated, but the MQMD CorrelId is the same as the original message identifier. By default (if no options are set), the MQReply node generates a new MsgId field in the reply message, and copies the MsgId from the input message to the CorrelId field in the reply message. If the original message identifier is not stored here, it will be lost in the next fan-in flow, so it is stored to MQ with the corresponding new MsgId generated by the MQReply node.

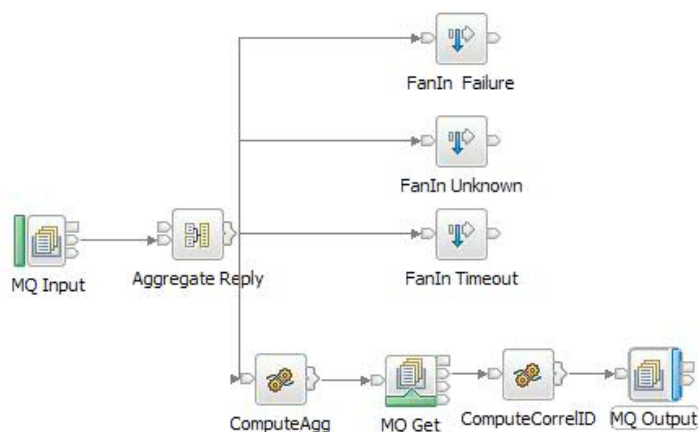
Aggregate Request-Reply flow



- **Aggregate Fan-In flow**

This flow receives all of the replies from the Request-Reply flow, and aggregates them into a single output message. The aggregate name for AggregateControl and AggregateReply nodes must have the same value. A Compute node is added to tweak the response message into a format in which it can be written out to a queue. Next, you need to recover the original message identifier, which is saved by the aggregate Request-Reply flow.

Aggregate fan-in flow



MQ nodes

This table lists all of the queues used in the above flows:

MQ node descriptions

Queue name	Flow / Node	Comments
PROVIDER	Facade Request / Write Request	Connection point between Facade Request and Fan-Out flows. The message from the SOAP request is sent to the Fan-Out flow by this queue.
	Aggregate Fan-Out / MQ Input	
STORE	Facade Request / SaveToStore	This queue is used to save and retrieve the SOAP reply ID, which is correlated to the original MQ message identifier.
	Facade Response / Recover Reply ID	
AGG_REQUEST	Aggregate Fan-Out / MQ Output	The divided sub-requests are put into this queue, and then the Request-Reply flow reads and processes them in parallel.
	Aggregate Request-Reply / MQ Input	
STORE1	Aggregate Request-Reply / Save CorrelId to MQ	The Request-Reply flow needs to save the original message identifier, which is correlated with the SOAP Reply ID. The Fan-In flow needs to recover the original message identifier and use it to update the output message identifier. Then the output message is sent to Facade Response flow.
	Aggregate Fan-In / MQ Get	
AGG_REPLY	Aggregate Fan-In / MQ Input	Reply-to queue of AGG_REQUEST. The results from the Request-Reply flow are put into this queue so the Fan-In flow can aggregate the result.
RESPONSE	Aggregate Fan-In / MQ Output	Connection point between Aggregate Fan-In and Facade Response. The aggregation result message is sent to the Facade Response flow to build the SOAP response message.
	Facade Response / Get Response	

Downloadable resources

Description	Name	Size
Code sample	wmb_parallel_aggregate_samples.zip	82 KB

Related topics

- **WebSphere Message Broker resources**

- [WebSphere Message Broker V7 information center](#)
A single Web portal to all WebSphere Message Broker V7 documentation, with conceptual, task, and reference information on installing, configuring, and using your WebSphere Message Broker environment.
- [WebSphere Message Broker developer resources page](#)
Technical resources to help you use WebSphere Message Broker for connectivity, universal data transformation, and enterprise-level integration of disparate services, applications, and platforms to power your SOA.
- [WebSphere Message Broker product page](#)
Product descriptions, product news, training information, support information, and more.
- [What's new in WebSphere Message Broker V7](#)
WebSphere Message Broker V7 provides universal connectivity with its ability to route and transform messages from anywhere to anywhere. Through its simple programming model and a powerful operational management interface, it makes complex application integration solutions much easier to develop, deploy, and maintain. This article describes the major enhancements in V7.
- [Download free trial version of WebSphere Message Broker V7](#)
WebSphere Message Broker V7 is an ESB built for universal connectivity and transformation in heterogeneous IT environments. It distributes information and data generated by business events in real time to people, applications, and devices throughout your extended enterprise and beyond.
- [WebSphere Message Broker documentation library](#)
WebSphere Message Broker specifications and manuals.
- [WebSphere Message Broker forum](#)
Get answers to your technical questions and share your expertise with other Message Broker users.
- [WebSphere Message Broker support page](#)
A searchable database of support problems and their solutions, plus downloads, fixes, and problem tracking.

- **WebSphere resources**

- [developerWorks WebSphere developer resources](#)
Technical information and resources for developers who use WebSphere products. developerWorks WebSphere provides product downloads, how-to information, support resources, and a free technical library of more than 2000 technical articles, tutorials, best practices, IBM Redbooks, and online product manuals.
- [developerWorks WebSphere application integration developer resources](#)
How-to articles, downloads, tutorials, education, product info, and other resources to help you build WebSphere application integration and business integration solutions.
- [developerWorks WebSphere business process management developer resources](#)
WebSphere BPM how-to articles, downloads, tutorials, education, product info, and other resources to help you model, assemble, deploy, and manage business processes.
- [Most popular WebSphere trial downloads](#)

No-charge trial downloads for key WebSphere products.

- [WebSphere forums](#)

Product-specific forums where you can get answers to your technical questions and share your expertise with other WebSphere users.

- [WebSphere on-demand demos](#)

Download and watch these self-running demos, and learn how WebSphere products and technologies can help your company respond to the rapidly changing and increasingly complex business environment.

- [developerWorks WebSphere weekly newsletter](#)

The developerWorks newsletter gives you the latest articles and information only on those topics that interest you. In addition to WebSphere, you can select from Java, Linux, Open source, Rational, SOA, Web services, and other topics. Subscribe now and design your custom mailing.

- [WebSphere-related books from IBM Press](#)

Convenient online ordering through Barnes & Noble.

- [WebSphere-related events](#)

Conferences, trade shows, Webcasts, and other events around the world of interest to WebSphere developers.

- **developerWorks resources**

- [Trial downloads for IBM software products](#)

No-charge trial downloads for selected IBM® DB2®, Lotus®, Rational®, Tivoli®, and WebSphere® products.

- [developerWorks blogs](#)

Join a conversation with developerWorks users and authors, and IBM editors and developers.

- [developerWorks cloud computing resources](#)

Access the IBM or Amazon EC2 cloud, test an IBM cloud computing product in a sandbox, see demos of cloud computing products and services, read cloud articles, and access other cloud resources.

- [developerWorks tech briefings](#)

Free technical sessions by IBM experts to accelerate your learning curve and help you succeed in your most challenging software projects. Sessions range from one-hour virtual briefings to half-day and full-day live sessions in cities worldwide.

- [developerWorks podcasts](#)

Listen to interesting and offbeat interviews and discussions with software innovators.

- [developerWorks on Twitter](#)

Check out recent Twitter messages and URLs.

- [IBM Education Assistant](#)

A collection of multimedia educational modules that will help you better understand IBM software products and use them more effectively to meet your business requirements.

© Copyright IBM Corporation 2011

(www.ibm.com/legal/copytrade.shtml)

[Trademarks](#)

(www.ibm.com/developerworks/ibm/trademarks/)