

WMB **AnyQmgrInputNode**

10 April 2006

Shravan K Kudikala
Staff Software Engineer
IBM
India

shravankk@in.ibm.com

Property of IBM

Take Note!

Before using this report be sure to read the general information under "Notices".

Fourth Edition, April 2006

This edition applies to Version 2.1 of AnyQmgrInputNode and to all subsequent releases and modifications unless otherwise indicated in new editions.

Third Edition, July 2005

This edition applies to Version 2.0 of AnyQmgrInputNode and to all subsequent releases and modifications unless otherwise indicated in new editions.

Second Edition, April 2005

This edition applies to Version 1.6 of AnyQmgrInputNode and to all subsequent releases and modifications unless otherwise indicated in new editions.

First Edition, June 2004

This edition applies to Version 1.0 of AnyQmgrInputNode and to all subsequent releases and modifications unless otherwise indicated in new editions.

© Copyright International Business Machines Corporation 2004.

All rights reserved.

Note to US Government Users -- Documentation related to restricted rights -- Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule contract with IBM Corp.

Table of Contents

<i>NOTICES</i>	<i>---</i>	<i>04</i>
<i>SUMMARY OF AMENDMENTS</i>	<i>---</i>	<i>05</i>
<i>PREFACE</i>	<i>---</i>	<i>06</i>
<i>OVERVIEW AND CONFIGURATION</i>	<i>---</i>	<i>07</i>
<i>INSTALLING THE PLUG-IN NODE</i>	<i>---</i>	<i>13</i>
<i>MODIFYING THE NODE</i>	<i>---</i>	<i>14</i>
<i>ACKNOWLEDGEMENTS</i>	<i>---</i>	<i>15</i>

Notices

The following paragraph does not apply in any country where such provisions are inconsistent with local law.

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore this statement may not apply to you.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates.

Any reference to an IBM licensed program or other IBM product in this publication is not intended to state or imply that only IBM's program or other product may be used. Any functionally equivalent program that does not infringe any of the intellectual property rights may be used instead of the IBM product.

Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, New York 10594, USA.

The information contained in this document has not be submitted to any formal IBM test and is distributed AS-IS. The use of the information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item has been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Trademarks and service marks

The following terms, used in this publication, are trademarks of the IBM Corporation in the United States or other countries or both:

- AIX
- IBM
- WebSphere MQ
- WebSphere Business Integration Message Broker (WBIMB)

The following terms are trademarks of other companies:

- Windows 2000/XP Microsoft Corporation

Summary of Amendments

Date	Changes	Version
22 June 2004	Initial release	1.0
02 April 2005	Second release	1.6
14 July 2005	Third release	2.0
10 April 2006	Fourth release	2.1

Changes Brought From V 1.0 to V2.0

V 1.0	-	Base code release with XML and BLOB domains.
V 1.1	-	XMLNS domain. Description : Now supports the XMLNS messages. This is done on a customer's request.
V 1.2	-	MRM domain Description : Now supports the MRM messages. If MRM domain is used, messageSet, messageType and messageFormat parameters must be set. This is provided on the request of an IBM internal customer.
V 1.3	-	Defect fixed with respect to 'firstParserClassName'. Description : This should/must be set only in the input node constructor. If not the first message would fail if the domain is not BLOB. The problem is corrected now.
V 1.4	-	Operating System logging (Event Log / Syslog). Description : Now the node has the ability to log messages directly into the Operating System log. On windows the logging is done to the Event Viewer and on Unix it is done into the Syslog / Userlog. On a production environment where node tracing is not encouraged, this feature helps users to find when there is a problem within the node. There is no specific option to enable or disable this feature. By default node initialization / deletion information and node processing failures are logged into the Operating System log. This is provided on the request off a customer who wanted a demo on the OS logging mechanism.
V 1.5	-	Supports toolkit version 5.1 Description : A separate plug-in representation folder is provided for V5.1 toolkit users. Directly copy the V5.1 toolkit related folder into the eclipse \ plugins directory in the same way as V5.0.
V 1.6	-	Multiple instances support. Description : Users can now use the 'multiple instances' option for the input node effectively. In the earlier versions, though 'multiple instances' option is set, only one thread is allocated / used internally. This problem is rectified and users can notice multiple threads when tracing option is set.
V 1.7	-	Defect Fixed. All messages received will be under syncpoint processing.
V 1.8	-	Message Transaction support. Description : Users now can process the message with or without transaction ability by choosing the new option available under the 'Advanced' option.
V1.9	-	Node attributes are shifter to common class NodeAttributes.java
V2.0	-	More efficient OS logging. Description : Users will now notice the complete exception stack trace in the operating system log whenever there is a failure in the message processing.
V2.1	-	WMB V6 Support up gradation

Preface

This Support Pac runs on WebSphere Business Integration Message Broker V 5.0 and V6.0. The plug-in node has been developed in Java and hence it can be used on any platform supported by the WBIMB V5/V6 brokers.

The AnyQmgrInputNode plug-in node was developed for the following reasons:

- ❖ To read messages from more than one queue (so that workload can be distributed at WebSphere MQ).
- ❖ To read messages from any queue manager (local or remote).
- ❖ To make this as an example to show how to develop custom java plug-in INPUT nodes in WBIMB.

The plug-in node was upgraded for the following reasons:

- ❖ To extend the domain range from XML, BLOB to XMLNS and MRM domains.
- ❖ To demonstrate how to log messages from the node into the operating system log.
- ❖ To extend the node for demonstrating how to support 'multiple instances' feature.
- ❖ To demonstrate how to handle the message transaction under syncpoint.

Overview and Configuration

Overview:

The AnyQmgrInputNode node is developed to overcome the restrictions of IBM primitive input node that reads messages only from one queue of the broker's queue manager.

The AnyQmgrInputNode provides a way to read messages from more than one queue in a sequence from any queue manager either local or remote, but doesn't provide all the functionalities of a built-in node like "Advanced and Validation" group attribute properties of built-in input node.

Under the following situations the AnyQmgrInputNode code can be modified to custom needs.

- a) If you want to write a better code to do Queue / Queue Manager Operations.
- b) If you want to read messages by spanning threads for each queue instead of reading one queue after the other in a cyclic way.
- c) If you want to read messages from other local/remote messaging sources like MQ Everyplace along with WMQ.
- d) If you want to read set of messages and form a single message before propagating it to the next node.
- e) If you want to use your own custom input node parser or other parsers of WBIMB in a customized way.
- f) If you want to do a better Exception handling mechanism.
- g) If you have any other needs or any other ways of making this node more reliable, robust and efficient.

Please read the documents/manuals provided with the product WBIMB to know more about input plug-in node development though the entire code is well documented for a better understanding.

The AnyQmgrInputNode node is developed to receive messages for a message flow from WebSphere MQ message queue defined on the specified local/remote queue manager using the WebSphere MQ Java application programming interfaces. This node currently reads 5 messages per queue before switching to the next queue. If in case the queue has less than or equal to 5 messages then control is passed after reading all the messages in that queue. But the number of messages to be read can be altered by setting appropriate value for "QUEUE_BATCH_SIZE" in "queueOps.java" file.

The node uses MQGET (WebSphere MQ API) to read a message from specified queue(s), and establish the processing environment for the message. The node is tested with multiple instances enabled for the message flow and found to work as desired. For reading messages from remote queues a listener must be running on the remote queue manager.

The AnyQmgrInputNode node handles messages in the following message domains only:

- ✓ BLOB

- ✓ XML
- ✓ XMLNS
- ✓ MRM

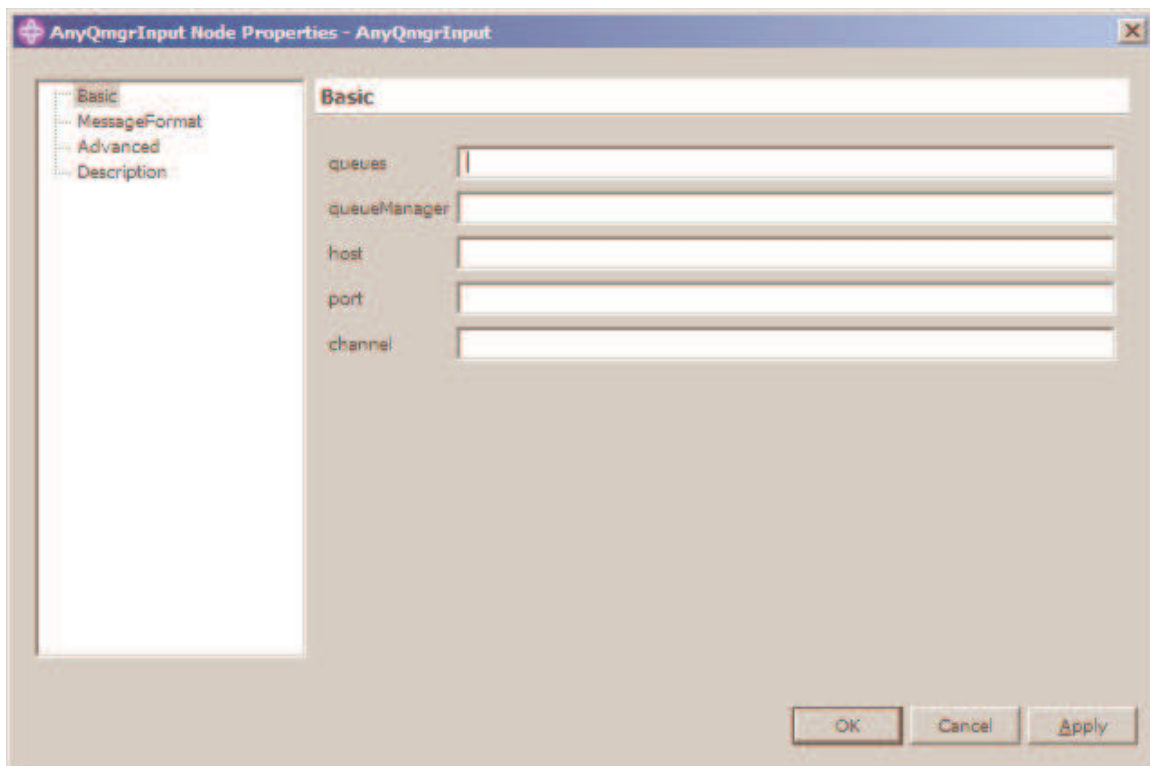
But you can modify the code to handle other domains similar to that of built-in input node, like JMSStream etc. But properties related to the domain need to be set in the code as described in the WBIMB product “plug-in development” help document.

Configuring the AnyQmgrInputNode node:

AnyQmgrInputNode version 2.1 configuration is different from its earlier version 1.6. The original v1.0 and last upgraded v1.6 shipped earlier, will not be provided separately through this latest version. The current document describes only the latest version. When you have put an instance of the AnyQmgrInputNode node into a message flow you can configure it by right-clicking the node and selecting the “Properties” option. The node's properties are divided into three groups, (i) Basic, (ii) MessageFormat and (iii) Advanced.

Configure the AnyQmgrInputNode node properties in the following way:

The Basic group properties are displayed in the properties dialog as shown in the below figure.

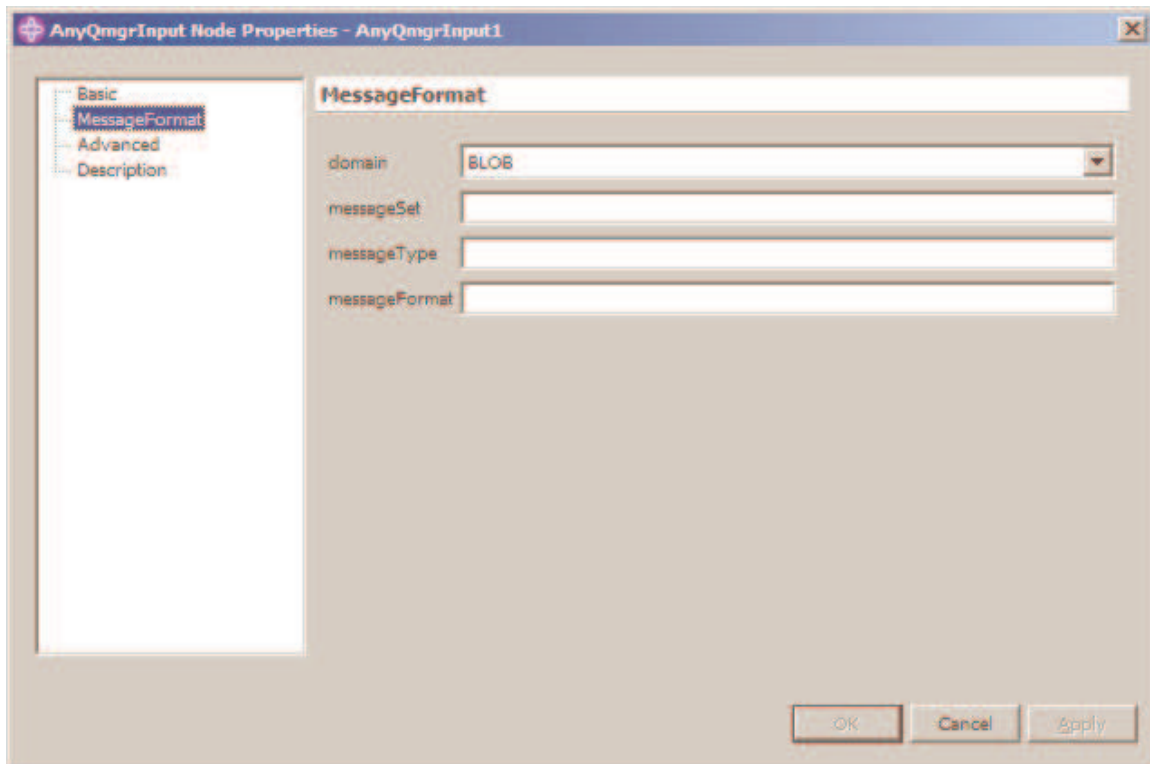


- ✓ **Property queues:** Enter the name of the queue(s) that the message flow would receive messages from. If multiple queue names are specified then all the queue names are to be separated by comma i.e., “,” would serve as a delimiter. It is assumed that all the specified WebSphere MQ queue(s) are already defined on the queue manager. This property is mandatory and at least one queue name must be specified. Please don't

specify the same queue name multiple times. The node doesn't take care of validating this and assumes all names are unique.

- ✓ **Property queueManager:** Enter the name of the queue manager. The node would look for all the listed queue(s) in this queue manager. If queue manager name is not specified then by default, the node would consider the property as broker's queue manager. This property is optional.
- ✓ **Property host:** Enter the hostname/IP address on which the queue manager is running. If the queue manager is not default and is not located on the box where the broker is running then this must be specified where the remote queue manager is running. This field is optional if the queue manager is the "default" (broker queue manager) or if it is residing on the same box as the broker.
- ✓ **Property port:** Enter the port number on which the remote queue manager is listening. This property is mandatory only if the queue manager is remote and *hostName* is specified.
- ✓ **Property channel:** Enter the channel name that needs to be used to connect to the queue manager. This can be any arbitrary name used conveniently but must comply with the channel name rules of WebSphere MQ. This property becomes mandatory along with *hostName* and *port* if the queue manager is remote.

The MessageFormat group properties are displayed in the properties dialog as shown in the below figure.

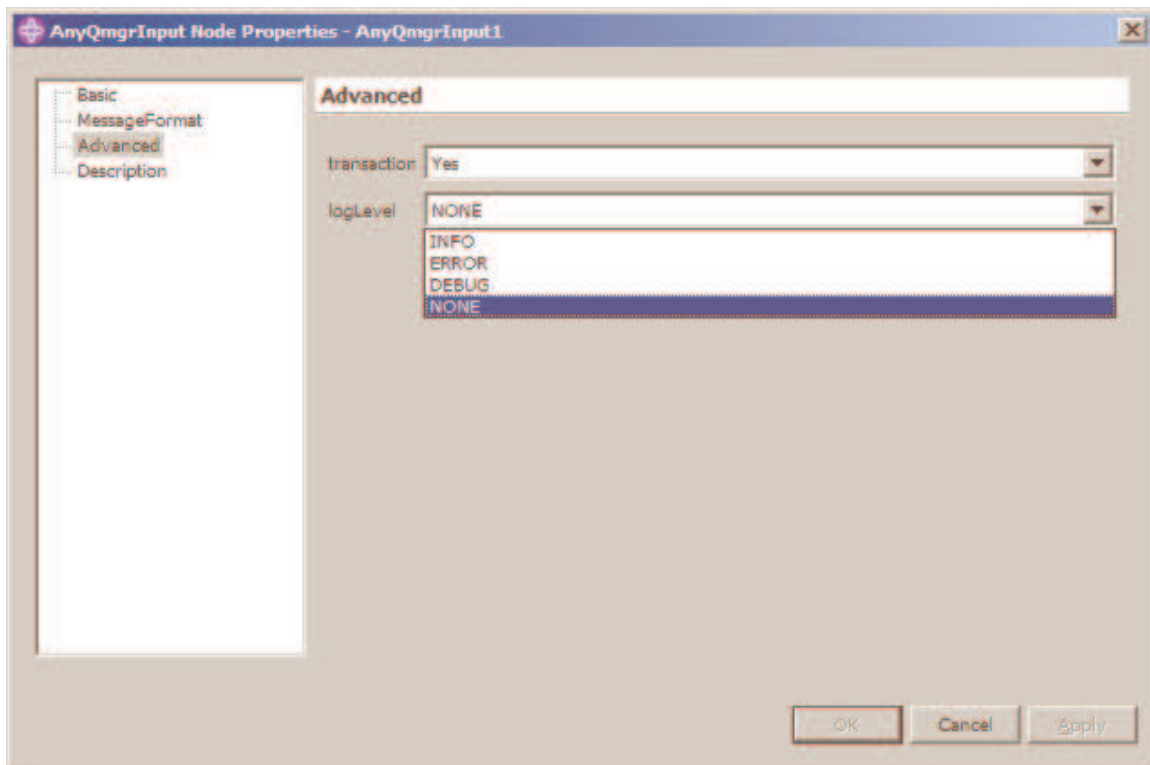


- ✓ **Property domain:** Choose the appropriate message domain depending on the incoming message. The default is BLOB. You can choose XML, XMLNS or MRM. If you are using

the MRM parser, then the other remaining properties (messageSet, messageType and messageProperty) are mandatory.

- ✓ *Property **messageSet***: When the domain is chosen as MRM parser then this property is mandatory. Specify the message set name of the incoming message.
- ✓ *Property **messageType***: When the domain is chosen as MRM parser then this property is mandatory. Specify the message type of the incoming message.
- ✓ *Property **messageFormat***: When the domain is chosen as MRM parser then this property is mandatory. Specify the message format of the incoming message. This is the physical format that you have defined for this message set. If you have used the default names for the physical formats, then it would be one among these, CWF1 / XML1 / TDS1. If you have specified different (non-default) names for any of these formats, then the same name should be specified here.

The Advanced group properties are displayed in the properties dialog as shown in the below figure.



- ✓ *Property **transaction***: Choose the transaction property to set the priority of the message transaction. The default value is 'Yes'. The transaction value set here will have the following significance.

Yes : The incoming message is received and processed under syncpoint. If the message processing fails under this condition then the message is backed-out. The backed-out message is placed on the back-out queue (if one is defined) when the back-out count reaches the input queues back-out threshold. If the back-out queue is not available then the message will be placed on the dead letter queue (if one is defined) of the queue manager. If the dead letter queue is not specified then it is placed on the default dead letter queue of the queue manager i.e., SYSTEM.DEAD.LETTER.QUEUE. If for some

reason the message couldn't be put on dead letter queue then a fatal exception is raised and the execution group will be restarted resulting in the loss of the message. Customer can modify this and implement it in the desired way.

No : The incoming message is not received and processed under syncpoint.

Automatic : The incoming message is received under syncpoint if it is marked persistent, otherwise it is not.

- ✓ **Property logLevel**: Choose the logLevel to debug the node. By default this is "NONE". It is encouraged to set NONE on production systems as the log refreshing mechanism isn't present in this node. The node appends the information into a single log file, irrespective of number of instances of the flows that use this node. The log file "AnyQmgrInputNodeLogData.txt" can be located in <WBIMB-install-dir>\bin on windows and on Unix machines it can be located in the directory from where the broker start command is issued. The loglevel value set here will have the following significance.

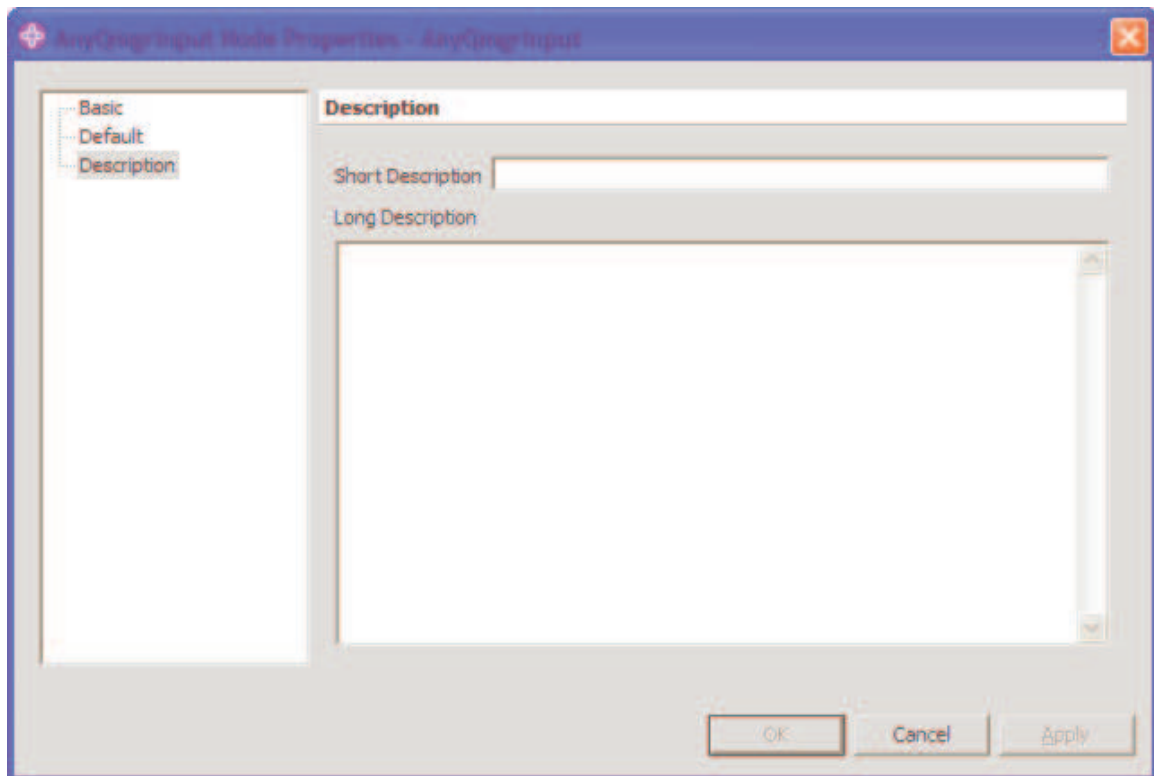
NONE: No logging is done.

ERROR: Log only if an error occurs in the node during message processing.

INFO : Log basic node operations and ERROR(s) if any.

DEBUG : Log very detailed information of the node, including every function in enters and exists along with ERROR(s).

The Description group properties are displayed in the properties dialog as shown in the below figure.



- ✓ **Properties Short/Long Description:** Select Description in the properties dialog navigator if you want to enter a short description, a long description, or both.

Click **Apply**. This makes the changes to the AnyQmgrInputNode node without closing the properties dialog. To apply the changes and close the properties dialog, click **OK**. If you press **Cancel**, the dialog is closed and all the changes that you have made to the properties are discarded.

Connecting the terminals:

AnyQmgrInputNode routes each message it retrieves successfully to the out terminal. If this fails, the message is routed to the failure terminal; you can connect nodes to this terminal to handle this condition. If you have not connected the failure terminal, the message is written to the Dead Letter Queue if one is defined.

Terminals and properties:

The AnyQmgrInputNode node terminals are described in the following **Table - 1**.

Terminal	Description
Failure	The output terminal to which the message is routed if an error occurred. If this is not connected then the message is dropped in Dead Letter Queue if one defined else the message is lost.
Out	The output terminal to which the message is routed if it is successfully retrieved from the WebSphere MQ queue.

Table – 1

Installing the plug-in node

❖ SupportPac contents

The downloaded **IA97.zip** file should be unzipped into a temporary directory

The following directories are created:

\AnyQmgrInputNode\Runtime\V5	-	Contains files meant for WBIMB V5 runtime.
\AnyQmgrInputNode\Runtime\V6	-	Contains files meant for WMB V6 runtime.
\AnyQmgrInputNode\Development\V5	-	Contains files meant for WMB V5 development.
\AnyQmgrInputNode\Development\V6	-	Contains files meant for WMB V6 development.
\AnyQmgrInputNode\IA97.pdf	-	V2.1 AnyQmgrInputNode manual

❖ Prerequisites

This Support Pac provides a plug-in node to be used with the IBM WebSphere Business Message Integrator Version 5. If any changes are to be made to the plug-in node, an appropriate Java Development Kit (JDK) V1.3 or above is required.

❖ Supported Platforms

This Support Pac has been developed and tested on Microsoft Windows 2000/XP and AIX V5.1 environments. But this plug-in node can be run on any environment that is supported by the WBIMB Broker V5/V6. The basic testing is done at WBIMB V5/V6 Gold, WBIMB V5 FP04 on WMQ 5.3 CSD07 and WMB V6 FP1 WMQ 5.3 FP12.

❖ Installing the plug-in node on a broker

Update the CLASSPATH environment variable on the WMB broker environment where the plug-in is going to be deployed by including the “**jplugin.jar, com.ibm.mq.jar**”, which are located at “<wbimb-install-dir>\classes”. Copy ‘**AnyQmgrInputNode.jar**’ file to <wbimb-install-dir>jplugin directory. This file is located under the extracted zip files. For WBIMB V5 runtime copy the file from “\AnyQmgrInputNode\Runtime\V5” directory and for WMB V6 runtime copy the file from “\AnyQmgrInputNode\Runtime\V6 “ directory.

You must stop and restart the broker to activate the new plug-in node.

❖ Integrating the plug-in node into the Windows WBIMB toolkit

if you are using the 5.1 toolkit, unzip the V5.1Toolkit.zip file located in “\AnyQmgrInputNode\Runtime\V5” directory into the <wbimb-toolkit-install-dir>\eclipse\plugins directory and restart the toolkit.

If you are using the 5.0 toolkit then unzip the file V5.0Toolkit.zip file located in “\AnyQmgrInputNode\Runtime\V5” directory into the <wbimb-toolkit-install-dir>\eclipse\plugins directory and restart the toolkit

If you are using the 6.0 toolkit then unzip the file V6.0Toolkit.zip file located in “\AnyQmgrInputNode\Runtime\V6” directory into the <wmb-toolkit-install-dir>\evtoolkit\ eclipse\plugins directory and restart the toolkit with “-clean” option as documented in the product manuals.

Modifying the node

Extracting and Compiling the plug-in node

How to get the source code:

In the node's earlier version 1.0, the source code and the toolkit representation are separately provided. In this version the complete plug-in development project is provided to the customer so that they can modify it easily and make further changes as they desire. Open the toolkit, use File-> Import -> Existing Project Into Workspace, choose the appropriate version (V5 or V6) project, (depending on WBIMB V5 or WMB V6) "AnyQmgrInputNode" and click ok. This would get the complete source code along with the plug-in toolkit representation files.

How to compile:

Open the java perspective and right click on the node project. Choose Export->Jar file. Click Next. Select only the package "com.ibm.Shra.Xternal.Plugins" from the "Select the resources to export". Mark the check box for the following options a) Export generated class files and resources b) Compress the contents to a jar file. For the option "Select the export Destination", use the browse button to point the jplugin directory located in <wmb-install-dir> along with the jar file name as 'AnyQmgrInputNode.jar'. It should be as "<wmb-install-dir>\jplugin\AnyQmgrInputNode.jar". Click Next -> Next. Now choose the radio buttons a) Generate the manifest file b) Seal some packages. Click Finish. This would compile and build the jar file into the '<wmb-install-dir>\jplugin\'. The options specified above while building the jar file are not mandatory and customer can choose their option as needed.

Representation of the node in the Toolkit

Please refer the product help / manuals inorder to know how to manipulate the node representation in the toolkit. All the files required for this are made available in the same project that is imported into the workspace.

Acknowledgements

Special thanks to all the customers who used the node, converted it into their required necessities and provided the feedback.

Thanks to all the customers who mailed me providing not only the feedback but also requested for new features to be brought into this node.

If the node users feel that a new feature into this node would be more useful to other customers too, then please send me a mail (shravankk@in.ibm.com), I shall try my best to incorporate it into the node.