Message Prioritization in Advanced Adapter Engine

Applies to:

SAP Net Weaver PI 7.1. For more information, visit the Service Bus-based Integration homepage

Summary

Often it becomes necessary to make sure that business critical messages are getting processed under high load condition. Through this guide Message Prioritization in Advanced adapter engine a newly introduced feature of PI 7.1 is explained that will help to avoid high priority messages getting stuck due to a backlog of other messages on the same node.

Author: Sunil Singh

Company: L&T Infotech

Created on: 25 March 2009

Author Bio

Sunil Singh is a SAP PI/XI Consultant in L&T InfoTech Ltd. His area of expertise includes SAP PI/XI, Java.

Table of Contents

Introduction

Prior to PI7.1 the message prioritization was available on Integration Engine which is restricted to messages with quality of service EO or EOIO. With PI 7.1 a provision is made to define the rule so as to assign the priority to a message having QoS EO, EOIO or BE.

Message Prioritization is designed to avoid high priority messages getting stuck due to a backlog of other messages on the same node. For message processing on the Adapter Engine we can define rules according to which messages with different priorities (low, normal, or high) are processed. It uses the attributes from the message header to execute the rules defined.

The Concept

Based on the rule defined in Message Prioritization of AE High/Medium/Low priority can be assigned to the messages per Sender/Receiver, Party/Component, Interface, and Namespace. If for one individual adapter type all the worker threads are occupied (at a given node), prioritization takes place.
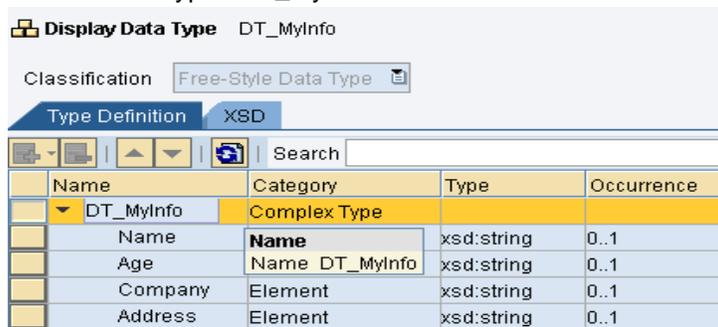
Business Scenario

In order to show the message prioritization taking place file to file scenario with some delay in message mapping is implemented so as to consume all the worker thread of an adapter.

Implementation

Design Steps

1. Create Data Type : DT_MyInfo

2. Create Message Type: **MT_MyInfo**
3. Create Outbound and Inbound Service Interface: **SI_MyInfo** and **SI_MyInfo_In**
4. Create a Message Mapping **MM_MyInfoToMyInfo** and write a UDF to provide some delay so that the queue threads get occupied at runtime and message prioritization takes place. (Note that delay is used just for demonstration purpose)
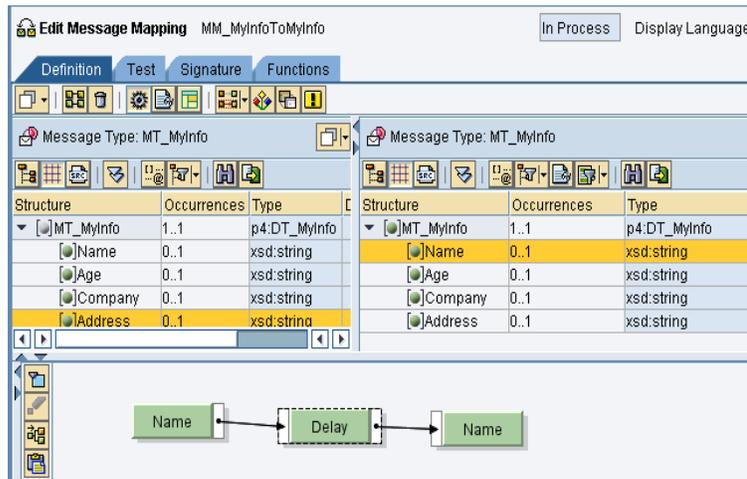
Below UDF function can be used to provide delay of 4000 ms.

try   {

    Thread.sleep(4000);

        }catch (Exception ie)

        {

    System.out.println(ie.getMessage());

        }

return (var1);

     5.   Create an operation mapping : OP_MyInfoToMyInfo

Configuration Steps

As mentioned above message prioritization takes place when all the worker threads are occupied. In order to reach this threshold I have sent the messages through three different sender components through file adapter.

    1.   Create three sender interfaces BC_File , BS_AAE, BS_AEE_1.
    2.   Create Sender and receiver file adapter for each sender component.
    3.   Create receiver determination, Interface Determination, Sender and Receiver Agreement for each.

RWB Steps

So as to message prioritization to take place for specific interface we have to define a role in RWB. To be able to define rules for a prioritized message processing, proceed as follows:

1. Open the Message Prioritization tab where we have to define rule for prioritization.

2.   Click on "create" and give the description of your rule and then Select Sender/Receiver, Party/Component, Interface, and Namespace for which rule needs to be configured. As shown below rules are configured where priority is defined based on Sender service.

	Description	Priority	Sender Partner	Sender Component	Receiver Partner	Receiver Component	Interface	Interface Namespace
⦿	New Rule	Low	*	BC_File	*	*	*	*
○	New Rule_low	Low	*	BS_AAE	*	BS_AAE	*	*
○	New Rule_Low1	High	*	BS_AEE_1	*	*	*	*

Line 1 / 3

3. Click on "Save" to save the rule we have configured.

As depicted above the messages sent by sender components BC_File and BS_AAE have low priority while the messages sent by sender component BS_AAE_1 have high priority.

File adapter is used to send and receive the messages by the above sender components.

Large number of messages is sent through above sender components until all the worker threads of sender file adapter get occupied.

To monitor the current status of file sender adapter go to RWB->Component Monitoring->Adapter Monitoring ->Engine Status

BcAdapterCall	Yes	0	0 / 0	10
BcAdapterRecv	Yes	0	0 / 0	10
BcAdapterRqst	Yes	0	0 / 0	10
BcAdapterSend	Yes	0	0 / 0	10
CIDXAdapterCall	Yes	0	0 / 0	10
CIDXAdapterRecv	Yes	0	0 / 0	10
CIDXAdapterRqst	Yes	0	0 / 0	10
CIDXAdapterSend	Yes	0	0 / 0	10
CIDX_http://sap.com/xi/XI/SystemCall	Yes	0	0 / 0	5
CIDX_http://sap.com/xi/XI/SystemRecv	Yes	0	0 / 0	5
CIDX_http://sap.com/xi/XI/SystemRqst	Yes	0	0 / 0	5
CIDX_http://sap.com/xi/XI/SystemSend	Yes	0	0 / 0	5
DispatchDisp	Yes	0	0 / 0	1
File_http://sap.com/xi/XI/SystemCall	Yes	0	0 / 0	5
File_http://sap.com/xi/XI/SystemRecv	Yes	0	0 / 0	5
File_http://sap.com/xi/XI/SystemRqst	Yes	0	0 / 0	5
File_http://sap.com/xi/XI/SystemSend	Yes	0	0 / 0	5

As shown in the figure above file sender adapter queue is assigned five threads through which the adapter can process the messages in parallel.

Large number of messages is sent through above sender components until all the worker threads of sender file adapter get occupied.

Run Time Observations

DispatchDisp	Yes	35	1 / 0	1
File_http://sap.com/xi/XI/SystemCall	Yes	0	0 / 0	5
File_http://sap.com/xi/XI/SystemRecv	Yes	0	0 / 0	5
File_http://sap.com/xi/XI/SystemRqst	Yes	0	0 / 0	5
File_http://sap.com/xi/XI/SystemSend	Yes	1	5 / 5	5
JDBC_http://sap.com/xi/XI/SystemCall	Yes	0	0 / 0	5

As shown in the figure above all the worker threads of sender file adapter are occupied, in such circumstances message prioritization taking place in Adapter engine can be observed

		Status	Start	End		Sender Component		Receiver Component	Interface	Type
	-	Delivering	23.03.2009 20:04:17	23.03.2009 20:05:08		BS_AEE_1		BS_AEE_1	MI_Receiver_out	XI Message
○	-	Delivering	23.03.2009 20:03:57	23.03.2009 20:04:50		BS_AEE_1		BS_AEE_1	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Delivering	23.03.2009 20:03:57	23.03.2009 20:04:50		BS_AEE_1		BS_AEE_1	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Successful	23.03.2009 20:03:57	23.03.2009 20:05:18		BS_AEE_1		BS_AEE_1	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Successful	23.03.2009 20:03:27	23.03.2009 20:05:08		BS_AEE_1		BS_AEE_1	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Successful	23.03.2009 20:03:17	23.03.2009 20:04:50		BS_AEE_1		BS_AEE_1	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	To Be Delivered	23.03.2009 20:03:07			BC_File		BC_File	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	To Be Delivered	23.03.2009 20:03:07			BS_AAE		BS_AAE	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	To Be Delivered	23.03.2009 20:03:07			BC_File		BC_File	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Successful	23.03.2009 20:02:37	23.03.2009 20:04:50		BS_AEE_1		BS_AEE_1	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Successful	23.03.2009 20:02:27	23.03.2009 20:04:12		BS_AEE_1		BS_AEE_1	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Successful	23.03.2009 20:02:17	23.03.2009 20:03:54		BS_AEE_1		BS_AEE_1	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	To Be Delivered	23.03.2009 20:02:17			BC_File		BC_File	urn:ltil:firstPI7Job MI_Receiver_out	XI Message

As shown in the figure above maximum amount of processing time is assigned to the message coming from BS_AAE_1 which is assigned high priority.

To cross verify the things we can change the prioritization as shown below wherein sender component BC_File is given high priority and other two are with low priority.

Prioritization Rules

☑ 3 of 3 rules saved

Create   Delete   Reset   Save

	Description	Priority	Sender Partner	Sender Component	Receiver Partner	Receiver Component	Interface	Interface Namespace
◉	New Rule	High ▾	*	BC_File	*	*	*	*
○	New Rule_low	Low ▾	*	BS_AAE	*	BS_AAE	*	*
○	New Rule_Low1	Low ▾	*	BS_AEE_1	*	*	*	*

Line 1 / 3

When messages were monitored in RWB->Message Monitoring it was observed that messages coming from BC_File were assigned maximum processing time as accepted.

		Status	Start	End		Sender Component		Receiver Component	Interface	Type
	-	Delivering	23.03.2009 20:13:17	23.03.2009 20:15:26		BC_File		BC_File	MI_Receiver_out	XI Message
○	-	Delivering	23.03.2009 20:13:17	23.03.2009 20:15:09		BC_File		BC_File	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	To Be Delivered	23.03.2009 20:12:57			BS_AAE		BS_AAE	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Delivering	23.03.2009 20:12:47	23.03.2009 20:15:09		BC_File		BC_File	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Delivering	23.03.2009 20:12:27	23.03.2009 20:14:41		BC_File		BC_File	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Successful	23.03.2009 20:12:27	23.03.2009 20:15:26		BC_File		BC_File	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Successful	23.03.2009 20:12:27	23.03.2009 20:15:09		BC_File		BC_File	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Successful	23.03.2009 20:11:57	23.03.2009 20:15:09		BC_File		BC_File	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Successful	23.03.2009 20:11:47	23.03.2009 20:15:09		BC_File		BC_File	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	To Be Delivered	23.03.2009 20:11:27			BS_AAE		BS_AAE	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Successful	23.03.2009 20:11:27	23.03.2009 20:14:41		BC_File		BC_File	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	Successful	23.03.2009 20:11:27	23.03.2009 20:14:30		BC_File		BC_File	urn:ltil:firstPI7Job MI_Receiver_out	XI Message
○	-	To Be Delivered	23.03.2009 20:11:27			BS_AEE_1		BS_AEE_1	urn:ltil:firstPI7Job MI_Receiver_out	XI Message

Conclusion

By configuring the rule for message prioritization on AE one can make sure that business critical data will not get blocked at high load condition.

Related Content.

https://www.sdn.sap.com/irj/scn/go/portal/prtroot/docs/library/uuid/70b8adc3-728c-2a10-7fad-d43f29074ef8

http://help.sap.com/saphelp_nwpi711/helpdata/en/48/cbb578cea80783e10000000a42189d/frameset.htm

SAP Note No. 1129604

For more information, visit the Service Bus-based Integration homepage

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.