# Using WebSphere Exits to Track Transactions Thru Composite Applications

By Bob Zerbe BMC Software

### Agenda

- Monitoring messages versus transactions
- Composite challenges
- WebSphere MQ API Exits
- WebSphere Message Broker API Exit
- WebSphere Application Server Approach
- Putting it to work

### **The Big Picture**



### **Traditional Approach**

- Servers
- Processes
- Applications
  - -Application Instrumentation
    - -App Code Reports Statistics
    - -Byte Code Instrumentation
  - -Application Server Monitoring

-App Environment (WAS, CICS, WPS, ...) Reports Statistics

-JMX, SMF, CEI

### **Traditional Approach**

- Strengths of Application Monitoring
  - Completely manageable by app/tech admin
  - No knowledge of other silos required
  - Data summarization limits impact
- Limits of Application Monitoring
  - One monitor per technology
  - Monitor puts hard limits on available information
  - Limited or no correlation of related components
  - Not granular enough
  - No knowledge of payload

## **Moving Beyond Silo Monitoring**

#### Individual Queue Monitoring – Not Enough



### **Service Delivery - Transactional**

#### **Delivering To SLA Performance**

- Did my message(s) get delivered?
- Are transactions processing within SLA rates?
- Did the transaction complete?
- Where is the problem?
- Where are my messages?
- Why didn't we know this before the customer did?
- What was the value of the transaction?

## **Corporate War Room**



### **Monitoring Flowing Data**



### **Monitoring Flowing Data**



### **Problem Management**

- Detect
- Isolate

### Diagnose

Avg RT = .75 sec Avail = 99.999% Avg RT = 2.35 sec Avail = 99.999%

Avg RT = 2.95 sec Avail = 99.900%



### **Problem Management**

#### • Detect:

Measure response time for critical transactions in each business service

#### Isolate:

Build aggregate metrics on response time (Average, Min, Max)

 Alert when there is a deviation by a specific service or transaction and identify where

### Diagnose:

- Elapsed time
- Use Transaction payload to provide better diagnostic information on individual transaction types

### **Collect And Store Data**

OrderNo	Web App	Msg Flow	BP Start	BP Order Entry	Mediation	SAP	Siebel	Mediation	BP Order Ship	Msg Flow	Ship App
1	1:23:12	1:23:13	1:23:15	1:23:15	1:23:16	1:23:19	1:23:19	1:23:27	1:23:29	1:23:30	1:23:31
2	1:26:33	1:26:34	1:26:35	1:26:36	1:26:38	1:26:39	1:26:41	1:26:44	1:26:46	1:26:47	1:26:48
3	1:26:41	1:26:41	1:26:42	1:26:43	1:26:44	1:26:45	1:26:45	1:26:46	1:26:47	1:26:48	1:26:49
4	1:27:11	1:27:13	1:27:14	1:27:16	1:27:18	1:27:19	1:27:19	1:27:22	1:27:24	1:27:25	1:27:27
5	1:29:29	1:29:31	1:29:32	1:29:32	1:29:34	1:29:36	1:29:35	1:29:41	1:29:59	1:30:01	1:30:02
6	1:31:50	1:31:52	1:31:53	1:31:54	1:31:55	1:31:56	1:31:56				
7	1:48:17	1:48:17	1:48:18	1:48:18	1:48:19	1:48:22	1:48:22	1:48:26	1:48:33	1:48:34	1:48:35
8	1:48:33	1:48:34	1:48:35	1:48:36	1:48:37	1:48:38	1:48:38	1:48:40	1:48:41	1:48:42	1:48:43
9	1:50:01	1:50:01	1:50:03	1:50:03	1:50:04	1:50:05	1:50:05	1:50:08	1:50:09	1:50:11	1:50:12

#### **How Do You Get There?**

### Use IBM supplied APIs\exits

- Why Use it?
- Configuring the Exit
- Caveats
- What can you get
- What happens when it runs?
- Considerations



- API Exit Uses
  - Beyond PCF...
  - Track individual messages through MQ
  - Track message latency
  - Identify various message sizes
  - Verify SLAs are being met
  - Interrogate\capture message content
    - Header info
    - Message body for BPM related work

#### Configuring the Exit

- Edit the WebSphere MQ qm.ini file
- -Add a stanza that includes:
  - Name of the API exit
  - Identifies the module and entry point of the API exit code to run
  - Optionally pass data with the exit
  - Identify the sequence of each exit in relation to other exits
- New stanza appears below
  - ApiExitLocal:
    - Sequence=300
    - Name=mypgm
    - Function=EntryPoint

Module=C:\Apps\BMTM\WebSphereMQ\_Runtime\exits\mypgm.dll

### WebSphere qm.ini for QMGR CSQ1

```
ExitPath:
 ExitsDefaultPath=C:\Apps\mqm\exits
 ExitsDefaultPath64=C:\Apps\mqm\exits64
InstanceData:
 Startup=1
 InstanceID=1351194071
#*
Log:
 LogPrimaryFiles=3
 LogSecondaryFiles=2
 LogFilePages=4096
 LogType=CIRCULAR
 LogBufferPages=0
 LogPath=C:\Apps\mqm\log\CSQ1
 LogWriteIntegrity=TripleWrite
```

\*#

Capitalware's MQ Technical Conference v2.0.1.4

### WebSphere qm.ini for QMGR CSQ1 cont'd.

Service:

- Name=AuthorizationService
- EntryPoints=14

ServiceComponent:

Service=AuthorizationService

Name=MQSeries.WindowsNT.auth.service

Module=amqzfu.dll

ComponentDataSize=0

#### ApiExitLocal:

Sequence=300

Name=mypgm

**Function=EntryPoint** 

Module=C:\Apps\BMTM\WebSphereMQ\_Runtime\exits\mypgm.dll

#### Caveats

- -Queue manager must be stopped then restarted (one time)
- -If queue manage fails to start, check program path and permissions
- -If multiple exits use sequence numbers
- Encryption and compression.
  - Messages that are encrypted must be decrypted before they reach the API to be processed
  - -To ensure that the exits are processed in the correct order,

- the *decryption API exit must have a larger sequence number* than the WebSphere MQ API exit.

-Compressed messages must be uncompressed before the WebSphere MQ API exit can read them.

- What does the exit provide?
  - Message output via put or put1 using a queue or topic
  - Message input via get or callback using a queue or topic.
  - Ability to see the message content and collect\act on it

#### Technology data available

Application Name Connection MQGMO\_BROWSE Message Size Queue Resolved Queue Resolved Queue Mgr. User Id Property Topic CICS/IMS ID Host Queue Manager CICS Task No. CICS Trans ID UOW

- •Use Message Parsers To Gather Info
  - Parse copy of messages in memory
  - Store what you need then discard copy
  - Useful for tracking messages from point to point or end to end in MQ
  - Duration of time on the queue prior to get
  - Where is my message now?
  - Useful if message content is important
     e.g.: account number and dollar amount\trade value

#### Exit Considerations

- Clustering: Configure exit for all QMGRs in cluster
- Message uniqueness: Message ID, Correlation ID, inside the body
- Shared queues versus dedicated queues
- Filter to collect message data on specific message types
- What queue managers and queues to apply this to
- Size of messages
- Volume of messages
- Where\how to store data
- Presentation interface for the data

### **Use Case: Transaction Rates & History**





### **Use Case: Transaction Rates & History**



### **Use Case: Message Size Tracking**

		T	ransactio	HOP1									
1							<b>*</b>						
23		X	<b>F</b> ()	PTD -	ø	10	-	X	C	Ø	0		
	414D	0	10:1	10:1	*		0	0	10:1	4			
	414D	0	10:1	10:1	-		0	0	10:1	1			
	414D	0	10:1	10:1	-		0	0	10:1	1			
	414D	0	10:1	10:1	1		0	0	10:1	1			
	414D	0	10:1	10:1	<b>*</b>		0	0	10:1	1			
	414D	0	10:1	10:1	1		0	0	10:1	1			
	414D	0	10:1	10:1	1		0	0	10:1	1			
	414D	0	10:1	10:1	×		0	0	10:1	4			
	414D	0	10:1	10:1	-		0	0	10:1	-			
	414D	0	10:1	10:1	1		0	0	10:1	1			
	414D	0	10:1	10:1	1		0	0	10:1	1			
	414D	0	10:1	10:1	-		0	0	10:1	1			
	414D	0	10:1	10:1	1		0	0	10:1	1			
	414D	0	10:1	10:1	1		0	0	10:1	1			
	414D	0	10:1	10:1	1		0	0	10:1	1			
	414D	0	10:1	10:1	1		0	0	10:1	1			

🖹 Transa	ction Payload	Data			_ 🗆 ×									
Txn Id:	144D512044455631202020202020202051C314072224EA0D													
Actvity:	HOP1													
Selection:														
	Name		Label	Value										
Applicatio	nName	0		WMQ_CHANNEL										
Message	Size	0		39										
Queue		0	0 TEST.TRN2											
Resolved	Queue	0		TEST.TRN2										

Transaction Payload Data													
Txn Id: 414D512D444556	414D512044455631202020202020202051C314072224E9ED												
ctvity: HOP1													
Selection:													
Name	Label	Value											
ApplicationName	0	WMQ_CHANNEL											
Message Size	0	39											
Queue	0	TEST.TRN1											
Resolved Queue	þ	TEST.TRN1											

### **Use Case: Where Did Message Go?**



### Use Case: Elapsed Time...How Long?



泉		)			ê	) 🤤			Ð																				
	1	Transaction USTOMN01 App							USTCMIN01 MO HUB NY							HUB UN1					GBTCMIN01_MQ								
87					1					1					1					?					?				
23	-	E	Q	10	=	Ξ	3	Ŷ	٩,	=	Ξ	۲	Q	٩,	-	Ξ	3	Ŷ	Q,	-	X	0	4	Q,		Ξ	0	Q	
1 10	MQ5	. 0	87		0	0	04:1	*		0	0	04:1	*		0	3	04:1	-		0	0		2						
10	MQ5.	. 0	- 27		0	0	04:1	1		0	0	04:1	1		0	3	04:1	1		0	203	04:1	. 🖌		0	0	04:1	1	
10	MQ5.	. 0	- 27		0	0	04:1	-		0	0	04:1	1		0	4	04:1	1		0	0		2						
10	MQ5.	. 0	87		0	0	04:1	1		0	0	04:1	1		0	3	04:1	1		0	220	04:1	. 🖌		0	-5	04:1	1	
10	MQ5.	. 315	1		0	0	04:1	1		0	0	04:1	1		0	3	04:1	-		0	217	04:1	. 🖌		0	-3	04:1	-	
10	MQ5.	. 337	1		0	0	04:1	1		0	0	04:1	1		0	3	04:1	1		0	218	04:1	. 🖌		0	-2	04:1	1	
1.23	MQ5.	. 313	1		0	0	04:1	1		0	0	04:1	1		0	3	04:1	~		0	216	04:1	. 🖌		0	-3	04:1	1	
10	MQ5.	. 329	1		0	0	04:1	1		0	0	04:1	1		0	3	04:1	1		0	228	04:1	. 🖌		0	-3	04:1	1	
10	MQ5.	. 0	27		0	0	04:1	1		0	0	04:1	1		0	4	04:1	-		0	0		2						
	MQ5.	. 0	87		0	0	04:1	1		0	0	04:1	1		0	4	04:1	1		0	202	04							
10	MQ5.	. 0	1																	0	0	04	Tran	sactio	n Payli	oad D	ata		
. 6	MQ5.	. 0	1																	0	0	04							
10	MQ5.	. 313	1		0	0	04:1	-		0	0	04:1	1		0	3	04:1	1		0	214	04	an take	MQS	_13488	62254	0120		
. 8	MQ5.	. 320	1		0	0	04:1	1		0	0	04:1	1		0	4	04:1	1		0	217	04	ALC: NO.						
10	MQ5.	. 322	1		0	0	04:1	1		0	0	04:1	1		0	3	04:1	~		0	213	04							
10	MQ5.	. 0	87		0	0	04:1	1		0	0	04:1	1		0	3	04:1	1		0	200	04		HUB I	CH				
1 10	MQ5.	. 0	2		0	0	04:1	~		0	0	04:1	1		0	3	04:1	~		0	0	- A	cowy:						

Name	Label	Value
Public2	þ	public data 2 Show This UC3
PublicI	0	public data 1 Show This UC3
YourSeq	0	pub1201208271117480000017181
Queue	0	INT/INT8/OH8/BMC/ZURJCH
Resolved Queue Manager	0	CHTCMB01
MySeq	0	pub1201208261207120000019678
Resolved Queue	0	SYSTEM. CLUSTER. TRANSMIT. QUEUE

- Uses
- Configuring the Exit
- Caveats
- What can you get
- What happens when it runs?
- Considerations



- Uses For Broker Exit
  - Track transactions through message flows
  - Track transactions from MQ to WMB and forward
  - Troubleshoot delays
  - Identify errors in messages and by value (content)
  - Part of composite application tracking
  - Available on Windows, AIX, Linux

- Configuring the Exit
  - Set up the Broker environment variables
  - Put the exit program in the target exit path
  - Run the Broker Command Console to issue commands
     mqsichangeflowuserexits BrokerName -e EGName
     f flowname -a pgmname
- Command to Verify the exit at EG Level:

mqsireportflowuserexits ESB -e Chain

BIP8854I: User Exits active for Execution Group 'Chain': BTMExtension.

BIP8071I: Successful command completion.

#### Use Broker Command Console (Windows)

- Accessories Administrative Tools BMC Middleware and Transaction Management BMC PATROL BMC ProactiveNet Server Extras and Upgrades FileZilla FTP Client IBM DB2 IBM Installation Manager IBM WebSphere Message Broker 8.0.0.0 Command Console .NET Programming APIs Java Programming APIs
  - IBM WebSphere Message Broker Toolkit

#### Environment Review

C:\Apps\WMB>mqsireportbroker ESB BIP8927I: Broker Name 'ESB' Install path = 'C:\Apps\WMB' Work path = 'C:\ProgramData\Application Data\IBM\MQSI' Broker UUID = '9da2bb2f-3fa8-4749-ae59-2d9cfa8a5ce8' Process id = '4820' Queue Manager = 'ESB' User lil path = '' User exit path = 'C:\Apps\BMTM\WMB\_Exit\_Runtime\_Toolkit'

```
Active user exits = "

LDAP principal = "

LDAP credentials = "

ICU converter path = "

Trusted (fastpath) Queue Manager application = 'false'
```

#### Environment Review continued

Active user exits = " LDAP principal = " LDAP credentials = " ICU converter path = " Trusted (fastpath) Queue Manager application = 'false' Configuration change timeout = '300' seconds Internal configuration timeout = '60' seconds Statistics major interval = '60' minutes Operation mode = 'advanced' Fixpack capability level = " (effective level 'unrestricted') Broker registry format = 'v8.0'Administration security = 'inactive' Multi-instance Broker = 'false' Shared Work Path = 'none' Start as WebSphere MQ Service = 'undefined' HTTP listener port = '7080'

BIP8071I: Successful command completion.

•What does the exit provide?

WebSphere Message Broker (WMB) Exit monitors the following events:

When a message is read into the input node of a flow
 When a message enters a WMB node through an input terminal

✓ When a message leaves a WMB node through an output terminaltopic.

**Additional capability** 

✓ Visibility into transaction level information

- validate transaction throughput, errors, unkowns
- Payload interrogation

What does the exit provide?

Visibility is at the wire level of the terminal NOT the node

Simple Message Flow With WMB Extension



Monitor the wires, not the node itself

Where Is The Transaction?



#### Caveats

- Message Broker must be down when configuring the exit (mqsistop brokername)
- Start Broker back up (mqsistart broker)
- Be aware of Broker behavior on failed messages
  - Message should roll all the way back thru the flow
- Use it tactically: don't try and follow\monitor every thing

Data Exposed



- Prior Diagram: 3 different types of data, (3 branches)
  - XFORM is transform
  - XFORM Canonical
    - standard format which uses common logic
    - SUB mail it out

There are many nodes in the flow not represented

Message Flow Depicting Success



### **WebSphere Application Server Method**

- Uses
- Configuring the Servlet filter
- What can you get



### **WebSphere Application Server Method**

- Uses A Servlet Filter
  - Servlet Filter communicates transaction information to A central source
  - Extracts information from HTTP requests\responses.
  - User configuration is required to determine which requests and responses to monitor, and which information should be published to Central source.
- Part of the Java Servlet Specification
  - Servlet Specification Defined by Sun
  - Part of Java Enterprise Edition
  - Filters in specification since V2.3 2000

### **WebSphere Application Server Method**

- Configuring the Servlet
  - Install the BTM JEE Service
  - Install the BTM MBean into the App Server
    - (Java EE Application Server or Servlet Container
  - Install the Servlet Filter into the App Server
  - Register the MBean
  - Declare the BTM Servlet Filter to the servlet container
    - via the web deployment descriptor (web.xml).
  - Restart the application server

# Putting it altogether

### **Enable Exits, Collect And Use The Data**



### **Using\Presenting The Data**



### **Using\Presenting The Data**





### **Using\Presenting The Data**



#### **Understanding The Data**



### **Understanding The Data**

	Average Completed		Completed Completed		Daily Daily		Daily	Failed	In	Maximum	Minimum	
	Elapsed	Transactions	Transactions	Transactions	Average	Maximum	Minimum	Transact	Progress	Elapsed	Elapsed	
	Time		Over	Under	Elapsed	Elapsed	Elapsed	ions	Transacti	Time	Time	
			Inresnola	Inresnola	Time	Time	Time		ons			
	Avg	Total	Total	Total	Total	Total	Total	Total	Last	Max	Min	
Dec 5, 2013 11:00 AM	2,069.	4,640.	1,434.	3,206.	24,852.	48,720.	12,528.	0.	45.	3,727.	1,067.	
Dec 5, 2013 12:00 PM	2,084.	4,660.	1,517.	3,143.	24,855.	48,720.	12,528.	0.	47.	3,453.	1,076.	
Dec 5, 2013 1:00 PM	2,056.	4,691.	1,382.	3,309.	24,853.	48,720.	12,528.	0.	8.	3,479.	1,066.	
Dec 5, 2013 2:00 PM	2,060.	4,685.	1,440.	3,245.	24,846.	48,720.	12,528.	0.	6.	3,557.	1,052.	
Dec 5, 2013 3:00 PM	2,063.	4,684.	1,441.	3,243.	24,840.	48,720.	12,528.	0.	8.	3,440.	1,059.	
Dec 5, 2013 4:00 PM	2,085.	4,658.	1,490.	3,168.	24,845.	49,256.	12,528.	0.	7.	4,127.	1,063.	
Dec 5, 2013 5:00 PM	2,068.	4,642.	1,415.	3,227.	24,852.	49,524.	12,528.	0.	45.	3,518.	1,045.	
Dec 5, 2013 6:00 PM	2,063.	4,686.	1,427.	3,259.	24,915.	41,079.	12,906.	0.	44.	3,465.	1,068.	
Dec 5, 2013 7:00 PM	2,063.	4,684.	1,380.	3,304.	24,768.	43,406.	12,716.	0.	8.	3,631.	1,048.	
Dec 5, 2013 8:00 PM	2,071.	4,669.	1,443.	3,226.	24,813.	44,972.	12,576.	0.	7.	3,831.	1,052.	
Dec 5, 2013 9:00 PM	2,048.	4,699.	1,343.	3,356.	24,755.	46,148.	12,536.	0.	6.	3,853.	1,040.	
Dec 5, 2013 10:00 PM	2,046.	4,702.	1,352.	3,350.	24,717.	46,236.	12,480.	0.	7.	3,567.	1,068.	
Dec 5, 2013 11:00 PM	2,047.	4,696.	1,339.	3,357.	24,682.	46,236.	12,480.	0.	7.	3,536.	1,044.	
Dec 6, 2013 12:00 AM	2,036.	4,673.	1,337.	3,336.	24,665.	46,236.	12,480.	0.	47.	3,710.	1,055.	
Dec 6, 2013 1:00 AM	2,046.	4,701.	1,310.	3,391.	24,636.	46,236.	12,480.	0.	7.	3,628.	1,064.	
Dec 6, 2013 2:00 AM	2,058.	4,646.	1,378.	3,268.	24,627.	46,236.	12,480.	0.	7.	3,626.	1,059.	
Dec 6, 2013 3:00 AM	2,068.	4,710.	1,429.	3,281.	24,648.	46,236.	12,480.	0.	6.	3,445.	1,069.	
Dec 6, 2013 4:00 AM	2,055.	4,695.	1,401.	3,294.	24,660.	46,705.	12,480.	0.	8.	3,920.	1,070.	
Dec 6, 2013 5:00 AM	2,074.	4,668.	1,470.	3,198.	24,673.	47,040.	12,480.	0.	6.	3,613.	1,061.	
Dec 6, 2013 6:00 AM	2,061.	4,641.	1,390.	3,251.	24,680.	47,040.	12,480.	0.	47.	3,499.	1,044.	
Dec 6, 2013 7:00 AM	2,055.	4,692.	1,367.	3,325.	24,684.	47,040.	12,480.	0.	6.	3,844.	1,066.	
Dec 6, 2013 8:00 AM	2,078.	4,652.	1,473.	3,179.	24,689.	47,040.	12,480.	0.	8.	3,490.	1,064.	
Dec 6, 2013 9:00 AM	2,068.	4,676.	1,436.	3,240.	24,700.	47,040.	12,480.	0.	6.	3,623.	1,052.	
Dec 6, 2013 10:00 AM	2,060.	393.	117.	276.	2,059.	3,920.	1,040.	0.	6.	3,412.	1,131.	
Avg	2,061.75	4,497.63	1,354.63	3,143.	23,804.75	45,051.08	12,051.25	0.		3,624.75	1,061.79	
Min	2,036.	393.	117.	276.	2,059.	3,920.	1,040.	0.		3,412.	1,040.	
Max	2,085.	4,710.	1,517.	3,391.	24,915.	49,524.	12,906.	0.		4,127.	1,131.	

### Summary

- IBM provides methods for accessing core technologies to solve problems
- Knowing how to enable exits is just the beginning
- Putting it all together and managing it is a challenge
- Vendors can provide various levels of access
- Common support:
  - MQ, WMB, DataPower, JMS, HTTP
  - All data is fed to the central service tier using a pub/sub architecture for correlation
  - Standard graphs, reports and views add value and insight by formalizing the data
  - Use Silo monitors in conjunction with transaction monitoring

### Summary

- Common use cases for Transaction Monitoring
  - Latency decomposition of composite applications: hop to hop or end to end
  - Service Level Monitoring and Trending
  - Failed Transaction Detection
  - Interrogate\capture payload for BPM related work
  - Prioritization of Alert Responses
- Benefits of exploiting these methods are considerable
  - End of war room sessions
  - Improved productivity
  - SLA compliance
  - Better throughput

## **Questions & Answers**

