



Advanced Monitoring Asset for IBM Integration Bus

*Technical documentation:
installation, configuration and usage
V1.0*

Patrick Marie

IBM Cloud Services consultant

September 2017

*Products needed: IBM Integration Bus 10.0, IBM MQ, IBM DB2, IBM WebSphere
Application Server or Liberty profile, HighCharts JS*

Environments: Linux, Windows

Versions

Version	Author	Date	Contents / Changes
V1.0	Patrick Marie, IBM	09/17	Initial version

Table of contents

Overview	5
Business flows meaning.....	5
Functionalities provided by the asset.....	5
Architecture of the asset.....	6
Installing and configuring the asset.....	9
Environments and products needed by the asset.....	9
Configuring IBM Integration Bus and MQ for monitoring.....	9
Enabling monitoring in the message flows	9
Activating monitoring for the flows in the integration server.....	14
Configuring the MQ components to gather monitoring events.....	15
Configuring the database.....	22
DB2.....	22
Oracle	23
Installing the asset modules on the application server	24
WebSphere Application Server	24
WebSphere Liberty Profile.....	35
Miscellaneous.....	38
Security	38
High availability.....	38
Supported browsers	38
Native language support.....	38
Using the GUI.....	39
Querying/listing business flow instances.....	39
Displaying statistics about the business flow instances	48
Statistics on throughputs by periods of time	49
Statistics about throughputs by nodes/servers or by business flow types	51
Statistics about processing times.....	52
Appendix.....	55
Material provided.....	55

Overview

The main purpose of this asset is to enhance the functionality of IBM Integration Bus named “Business Transaction Monitoring”. In other words, this application is designed to provide advanced monitoring functionalities for the business flows (or business transactions) of IBM Integration Bus, and also some statistics.

Business flows meaning

What is named “business flows” in the asset and in this documentation is the same as “business transactions” in the IBM Integration Bus documentation. For me, the word “transaction” has another meaning and I found “business flow” more suitable.

To be more explicit, a business flow is a series of actions which are related to the processing of a common set of data by IBM Integration Bus. It may correspond to an IBM Integration Bus message flow instance or it may encompass several IBM Integration Bus message flow instances participating in the processing of the same data in sequence. It could even involve actions performed outside of IBM Integration Bus, if they also participate in the same processing of data.

Functionalities provided by the asset

The main functionalities provided by this application are the following:

- Gather data about the business flow instances executed by IBM Integration Bus from the monitoring events generated by the IBM Integration Bus flows as MQ messages and store them into a database.
- Provide a Web user interface enabling administrators to query, display and analyse these data.

This GUI offers the following functionalities:

- Query the business flow data stored in the database with filtering them using various criteria (period of time, integration node, integration server, type of business flow, business data) and display the matching instances in a table.

Advanced monitoring of IBM Integration Bus business flows

See all business flows Start: 2017/01/19 00:00:00 (CET) Business data (name, value): Max number of business flows per page: 7 Logout

See all failed business flows End: 2017/08/06 15:48:01 (CEST) ☒ Now

Monitor by integration nodes/servers

Monitor by business flow types

All business flows for the period from **January 19, 2017 12:00:00 AM CET to August 6, 2017 3:48:01 PM CEST** (from 15 to 21 of 10010):

Limit to failed flows Order: select a sort method... Statistics: select type and display...

ID	Status	Start time	End time	Business flow type	Node / Server	Business data
414d512049423130514d47525f537475bfe1582176511c	Completed	2017-02-20 09:34:44.776	2017-02-20 09:34:44.776	Small Business Appli	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain
414d512049423130514d47525f537475bfe1582176511e	Failed	2017-02-20 09:34:44.809	2017-02-20 09:34:44.811	Small Business Appli	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain
414d512049423130514d47525f537475bfe15821765120	Completed	2017-02-20 09:34:44.844	2017-02-20 09:34:44.853	Retail Order	Start: IB10NODE_Stud / default End: IB10NODE_Stud / default	orderId: 2 customerNumber: 5555 itemNumber: 1234-6789-1212 customerName: ChocolateFan requestType: orderItem customerName: CoffeeFan itemName: ColombianSupreme
414d512049423130514d47525f537475bfe15821765122	Completed	2017-02-20 09:34:44.883	2017-02-20 09:34:44.888	Retail Order	Start: IB10NODE_Stud / default End: IB10NODE_Stud / default	orderId: 2 customerNumber: 5670 itemNumber: 3529-9999-1937 customerName: TeaFan requestType: orderItem customerName: CeylonBlackTea
414d512049423130514d47525f537475bfe15821765124	Completed	2017-02-20 09:34:44.918	2017-02-20 09:34:44.921	Retail Order	Start: IB10NODE_Stud2 / default End: IB10NODE_Stud2 / default	orderId: 3 customerNumber: 4785 itemNumber: 9876-5342-3847 customerName: TeaFan requestType: orderItem customerName: CeylonBlackTea
414d512049423130514d47525f537475bfe15821765126	Completed	2017-02-20 09:34:44.95	2017-02-20 09:35:02.001	Multi Broker Transaction	Start: IB10NODE_Stud / default Other: IB10NODE_Stud / default2 End: IB10NODE_Stud2 / default End: IB10NODE_Stud2 / default2	orderId: 2 customerNumber: 5555 itemNumber: 1234-6789-1212 customerName: ChocolateFan requestType: orderItem customerName: Chocolate chip cookies

Go to preceding Go to beginning Go to line Go to next Go to end Home

- Display statistics as bar charts about the instances retrieved by a query.

Various statistics are offered:

- About throughputs by periods of time.
- About throughputs by nodes/servers and by types of business flow.
- About processing times by periods of time.

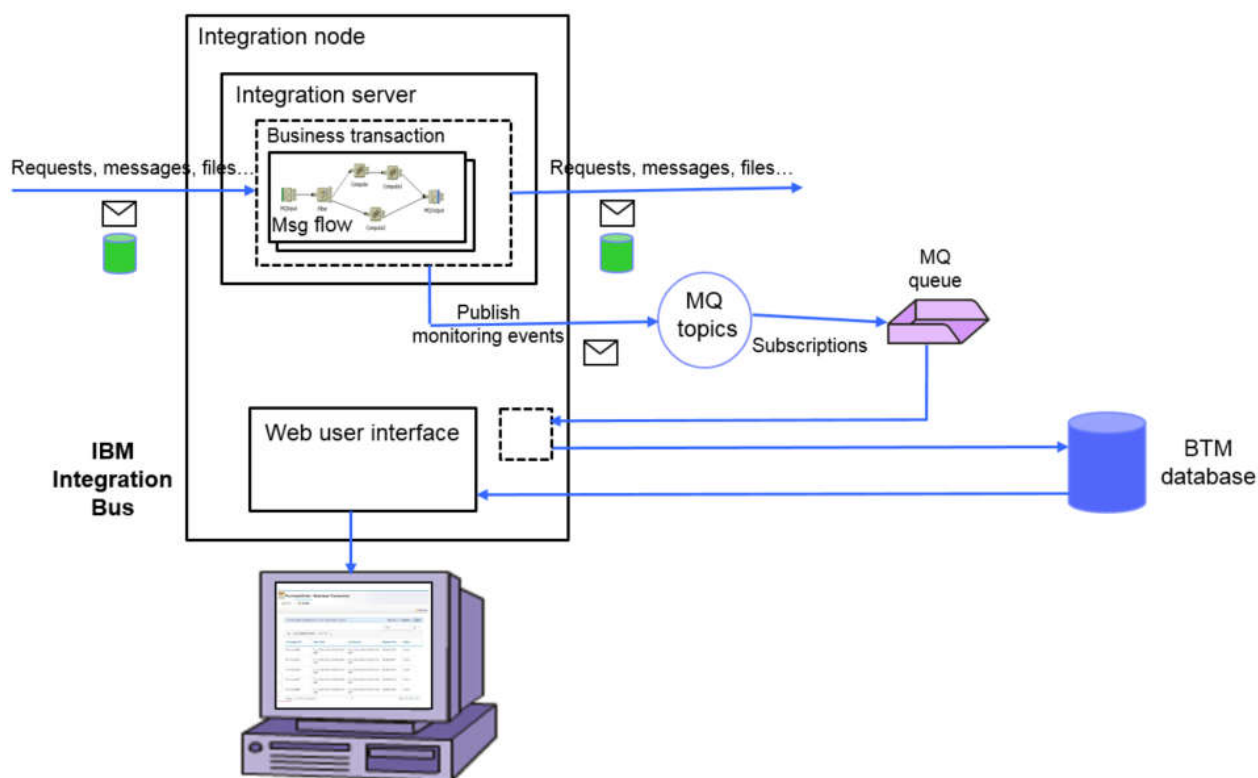


Moreover:

- Monitoring events received from several integration nodes and servers can be mixed in the same database and queried, viewed and analysed together from a single GUI.
- There is no “monitoring model” describing how the events received by the asset must be used. In other words, the monitoring model is generic and implicit and the monitoring events contain all the information needed to use them for monitoring. The consequence of this feature is that it is very fast and easy to have new flows monitored: as soon as they are deployed and ready to publish monitoring events, they can be monitored by the Advanced Monitoring Asset.

Architecture of the asset

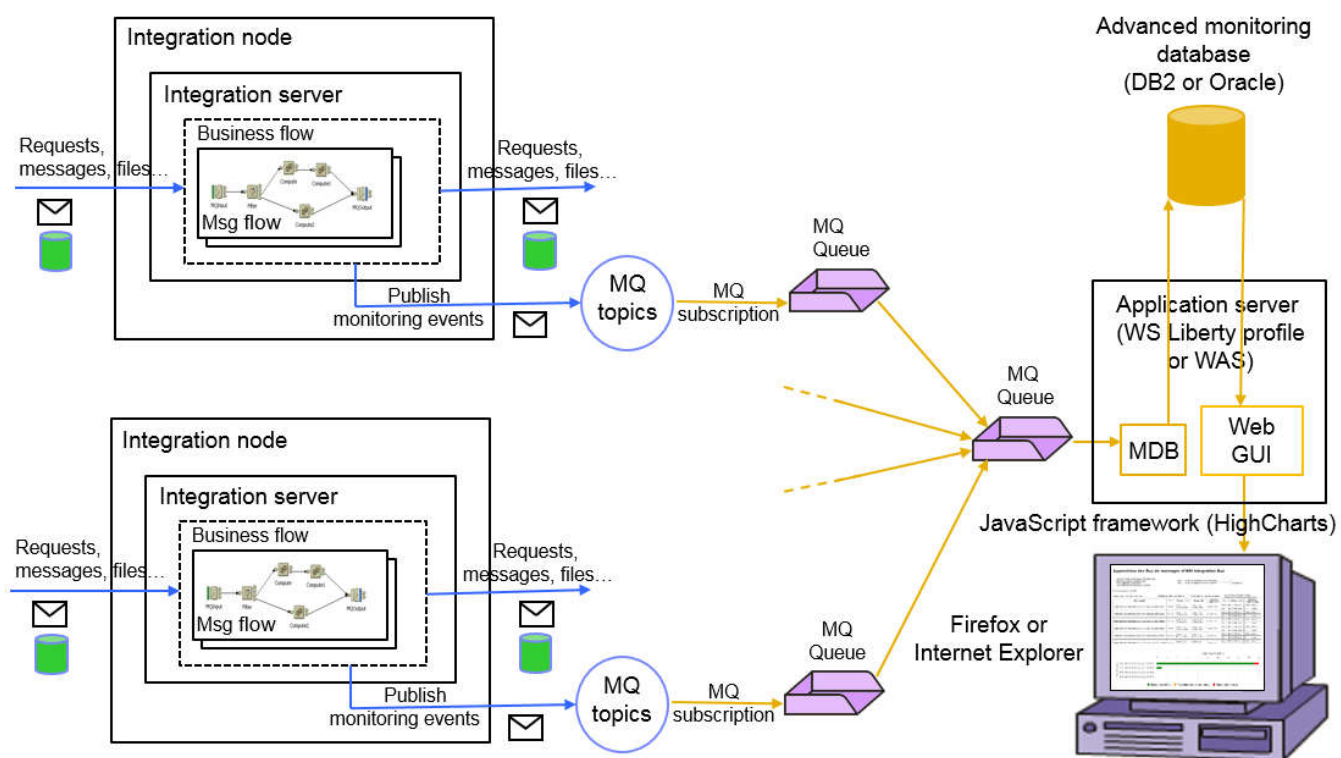
The architecture is derived from the architecture of the native “Business Transaction Monitoring” functionality of IBM Integration Bus, represented here under:



The two architectures share a common basis, which is the mechanism used by IBM Integration Bus to publish monitoring events to MQ topics, but they differ in the way how these events are used: the Advanced Monitoring Asset subscribes to a different (more generic) topic, stores data into database tables with a different structure and the GUI is also different.

The asset also adds the capability to gather events coming from several different integration nodes and servers.

The architecture of the Advanced Monitoring Asset is the following:



Installing and configuring the asset

Environments and products needed by the asset

The following environments are supported by the asset:

- A version of Windows 64-bit where the products here under are supported.
- Or a version of Linux 64-bit where the products here under are supported.

And the following products are needed:

- IBM Integration Bus v10.0.
- IBM MQ v7.1 or higher.
- IBM DB2 v9.0 or higher.
- An application server using Java 1.7 or higher:
 - WebSphere Liberty Profile v 8.5 or higher.
 - Or WebSphere Application Server v8.5 or higher.
- HighCharts JS:

HighCharts JS is a JavaScript framework designed to draw charts in Web applications. It does not need to be installed: some of its files are included in the asset. However, it belongs to the Highsoft Company and, in order to be allowed to use these files, a HighCharts JS license is included in the asset in OEM.

Configuring IBM Integration Bus and MQ for monitoring

The generation of the monitoring events is based on the capability of IBM Integration Bus to emit events for monitoring when message flows are executed. This capability is described at:

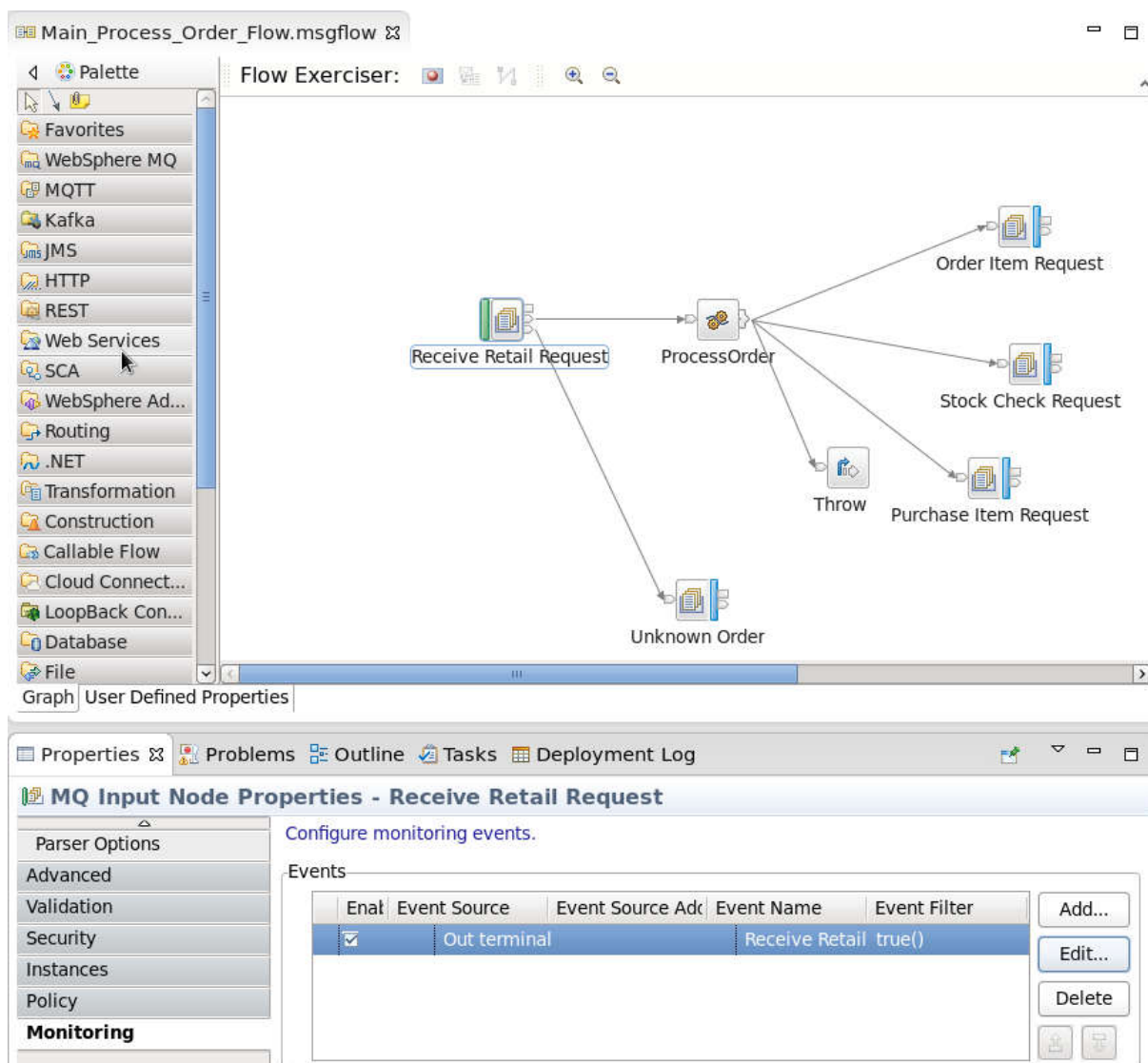
https://www.ibm.com/support/knowledgecenter/en/SSMKHH_10.0.0/com.ibm.etools.mft.doc/ac37860.htm. This foundation layer is also shared with the Business Transaction Monitoring functionality of IBM Integration Bus. It requires IBM MQ to be installed and a queue manager to be associated with the integration node whose flows are to be monitored.

Enabling monitoring in the message flows

This is done as described at:

https://www.ibm.com/support/knowledgecenter/en/SSMKHH_10.0.0/com.ibm.etools.mft.doc/ac37850.htm, with some specifics which will be explained.

To access the monitoring properties, open the message flow with the editor in the Toolkit, select the node and retrieve the “Monitor” tab in the Properties view:



Then click “Add” to add an event and define its properties:

Edit event

Basic | Correlation | Transaction

Event Source
Select the source of the event.
Out terminal

Event Source Address
The integration node identifies an event source using an event source address. Use this value when you enable and disable event sources using runtime commands.
Receive Retail Request.terminal.out

Event Name
Provide the name by which events emitted from this source are to be known. Specify either a literal name, or the location of a character field in the message tree or elsewhere in the message assembly.
☒ Literal Retail Order#S#Retail request received
☐ Data location Edit...

Event Filter
Provide an expression to control whether the event is emitted. The expression must evaluate to true or false, and can reference fields in the message tree or elsewhere in the message assembly.
If you do not specify a value, the value true() is used.
true() Edit...

Event Payload
Most events need to contain data taken from fields in the message tree or from elsewhere in the message assembly. Data taken from simple fields or complex fields appears in the event in XML character format. An event can also contain bitstream data, which appears in the event as hexadecimal bytes.

Data location	
\$Root/XMLNSC/btm_retail/orderId	Add...
\$Root/XMLNSC/btm_retail/requestType	Edit...
\$Root/XMLNSC/btm_retail/customerNumber	Delete
\$Root/XMLNSC/btm_retail/customerName	

☐ Include bitstream data in payload

Content: Encoding:

Cancel OK

The way to define the event properties is identical to the way you would do it for the “Business Transaction Monitoring” native functionality of IBM Integration Bus. The only specifics of the Advanced Monitor application are described here under:

- For a common flow instance, it is possible to correlate together events coming from several different integration nodes and servers.
- It is possible to have several start events and several end or failed events for a business flow instance.
- The event name is also used to hold the business flow type and the status (in addition to the event name).

Edit event

Basic Correlation Transaction

Event Source
Select the source of the event.
Out terminal

Event Source Address
The integration node identifies an event source using an event source address. Use this value when you enable and disable event sources using runtime commands.
Receive Retail Request.terminal.out

Event Name
Provide the name by which events emitted from this source are to be known. Specify either a literal name, or the location of a character field in the message tree or elsewhere in the message assembly.
☒ Literal Retail Order#5#Retail request received
☐ Data location Edit...

Event Filter
Provide an expression to control whether the event is emitted. The expression must evaluate to true or false, and can reference fields in the message tree or elsewhere in the message assembly.
If you do not specify a value, the value true() is used.
true() Edit...

Event Payload
Most events need to contain data taken from fields in the message tree or from elsewhere in the message assembly. Data taken from simple fields or complex fields appears in the event in XML character format. An event can also contain bitstream data, which appears in the event as hexadecimal bytes.

Data location Add... Edit... Delete

\$Root/XMLNSC/btm_retail/orderId
\$Root/XMLNSC/btm_retail/requestType
\$Root/XMLNSC/btm_retail/customerNumber
\$Root/XMLNSC/btm_retail/customerName

☐ Include bitstream data in payload

Content Encoding

Cancel OK

More specifically, the value of that field can contain several parts separated by '#' characters.

- When the field contains 3 parts (that is 2 '#'):
 - The first part is the business flow type.
 - The second part is the status ("S" for "Started", "E" for "Ended", "F" for "Failed" and "P" or any other value for "In process").
 - The third part is the event name.
- When the field contains 2 parts (that is one '#'):
 - The first part is the status ("S" for "Started", "E" for "Ended", "F" for "Failed" and "P" or any other value for "In process").
 - The second part is the event name.
- When the field contains just one part (that is no '#'):
 - The single part is the event name.

It is not needed for all events to include a business flow type and a status:

- When the flow type is not included, it is obtained from another event participating in the same business flow instance (at least one of the events must provide it and it is recommended to provide it in the start event). When several events belonging to a common business flow instance provide contradictory information about the business flow type, the value provided by the last start event takes precedence.

- When the status is not included, “In process” is assumed.
- The events are correlated using the “global transaction correlator” setting only (the other correlators are not taken into account).

Edit event

Basic Correlation Transaction

Event Correlation
 A monitoring application uses event correlators to match events emitted by the same, or related, business transactions. A local transaction correlator links the events emitted by a single invocation of a message flow. A parent transaction correlator links the events from a message flow to a parent message flow or an external application. A global transaction correlator links events from a message flow to one or more related message flows or external applications. An event must contain a local transaction correlator, but need not contain a parent transaction correlator or global transaction correlator.

Local transaction correlator:
☒ Automatic ☐ Specify location of correlator

Description:
 The local correlator used by the most recent event for this message flow invocation will be used. If no local correlator exists yet, a new unique value will be generated.

Parent transaction correlator:
☒ Automatic ☐ Specify location of correlator

Description:
 The parent correlator used by the most recent event for this message flow invocation will be used. If no correlator exists yet, no parent correlator will be used.

Global transaction correlator:
☐ Automatic ☒ Specify location of correlator

Description:
 The global correlator will be read from the specified location in the message tree. Ensure the specified location contains a suitable global correlator value.

\$Root/MQMD/MsgId Edit...

Cancel OK

- Data taken from the fields of the processed message can be included in the event, but this is not mandatory. When an event does include data, these data are updated in the database for the business flow instance and they replace entirely the preceding set of data (provided by a previous event), if any. When an event does not include data, the preceding set of data recorded for the business flow instance, if any, is left unchanged.

Edit event

Basic Correlation Transaction

Event Source
Select the source of the event.
Out terminal

Event Source Address
The integration node identifies an event source using an event source address. Use this value when you enable and disable event sources using runtime commands.
Receive Retail Request.terminal.out

Event Name
Provide the name by which events emitted from this source are to be known. Specify either a literal name, or the location of a character field in the message tree or elsewhere in the message assembly.
☒ Literal Retail Order#5#Retail request received
☐ Data location Edit...

Event Filter
Provide an expression to control whether the event is emitted. The expression must evaluate to true or false, and can reference fields in the message tree or elsewhere in the message assembly.
If you do not specify a value, the value true() is used.
true() Edit...

Event Payload
Most events need to contain data taken from fields in the message tree or from elsewhere in the message assembly. Data taken from simple fields or complex fields appears in the event in XML character format. An event can also contain bitstream data, which appears in the event as hexadecimal bytes.

Data location

- \$Root/XMLNSC/btm_retail/orderid
- \$Root/XMLNSC/btm_retail/requestType
- \$Root/XMLNSC/btm_retail/customerNumber
- \$Root/XMLNSC/btm_retail/customerName

☐ Include bitstream data in payload

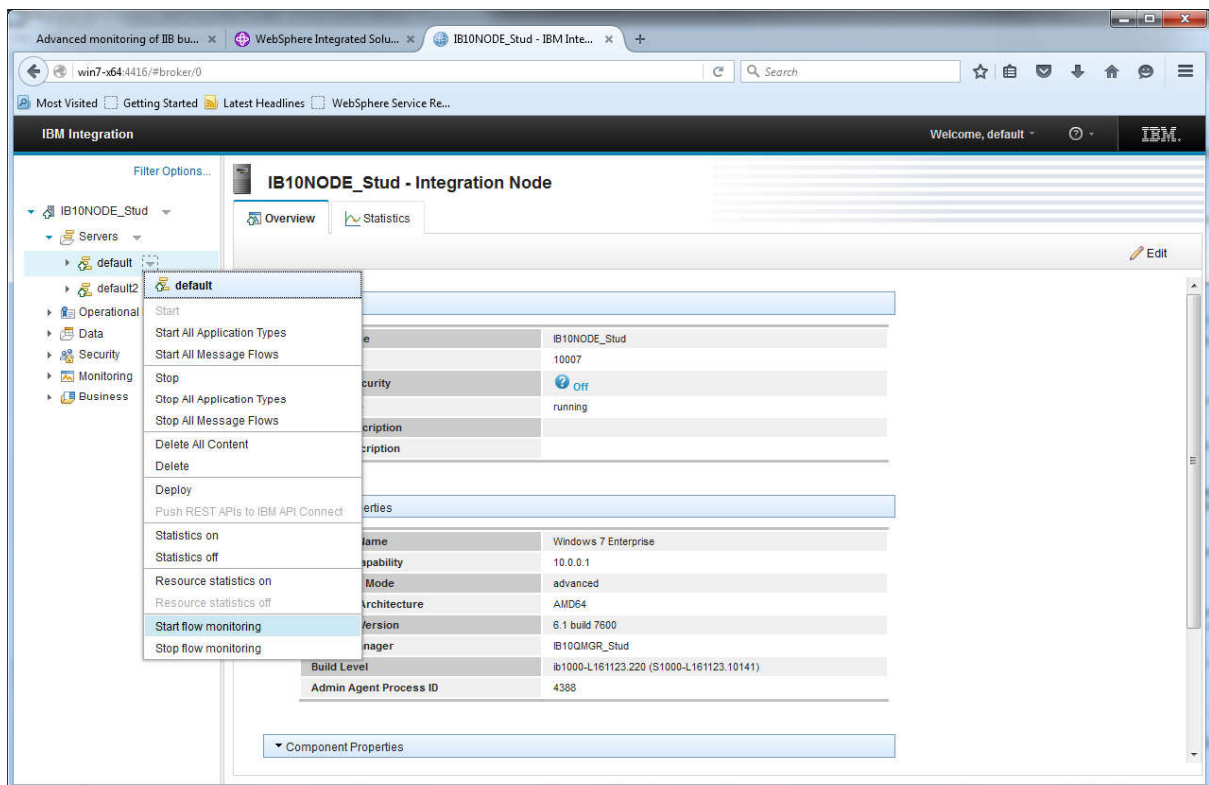
Content Encoding

Cancel OK

Activating monitoring for the flows in the integration server

By default, monitoring is not activated for the flows hosted by an integration server. An activation operation must be executed for that purpose.

This can be done using the IBM Integration Bus Web admin interface:



In the interface, retrieve the integration server, the application or the flow and, with the mouse, right click on it and select “Start flow monitoring”.

This can also be done using a command. For example, to enable monitoring for a single message flow:

```
mqsichangeflowmonitoring <node> -c active -e <server> -f <message_flow>
```

Or, to monitor all the message flows of an integration server:

```
mqsichangeflowmonitoring <node> -c active -e <server> -j
```

Configuring the MQ components to gather monitoring events

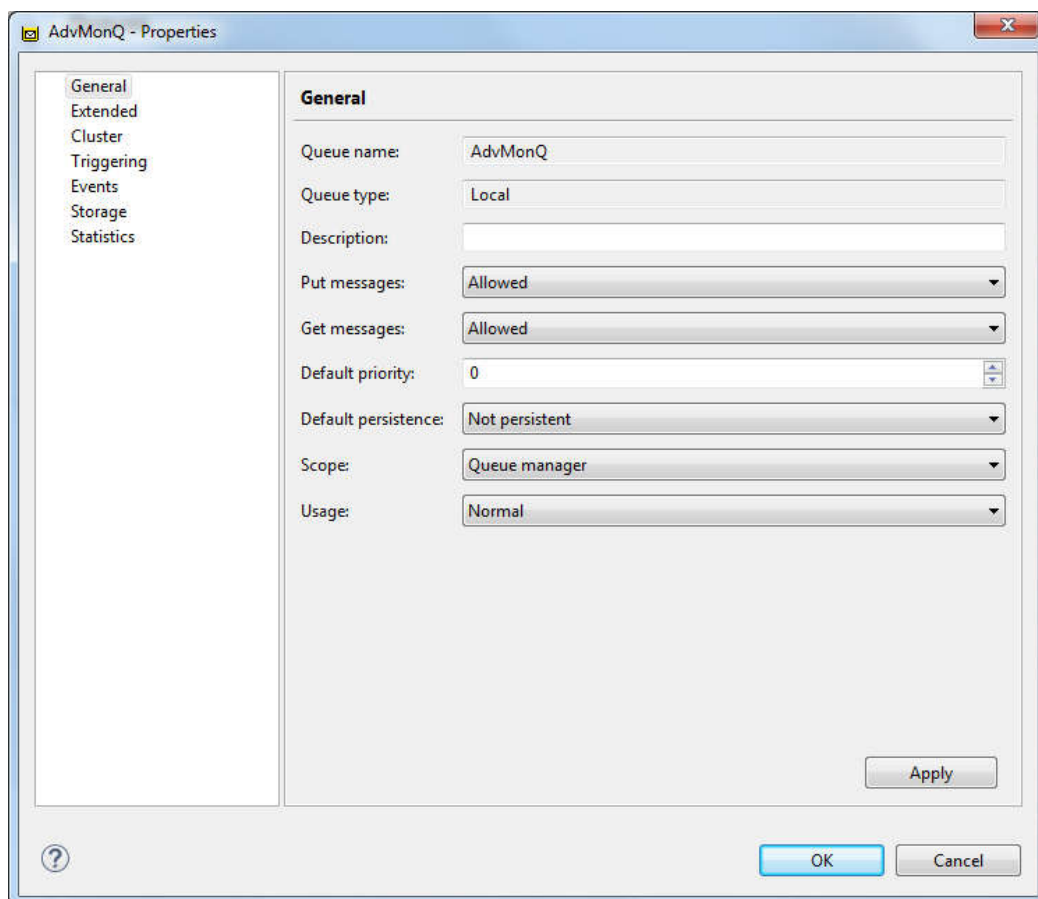
When several IBM Integration Bus nodes associated to different queue managers are involved in the configuration, a “target” queue manager will receive the monitoring events from the various “source” queue managers associated with IBM Integration Bus nodes and make them available to the Advanced Monitoring Asset. When one single IBM Integration Bus is involved, only one queue manager is needed and the configuration is simpler.

All the components described here under can be created either using the MQ Explorer or using “runmqsc” commands or scripts.

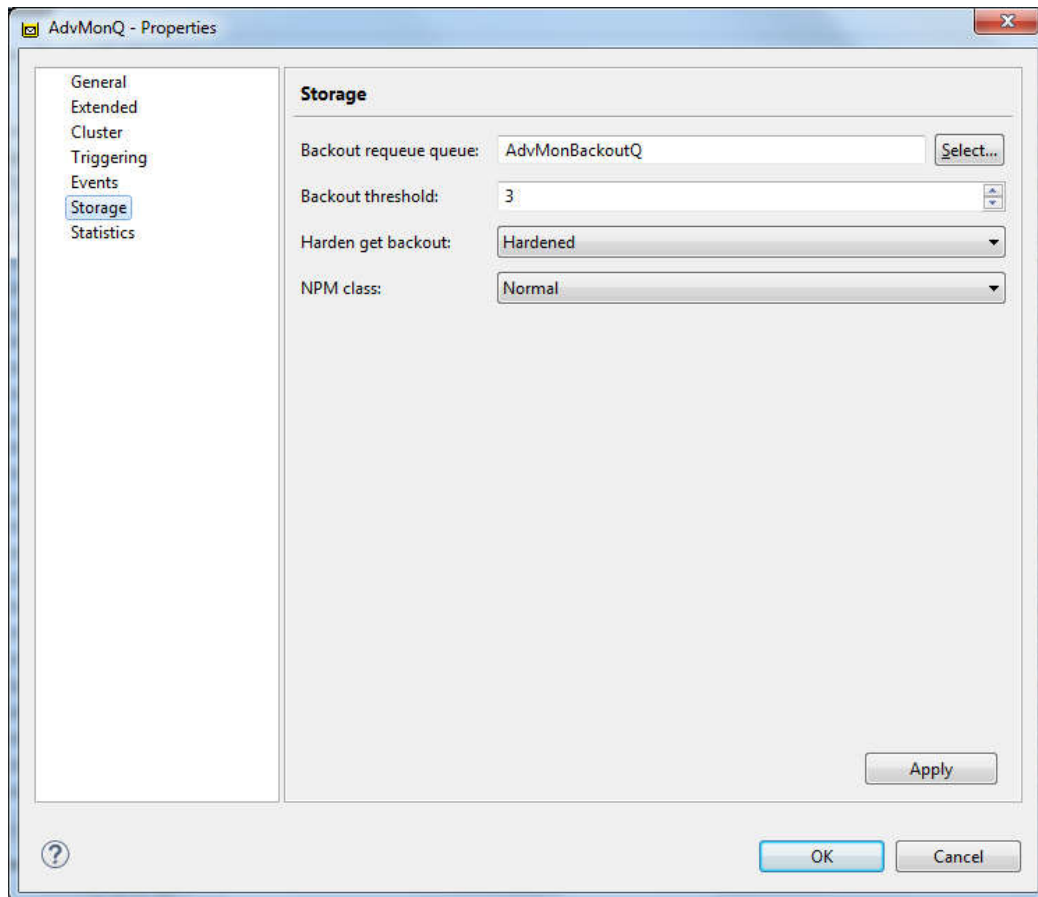
Multi node configuration

Create a “target” queue manager which will receive the monitoring events from the “source” queue managers associated to IBM Integration Bus nodes and make them available to the Advanced Monitoring Asset. Name this queue manager with any name that you like (I shall use “IB10QMGR_AdvMon” in my examples). On this queue manager create the following objects:

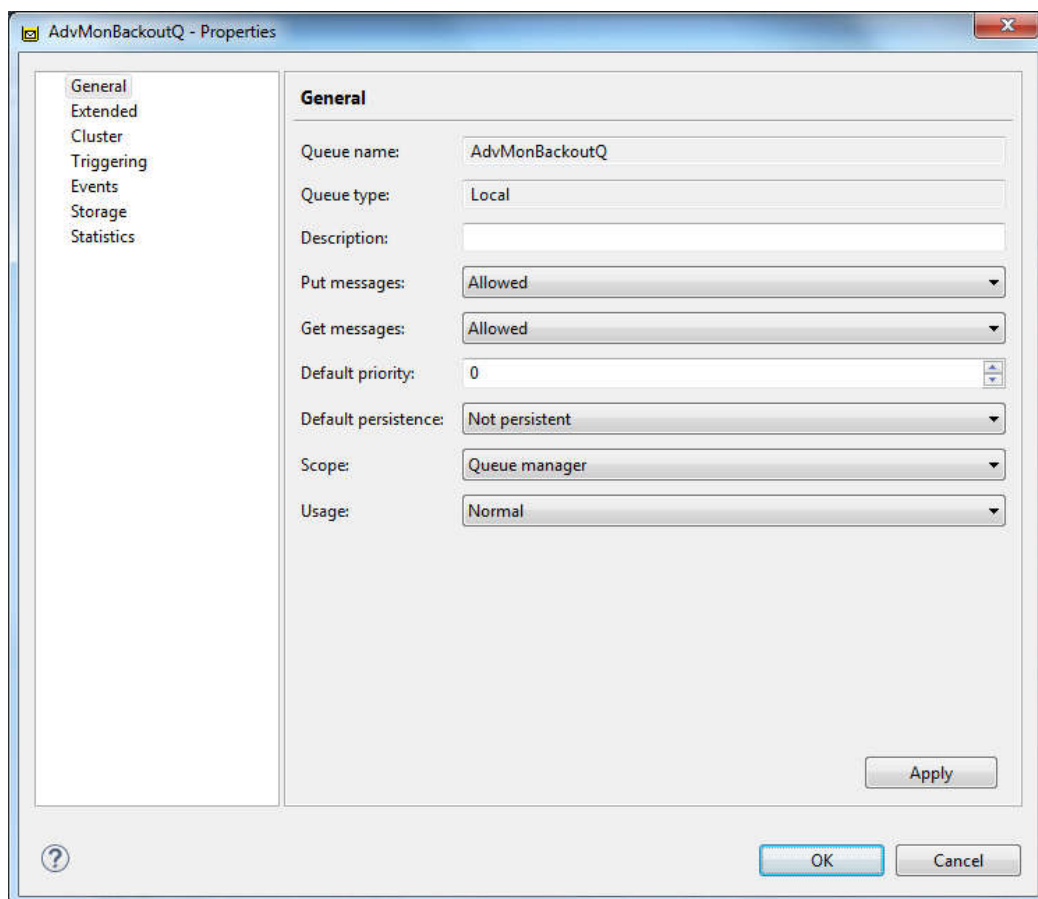
- A local queue with any name that you like (I shall use “AdvMonQ”) to be used as the target queue.



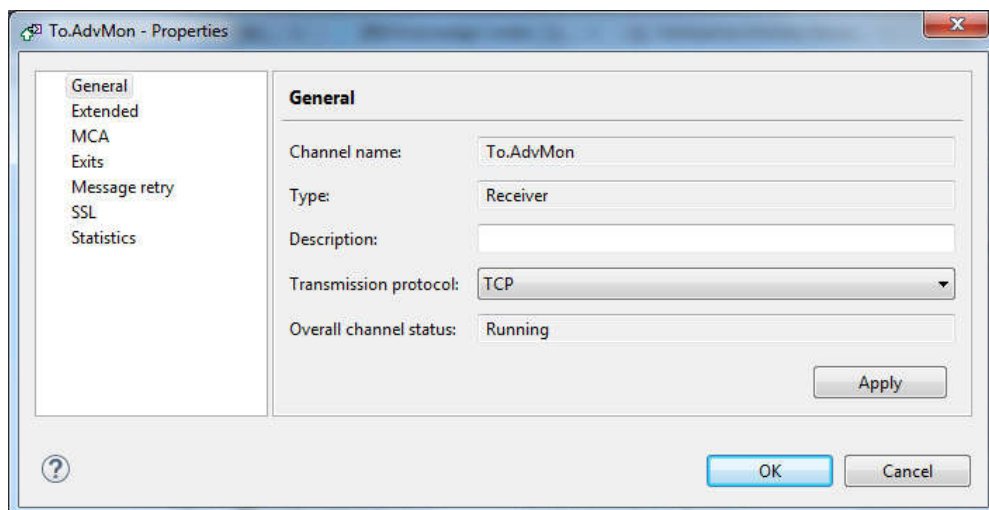
Also define a backout requeue queue and a backout threshold (so that, if the MDB part of the asset is not able to store the data extracted from an event message into the database, only a limited number of retries is attempted):



The corresponding backout requeue queue must also be defined :

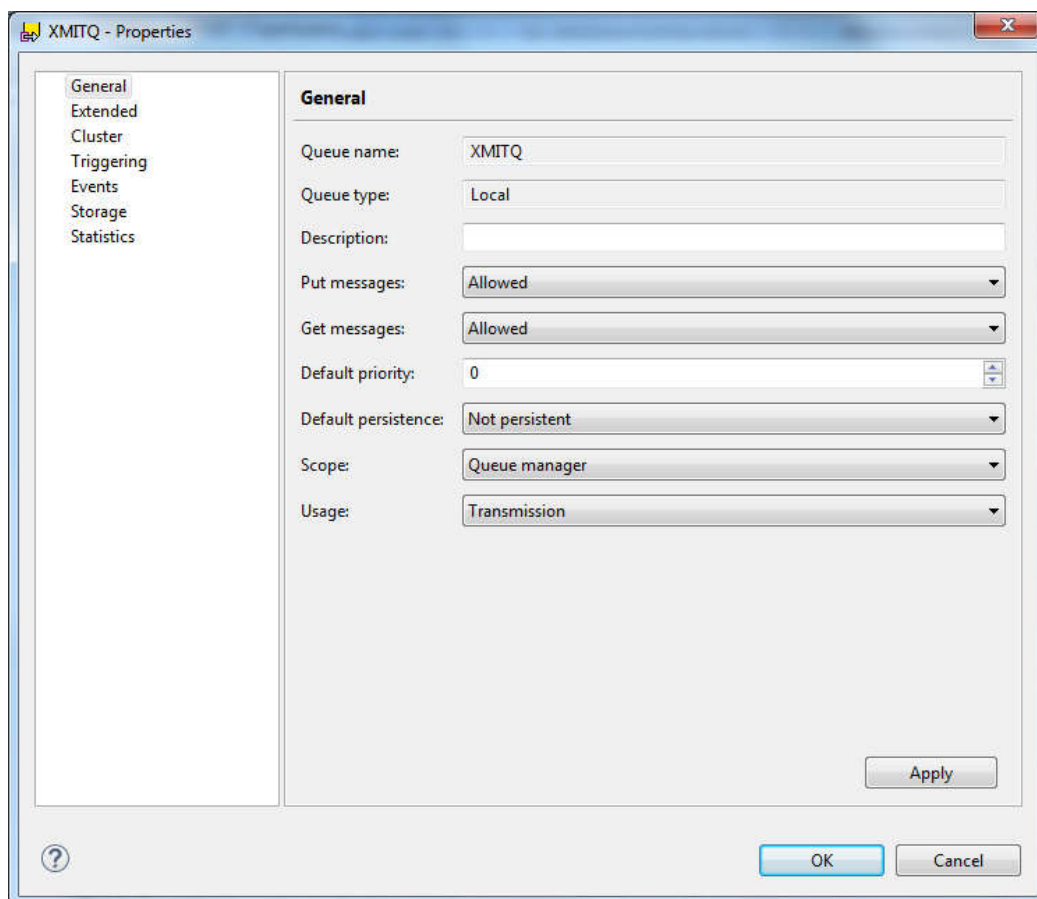


- Also create a receiver channel with any name that you like (I shall use “To.AdvMon”).



On the “source” queue manager associated to one of the integration nodes hosting the flows to be monitored, create the following objects:

- A local transmission queue with any name that you like (I shall use “XMITQ”).

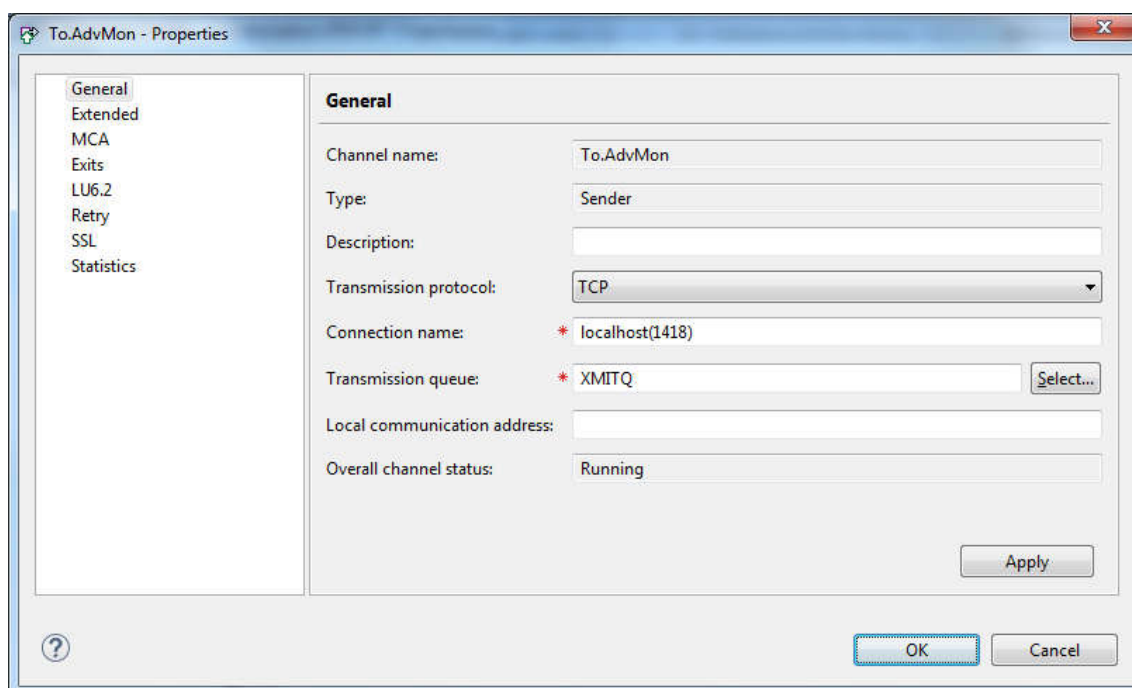


The image shows the 'XMITQ - Properties' dialog box. On the left is a tree view with the following items: General (selected), Extended, Cluster, Triggering, Events, Storage, and Statistics. The main area is titled 'General' and contains the following fields:

- Queue name: XMITQ
- Queue type: Local
- Description: (empty text box)
- Put messages: Allowed (dropdown)
- Get messages: Allowed (dropdown)
- Default priority: 0 (spin box)
- Default persistence: Not persistent (dropdown)
- Scope: Queue manager (dropdown)
- Usage: Transmission (dropdown)

At the bottom right are 'Apply', 'OK', and 'Cancel' buttons. A help icon (?) is at the bottom left.

- A sender channel with the same name that was used for the receiver channel on the previously defined “target” queue manager (I shall use “To.AdvMon”) and associate it with the previously defined local transmission queue and the “target” queue manager as destination (define its hostname and listener port in the “Connection name” field).



The image shows the 'To.AdvMon - Properties' dialog box. On the left is a tree view with the following items: General (selected), Extended, MCA, Exits, LU6.2, Retry, SSL, and Statistics. The main area is titled 'General' and contains the following fields:

- Channel name: To.AdvMon
- Type: Sender
- Description: (empty text box)
- Transmission protocol: TCP (dropdown)
- Connection name: * localhost(1418)
- Transmission queue: * XMITQ (with a 'Select...' button)
- Local communication address: (empty text box)
- Overall channel status: Running

At the bottom right are 'Apply', 'OK', and 'Cancel' buttons. A help icon (?) is at the bottom left.

- A remote queue with any name that you like (I shall use “AdvMonQ”) and make it point to the remote queue, the “target” queue manager and the previously defined transmission queue.

The screenshot shows the 'AdvMonQ - Properties' dialog box with the 'General' tab selected. The dialog has a sidebar on the left with tabs: General, Extended, Cluster, and Statistics. The main area contains the following fields:

- Queue name: AdvMonQ
- Queue type: Remote
- Description: (empty)
- Put messages: Allowed (dropdown)
- Default priority: 0 (spin box)
- Default persistence: Not persistent (dropdown)
- Scope: Queue manager (dropdown)
- Remote queue: AdvMonQ
- Remote queue manager: IB10QMGR_AdvMon
- Transmission queue: XMITQ (with a 'Select...' button next to it)

At the bottom right are 'Apply', 'OK', and 'Cancel' buttons. A help icon (?) is at the bottom left.

- A subscription with any name that you like (I shall use “AdvMonSub”) with topic string “\$SYS/Broker/<node>/Monitoring/+/+” (<node> must be replaced with the name of the local IBM Integration Bus node hosting the flows to be monitored) and pointing to the remote queue “AdvMonQ” as destination. This subscription will catch the monitoring events published on the topic and will forward them to the “target” queue (“AdvMonQ” in our case) on the “target” queue manager (through the remote queue also named “AdvMonQ” on the “source” queue manager).

AdvMonSub - Properties

General

Subscription name: AdvMonSub

Topic

Topic name:

Topic string: SSYS/Broker/IB10NODE_Stud/Monitoring/+/+

Wildcard usage: Topic level wildcard

Scope: All

Destination

Destination class: Provided

Destination queue manager:

Destination name: * AdvMonQ

Correlation identifier:

00000	41	4D	51	20	49	42	31	30	--	51	4D
00010	BF	EF	A1	58	21	6E	98	90	--		

Edit...

Durable: Yes

Type: Admin

Properties: Message properties

User data:

Selector:

Selector type: None

Apply

OK Cancel

For the other IBM Integration Bus nodes to be monitored together with the one you just created the configuration for, just repeat the previous operations for the queue managers associated to these additional nodes:

- Create a transmission queue.
- Create a sender channel to have the event messages routed to the “target” queue on the “target” queue manager.
- Create a remote queue.
- Create a subscription.

Single node configuration

When one single IBM Integration Bus node and one single queue manager is involved in the configuration, it is not necessary to create a “target” queue manager, nor to define channels, nor a transmission queue, nor a remote queue. No additional queue manager needs to be created. The only components to be created are therefore:

- A local “target” queue (where events will be made available to the asset).
- The subscription pointing to the local “target” queue as destination.

Configuring the database

Two tables are needed for storing the event data and the business flow data. These tables can be hosted by either a DB2 database or an Oracle database.

DB2

Create a new DB2 database with any name that you like (I used “ADVMON”) which will host the data for monitoring the flows. The default options can be used for this operation.

Then customize for your environment and execute the following SQL script to create the tables and the index:

```
CONNECT TO <database_name> USER <db2_user> USING <db2_password>;

DROP TABLE <schema>.WMB_MSGS;

CREATE TABLE <schema>.WMB_MSGS
(WMB_MSGKEY          VARCHAR(100) NOT NULL,
 TRANSAC_NAME        VARCHAR(100),
 EVENT_NAME          VARCHAR(100),
 EVENT_TYPE          CHAR(1),
 MSGFLOW_NAME        VARCHAR(100),
 APPL_NAME           VARCHAR(100),
 NODE_NAME           VARCHAR(100),
 NODE_TYPE           VARCHAR(100),
 TERMINAL_NAME       VARCHAR(100),
 EVENT_TIMESTAMP     TIMESTAMP,
 GLOBAL_TRANSACTION_ID VARCHAR(100),
 IN_IS               VARCHAR(100),
 APPL_DATA           VARCHAR(1000)
)
DATA CAPTURE NONE;

ALTER TABLE <schema>.WMB_MSGS ADD CONSTRAINT WMB_MSGS_PK PRIMARY KEY
(WMB_MSGKEY);

DROP TABLE <schema>.WMB_BUSTRANS;

CREATE TABLE <schema>.WMB_BUSTRANS
(GLOBAL_TRANSACTION_ID VARCHAR(100) NOT NULL,
 TRANSAC_NAME          VARCHAR(100),
 START_TIMESTAMP       TIMESTAMP,
 END_TIMESTAMP         TIMESTAMP,
 STATUS               CHAR(1),
 IN_IS_S              VARCHAR(100),
 IN_IS_E              VARCHAR(100),
 IN_IS_P              VARCHAR(1000),
 LAST_UPDT_TIMESTAMP  TIMESTAMP,
 LAST_DATA_UPDT_TSP   TIMESTAMP,
 APPL_DATA            VARCHAR(1000)
)
DATA CAPTURE NONE;

ALTER TABLE <schema>.WMB_BUSTRANS ADD CONSTRAINT WMB_BUSTRANS_PK PRIMARY
KEY (GLOBAL_TRANSACTION_ID);

DROP INDEX <schema>.WMB_BTRN_IDX;
```

```
CREATE INDEX <schema>.WMB_BTRN_IDX on <schema>.WMB_BUSTRANS
(GLOBAL_TRANSACTION_ID, TRANSAC_NAME, START_TIMESTAMP, END_TIMESTAMP,
STATUS, IN_IS_S, IN_IS_E);

TERMINATE;
```

In the previous script, <database_name>, <db2_user>, <db2_password> and <schema> must be replaced with the values suitable for your environment. Some further customizations might also be desirable.

Oracle

Create a new Oracle database with any name that you like (I used “ADVMON”) which will host the data for monitoring the flows. The default options can be used for this operation.

Then customize for your environment and execute the following SQL script to create the tables and the index:

```
DROP TABLE <schema>."WMB_MSGS";

CREATE TABLE <schema>."WMB_MSGS"
("WMB_MSGKEY" VARCHAR2(100 BYTE) NOT NULL ENABLE,
 "TRANSAC_NAME" VARCHAR2(100 BYTE),
 "EVENT_NAME" VARCHAR2(100 BYTE),
 "EVENT_TYPE" VARCHAR2(1 BYTE),
 "MSGFLOW_NAME" VARCHAR2(100 BYTE),
 "APPL_NAME" VARCHAR2(100 BYTE),
 "NODE_NAME" VARCHAR2(100 BYTE),
 "NODE_TYPE" VARCHAR2(100 BYTE),
 "TERMINAL_NAME" VARCHAR2(100 BYTE),
 "EVENT_TIMESTAMP" TIMESTAMP (6),
 "GLOBAL_TRANSACTION_ID" VARCHAR2(100 BYTE),
 "IN_IS" VARCHAR2(100 BYTE),
 "APPL_DATA" VARCHAR2(1000 BYTE),
 CONSTRAINT "WMB_MSGS_PK" PRIMARY KEY ("WMB_MSGKEY"));

DROP TABLE <schema>."WMB_BUSTRANS";

CREATE TABLE <schema>."WMB_BUSTRANS"
("GLOBAL_TRANSACTION_ID" VARCHAR2(100 BYTE) NOT NULL ENABLE,
 "TRANSAC_NAME" VARCHAR2(100 BYTE),
 "START_TIMESTAMP" TIMESTAMP (6),
 "END_TIMESTAMP" TIMESTAMP (6),
 "STATUS" VARCHAR2(1 BYTE),
 "IN_IS_S" VARCHAR2(100 BYTE),
 "IN_IS_E" VARCHAR2(100 BYTE),
 "IN_IS_P" VARCHAR2(1000 BYTE),
 "LAST_UPDT_TIMESTAMP" TIMESTAMP (6),
 "LAST_DATA_UPDT_TSP" TIMESTAMP (6),
 "APPL_DATA" VARCHAR2(1000 BYTE),
 CONSTRAINT "WMB_BUSTRANS_PK" PRIMARY KEY ("GLOBAL_TRANSACTION_ID"));

DROP INDEX <schema>."WMB_BTRN_IDX";

CREATE UNIQUE INDEX <schema>."WMB_BTRN_IDX" ON "ADVUSER"."WMB_BUSTRANS"
("GLOBAL_TRANSACTION_ID", "TRANSAC_NAME", "START_TIMESTAMP",
"END_TIMESTAMP", "STATUS", "IN_IS_S", "IN_IS_E");
```

In the previous script, <schema> must be replaced with the value suitable for your environment. Some further customizations might also be desirable.

Installing the asset modules on the application server

The asset is a Java application which must be executed in an application server. It can be installed on either:

- WebSphere Application Server
- WebSphere Liberty profile

WebSphere Application Server

The WAS server having been started, with the WAS Admin console, define the following settings:

- Define a JDBC provider.

In the case of DB2, create a JDBC driver of type “DB2 Universal JDBC Driver Provider (XA)” pointing to directory containing the JDBC driver JAR files.

[JDBC providers](#) > **DB2 Universal JDBC Driver Provider (XA)**

Use this page to edit properties of a Java Database Connectivity (JDBC) provider. The JDBC provider object encapsulates the specific JDBC driver implementation class for access to the specific vendor database of your environment.

Configuration

General Properties

* **Scope**
cells:win7-x64Node02Cell:nodes:win7-x64Node02:servers:server1

* **Name**
DB2 Universal JDBC Driver Provider (XA)

Description
Two-phase commit DB2 JCC provider that supports JDBC 3.0. Data sources that use this provider support the use of XA to perform 2-phase commit processing. Use of driver type 2 on the application server for z/OS is not supported for data sources created under this provider.

Class path
\${DB2UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc.jar
\${UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc_license_cu.jar
\${DB2UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc_license_cisuz.jar

Native library path
\${DB2UNIVERSAL_JDBC_DRIVER_NATIVEPATH}

☐ Isolate this resource provider

* **Implementation class name**
com.ibm.db2.jcc.DB2XADataSource

Additional Properties

- [Data sources](#)
- [Data sources \(WebSphere Application Server V4\)](#)

Apply OK Reset Cancel

Or, in the case of Oracle, create a JDBC driver of type “Oracle JDBC Driver (XA)” pointing to directory containing the JDBC driver JAR files.

Advanced Monitoring Asset for IBM Integration Bus

[JDBC providers](#) > [Oracle JDBC Driver \(XA\)](#)

Use this page to edit properties of a Java Database Connectivity (JDBC) provider. The JDBC provider object encapsulates the specific JDBC driver implementation class for access to the specific vendor database of your environment.

Configuration

General Properties	Additional Properties
<p>* Scope</p> <p>cells:win7-x64Node02Cell:nodes:win7-x64Node02:servers:server1</p>	<p>» Data sources</p> <p>» Data sources (WebSphere Application Server V4)</p>
<p>* Name</p> <p>Oracle JDBC Driver (XA)</p>	
<p>Description</p> <p>Oracle JDBC Driver (XA)</p>	
<p>Class path</p> <p>\${ORACLE_JDBC_DRIVER_PATH}/ojdbc6.jar</p>	
<p>Native library path</p>	
<p><input type="checkbox"/> Isolate this resource provider</p>	
<p>* Implementation class name</p> <p>oracle.jdbc.xa.client.OracleXADataSource</p>	
<p>Apply OK Reset Cancel</p>	

- Create an authentication alias for connecting to the database.

In the case of DB2, create an authentication alias with the credentials (userid and password) to connect to DB2.

[Global security](#) > [JAAS - J2C authentication data](#) > [win7-x64Node02/db2alias](#)

Specifies a list of user identities and passwords for Java(TM) 2 connector security to use.

General Properties

* Alias

win7-x64Node02/db2alias

* User ID

db2admin

* Password

Description

Apply OK Reset Cancel

Or, in the case of Oracle, create an authentication alias with the credentials (userid and password) to connect to Oracle.

[Data sources](#) > [ADVMOND](#) > [JAAS - J2C authentication data](#) > [win7-x64Node02/OracleAlias](#)

Specifies a list of user identities and passwords for Java(TM) 2 connector security to use.

General Properties

* Alias

win7-x64Node02/OracleAlias

* User ID

ADVUSER

* Password

Description

Apply OK Reset Cancel

- Create a datasource.

In the case of DB2, create a new JDBC datasource using the previously created JDBC driver and the following properties:

- Name: “ADVMON”.
- JDBC name: “jdbc/ADVMON”. This name is mandatory, because it is referred by the asset’s code.
- Database name: The name of the database that you created in DB2 (“ADVMON” in my case).
- Server name: The hostname or IP address of the server hosting DB2.
- Port number: The port of the DB2 server.
- Authentication alias: The previously created alias with the DB2 credentials.

Data sources > ADVMOND

Use this page to edit the settings of a datasource that is associated with your selected JDBC provider. The datasource object supplies your application with connections for accessing the database.

Configuration

[Test connection](#)

General Properties

Scope

cells:win7-x64Node02Cell:nodes:win7-x64Node02:servers:server1

Provider

DB2 Universal JDBC Driver Provider (XA)

Name

ADVMON

JNDI name

jdbc/ADVMON

☒ Use this data source in container managed persistence (CMP)

Description

DB2 Universal Driver Datasource

Category

Data store helper class name

☒ Select a data store helper class

Data store helper classes provided by WebSphere Application Server

- DB2 Universal data store helper**
(com.ibm.websphere.rsadapter.DB2UniversalDataStoreHelper)
- DB2 for iSeries data store helper
(com.ibm.websphere.rsadapter.DB2AS400DataStoreHelper)

☐ Specify a user-defined data store helper

Enter a package-qualified data store helper class name

Additional Properties

- [Connection pool properties](#)
- [WebSphere Application Server data source properties](#)
- [Custom properties](#)

Related Items

- [JAAS - J2C authentication data](#)

Security settings

Select the authentication values for this resource.

Authentication alias for XA recovery

win7-x64Node02/db2alias

Component-managed authentication alias

win7-x64Node02/db2alias

Mapping-configuration alias

(none)

Container-managed authentication alias

(none)

Common and required data source properties

Name	Value
Driver type	4
Database name	ADVMON
Server name	localhost
Port number	50000

Or, in the case of Oracle, create a new JDBC datasource using the previously created JDBC driver and the following properties:

- Name: "ADVMON".
- JDBC name: "jdbc/ADVMON". This name is mandatory, because it is referred by the asset's code.
- Database name: The name of the database that you created in Oracle ("ADVMON" in my case).
- Authentication alias: The previously created alias with the Oracle credentials.
- URL: "jdbc:oracle:thin:@//localhost:<oracle_port>/<database_name>" (in my case: "jdbc:oracle:thin:@//localhost:1521/ADVMON").

Data sources > ADVMONO

Use this page to edit the settings of a datasource that is associated with your selected JDBC provider. The datasource object supplies your application with connections for accessing the database.

Configuration

Test connection

General Properties

Scope

cells:win7-x64Node02Cell:nodes:win7-x64Node02:servers:server1

Provider

Oracle JDBC Driver (XA)

Name

ADVMON

JNDI name

jdbc/ADVMON

☒ Use this data source in container managed persistence (CMP)

Description

New JDBC Datasource

Category

Data store helper class name

☒ Select a data store helper class

Data store helper classes provided by WebSphere Application Server

Oracle11g data store helper

(com.ibm.websphere.rsadapter.Oracle11gDataStoreHelper)

☐ Specify a user-defined data store helper

Enter a package-qualified data store helper class name

Security settings

Select the authentication values for this resource.

Authentication alias for XA recovery

win7-x64Node02/OracleAlias

Component-managed authentication alias

win7-x64Node02/OracleAlias

Mapping-configuration alias

(none)

Container-managed authentication alias

(none)

Common and required data source properties

Name	Value
URL	jdbc:oracle:thin:@//localhost:

Additional Properties

Connection pool properties

WebSphere Application Server data source properties

Custom properties

Related Items

JAAS - J2C authentication data

Apply

OK

Reset

Cancel

- Create a JMS queue with the following properties :
 - JMS provider: WebSphere MQ.
 - Name: "AdvMonQ".
 - JNDI name: "jms/AdvMonQ". This name is mandatory, because it is referred by the asset's code.
 - Queue name: The name that you used for the target queue ("AdvMonQ" in my case).
 - Queue manager name: The name that you used for the "target" queue manager ("IB10QMGR_AdvMon" in my case).

Queues > AdvMonQ

Queue destinations provided for point-to-point messaging by the WebSphere MQ messaging provider. Use WebSphere MQ queue destination administrative objects to manage queue destinations for the WebSphere MQ messaging provider.

Configuration

General Properties

Administration

Scope

Node=win7-x64Node02,Server=server1

Provider

WebSphere MQ messaging provider

* Name

AdvMonQ

* JNDI name

jms/AdvMonQ

Description

WebSphere MQ Queue

* Queue name

AdvMonQ

Queue manager or Queue sharing group name

IB10QMGR_AdvMon

Apply

OK

Reset

Cancel

Additional Properties

- Advanced properties
- WebSphere MQ Queue Connection Properties
- Custom properties

- Select the “WebSphere MQ Queue Connection Properties” link:
 - Queue manager host: The hostname of the server hosting the “target” queue manager (named “IB10QMGR_AdvMon” in my case).
 - Queue manager port: The port of the listener of the “target” queue manager.
 - Server connection channel name: The server connection channel to be used (for example: “SYSTEM.DEF.SVRCONN”).

Queues > AdvMonQ > WebSphere MQ connection properties

Use this panel to specify how to connect to the queue manager that hosts the queue. The system uses these connection properties to retrieve, display and update the queue configuration details that are shown on the WebSphere MQ queue settings panel.

Configuration

General Properties

Additional Properties

- MQ Config

Queue manager host

localhost

Queue manager port

1418

Server connection channel name

SYSTEM.DEF.SVRCONN

User ID

Password

Apply

OK

Reset

Cancel

- Create a JMS activation specification with the following properties:

- JMS provider: WebSphere MQ.
- Name: “AdvMonAS”.
- JNDI name: “jms/AdvMonAS”. This name is mandatory, because it is referred by the asset’s code.
- Queue manager name: The name that you used for the “target” queue manager (in my case: “IB10QMGR_AdvMon”).
- Transport: Client.
- Queue manager host: The hostname of the server hosting the “target” queue manager (named “IB10QMGR_AdvMon” in my case).
- Queue manager port: The port of the listener of the “target” queue manager.
- Server connection channel name: The server connection channel to be used (for example: “SYSTEM.DEF.SVRCONN”).
- Destination JNDI name: “jms/AdvMonQ”. This name is mandatory, because it is referred by the asset’s code.
- Destination type: Queue.

[Activation specifications](#) > **AdvMonAS**

WebSphere MQ Activation Specification

Configuration

General Properties

Administration

Scope

Node=win7-x64Node02,Server=server1

Provider

WebSphere MQ Resource Adapter

* Name

AdvMonAS

* JNDI name

jms/AdvMonAS

Description

Additional Properties

Advanced properties

Broker properties

Custom properties

Client transport properties

Related Items

JAAS - J2C authentication data

Connection

Queue manager

IB10QMGR_AdvMon

Transport

Client

Enter host and port information in the form of separate hostname and port values

* Hostname

localhost

Port

1418

Enter host and port information in the form of a connection name list

Connection name list

Server connection channel

SYSTEM.DEF.SVRCONN

Use SSL to secure communication with WebSphere MQ

Centrally managed

Specific configuration

SSL configuration

NodeDefaultSSLSettings

Destination

* Destination JNDI name

jms/AdvMonQ

Message selector

Destination type

Queue

Durable subscription

Subscription name

Advanced

Client ID

Allow cloned durable subscriptions

Provider version

Security settings

Select the authentication values for this resource.

Authentication alias

(none)

Apply

OK

Cancel

- And, finally, create a namespace binding with the following properties:
 - Binding identifier: “GeneralParametersFileBinding”.
 - Name in namespace: “GeneralParametersFile” (this name cannot be changed).
 - String value: Any path and file name that you like (I used “D:\AdvMon\GeneralParameters.properties”).

[Name Space Bindings](#) > **GeneralParametersFileBinding**

Use this page to configure a name binding of a constant string value.

Configuration

General Properties

* Scope

* Binding type

* Binding identifier

* Name in name space relative to lookup name prefix 'cell/nodes/win7-x64Node02/servers/server1/'

* String value

Also create a file with these name and path and with the following parameters (which values can be adjusted for your needs).

In the case of DB2:

```
# General parameters

DBType = DB2
DBSchema = DB2INST1
MaxBusFlowLineNumber = 20000
MaxTimeLineNumber = 10000
ImagesInBackground = true
BackgroundColor = #c7d7ee
```

And, in the case of Oracle:

```
# General parameters

DBType = Oracle
DBSchema = ADVUSER
MaxBusFlowLineNumber = 20000
MaxTimeLineNumber = 10000
ImagesInBackground = true
BackgroundColor = #c7d7ee
```


These are the general parameters of the asset, with their default values, which you can change if you wish. Their meaning is the following:

- DBType: The database type: DB2 or Oracle.
- DBSchema: The schema of the two tables in the DB2 or Oracle database.
- MaxBusFlowLineNumber: Maximum number of lines which can be retrieved at a time from the database when searching business flow instances. When more than <MaxBusFlowLineNumber> matching the search criteria are present in the database, only the first <MaxBusFlowLineNumber> are retrieved by the query.
- MaxTimeLineNumber: Maximum number of lines which can be displayed for the periods of time when displaying bar chart statistics. If more than <MaxTimeLineNumber> lines should be displayed for the selected period of time, only the first <MaxTimeLineNumber> are displayed in the chart.
- ImagesInBackground: By default, the GUI part of the asset displays pages with images of modern buildings in the background. If you prefer to have a plain color background, set this parameter to “false”.
- BackgroundColor: If you set the previous parameter to false (use a plain color rather than images for the background), select the background color to be used here. The default value is light blue.

Then, using either the admin console or “wsadmin” commands, deploy the “IIBAdvMonMDBWeb.ear” and the “IIBAdvMonGUIWeb.ear” files.

When deploying “IIBAdvMonMDBWeb.ear”, map the activation specification and queue to be used by the asset with “jms/AdvMonAS” and “jms/AdvMonQ”:

Specify options for installing enterprise applications and modules.

Step 3: Bind listeners for message-driven beans

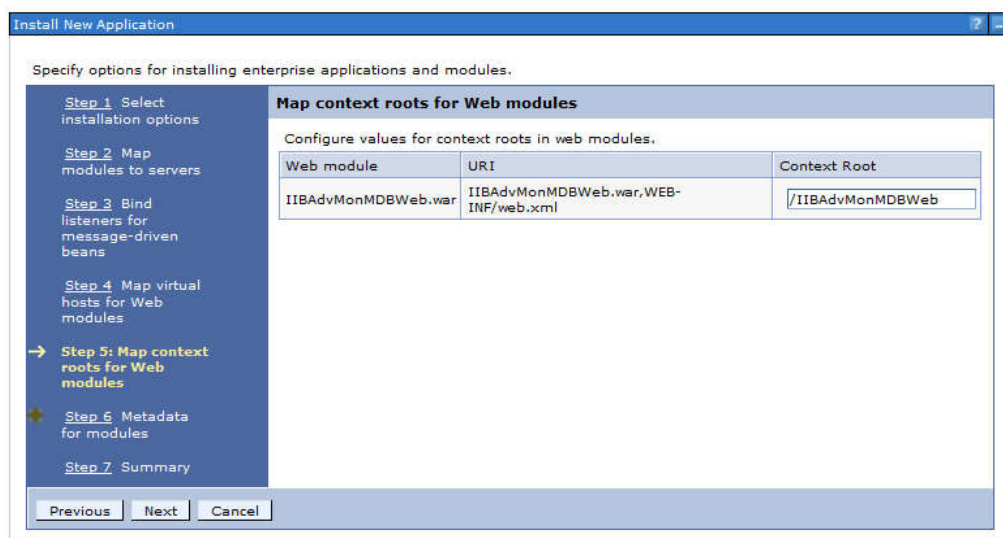
Each message-driven enterprise bean in your application or module must be bound to a listener port name or to an activation specification JNDI name. When a message-driven enterprise bean is bound to an activation specification JNDI name you can also specify the destination JNDI name and authentication alias.

☐ Apply Multiple Mappings

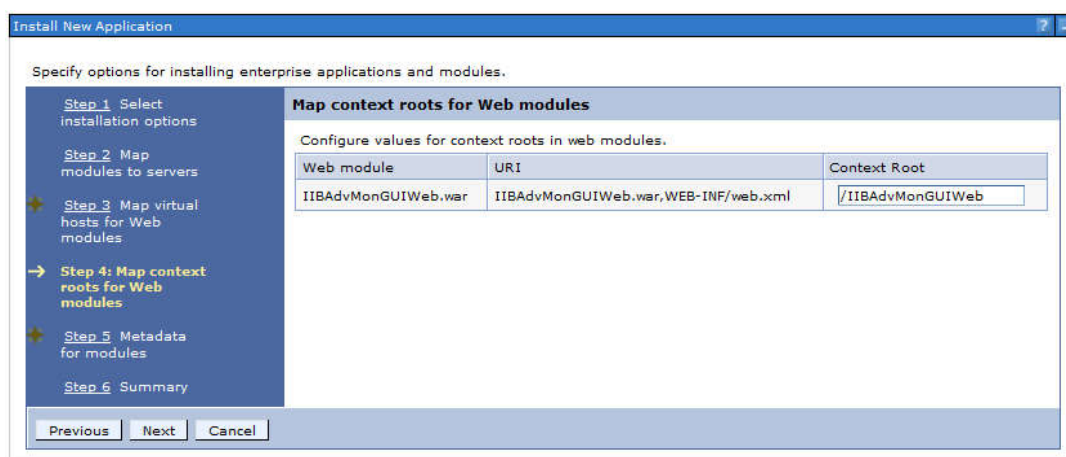
Select	Module	Bean	URI	Messaging type	Listener Bindings
<input type="checkbox"/>	IIBAdvMonMDBWeb.war	MonEventLoaderMDB	IIBAdvMonMDBWeb.war, WEB-INF/ejb-jar.xml	javax.jms.MessageListener	<input type="radio"/> Listener port Name <input type="radio"/> Activation Specification Target Resource JNDI Name jms/AdvMonAS Destination JNDI name jms/AdvMonQ ActivationSpec authentication alias

Previous Next Cancel

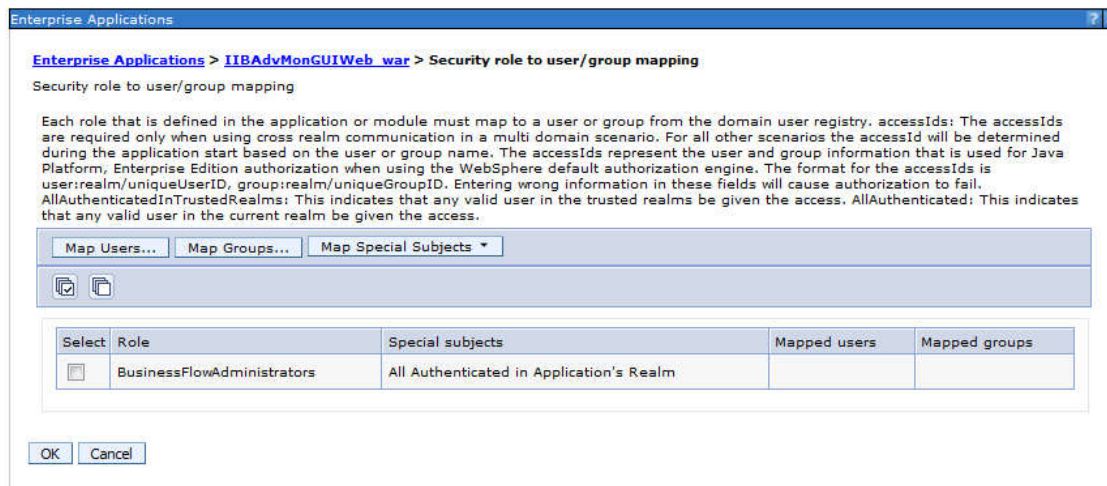
Then map the context root “IIBAdvMonMDBWeb” for the module:



When deploying “IIBAdvMonGUIWeb.ear”, map the context root “IIBAdvMonGUIWeb” for the module:



When the two applications are deployed, open the settings of the “IIBAdvMonGUIWeb” application with the Admin Console and map the suitable users to the role “BusinessFlowAdministrators”.



Then start the two Web modules and check the log file ("SystemOut.log") to make sure that there is not any error.

WebSphere Liberty Profile

If not already installed, add the following features to your Liberty server (using the "installUtility" command):

- javaee-7.0,
- wmqJmsClient-2.0.

Install the asset as described below. First open the "server.xml" file of the Liberty server and add the following settings:

- Add a JDBC driver and a datasource definitions:

In the case of DB2, define a JDBC driver and a JDBC datasource for the DB2 database:

```
<dataSource connectionManagerRef="ConMgr" id="ADVMON"
jndiName="jdbc/ADVMON" type="javax.sql.DataSource">
  <jdbcDriver>
    <library name="DB2JCC4Library">
      <fileset dir="<db2_path>\java" includes="db2jcc4.jar,
db2jcc_license_cu.jar"/>
    </library>
  </jdbcDriver>
  <properties.db2.jcc databaseName="<database_name>"
user="<db2_userid>" password="<db2_password>"/>
</dataSource>

<connectionManager id="ConMgr" maxPoolSize="<pool_size>"/>
```

In the above statements, replace <db2_path>, <database_name>, <db2_userid>, <db2_password> and <pool_size> with the values suitable for your environment.

Or, in the case of Oracle, define a JDBC driver and a JDBC datasource for the Oracle database:

```
<dataSource connectionManagerRef="ConMgr" id="ADVMON"
jndiName="jdbc/ADVMON">
  <jdbcDriver>
    <library name="OracleLib">
```

```

        <fileset dir="<oracle_path>\jlib"
includes="ojdbc6.jar"/>
        </library>
        </jdbcDriver>
        <properties.oracle
URL="jdbc:oracle:thin:@//localhost:1521/<database_name>"
user="<oracle_userid>" password="<oracle_password>" />
        </dataSource>

        <connectionManager id="ConMgr" maxPoolSize="<pool_size>"/>

```

In the above statements, replace `<oracle_path>`, `<database_name>`, `<oracle_userid>`, `<oracle_password>` and `<pool_size>` with the values suitable for your environment.

- Next, define the MQ queue and activation specification:

```

        <variable name="wmqJmsClient.rar.location"
value="<mq_path>\java\lib\jca\wmq.jmsra.rar"/>

        <jmsQueue id="jms/AdvMonQ" jndiName="jms/AdvMonQ">
        <properties.wmqJms baseQueueManagerName="<queue_manager_name>"
baseQueueName="<queue_name>"/>
        </jmsQueue>

        <jmsActivationSpec id="IIBAdvMonMDBWeb/MonEventLoaderMDB">
        <properties.wmqJms brokerVersion="1" destinationRef="jms/AdvMonQ"
maxPoolDepth="<pool_size>" port="<mq_port>" providerVersion="7"
queueManager="<queue_manager_name>" shareConvAllowed="false"
subscriptionDurability="NonDurable"/>
        </jmsActivationSpec>

```

In the above statements, replace `<mq_path>`, `<queue_manager_name>`, `<queue_name>`, `<pool_size>` and `<mq_port>` with the values suitable for your environment.

- Then create a JNDI entry for the external parameter file:

```

        <jndiEntry jndiName="GeneralParametersFile" value="<path>
\GeneralParameters.properties" />

```

In the previous statement, replace `<path>` with the value suitable for your environment.

Also create a file with this name and path and with the following contents (adjust parameter values as needed for your environment).

In the case of DB2:

```

# General parameters

DBType = DB2
DBSchema = DB2INST1
MaxBusFlowLineNumber = 20000
MaxTimeLineNumber = 10000
ImagesInBackground = true
BackgroundColor = #c7d7ee

```

And, in the case of Oracle :

```
# General parameters

DBType = Oracle
DBSchema = ADVUSER
MaxBusFlowLineNumber = 20000
MaxTimeLineNumber = 10000
ImagesInBackground = true
BackgroundColor = #c7d7ee
```

These are the general parameters of the asset (with the values in used in my environment). Their meaning is the following:

- DBType: The database type: DB2 or Oracle.
 - DBSchema: The schema of the two tables in the DB2 or Oracle database.
 - MaxBusFlowLineNumber: Maximum number of lines which can be retrieved from the database when searching business flow instances. When more than <MaxBusFlowLineNumber> matching the search criteria are present in the database, only the first <MaxBusFlowLineNumber> are retrieved by the query.
 - MaxTimeLineNumber: Maximum number of lines which can be displayed for the periods of time when displaying bar chart statistics. If more than <MaxTimeLineNumber> lines should be displayed for the selected period of time, only the first <MaxTimeLineNumber> are displayed in the chart.
 - ImagesInBackground: By default, the asset displays pages with images about buildings in the background. If you prefer to have a plain color background, set this parameter to “false”.
 - BackgroundColor: If you set the previous parameter to “false” (use a plain color rather than images for the background), select the background color here. The default value is light blue.
- Copy the files “IIBAdvMonMDBWeb.war” and “IIBAdvMonUIWeb.war” to the “apps” directory of the Liberty server.
 - Also add the following statements into the “server.xml” file:

```
<webApplication id="IIBAdvMonMDBWeb" location="IIBAdvMonMDBWeb.war"
name="IIBAdvMonMDBWeb"/>
```

```
<webApplication id="IIBAdvMonUIWeb" location="IIBAdvMonUIWeb.war"
name="IIBAdvMonUIWeb"/>
```

- Finally, add the security role “BusinessFlowAdministrators” to the “IIBAdvMonGUIWeb” application and map the suitable users or groups to this role :

```
<webApplication id="IIBAdvMonGUIWeb" location="IIBAdvMonGUIWeb.war"
name="IIBAdvMonGUIWeb">
  <application-bnd>
    <security-role name="BusinessFlowAdministrators">
      <special-subject type="ALL_AUTHENTICATED_USERS" />
    </security-role>
  </application-bnd>
</webApplication>
```

When this is done, save and close the “server.xml” file, start your Liberty profile server and check the log file (“console.log”) to make sure that there is not any error and that the two Web modules are up and running.

Miscellaneous

Security

The Advanced Monitoring Asset is made of two parts:

- “IIBAdvMonMDBWeb”, a Web application containing a message driven bean.
- “IIBAdvMonGUIWeb”, a “classical” Web application holding servlets and JSP.

“IIBAdvMonMDBWeb” is a module running in the background, with no user interface, which does not need to be secured.

On the other hand, “IIBAdvMonGUIWeb” is a user interface which needs to be secured, so that only authorized users can access the information about the business flows. A security role “BusinessFlowAdministrators” has been defined for that purpose in the “IIBAdvMonGUIWeb” module. The users or groups to be authorized must be mapped to this role when the application is deployed.

High availability

This asset is made of two parts:

- “IIBAdvMonMDBWeb”, a Web application containing a message driven bean.
- “IIBAdvMonGUIWeb”, a “classical” Web application holding servlets and JSP.

These two modules can be deployed to a cluster to enable high availability and workload management.

Session affinity must be set for “IIBAdvMonGUIWeb”, which stores and uses data in the session. This will enable the plugin to route all requests related to the same session to the cluster member hosting this session.

Supported browsers

The browsers which have been successfully tested are:

- Firefox.
- Internet Explorer

Chrome is not supported because it does not support the `<input type = "datetime-local">` HTML tag yet.

No other browser has been tested.

Native language support

The following languages are supported by the GUI part of the Advanced Monitoring Asset:

- English,
- French,
- German,
- Spanish.

The language used is the one defined in the browser.

For the other languages, the dialogs are displayed in English (used as default language).

Using the GUI

The GUI can be accessed using a browser (either Firefox or Internet Explorer). The URL to be entered is: <http://<hostname>:<port>/IIBAdvMonGUIWeb/AdvMonitor> and it opens the main menu:

Advanced monitoring of IBM Integration Bus business flows

See all business flows
See all failed business flows
Monitor by integration nodes/servers
Monitor by business flow types

Start: 2017/02/19 00:00:00 (CET)
End: 2017/08/04 17:43:03 (CEST) ☒ Now

Business data (name, value):

Max number of business flows per page: 7

Welcome in the Advanced Monitoring application for IBM Integration Bus business flows!

Please select an action in the above menu.

Querying/listing business flow instances

On the left of the main menu, there is a choice between 4 different query actions:

- Use “See all business flows” to select all business flow instances matching the filter criteria.

Advanced monitoring of IBM Integration Bus business flows

See all business flows
See all failed business flows
Monitor by integration nodes/servers
Monitor by business flow types

Start: 2017/01/19 00:00:00 (CET)
End: 2017/08/04 15:48:01 (CEST) ☒ Now

Business data (name, value):

Max number of business flows per page: 7

All business flows for the period from **January 19, 2017 12:00:00 AM CET to August 6, 2017 3:48:01 PM CEST** (from 15 to 21 of 10010):

Limit to failed flows

Id	Status	Start time	End time	Business flow type	Node / Server	Business data
4140512049423130514447525f532475bfeaf1582176511c	Completed	2017-02-20 09:34:44.776	2017-02-20 09:34:44.776	Small Business Appl	Start: IB10NODE_Stud / default End: IB10NODE_Stud / default	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Details/Geography/Item/Country: UK Details/Geography/Item/Country: Germany Details/Geography/Item/Country: Spain
4140512049423130514447525f532475bfeaf1582176511e	Failed	2017-02-20 09:38:44.809	2017-02-20 09:34:44.81	Small Business Appl	Start: IB10NODE_Stud / default End: IB10NODE_Stud / default	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Details/Geography/Item/Country: UK Details/Geography/Item/Country: Germany Details/Geography/Item/Country: Spain
4140512049423130514447525f532475bfeaf15821765120	Completed	2017-02-20 09:34:44.844	2017-02-20 09:34:44.853	Retail Order	Start: IB10NODE_Stud / default End: IB10NODE_Stud / default	orderId: 2 customerNumber: 5555 itemNumber: 1234-6789-1212 requestType: orderItem customerName: ChocolateFan itemName: Chocolate chip cookies
4140512049423130514447525f532475bfeaf15821765122	Completed	2017-02-20 09:34:44.883	2017-02-20 09:34:44.888	Retail Order	Start: IB10NODE_Stud / default End: IB10NODE_Stud / default	orderId: 2 customerNumber: 5670 itemNumber: 3529-9999-1937 requestType: orderItem customerName: CoffeeFan itemName: ColumbianSupreme
4140512049423130514447525f532475bfeaf15821765124	Completed	2017-02-20 09:34:44.918	2017-02-20 09:34:44.921	Retail Order	Start: IB10NODE_Stud2 / default End: IB10NODE_Stud2 / default	orderId: 3 customerNumber: 4785 itemNumber: 9876-5342-3847 requestType: orderItem customerName: TeaFan itemName: CeylonBlackTea
4140512049423130514447525f532475bfeaf15821765124	Completed	2017-02-20 09:34:44.95	2017-02-20 09:35:02.001	Multi Broker Transaction	Start: IB10NODE_Stud / default Other: IB10NODE_Stud / default End: IB10NODE_Stud2 / default	
4140512049423130514447525f532475bfeaf15821765126	Completed	2017-02-20 09:34:44.986	2017-02-20 09:34:44.99	Retail Order	Start: IB10NODE_Stud / default End: IB10NODE_Stud / default	orderId: 2 customerNumber: 5555 itemNumber: 1234-6789-1212 requestType: orderItem customerName: ChocolateFan itemName: Chocolate chip cookies

- Use “See all failed business flows” to select all failed business flow instances matching the filter criteria.

Advanced Monitoring Asset for IBM Integration Bus

Advanced monitoring of IBM Integration Bus business flows

[See all business flows](#)
[See all failed business flows](#)
[Monitor by integration nodes/servers](#)
[Monitor by business flow types](#)

Start: 2017/02/10 00:00:00 (CET) End: 2017/08/12 22:56:03 (CEST) ☒ Now

Business data (name, value):

Max number of business flows per page: [Logout](#)

All failed business flows for the period from **February 10, 2017 12:00:00 AM CET** to **August 12, 2017 10:56:03 PM CEST** (from 6 to 10 of 1001):

Id	Status	Start time	End time	Business flow type	Node / Server	Business data
414d512049423130514d47525f537475bfe1582176516a	Failed	2017-02-20 09:34:46.256	2017-02-20 09:34:46.257	Small Business Appl	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Value: 0 Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain
414d512049423130514d47525f537475bfe15821765178	Failed	2017-02-20 09:34:46.011	2017-02-20 09:34:46.012	Small Business Appl	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Value: 0 Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain
414d512049423130514d47525f537475bfe1582176518a	Failed	2017-02-20 09:34:46.975	2017-02-20 09:34:46.976	Small Business Appl	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Value: 0 Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain
414d512049423130514d47525f537475bfe1582176519c	Failed	2017-02-20 09:34:47.333	2017-02-20 09:34:47.333	Small Business Appl	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Value: 0 Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain
414d512049423130514d47525f537475bfe158217651ae	Failed	2017-02-20 09:34:47.698	2017-02-20 09:34:47.699	Small Business Appl	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Value: 0 Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain

[Go to preceding](#) [Go to beginning](#) [Go to line](#) [Go to next](#) [Go to end](#) [Home](#)

- Use “Monitor by integration nodes/servers” to select the business flow instances by integration nodes, then by integration servers, then by business flow types (or by part of these criteria).

Advanced monitoring of IBM Integration Bus business flows

[See all business flows](#)
[See all failed business flows](#)
[Monitor by integration nodes/servers](#)
[Monitor by business flow types](#)

Start: 2017/01/19 00:00:00 (CET) End: 2017/08/06 15:55:19 (CEST) ☒ Now

Business data (name, value):

Max number of business flows per page: [Logout](#)

All integration nodes:

Node "IB10NODE_Stud" selected:

Server "default" selected:

Business flow types: Multi Broker Transaction, Retail Order

Business flows of node "IB10NODE_Stud", of server "default" and of business flow type "Multi Broker Transaction" for the period from **January 19, 2017 12:00:00 AM CET** to **August 6, 2017 3:55:19 PM CEST** (from 6 to 10 of 2002):

Id	Status	Start time	End time	Business flow type	Node / Server	Business data
414d512049423130514d47525f537475bfe15821765136	Completed	2017-02-20 09:34:45.306	2017-02-20 09:35:02.05	Multi Broker Transaction	Start: IB10NODE_Stud / default Other: IB10NODE_Stud2 / default End: IB10NODE_Stud2 / default	
414d512049423130514d47525f537475bfe1582176513a	Completed	2017-02-20 09:34:45.449	2017-02-20 09:35:02.068	Multi Broker Transaction	Start: IB10NODE_Stud / default Other: IB10NODE_Stud2 / default End: IB10NODE_Stud2 / default	
414d512049423130514d47525f537475bfe15821765148	Completed	2017-02-20 09:34:45.663	2017-02-20 09:35:02.087	Multi Broker Transaction	Start: IB10NODE_Stud / default Other: IB10NODE_Stud2 / default End: IB10NODE_Stud2 / default	
414d512049423130514d47525f537475bfe15821765150	Completed	2017-02-20 09:34:45.809	2017-02-20 09:35:02.108	Multi Broker Transaction	Start: IB10NODE_Stud / default Other: IB10NODE_Stud2 / default End: IB10NODE_Stud2 / default	
414d512049423130514d47525f537475bfe1582176515a	Completed	2017-02-20 09:34:46.025	2017-02-20 09:35:02.124	Multi Broker Transaction	Start: IB10NODE_Stud / default Other: IB10NODE_Stud2 / default End: IB10NODE_Stud2 / default	

[Go to preceding](#) [Go to beginning](#) [Go to line](#) [Go to next](#) [Go to end](#) [Home](#)

- Use “Monitor by business flow types” to select the business flow instances by business flow types, then by integration nodes, then by integration servers (or by part of these criteria).

Advanced monitoring of IBM Integration Bus business flows

See all business flows
See all failed business flows
Monitor by integration nodes/servers
Monitor by business flow types

Start: 2017/01/19 00:00:00 (CET)
End: 2017/08/06 15:59:33 (CEST) ☒ Now

Business data (name, value):

Max number of business flows per page: 5

All types of business flows:

Business flow types
Multi Broker Transaction
Retail Order
Small Business Appli

Business flow type "Small Business Appli" selected:

Nodes
IB10NODE_Stud

Servers
default2

Business flows of type "Small Business Appli", of node "IB10NODE_Stud" and server "default2" for the period from January 19, 2017 12:00:00 AM CET to August 6, 2017 3:59:33 PM CEST (from 11 to 15 of 2002):

Limit to failed flows: Order: select a sort method... Statistics: select type and display...

Id	Status	Start time	End time	Business flow type	Node / Server	Business data
414d512049423130514447525f537475bfe15821765164	Completed	2017-02-20 09:34:46.22	2017-02-20 09:34:46.221	Small Business Appli	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain
414d512049423130514447525f537475bfe15821765166	Failed	2017-02-20 09:34:46.256	2017-02-20 09:34:46.257	Small Business Appli	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain
414d512049423130514447525f537475bfe15821765176	Completed	2017-02-20 09:34:46.575	2017-02-20 09:34:46.575	Small Business Appli	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain
414d512049423130514447525f537475bfe15821765178	Failed	2017-02-20 09:34:46.611	2017-02-20 09:34:46.612	Small Business Appli	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain
414d512049423130514447525f537475bfe15821765188	Completed	2017-02-20 09:34:46.94	2017-02-20 09:34:46.94	Small Business Appli	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain

Go to preceding
Go to next
Go to beginning
Go to end
Home

In the middle of the main menu, there are input fields for filtering the business flow instances to be retrieved:

- The period of time when the flow instances to be retrieved were started. This period of time is specified using a start date/time and an end date/time. The "Now" check box can be used to specify that the end time is the current date/time.
- Business data. This type of filter is optional. It enables to filter the business flow instances using data available in the event message payload. The values to be entered for a data filter are:
 - The data name on the left.
 - The data value on the right. The data value can include wildcard characters: _ (underscore) for a single character and % (percent) for any number of characters (1 or more). These special characters can also be escaped by adding a \ (back slash) in front of them.

Up to 3 data filters can be defined. Both simple and complex data can be used for filtering. For simple data, the name to be used is a plain data name (for example: "customerId"). For complex data, the name must include a path to the single data (for example: "Details/Geography/Country/City").

Advanced monitoring of IBM Integration Bus business flows

See all business flows
See all failed business flows
Monitor by integration nodes/servers
Monitor by business flow types

Start: 2017/01/19 00:00:00 (CET)
End: 2017/08/06 15:48:01 (CEST) ☒ Now

Business data (name, value):

Max number of business flows per page: 7

Advanced Monitoring Asset for IBM Integration Bus

On the right of the main menu, there is an input field to specify the maximum number of business flow instances to be displayed as a table on a page (when needed, actions are provided at the bottom of the table to scroll forward or backward in the list of retrieved business flow instances).

After a query, the list of retrieved business flow instances (or part of it) is displayed as a table. The instances are sorted by start times.

Advanced monitoring of IBM Integration Bus business flows

Start: 2017/01/19 00:00:00 (CET) Business data (name, value):
 End: 2017/08/06 15:48:01 (CEST) ☒ Now

Max number of business flows per page: 7

All business flows for the period from **January 19, 2017 12:00:00 AM CET** to **August 6, 2017 3:48:01 PM CEST** (from 15 to 21 of 10010):

Id	Status	Start time	End time	Business flow type	Node / Server	Business data
414d512049423130514d47525f537475b6fa1582176511c	Completed	2017-02-20 09:34:44.776	2017-02-20 09:34:44.776	Small Business Appli	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain Value: 1
414d512049423130514d47525f537475b6fa1582176511a	Failed	2017-02-20 09:34:44.809	2017-02-20 09:34:44.81	Small Business Appli	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2	Id: 12345 Details/Geography/Item/City: Paris Details/Geography/Item/Country: France Details/Geography/Item/City: London Details/Geography/Item/Country: UK Details/Geography/Item/City: Berlin Details/Geography/Item/Country: Germany Details/Geography/Item/City: Madrid Details/Geography/Item/Country: Spain Value: 0
414d512049423130514d47525f537475b6fa15821765120	Completed	2017-02-20 09:34:44.844	2017-02-20 09:34:44.853	Retail Order	Start: IB10NODE_Stud / default End: IB10NODE_Stud / default	orderId: 2 customerNumber: 5555 itemNumber: 1234-6789-1212 requestType: orderItem customerName: ChocolateFan itemName: Chocolate chip cookies
414d512049423130514d47525f537475b6fa15821765122	Completed	2017-02-20 09:34:44.883	2017-02-20 09:34:44.888	Retail Order	Start: IB10NODE_Stud / default End: IB10NODE_Stud / default	orderId: 2 customerNumber: 5670 itemNumber: 3529-9999-1937 requestType: orderItem customerName: CoffeeFan itemName: ColumbianSupreme
414d512049423130514d47525f537475b6fa15821765124	Completed	2017-02-20 09:34:44.918	2017-02-20 09:34:44.921	Retail Order	Start: IB10NODE_Stud2 / default End: IB10NODE_Stud2 / default	orderId: 3 customerNumber: 4785 itemNumber: 9876-5342-3847 requestType: orderItem customerName: TeaFan itemName: CeylonBlackTea
414d512049423130514d47525f537475b6fa15821765124	Completed	2017-02-20 09:34:44.95	2017-02-20 09:35:02.001	Multi Broker Transaction	Start: IB10NODE_Stud / default Other: IB10NODE_Stud / default2 End: IB10NODE_Stud2 / default	
414d512049423130514d47525f537475b6fa15821765126	Completed	2017-02-20 09:34:44.986	2017-02-20 09:34:44.99	Retail Order	Start: IB10NODE_Stud / default End: IB10NODE_Stud / default	orderId: 2 customerNumber: 5555 itemNumber: 1234-6789-1212 requestType: orderItem customerName: ChocolateFan itemName: Chocolate chip cookies

The columns contain the following data:

- Id: The Id of the business flow instance, defined as the “Global transaction corellator” property of the message flow nodes (monitoring tab).

Edit event

Basic Correlation Transaction

Event Correlation

A monitoring application uses event correlators to match events emitted by the same, or related, business transactions. A local transaction correlator links the events emitted by a single invocation of a message flow. A parent transaction correlator links the events from a message flow to a parent message flow or an external application. A global transaction correlator links events from a message flow to one or more related message flows or external applications. An event must contain a local transaction correlator, but need not contain a parent transaction correlator or global transaction correlator.

Local transaction correlator:

☒ Automatic ☐ Specify location of correlator

Description
The local correlator used by the most recent event for this message flow invocation will be used. If no local correlator exists yet, a new unique value will be generated.

Parent transaction correlator:

☒ Automatic ☐ Specify location of correlator

Description
The parent correlator used by the most recent event for this message flow invocation will be used. If no correlator exists yet, no parent correlator will be used.

Global transaction correlator:

☐ Automatic ☒ Specify location of correlator

Description
The global correlator will be read from the specified location in the message tree. Ensure the specified location contains a suitable global correlator value.

\$Root/MQMD/MsgId Edit...

Cancel OK

- Status: The current status of the business flow instance. Either “Started”, “In process”, “Completed” or “Failed”. This status is defined in the “Event name” property of message flow nodes (monitoring tab) and extracted from the received monitoring events.

Edit event

Basic | Correlation | Transaction

Event Source
Select the source of the event.
Out terminal

Event Source Address
The integration node identifies an event source using an event source address. Use this value when you enable and disable event sources using runtime commands.
MQ Input.terminal.out

Event Name
Provide the name by which events emitted from this source are to be known. Specify either a literal name, or the location of a character field in the message tree or elsewhere in the message assembly.
☒ Literal Small Business Appli#S#Small Business Flow started
☐ Data location Edit...

Event Filter
Provide an expression to control whether the event is emitted. The expression must evaluate to true or false, and can reference fields in the message tree or elsewhere in the message assembly. If you do not specify a value, the value true() is used.
true() Edit...

Event Payload
Most events need to contain data taken from fields in the message tree or from elsewhere in the message assembly. Data taken from simple fields or complex fields appears in the event in XML character format. An event can also contain bitstream data, which appears in the event as hexadecimal bytes.

Data location	
\$Root/JSON/Data/Test/Id	Add...
\$Root/JSON/Data/Test/Value	Edit...
\$Root/JSON/Data/Test/Details	Delete

☐ Include bitstream data in payload

Content: Encoding:

Cancel OK

When this field contains three parts separated by “#”, the second part is taken as the status. When this field contains two parts separated by a “#”, the first part is taken as the status. Otherwise, when the field contains no “#”, the status is unknown and regarded as “in process”.

- Start time: The date and time when the (first) start event of the business flow instance occurred.
- End time: The date and time when the (last) end or failure event of the business flow instance occurred.
- Business flow type: Name of the business flow which the instance belongs to. This name is defined in the “Event name” property of message flow nodes (monitoring tab) and extracted from the received monitoring events.

Edit event

Basic Correlation Transaction

Event Source
Select the source of the event.
Out terminal

Event Source Address
The integration node identifies an event source using an event source address. Use this value when you enable and disable event sources using runtime commands.
MQ Input.terminal.out

Event Name
Provide the name by which events emitted from this source are to be known. Specify either a literal name, or the location of a character field in the message tree or elsewhere in the message assembly.
☒ Literal Small Business Appl#S#Small Business Flow started
☐ Data location Edit...

Event Filter
Provide an expression to control whether the event is emitted. The expression must evaluate to true or false, and can reference fields in the message tree or elsewhere in the message assembly. If you do not specify a value, the value true() is used.
true() Edit...

Event Payload
Most events need to contain data taken from fields in the message tree or from elsewhere in the message assembly. Data taken from simple fields or complex fields appears in the event in XML character format. An event can also contain bitstream data, which appears in the event as hexadecimal bytes.

Data location	
\$Root/JSON/Data/Test/Id	Add...
\$Root/JSON/Data/Test/Value	Edit...
\$Root/JSON/Data/Test/Details	Delete

☐ Include bitstream data in payload

Content: Encoding:

Cancel OK

When this field contains three parts separated by '#', the first part is taken as the business flow name. Otherwise, when this field contains less than two '#', the flow name is not provided by the event (but it may be retrieved from other events correlated with this one).

- **Node / Server:** Contains the nodes/servers where the message flow instances belonging to the business flow instance executed and emitted monitoring events. The cell can be splitted into 3 parts:
 - The node(s)/server(s) from which start events were received.
 - The node(s)/server(s) from which end or failure events were received.
 - The node(s)/server(s) from which other events were received.

Id	Status	Start time	End time	Business flow type	Node / Server	
414d512049423130514d47525f53747509fc9c5892ec3b21	Completed	2017-02-19 16:09:54.932	2017-02-19 16:09:55.277	Retail Order	Start: IB10NODE_Std / default End: IB10NODE_Std / default	order custo item
414d512049423130514d47525f53747509fc9c5892ec3b21	Completed	2017-02-19 16:09:54.953	2017-02-19 16:09:55.154	Multi Broker Transaction	Start: IB10NODE_Std / default Other: IB10NODE_Std2 / default End: IB10NODE_Std2 / default	
414d512049423130514d47525f53747509fc9c5892ec3b21	Completed	2017-02-19 16:09:54.962	2017-02-19 16:09:54.965	Small Business Appli	Start: IB10NODE_Std / default2 End: IB10NODE_Std / default2	Id: 12 Detail Detail

- Business data: The business data defined as the “Event payload” property of the message flow nodes (monitoring tab) and extracted from the payload of the message processed by the business flow instance.

Edit event

Basic

Correlation

Transaction

Event Source

Select the source of the event.

Out terminal

Event Source Address

The integration node identifies an event source using an event source address. Use this value when you enable and disable event sources using runtime commands.

MQ Input.terminal.out

Event Name

Provide the name by which events emitted from this source are to be known. Specify either a literal name, or the location of a character field in the message tree or elsewhere in the message assembly.

Literal

Small Business Appli#S#Small Business Flow started

Data location

Edit...

Event Filter

Provide an expression to control whether the event is emitted. The expression must evaluate to true or false, and can reference fields in the message tree or elsewhere in the message assembly. If you do not specify a value, the value true() is used.

true()

Edit...

Event Payload

Most events need to contain data taken from fields in the message tree or from elsewhere in the message assembly. Data taken from simple fields or complex fields appears in the event in XML character format. An event can also contain bitstream data, which appears in the event as hexadecimal bytes.

Data location

Add...

\$Root/JSON/Data/Test/Id

Edit...

\$Root/JSON/Data/Test/Value

Delete

\$Root/JSON/Data/Test/Details

Include bitstream data in payload

Content

Encoding

?

Cancel

OK

Several data can be displayed and they can be of simple or complex type. For simple type data, only the name and value are displayed (for example: “customerNumber: 7654”). For complex type data, the name includes a path (for example: “Details/Geography/Country/City: Paris”).

Completed business flow instances are displayed in green, while failed business flow instances are displayed in red, and in process (or in doubt) business flow instances in orange.

At the bottom of the list, when needed, there are actions to scroll forward and backward in the list of retrieved flow instances:

- Go to preceding: To display the flow instances just preceding those currently displayed.
- Go to next: To display the flow instances immediately following those currently displayed.
- Go to beginning: To go to the beginning of the list.
- Go to end: To go to the end of the list.
- Go to line: To go to a specific line number in the list.

#14d51204e923130214057525f53247586a5a858224f6808	Completed	2017-02-20 09:34:45.629	2017-02-20 09:34:45.632	Retail Order	Start: IB10NODE_Stud2 / default End: IB10NODE_Stud2 / default	orderId: 3 customerNumber: 4785 itemNumber: 9976-5342-3847	requestType: orderItem customerName: TeaFan itemName: CeylonBlackTea
#14d51204e923130214057525f53247586a5a858224f6808	Completed	2017-02-20 09:34:45.663	2017-02-20 09:35:02.067	Multi Broker Transaction	Start: IB10NODE_Stud / default Other: IB10NODE_Stud / default2 End: IB10NODE_Stud2 / default2		

[Go to preceding](#)
[Go to next](#)
[Go to beginning](#)
[Go to end](#)
[Go to line](#)
[Home](#)

By default, the list of instances is sorted by start times (ascending order). It is possible to select other sort methods:

- Sort by start times (ascending). The method used by default.
- Sort by start times (descending).
- Sort by end times (ascending).
- Sort by end times (descending).
- Sort by status (ascending).
- Sort by status (descending).
- Sort by flow types (ascending).
- Sort by flow types (descending).
- Sort by start nodes / servers (ascending).
- Sort by start nodes / servers (descending).
- Sort by end nodes / servers (ascending).
- Sort by end nodes / servers (descending).

Advanced Monitoring Asset for IBM Integration Bus

All business flows for the period from **February 9, 2017 12:00:00 AM CET** to **August 9, 2017 9:53:00 PM CEST** (from 1 to 20 of 10010):

Limit to failed flows	Order: select a sort method...	Statistics: select type and display...			
	Order: select a sort method...				
	Sort by start times - Ascending				
	Sort by end times - Ascending				
	Sort by end times - Descending				
	Sort by status - Ascending				
	Sort by status - Descending				
	Sort by flow types - Ascending				
	Sort by flow types - Descending				
	Sort by start nodes / servers - Ascending				
	Sort by start nodes / servers - Descending				
	Sort by end nodes / servers - Ascending				
	Sort by end nodes / servers - Descending				
414d512049423130514d47525f537475bfefa1582176510a	Completed	2017-02-20 09:34:44.237	2017-02-20 09:34:44.337	Retail Order	Start: IB10NODE_Stud / default End: IB10NODE_Stud / default
414d512049423130514d47525f537475bfefa1582176510a	Completed	2017-02-20 09:34:44.306	2017-02-20 09:34:44.353	Retail Order	Start: IB10NODE_Stud / default End: IB10NODE_Stud / default
414d512049423130514d47525f537475bfefa1582176510a	Completed	2017-02-20 09:34:44.347	2017-02-20 09:34:44.354	Retail Order	Start: IB10NODE_Stud / default End: IB10NODE_Stud / default
414d512049423130514d47525f537475bfefa1582176510a	Completed	2017-02-20 09:34:44.388	2017-02-20 09:35:01.935	Multi Broker Transaction	Start: IB10NODE_Stud / default Other: IB10NODE_Stud / default2 End: IB10NODE_Stud2 / default2
414d512049423130514d47525f537475bfefa1582176510a	Completed	2017-02-20 09:34:44.423	2017-02-20 09:34:44.426	Small Business Appli	Start: IB10NODE_Stud / default2 End: IB10NODE_Stud / default2

When a sort method is selected, it is kept for the following queries (until another sort method is selected).

By selecting the id (first column) of a business flow instance in the list, it is possible to display all the events which have been received for this instance as another table below.

Events received for business flow "414d512049423130514d47525f537475bfefa15821765112" (5):

Id	Event time	Event name	Event type	Node / Server	Application	Flow	Node	Node type	Terminal	Business data
ID: 414d512049423130514d47525f537475bfefa158216f2e69	2017-02-20 09:34:44.596	MQ Input.OutTerminal	Démarre	IB10NODE_Stud / default	MultiBrokerTransactionApplication1	MultiBrokerTransactionFlow1	MQ Input	ComibmMQInputNode	out	
ID: 414d512049423130514d47525f537475bfefa158216f2e6e	2017-02-20 09:34:44.596	MQ Output.InTerminal	En cours	IB10NODE_Stud / default	MultiBrokerTransactionApplication1	MultiBrokerTransactionFlow1	MQ Output	ComibmMQOutputNode	in	
ID: 414d512049423130514d47525f537475bfefa15821704d6f	2017-02-20 09:34:44.597	MQ Output.InTerminal	En cours	IB10NODE_Stud / default2	MultiBrokerTransactionApplication2	MultiBrokerTransactionFlow2	MQ Output	ComibmMQOutputNode	in	
ID: 414d512049423130514d47525f537475bfefa15821704d74	2017-02-20 09:35:01.944	MQ Output.InTerminal	En cours	IB10NODE_Stud2 / default	MultiBrokerTransactionApplication3	MultiBrokerTransactionFlow3	MQ Output	ComibmMQOutputNode	in	
ID: 414d512049423130514d47525f537475bfefa15821704d74	2017-02-20 09:35:01.948	MQ Output.InTerminal	Completed	IB10NODE_Stud2 / default2	MultiBrokerTransactionApplication4	MultiBrokerTransactionFlow4	MQ Output	ComibmMQOutputNode	in	

Displaying statistics about the business flow instances

From a list of business flow instances resulting of a query and displayed below the main menu, it is possible to request several kinds of statistics.

All business flows for the period from **February 9, 2017 12:00:00 AM CET** to **August 9, 2017 9:53:00 PM CEST** (from 1 to 20 of 10010):

Limit to failed flows	Order: select a sort method...	Statistics: select type and display...		
	Statistics: select type and display...			
	Statistics on throughputs by periods of time			
	Statistics on throughputs by nodes/servers and flow types			
	Statistics on processing times			
414d512049423130514d47525f537475bfefa15821765102	Completed	2017-02-20 09:34:44.306	2017-02-20 09:34:44.337	Retail Order
414d512049423130514d47525f537475bfefa15821765104	Completed	2017-02-20 09:34:44.306	2017-02-20 09:34:44.353	Retail Order

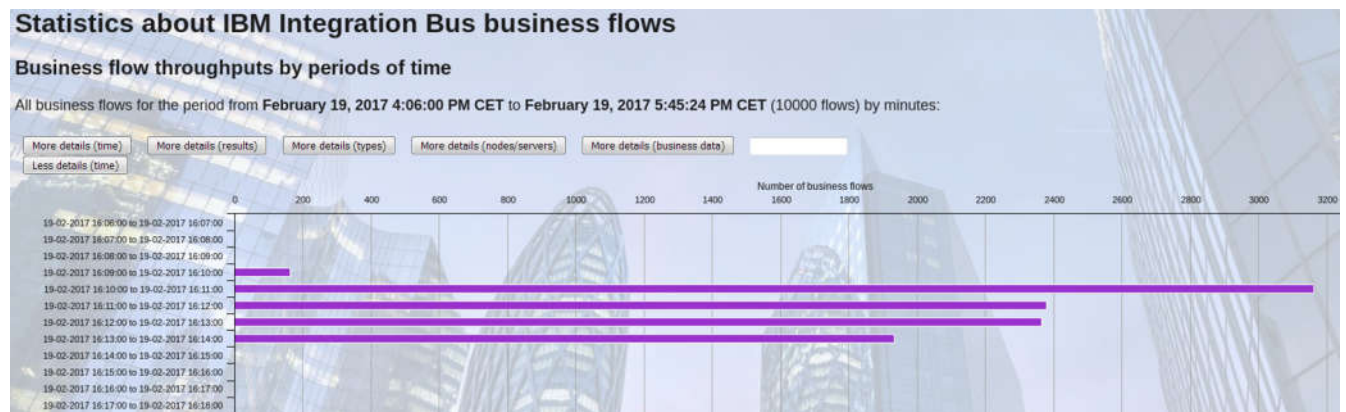
These statistics are always calculated from the business flow instances retrieved by the last query (the full list of results, not the small part of it displayed as a table).

The statistics are displayed as bar charts (built using the HighCharts JS framework). With any of these bar charts:

- By pointing the mouse to a bar, you can get details about this bar displayed in a balloon.
- At the bottom of the chart, there is a legend with several entries describing the various colors used to display bars in the chart. By clicking on one of the legend entries, you can hide all the corresponding bars from the chart. If you want to display the hidden bars again, just click again at the same place.

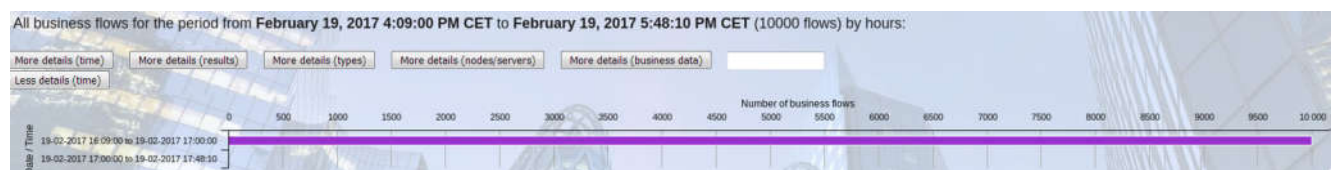
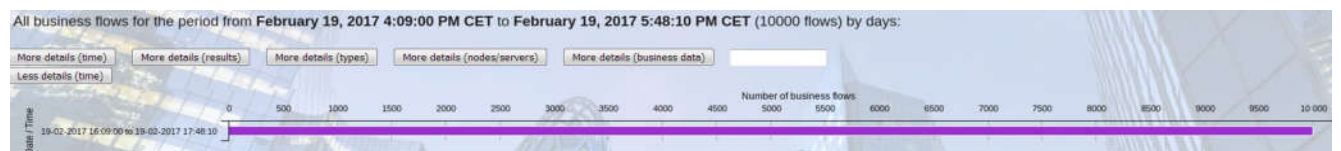
Statistics on throughputs by periods of time

The first kind of statistics displays statistics about the numbers of business flow instances which were received and processed by periods of time.

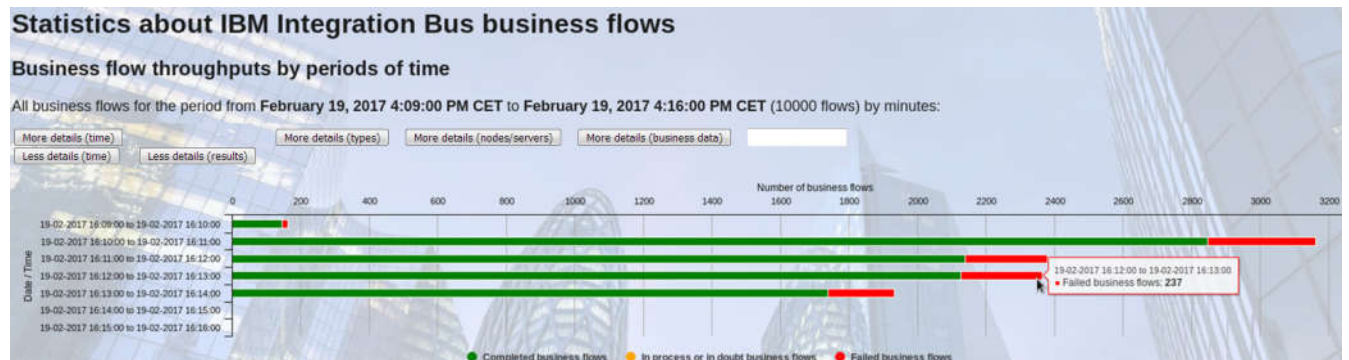


There are several actions on top of the bar chart to change the view:

- “More details (time)” and “Less details (time)” enable you to change the period of time, which can be seconds, minutes, hours, days, weeks, months or years.



- “More details (results)” lets you make a difference between the business flows instances which completed without error (displayed in green), those which ended in error (in red) and those which are still in process or in doubt (in orange).



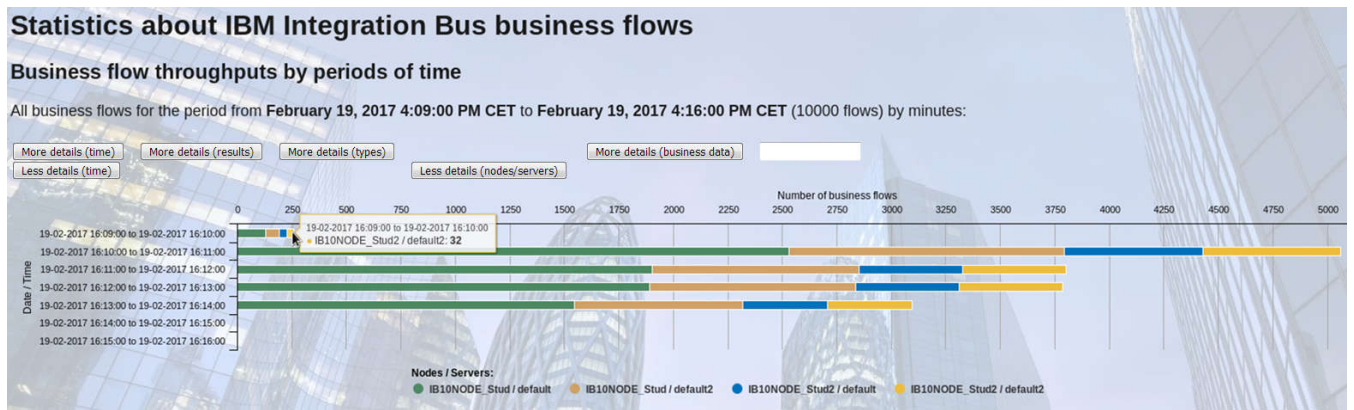
Clicking then on “Less details (results)” returns you to the base view (one single color).

- “More details (types)” lets you make a difference between the business flows instances belonging to different business flow types.



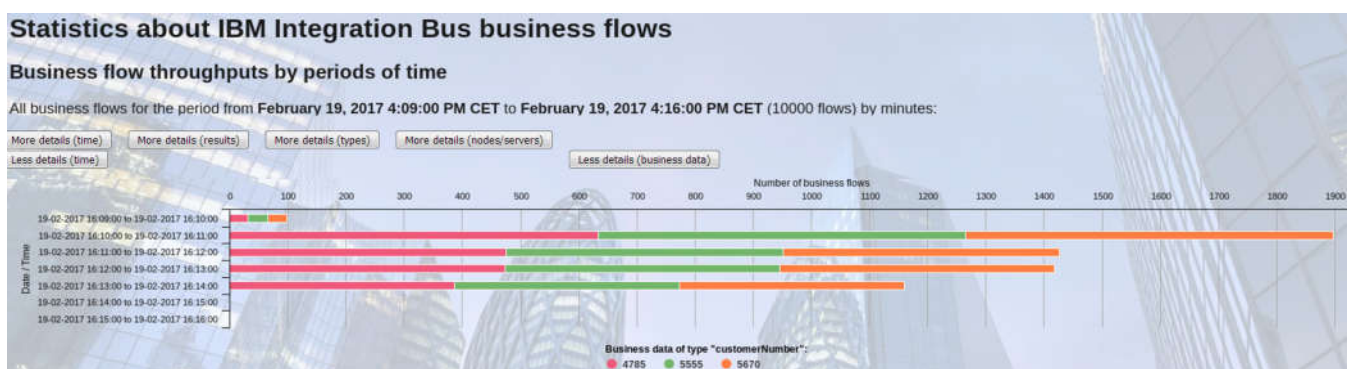
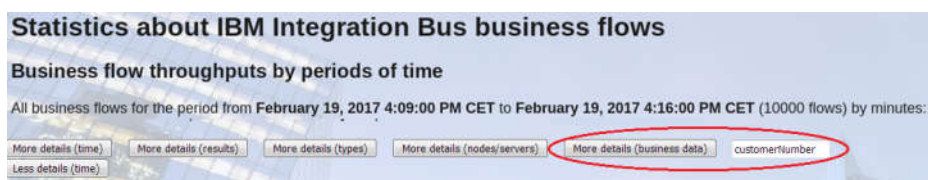
Clicking then on “Less details (types)” returns you to the base view (one single color).

- “More details (nodes/servers)” lets you make a difference between the business flows instances having executed on different node/server pairs. The first time which you click on this action, you get the throughputs splitted by nodes. If you click a second time on the same action, you get the throughputs splitted by nodes and servers.



Clicking then on “Less details (nodes/servers)” returns you to the previous view (split by nodes or no split).

- “More details (business data)” lets you make a difference between the business flows instances having executed by business data. The name of the business data do be used for the split must be entered into the field next. Simple or complex data can be used. In the case of a complexe data, its full path must be entered (for example “Details/Geography/Country/City”).



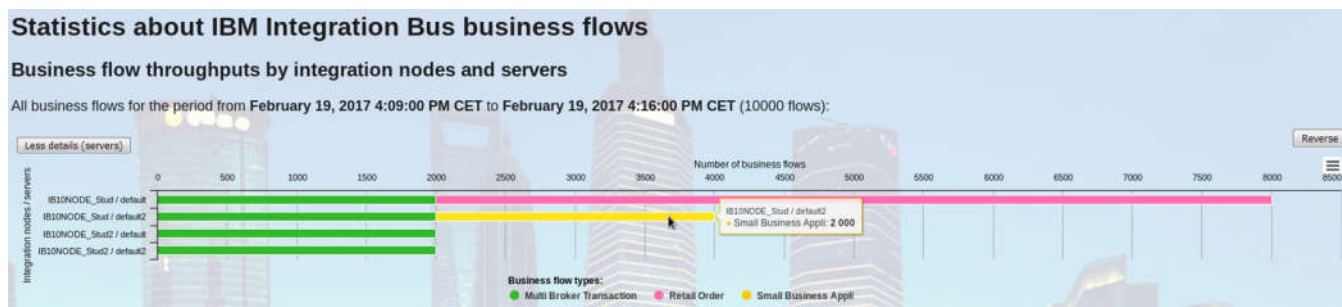
Clicking then on “Less details (business data)” returns you to the previous view (no split by data).

Statistics about throughputs by nodes/servers or by business flow types

The second kind of statistics displays statistics about the numbers of business flow instances which were received and processed by node/server pairs and by business flow types.

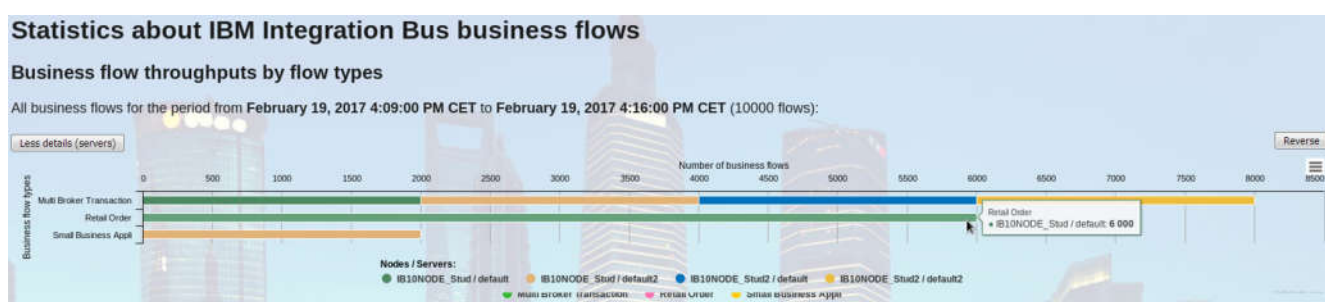
The first view which gets displayed presents the nodes as lines and the business flow types as different colors.

- By clicking on “More details”, you can have the servers also taken into account for the lines of the chart.



Clicking on “Less details (servers)” then returns you to the previous view.

- By clicking on “Reverse” (on the right side), you can switch to a display with business flow types represented as lines and nodes (or nodes/servers) represented as different colors.



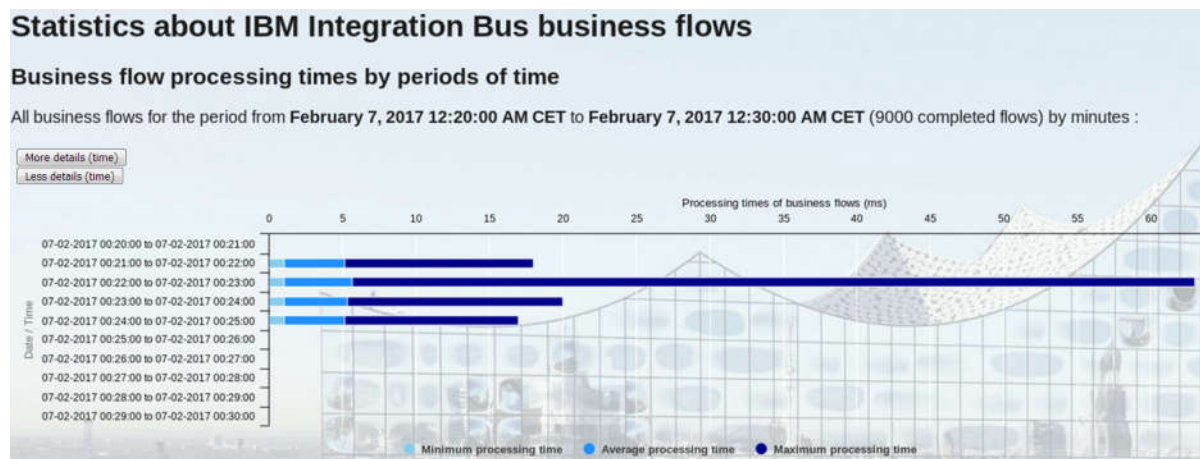
Statistics about processing times

The third and last kind of statistics displays statistics about the processing times of the business flow instances by periods of time. For a period of time, the following values are presented using different colors:

- Minimum processing time (in milliseconds) for the period (in light blue).
- Average processing time (in milliseconds) for the period (in medium blue).
- Maximum processing time (in milliseconds) for the period (in dark blue).

The three kinds of bars overlap each other, which means that one bar usually hides part of another. The following situations may occur:

- If you do not see the minimum processing time bar for a period, it means that the minimum processing time for that period was either 0 or a very small value (compared to the other values displayed on the chart).
- If you do not see the average processing time bar for a period, it means either that it is hidden behind the minimum processing time bar or that the average processing time for that period was 0 or a very small value.
- If you do not see the maximum processing time bar for a period, it means either that it is hidden behind the minimum processing time bar or the average processing bar or that the maximum processing time for that period was 0 or a very small value.



“More details (time)” and “Less details (time)” let you switch between different periods of time (seconds, minutes, hours, days, weeks, months or years) for the lines of the chart.

Appendix

Material provided

The following material is provided to install the asset and learn to use it faster:

- “IIBAdvMonMDBWeb.war” and “IIBAdvMonGUIWeb.war”: The files to be used to install the two parts of the asset (the Message Driven Bean part and the GUI part). The instructions to install these modules are provided in the chapter “Installing the asset modules on the application server”.
- “CreateAdvMonDB_DB2.sql”: A sample DB2 script for creating the DB2 tables and index needed by the asset (if DB2 is used).
- “CreateAdvMonDB_Oracle.sql”: A sample Oracle script for creating the Oracle tables and index needed by the asset (if Oracle is used).
- Three sample flows ready for monitoring with the asset:
 - “BTM_RETAIL_APP.zip” is a sample provided by IBM Integration Bus (slightly adapted for the asset). It includes a set of flows designed to work together. Instructions for installing and using this sample are provided in the IBM Integration Bus Toolkit: open the Tutorials gallery and select the “Business transaction monitoring” entry. Then follow the instructions provided by this entry but, at the “Import projects” stage, import the Project Interchange File (“BTM_RETAIL_APP.zip”) provided with the asset rather than the one provided by the Toolkit.

Here are the contents of 3 messages (also provided as XML files) which can be injected into this sample:

```
<btm_retail><orderId>2</orderId><requestType>orderItem</requestType><customerNumber>5555</customerNumber><customerName>ChocolateFan</customerName><itemName>Chocolate chip</itemName><flows></flows><requestTime></requestTime><processTime></processTime><responseTime></responseTime></btm_retail>
```

```
<btm_retail><orderId>2</orderId><requestType>orderItem</requestType><customerNumber>5670</customerNumber><customerName>CoffeeFan</customerName><itemName>ColumbianSupreme</itemName><flows></flows><requestTime></requestTime><processTime></processTime><responseTime></responseTime></btm_retail>
```

```
<btm_retail><orderId>3</orderId><requestType>orderItem</requestType><customerNumber>4785</customerNumber><customerName>TeaFan</customerName><itemName>CeylonBlackTea</itemName><flows></flows><requestTime></requestTime><processTime></processTime><responseTime></responseTime></btm_retail>
```

- “MultiBrokerTransactionApplication.zip” includes four flows (“MultiBrokerTransactionApplication1”, “MultiBrokerTransactionApplication2”, “MultiBrokerTransactionApplication3” and “MultiBrokerTransactionApplication4”) designed for working in sequence on the same message and able to execute on different integration servers, belonging to a common integration node or to different integration nodes.

When installing this sample, you will have to create the input and output queues used by these flows: “MQTQIn”, “MQTQIn2”, “MQTQInter”, “MQTQIn3”, “MQTQOut”:

- If one flow and the next are executed on the same node (either on the same server or not), the queue used to pass the message on can be a plain local queue.
- If one flow and the next are executed on different nodes, the queue used to pass the message on must be a remote queue.

If several nodes are used, you will also have to define a pair of channels and a transmission queue.

This is the contents of a message (also provided as an XML file) which can be injected into this sample:

```
<Test><Value>1</Value></Test>
```

- “Small BusinessApp.zip” includes a single flow which either succeeds (when the contents of the field “Test\Value” is neither null no equal to 0) or fails (when the contents of the field “Test\Value” is null or equal to 0).

When installing this sample, you will have to create the input and output queues used by the flow: “SBFInQ”, “SBFOutQ” and “SBFErrQ”.

These are the contents of two messages (also provided as an JSON files) which can be injected into this sample and which completes with success (the first one, because the “Value” field contains “1”) or failure (the second one, because the “Value” field contains “0”):

```
{"Test": {"Id": "12345", "Value": "1", "Details": {"Geography": [{"City": "Paris", "Country": "France"}, {"City": "London", "Country": "UK"}, {"City": "Berlin", "Country": "Germany"}, {"City": "Madrid", "Country": "Spain"}]}}
```

```
{"Test": {"Id": "12345", "Value": "0", "Details": {"Geography": [{"City": "Paris", "Country": "France"}, {"City": "London", "Country": "UK"}, {"City": "Berlin", "Country": "Germany"}, {"City": "Madrid", "Country": "Spain"}]}}
```

In order to preserve the resources of the planet, this document was entirely composed with the « Garamond » font (except the commands and the texts included in images). See: <http://edition.cnn.com/2014/03/27/living/student-money-saving-typeface-garamond-schools/>.

Of course, it is even better not to print this document at all.