



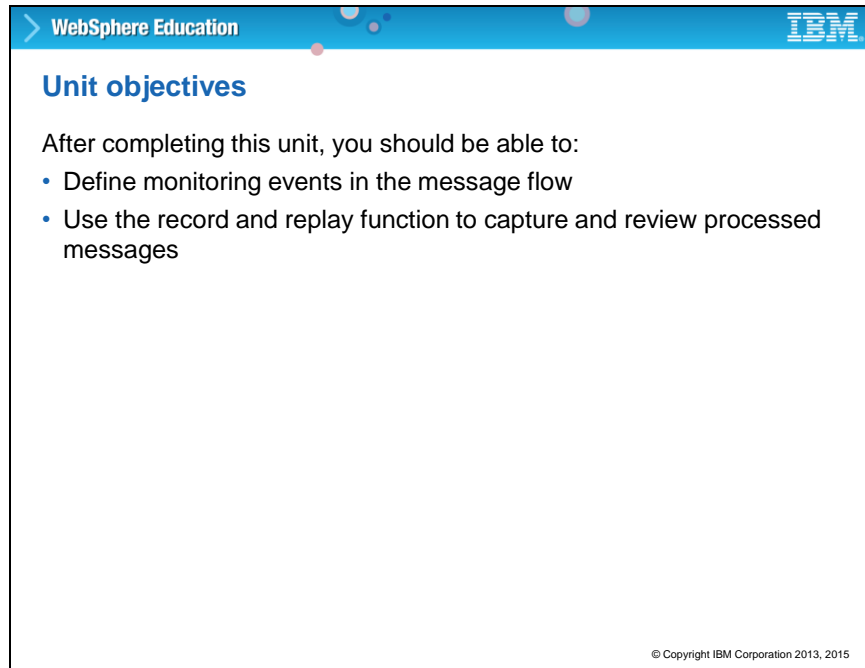
## Slide 1


> WebSphere Education 

# Monitoring message flow events



© Copyright IBM Corporation 2013, 2015  
Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

A presentation slide with a blue header bar containing the text 'WebSphere Education' and the IBM logo. The main content area is white and contains the title 'Unit objectives' in blue. Below the title, it states 'After completing this unit, you should be able to:' followed by a bulleted list of two items. The footer contains a small copyright notice.

> WebSphere Education 

### Unit objectives

After completing this unit, you should be able to:

- Define monitoring events in the message flow
- Use the record and replay function to capture and review processed messages

© Copyright IBM Corporation 2013, 2015

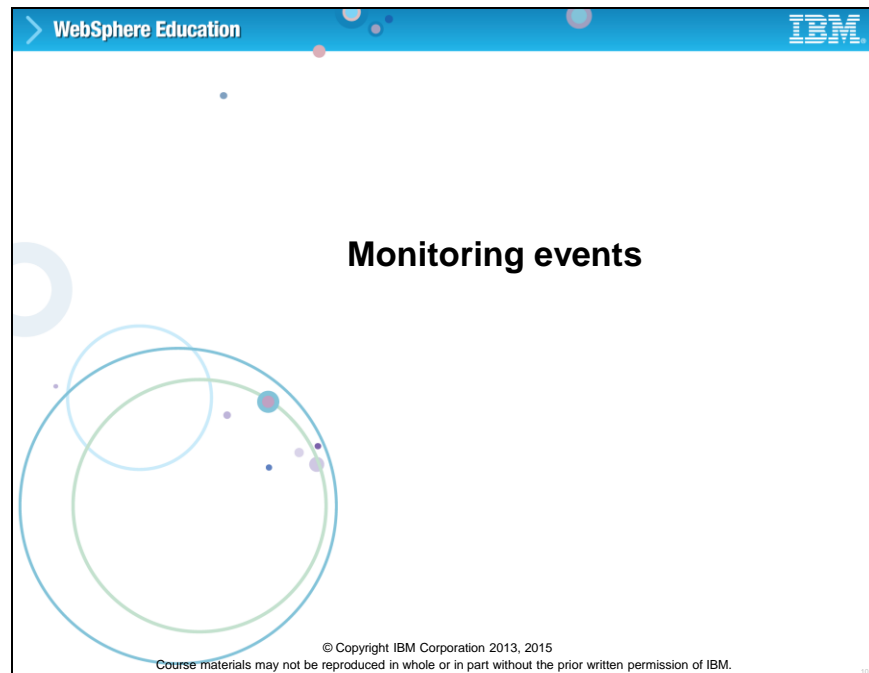
## Unit objectives

Integration nodes use a publish/subscribe broker to publish events in response to changes in configuration, state, or operational status. You can monitor these events by subscribing to the topics. For audit or problem determination purposes, you can record data to a database, then view it, and replay it. This unit describes the IBM Integration Bus tools that are available for monitoring message flow events and analyzing message data.

After completing this unit, you should be able to:



- Define monitoring events in the message flow
- Use the record and replay function to capture and review processed messages

## Slide 3



### Topic 1: Monitoring events

Message flows can be configured to emit events. The events can be read and used by other applications for transaction monitoring and auditing, and business process monitoring. This topic introduces event monitoring.

 WebSphere Education 

## Event monitoring and auditing

- Generates monitoring and audit events from message flows
  - Operationally enable, disable, and change event production in the Integration Toolkit Message Flow editor or by using commands
  - Events are published on well-known topics over IBM MQ transport or MQTT for multiple concurrent consumers
  - Events are optionally produced within same transaction sync point for optimum performance
- Provides noninvasive event monitoring profile
  - Configurable service overrides the monitoring properties of a message flow
  - Can be used to customize events without deploying the message flow again
- Integrates with IBM Business Monitor
  - Monitor and analyze key performance indicators
  - Automatic generation of monitor model
  - Comprehensive sample built-in

© Copyright IBM Corporation 2013, 2015

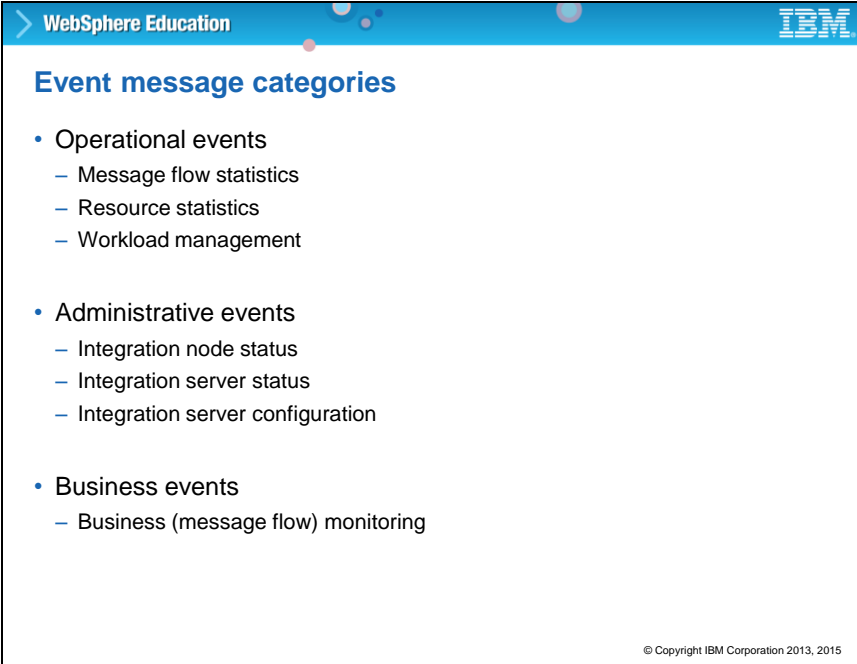
## Event monitoring and auditing

Message flows can be configured to emit events. Other applications can read and use these events for transaction monitoring, auditing, and business process monitoring. For example, it might be necessary to capture the first three fields of an input message at a certain point in the flow. Event generation can be configured by using the Integration Toolkit and commands.

Events are published over a publish/subscribe network as a well-defined topic. You can also generate an event at design-time or operationally at any point throughout the flow, containing the entire payload or any piece of it.

A monitoring profile configurable service can be used to customize events after a message flow is deployed, but without redeploying the flow.

You can also monitor message flows by using IBM Business Monitor. IBM Business Monitor is business activity monitoring software that provides a real-time view of your business processes and operations. It contains personalized business dashboards that calculate and display key performance indicators and metrics that are derived from business processes, business activity data, and business events. IBM Business Monitor can be used to trend the data, for example, analyzing the volume of changes, or breaking out the information by geographical area.



The slide is titled "Event message categories" and is part of a "WebSphere Education" presentation, as indicated by the header. It lists three main categories of events: Operational events, Administrative events, and Business events. Each category has a list of sub-points. The IBM logo is visible in the top right corner. A copyright notice for IBM Corporation 2013, 2015 is at the bottom right.



- Operational events
  - Message flow statistics
  - Resource statistics
  - Workload management
- Administrative events
  - Integration node status
  - Integration server status
  - Integration server configuration
- Business events
  - Business (message flow) monitoring

© Copyright IBM Corporation 2013, 2015

## Event message categories

Integration Bus event messages are categorized as operational events, administrative events, and business events. This unit focuses on the business events that a message flow can generate.

If you want to receive event messages for any of these events (which include viewing statistics in the IBM Integration web user interface), you must ensure that event publication is enabled. You also must ensure that either an MQTT or MQ publish/subscribe broker is configured.



### Configuring the publication of event messages

- Integration node event messages can be published to the following publish/subscribe brokers:
  - Built-in MQTT publish/subscribe broker
  - MQTT publish/subscribe broker on an external MQTT server
  - IBM MQ publish/subscribe broker
  - Combination of IBM MQ and MQTT publish/subscribe brokers
- Default publish/subscribe broker depends on the Integration Bus deployment:
  - If IBM MQ is not installed, Operational and Administrative event messages are published to the built-in MQTT publish/subscribe broker
  - If IBM MQ is installed but a queue manager is not specified on the integration node, Operational and Administrative event messages are published to the built-in MQTT publish/subscribe broker
  - If IBM MQ is installed and a queue manager is specified on the integration node, Operational, Administrative, and Business event messages are published to the IBM MQ publish/subscribe broker

© Copyright IBM Corporation 2013, 2015



## Configuring the publication of event messages

Integration Bus publishes operational, administration, and business events on known topics.

Integration node event messages can be published to the following publish/subscribe brokers:

- The built-in MQTT publish/subscribe broker
- An MQTT publish/subscribe broker on an external MQTT server
- An MQ publish/subscribe broker
- A combination of IBM MQ and MQTT publish/subscribe brokers

By default, operational events and administration events are published to the built-in MQTT publish/subscribe broker and business events are published to MQ. If you want to publish Business events to MQTT, you must enable the publication of the BusinessEvents group to the built-in MQTT publish/subscribe broker.

 WebSphere Education 

## Subscribing to business event messages

- To receive business event messages from message flows, subscribe to the following topic:  
`topicRoot/Monitoring/IntServer/messageFlow`
  - For an MQTT publish/subscribe broker, `topicRoot` is `IBM/IntegrationBus`
  - For an IBM MQ publish/subscribe broker, `topicRoot` is `$SYS/Broker`
- Use the `enabled` property of the `mqsichangeproperties` command to enable or disable the publication of event messages by a specified publish/subscribe broker (OperationalEvents, AdminEvents, and BusinessEvents)

Example: `mqsichangeproperties IBNODE -b pubsub -o BusinessEvents/MQTT -n enabled -v true`

© Copyright IBM Corporation 2013, 2015

## Subscribing to business event messages

An application can subscribe to topics that return messages about the configuration, state, or operational status of your integration node, integration server, or message flows.

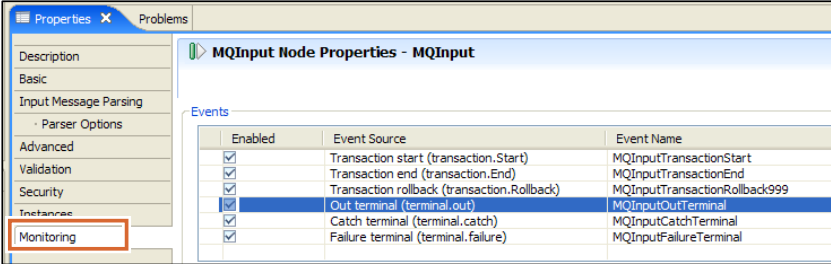
You can configure your message flow to emit event messages that can be used to support transaction monitoring and auditing, and business process monitoring.

When you use the built-in MQTT publish/subscribe broker, business monitoring events are not published by default. This example shows you how to use the `mqsichangeproperties` command to enable business event publication on the internal MQTT broker.

WebSphere Education IBM

## Message flow event monitoring configuration

- Every message flow node has a **Monitoring** tab to enable monitoring events
  - Transaction start, end, and rollback events
  - An event whenever a message is propagated from any terminal of any node
- Optionally, configure payload data, content style, identity, correlation, and sequencing data
- Configure event filters to specify conditions for event creation  
Example: Generate event when `msg.Price>100`



Enabled	Event Source	Event Name
<input checked="" type="checkbox"/>	Transaction start (transaction.Start)	MQInputTransactionStart
<input checked="" type="checkbox"/>	Transaction end (transaction.End)	MQInputTransactionEnd
<input checked="" type="checkbox"/>	Transaction rollback (transaction.Rollback)	MQInputTransactionRollback999
<input checked="" type="checkbox"/>	Out terminal (terminal.out)	MQInputOutTerminal
<input checked="" type="checkbox"/>	Catch terminal (terminal.catch)	MQInputCatchTerminal
<input checked="" type="checkbox"/>	Failure terminal (terminal.failure)	MQInputFailureTerminal

© Copyright IBM Corporation 2013, 2015

## Message flow event monitoring configuration

In a development environment, the monitoring requirements are defined in the Integration Toolkit. Every message processing node contains **Monitoring** properties that identify the event to generate and any additional information that the event message provides.


Developers can use the **Monitoring** properties to declare the event to see from a node, such as an input event, by checking the box next to the event. In the example, the MQ Input node has monitoring enabled at all points for a transaction: transaction start, transaction end, and transaction rollback.

It is also possible to declare sequencing information, correlating information, what the payload looks like, certain fields, and other information.

Event filters can be defined to generate events only when a certain condition applies, such as when a field in the message exceeds a specified value.

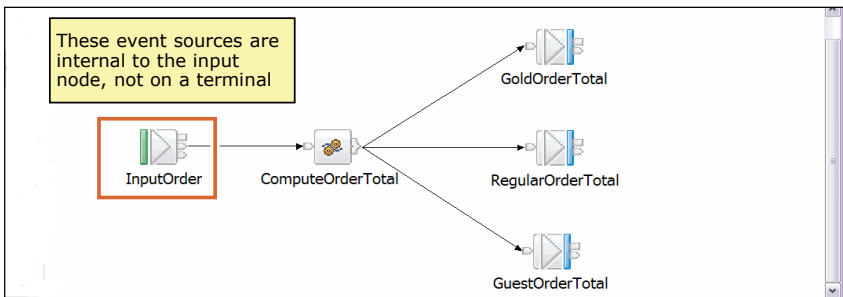


## Slide 9

WebSphere Education 

### Event monitoring support: Input node

These event sources are internal to the input node, not on a terminal



InputOrder → ComputeOrderTotal → GoldOrderTotal, RegularOrderTotal, GuestOrderTotal

Graph | User Defined Properties

Error Log | Tasks | Problems | Properties

Default Values for Message Flow Properties - TotalPurchaseOrderFlow

Description

Monitoring

Events

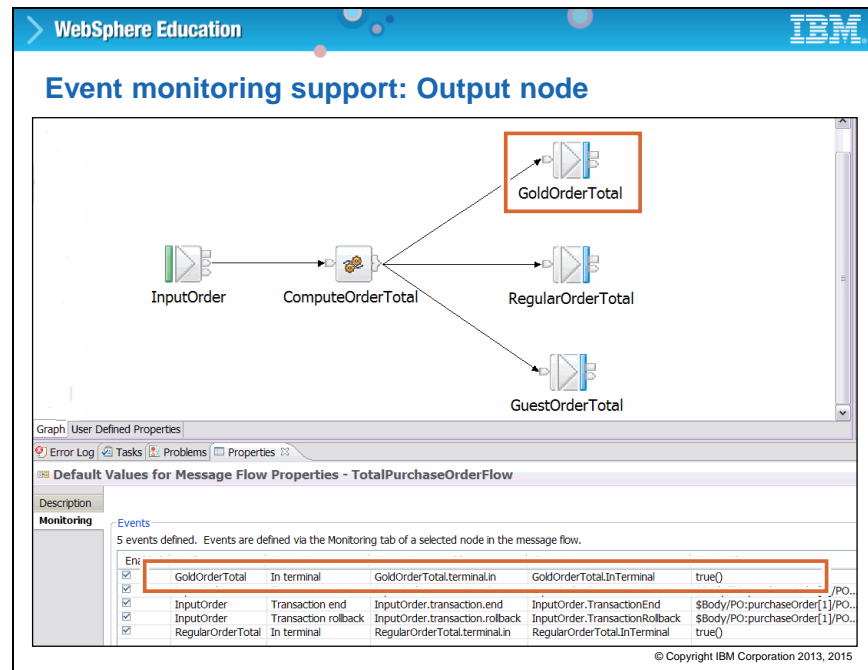
5 events defined. Events are defined via the Monitoring tab of a selected node in the message flow.

Enabled	Node	Event Source	Event Source Address	Event Name	Event Filter
<input checked="" type="checkbox"/>	GoldOrderTotal	In terminal	GoldOrderTotal.terminal.in	GoldOrderTotal.InTerminal	true()
<input checked="" type="checkbox"/>	InputOrder	Transaction start	InputOrder.transaction.start	InputOrder.TransactionStart	\$body/PO:purchaseOrder[1]/PC
<input checked="" type="checkbox"/>	InputOrder	Transaction end	InputOrder.transaction.end	InputOrder.TransactionEnd	\$body/PO:purchaseOrder[1]/PC
<input checked="" type="checkbox"/>	InputOrder	Transaction rollback	InputOrder.transaction.rollback	InputOrder.TransactionRollback	\$body/PO:purchaseOrder[1]/PC
<input checked="" type="checkbox"/>	RegularOrderTotal	In terminal	RegularOrderTotal.terminal.in	RegularOrderTotal.InTerminal	true()

© Copyright IBM Corporation 2013, 2015

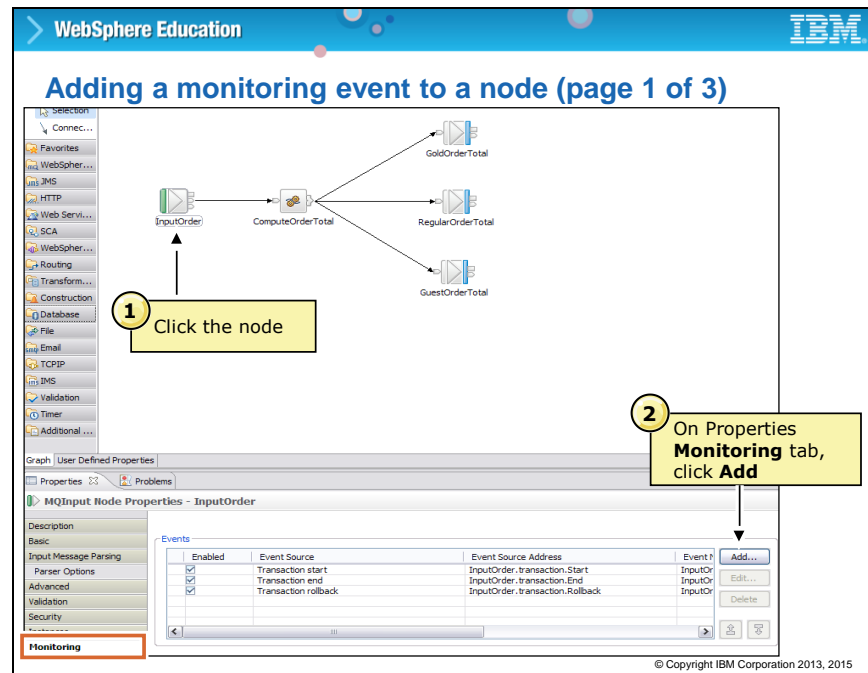
### Event monitoring support: Input node

This slide highlights the event definitions for the MQ Input node that is named **InputOrder**: transaction start, transaction end, and transaction rollback.



## Event monitoring support: Output node

This slide highlights the event that is defined for the MQ Output node that is named **GoldOrderTotal**. The only event that is defined in the node is the **In** terminal, which means that an event is emitted when the message reaches the **In** terminal of the node.

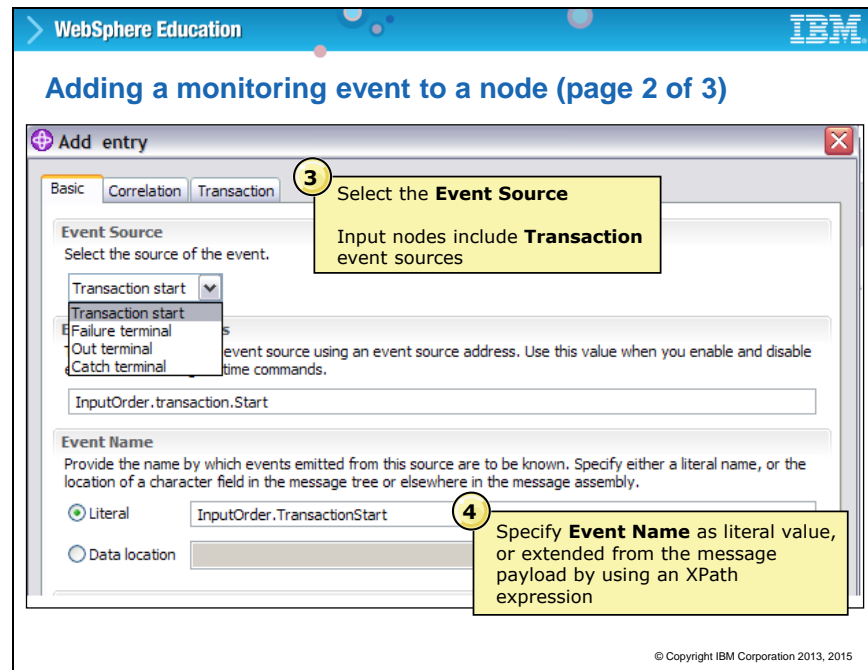


## Adding a monitoring event to a node (1 of 3)

This slide and the next series of slides show how to configure events in the Integration Toolkit.

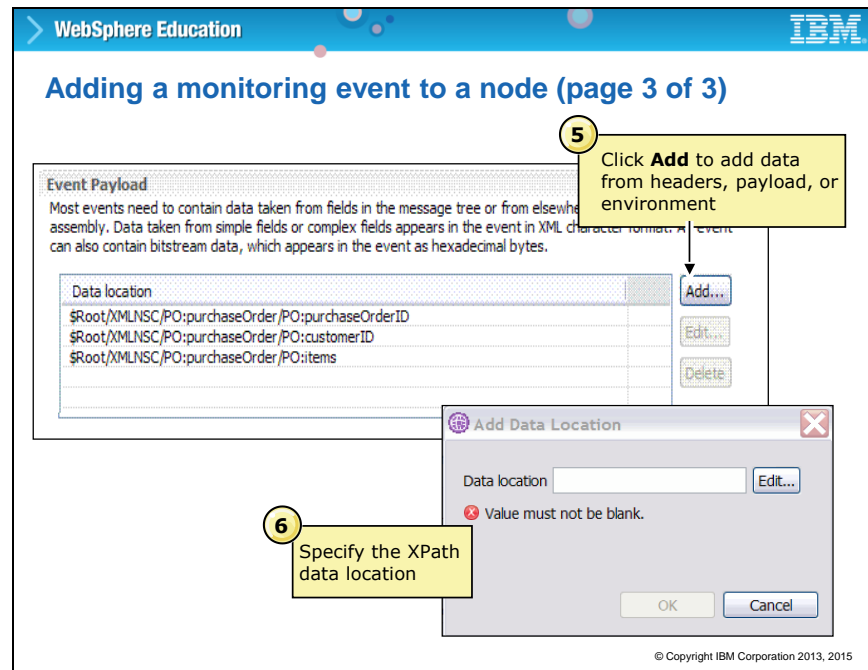
1. To add, edit, or delete an event for a node, click the node in the flow that generates the events, such as an input node. Click the **Monitoring** tab in the **Properties** view to show the event summary window.
2. To add an event, click **Add**.

You can also select any event in the summary and then click **Delete** to delete the event, or click **Edit**, to modify the event configuration.



### Adding a monitoring event to a node (1 of 3)

3. Select the event source from the **Event Source** list. The event source options depend on the node type and terminals. In the example that is shown here, the node that was selected was an input node with **Failure**, **Out**, and **Catch** terminals. The list contains an event for each output terminal. The list also contains an event for transaction start because the node is an input node.
4. The Integration Toolkit provides an event source literal name, also referred to as the event source address. It is the concatenation of the node name and the event. In this example, the node name is **InputOrder**. The event that is selected in the previous step was transaction start so the event source literal name is **InputOrder.TransactionStart**. The event source literal name is the name that is used to reference this event from a command. You have the option of overriding the Event Name with a literal value or an XPath expression that references an element in the message payload.



### Adding a monitoring event to a node (3 of 3)

The next steps are optional.

5. The **Event Payload** section identifies the payload information to include in the event message. If you want to include the message data with the event message, click **Add**, and then provide the XPath or ESQL path of the payload object in the **Add Data Location** window.
6. If you click **Edit** in **Add Data Location** window, the XPath builder opens with the message structure to assist with creating the XPath for the data.

Multiple XPath queries can be specified by repeating these steps.

WebSphere Education IBM

### Customizing a monitoring event: Bitstream data

**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/PO:purchaseOrder/PO:purchaseOrderID	
\$Root/XMLNSC/PO:purchaseOrder/PO:customerID	
\$Root/XMLNSC/PO:purchaseOrder/PO:items	

Add...  
Edit...  
Delete

☒ Include bitstream data in payload

Content: All  
Headers  
Body  
All

Encoding: base64Binary

1 Click to include bitstream data in the payload

2 Specify the bitstream content

© Copyright IBM Corporation 2013, 2015

### Customizing a monitoring event: Bitstream data

If you want an event to include the payload in the event message, click the **Include bitstream data in payload** check box in the **Event Payload** section. After the check box is selected, you can select the bitstream content as **headers only**, **body only**, or **all**. You can also select the encoding as **Cdata**, **base64 binary**, or **hex64 binary**.

Do not select **CData** for bitstream encoding when **Content** is set to **Headers** or **All** because headers can contain binary data. If you select **CData** for **Encoding**, set **Content** to **Body**.

**WebSphere Education** IBM

## Event filter

**Basic** | Correlation | Transaction

**Event Source**  
Select the source of the event.  
Transaction start

**Event Source Address**  
The broker identifies an event source using an event source address. Use this value when you enable and disable event sources using runtime commands.  
InputOrder.transaction.Start

**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: InputOrder.TransactionStart  
☐ 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.  
\$Root/XMLNSC/PO:purchaseOrder[1]/PO:customerType[1] != 'GUEST' Edit...

**Events**  
5 events defined. Events are defined via the Monitoring tab of a selected node in the message flow.

Enabled	Node	Event Source	Event Source Address	Event Name	Filter
<input checked="" type="checkbox"/>	GoldOrderTotal	In terminal	GoldOrderTotal.terminal.in	GoldOrderTotal.InTerminal	true()
<input checked="" type="checkbox"/>	InputOrder	Transaction start	InputOrder.transaction.Start	InputOrder.TransactionStart	\$Root/XMLNSC/PO:purchaseOrder[1]/PO:customerType[1] != 'GUE
<input checked="" type="checkbox"/>	InputOrder	Transaction end	InputOrder.transaction.End	InputOrder.TransactionEnd	\$Root/XMLNSC/PO:purchaseOrder[1]/PO:customerType[1] != 'GUE
<input checked="" type="checkbox"/>	InputOrder	Transaction rollback	InputOrder.transaction.Rollback	InputOrder.TransactionRollback	\$Root/XMLNSC/PO:purchaseOrder[1]/PO:customerType[1] != 'GUE
<input checked="" type="checkbox"/>	RegularOrderTotal	In terminal	RegularOrderTotal.terminal.in	RegularOrderTotal.InTerminal	true()

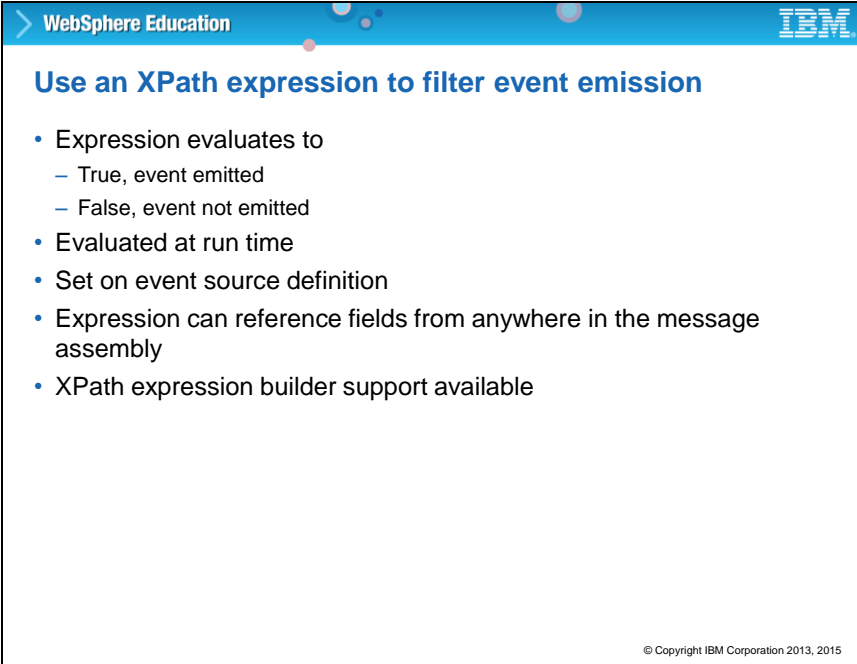
© Copyright IBM Corporation 2013, 2015

## Event filter


Integration Bus event filters can filter out events that do not match a business rule. So Integration Bus can filter events rather than emitting them to IBM Business Monitor, for example, and filtering them there. This process improves performance by eliminating an unnecessary event.

The event filter can be set to a number, Boolean, or string XPath expression that evaluates to a Boolean “true” or “false”. If the expression evaluates to “true”, then events are emitted. If the expression evaluates to “false”, then events are not emitted. The default setting is “true”.

As shown in the example, event filters are configured in the **Event Filters** section that is displayed when you add or edit events in the Integration Toolkit. When you define a filter, you can click **Edit** in the **Event Filter** section to start the XPath expression builder. The event filters for each event are shown on the **Monitoring** tab **Summary** table. In the example, the event filter for the node **GoldOrderTotal** In terminal event source is set to “true”. This value is the default, which causes an event to always be emitted.



The slide features a blue header bar with the text 'WebSphere Education' on the left and the IBM logo on the right. The main content area is white with a blue title and a bulleted list. A small copyright notice is at the bottom right.

> WebSphere Education 

### Use an XPath expression to filter event emission

- Expression evaluates to
  - True, event emitted
  - False, event not emitted
- Evaluated at run time
- Set on event source definition
- Expression can reference fields from anywhere in the message assembly
- XPath expression builder support available

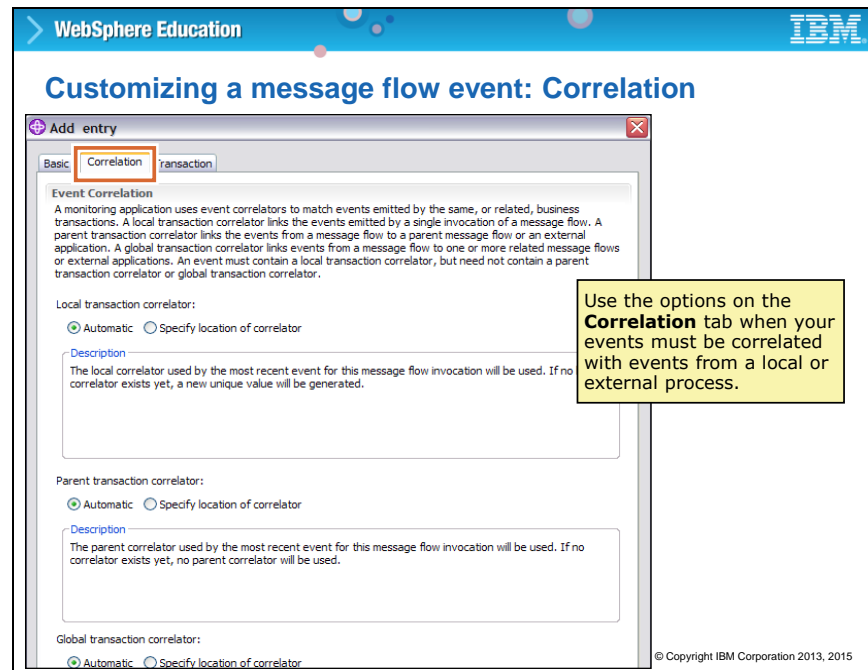
© Copyright IBM Corporation 2013, 2015

### Use an XPath expression to filter event emission

With Integration Bus filters, you can specify an XPath or ESQL query that is evaluated at run time to control whether the event should be produced.



Filters can be configured in the Integration Toolkit or in the exported XML file that contains event information.





## Customizing a message flow event: Correlation

Event configuration includes a **Correlation** tab. This option is used to associate events. Correlation provides three levels of transaction correlator: local, parent, and global.

 WebSphere Education 

### Transaction correlators

- A monitoring application uses event correlators to match events that are emitted by the same, or related, business transactions
- Integration Bus developer specifies location in the logical message tree for each correlator on the **Correlation** tab when defining an event in the Integration Toolkit
  - Local transaction correlator links the events that a single invocation of a message flow emits  
Example: `Customer ID`
  - Parent transaction correlator links the events from a message flow to a parent message flow or an external application  
Example: `Order ID`
  - Global transaction correlator links events from a message flow to one or more related message flows or external applications  
Example: `Stock Amount`
- An event must contain a Local transaction correlator, but need not contain a Parent transaction correlator or Global transaction correlator

© Copyright IBM Corporation 2013, 2015

## Transaction correlators

Local transaction correlator groups all monitoring events that are related to a specific message. You can select Automatic to automatically generate a correlation ID, or you can specify an ID. If you select Automatic, Integration Bus reuses the query results from the first evaluation of the XPath query.

Parent transaction correlator and Global transaction correlator are business-related and correlate events between different messages. For example, if you are processing invoice line items as a transaction, you can correlate to get the invoice number in the parent, then get the individual line item events. Using the same example, you can use global to identify the purchase order number and pick up the invoice events from all invoices that reference the purchase order.

WebSphere Education
IBM

## Unit of work

Edit entry

Basic
Correlation
Transaction

**Event Unit Of Work**

Choose whether the emission of monitoring events by a message flow is coordinated with the message flow transaction, or is in an independent unit of work, or is not in a unit of work. In general, if you want an event to be emitted only if the message flow transaction commits, select 'Message Flow'; if you want an event to be emitted regardless of whether the message flow transaction commits or rolls back, select 'None'; if you want a group of events to be emitted together regardless of whether the message flow transaction commits or rolls back, select 'Independent'.

☒ Message flow
☐ Independent
☐ None

**Select event unit of work**

**Description**

The event, and all other events with this setting, will be emitted only if the message flow commits its unit of work successfully.

- Use the options on the **Transaction** tab to control the emission of monitoring events by a message flow by selecting one of the following options:
  - Message flow:** Events are published based on success message flow
  - Independent:** Groups of events are published together regardless of success or failure of message flow
  - None:** Events are always published

© Copyright IBM Corporation 2013, 2015



## Unit of work

In Integration Bus, you can choose how event emission is coordinated by specifying a unit of work on the event monitoring **Transaction** tab.

When a message is processed, the updates are included in a unit of work that is referred to as the **Message flow** unit of work. It is committed if the message processing is successful, and it is rolled back if it fails.

The **Independent** unit of work is a separate unit of work that is created and committed regardless of whether the message is processed successfully or not. Use this option for events, such as those related to error paths, that must be published even if the flow fails.

If you do not want a monitoring event to be included in any unit of work, choose the **None** option.

 WebSphere Education 

### Message flow event monitoring profiles

- Configurable service that is used to customize events after a message flow was deployed, but without redeploying the flow  
Name: `DefaultMonitoringProfile`  
Property: `profileProperties`      Name of the monitoring profile
- An XML document specifies the event sources in a message flow that emits events, and the properties of those events
- Must conform to XML schema file `MonitoringProfile.xsd`



© Copyright IBM Corporation 2013, 2015

## Message flow event monitoring profiles

You or an administrator can use a monitoring profile configurable service to customize events after a message flow is deployed without redeploying the flow.

A monitoring profile is an XML file that lists the event sources in the message flow, and defines the properties of each event. A monitoring profile also gives an administrator the ability to change monitoring events that were configured in the message flow node Monitoring properties without changing or redeploying the message flow.

If the deployed message flow has event monitoring properties that are configured by using the Integration Toolkit, use the `mqsiexportflowmonitoring` command to create the equivalent monitoring profile XML file for the message flow. Use this profile as a starting point for creating other monitoring profiles.



## Example: Message flow event monitoring profile

Example: Include two simple fields from a message

```
<p:monitoringProfile p:version="2.0">
  <p:eventSource p:eventSourceAddress="MQInput.terminal.out">
    <p:applicationDataQuery>
      <p:simpleContent p:dataType="integer" p:name="InvoiceNumber">
        <p:valueQuery p:queryText="$Body/invoice/invoiceNo"/>
      </p:simpleContent>
      <p:simpleContent p:dataType="string" p:name="BatchID">
        <p:valueQuery p:queryText="$Body/batch/batchNo"/>
      </p:simpleContent>
    </p:applicationDataQuery>
  </p:eventSource>
</p:monitoringProfile>
```

© Copyright IBM Corporation 2013, 2015

### Example: Message flow event monitoring profile

This slide contains an example of an event monitoring profile XML file.

The profile in the example captures the invoice number and batch ID from the body of the logical message. The message is captured at the MQ Input node Out terminal (the event source).

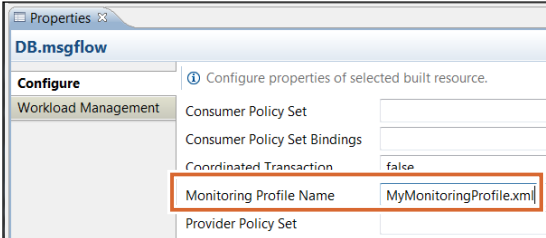
WebSphere Education

IBM

### Apply a message flow monitoring profile to a BAR file

Use the Integration Toolkit BAR File editor to apply a monitoring profile to one or more message flows:

1. Double-click the BAR file to open the BAR File editor
2. Click the **Manage** tab
3. Select the message flow to display the **Properties** view
4. In the **Monitoring Profile Name** field, enter the name of a monitoring profile
5. Save the BAR file
6. Deploy the BAR file
7. Activate monitoring for the flow with `mqsichangeflowmonitoring`



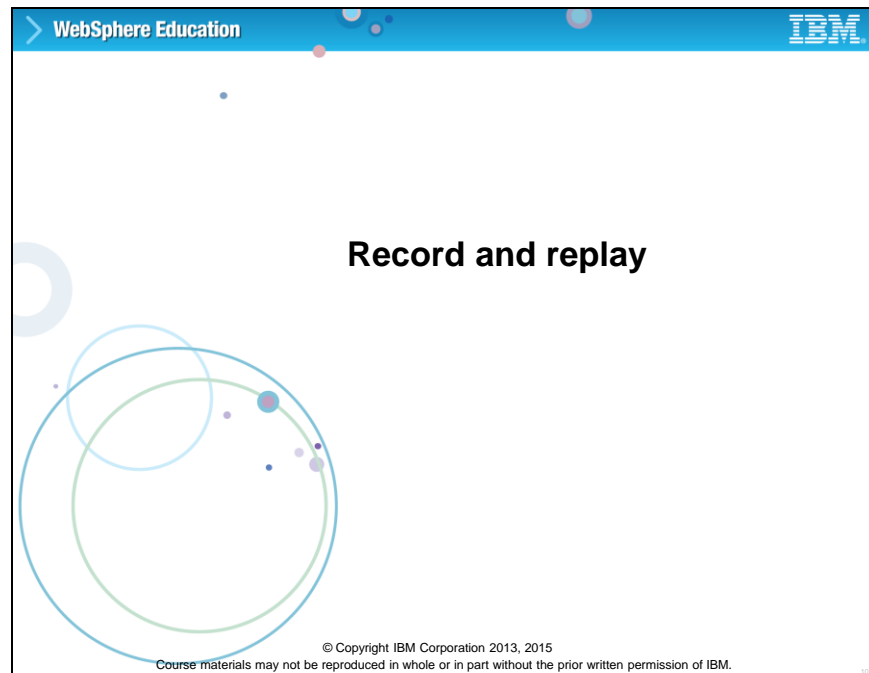
© Copyright IBM Corporation 2013, 2015

### Apply a message flow monitoring profile to a BAR file

A monitoring profile can be applied by modifying the BAR file in the Integration Toolkit BAR file editor.

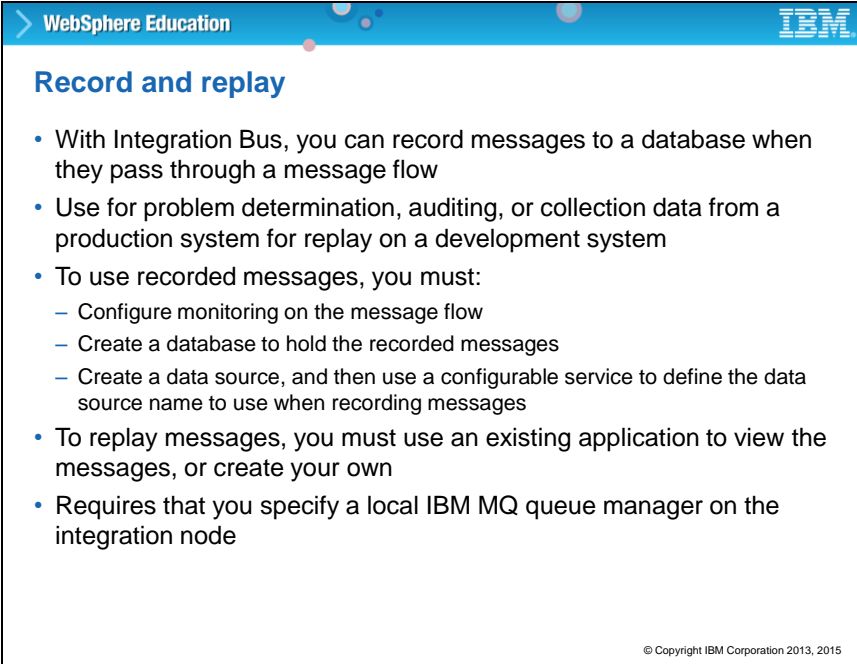
To apply a monitoring profile to a message flow in the BAR file editor, open the BAR file and go to the **Manage** tab. Select the message flow to display the message flow properties. In the **Monitoring Profile Name** field, enter the name of a monitoring profile.

After you save and deploy the BAR file, you must activate monitoring for the flow by using the `mqsichangeflowmonitoring` command.



## Topic 2: Record and replay

If you configured your message flow to emit event messages, you can record and replay in-flight data. This topic describes how to set up the Integration Bus environment for record and replay.



The slide is titled "Record and replay" and is part of a "WebSphere Education" presentation. It contains a bulleted list of instructions for using the record and replay feature. The list includes: recording messages to a database, using the feature for problem determination or auditing, requirements for using recorded messages (configuring monitoring, creating a database, and defining a data source), requirements for replaying messages (using an existing application or creating one), and the requirement to specify a local IBM MQ queue manager. The IBM logo is in the top right corner, and the copyright notice "© Copyright IBM Corporation 2013, 2015" is at the bottom right.

WebSphere Education

## Record and replay

- With Integration Bus, you can record messages to a database when they pass through a message flow
- Use for problem determination, auditing, or collection data from a production system for replay on a development system
- To use recorded messages, you must:
  - Configure monitoring on the message flow
  - Create a database to hold the recorded messages
  - Create a data source, and then use a configurable service to define the data source name to use when recording messages
- To replay messages, you must use an existing application to view the messages, or create your own
- Requires that you specify a local IBM MQ queue manager on the integration node

© Copyright IBM Corporation 2013, 2015

## Record and replay

While monitoring and auditing is restricted to capturing data at specific endpoints only, record and replay captures and manages message data as it is processed. The record and replay function gives you the ability to capture data that contains errors, correct it, and then resubmit it for processing.

At a high level, the following operations are the primary operations that are involved in record and replay:

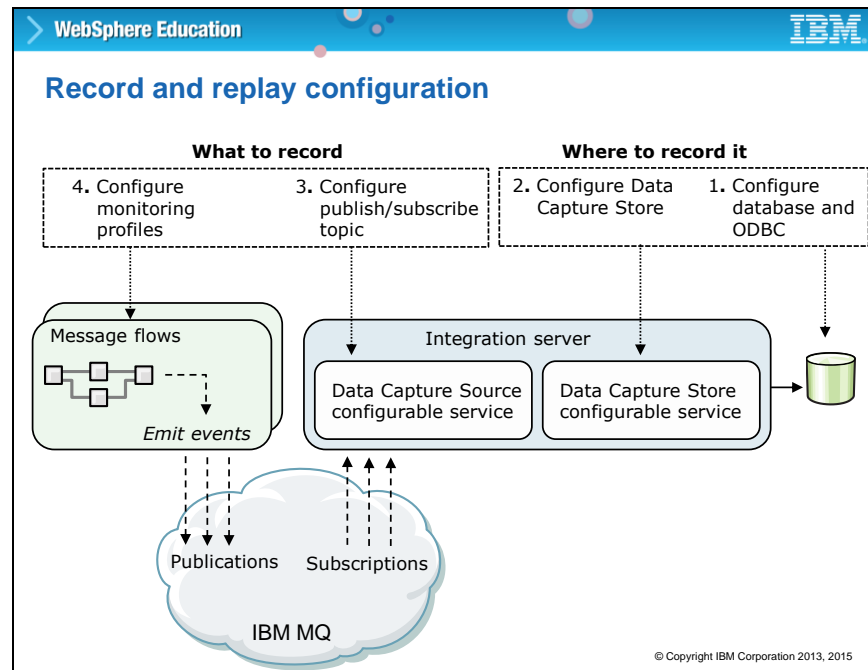
- Some type of trigger event occurs that causes message details to be enqueued. For example, some type of processing error occurs in a message flow.
- The integration node records the message in a DB2, Oracle, or Microsoft SQL Server database.
- Using Integration Bus and the Integration web user interface, you can query saved messages.

If authorized, you can replay messages to the same or a different source. For example, you can resubmit the corrected message to MQ for reprocessing.

If you want to control users' ability to record, view, and replay data, you must enable integration node administration security and also configure security for data capture.



Record and replay use MQ publish/subscribe, which requires that the integration node is associated with an MQ queue manager.



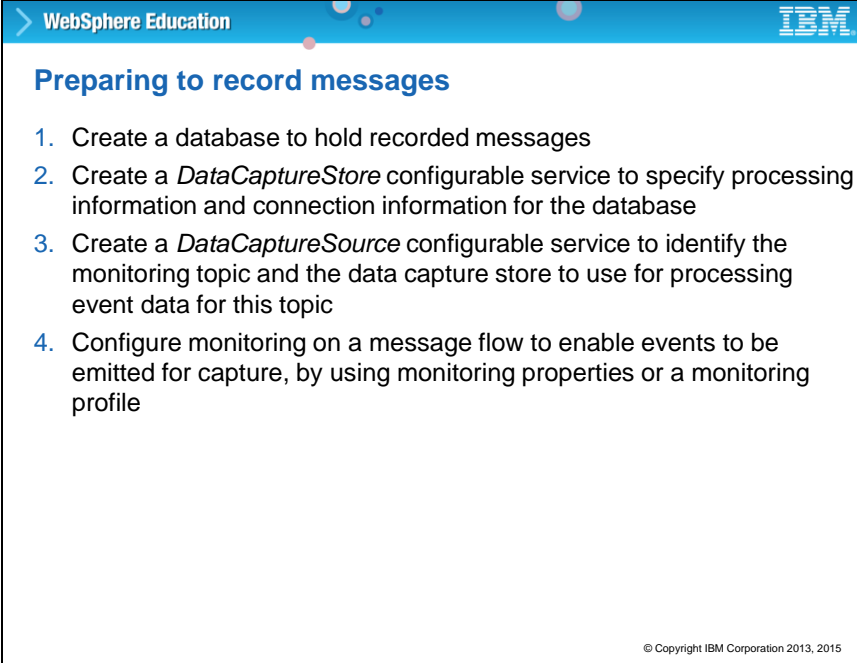
## Record and replay configuration

This slide summarizes the configuration requirements for record and replay.

Record and replay require an external database to store the transactions in a Data Capture Store. So the first steps are to define the database table, and the connection to the database from Integration Bus.

Record and replay also use the MQ publish/subscribe broker to subscribe to the event monitoring topics and capture the event messages. To identify the source of the data, create a DataCaptureSource configurable service.

These steps are described in detail next.



The slide is titled "Preparing to record messages" and is part of a WebSphere Education presentation. It contains a numbered list of four steps for setting up message recording. The IBM logo is visible in the top right corner of the slide frame.

1. Create a database to hold recorded messages
2. Create a *DataCaptureStore* configurable service to specify processing information and connection information for the database
3. Create a *DataCaptureSource* configurable service to identify the monitoring topic and the data capture store to use for processing event data for this topic
4. Configure monitoring on a message flow to enable events to be emitted for capture, by using monitoring properties or a monitoring profile

© Copyright IBM Corporation 2013, 2015

## Preparing to record messages

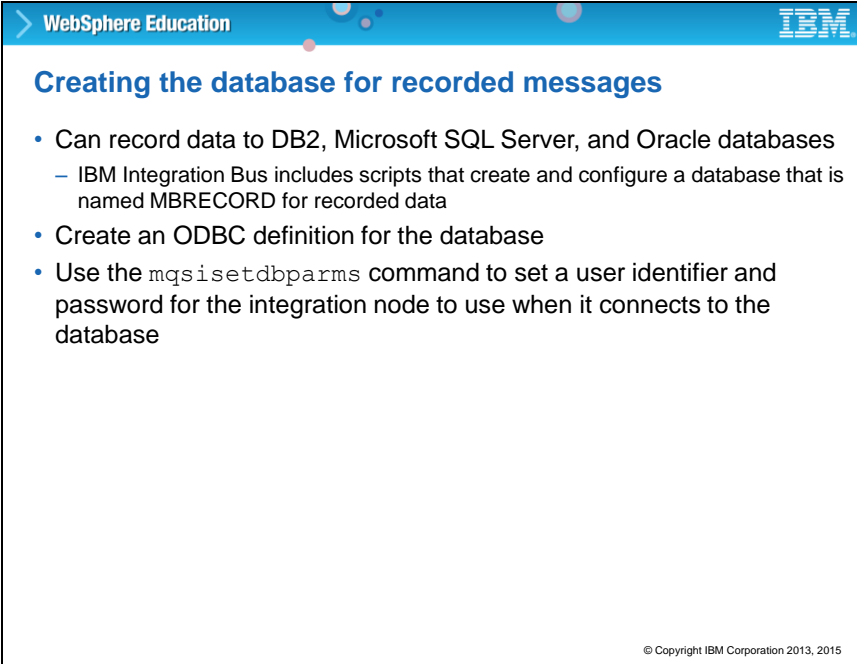
The slide lists the steps that must be completed before Integration Bus can record messages.

First, you must specify the database and create the database table to hold the recorded messages. You must also create an ODBC connection for the database.


Next, create a *DataCaptureStore* configurable service that identifies the database that holds the messages.

Then, create a *DataCaptureSource* configurable service to specify the monitoring topic that identifies the messages flows from which your data comes. The *DataCaptureSource* configurable service identifies the monitoring topics and data capture store. Multiple instances of the *DataCaptureSource* configurable service can use the same *DataCaptureStore* configurable service.

Finally, ensure that event monitoring is configured and enabled for the message flow.



The slide features a blue header bar with the text 'WebSphere Education' on the left and the IBM logo on the right. The main title 'Creating the database for recorded messages' is in blue. Below it, a bulleted list describes the process. The footer contains a small copyright notice.

> WebSphere Education 

### Creating the database for recorded messages

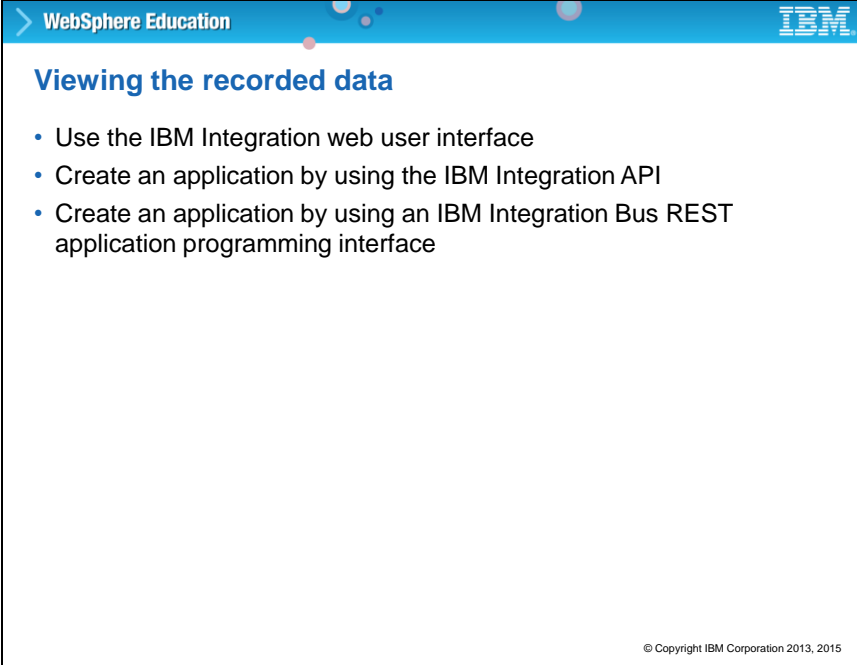
- Can record data to DB2, Microsoft SQL Server, and Oracle databases
  - IBM Integration Bus includes scripts that create and configure a database that is named MBRECORD for recorded data
- Create an ODBC definition for the database
- Use the `mqsisetdbparms` command to set a user identifier and password for the integration node to use when it connects to the database

© Copyright IBM Corporation 2013, 2015


## Creating the database for recorded messages

You can record event messages to DB2, Microsoft SQL Server, and Oracle databases.

A script is provided with Integration Bus that you can use to create the database and database tables. The script creates a database that is called MBRECORD with a default schema. You can run this script unmodified, or you can customize it.



The slide is titled "Viewing the recorded data" and is part of a "WebSphere Education" presentation. It features a blue header with the "WebSphere Education" text and the IBM logo. The main content is a bulleted list of three methods to view recorded data. The slide is framed by a thin black border.

> WebSphere Education 

### Viewing the recorded data

- Use the IBM Integration web user interface
- Create an application by using the IBM Integration API
- Create an application by using an IBM Integration Bus REST application programming interface

© Copyright IBM Corporation 2013, 2015

### Viewing the recorded data

You can use the Integration web user interface to review the recorded data, update, and requeue the message, or delete the message.

You can also use the IBM Integration API or the IBM Integration REST API to write your own message viewer application to fit your requirements.

WebSphere Education IBM

## IBM Integration web interface: Data viewer tab

**Data Capture Store**

Data viewer ▶ Replay list

Trades ▶ Mark for replay ▶ Customize ▶ Filter

*No filter applied*

	Event time	Local Transaction ID	Parent Transaction ID	Global Transaction ID	Data	Errors	Event name
<input type="checkbox"/>	2013-05-31 13:41:35.972	CG123456	BNY347290	IBM \$1000		-	
<input type="checkbox"/>	2013-05-31 13:41:35.834	CG123456	BNY347290	IBM \$1000		-	
<input type="checkbox"/>	2013-05-31 13:41:35.967	CG123456	BNY347290	IBM \$1000		-	
<input type="checkbox"/>	2013-05-31 13:41:44.661	CR100200-A	BNY809092	APPL \$500		-	Deciding customer type
<input type="checkbox"/>	2013-05-31 13:41:50.250	GU123456	BNY348475	MSFT \$5000		-	Deciding customer type
<input type="checkbox"/>	2013-05-31 13:41:50.251	GU123456	BNY348475	MSFT \$5000		-	Decision: Guest customer
<input type="checkbox"/>	2013-05-31 13:41:35.773	CG123456	BNY347290			-	Trade instruction received
<input type="checkbox"/>	2013-05-31 13:41:44.659	CR100200-A	BNY809092			-	Trade instruction received
<input type="checkbox"/>	2013-05-31 13:41:50.250	GU123456	BNY348475			-	Trade instruction received

© Copyright IBM Corporation 2013, 2015

## IBM Integration web interface: Data viewer tab

This slide shows an example of the record and replay Data viewer in the Integration web interface.

The Data viewer table contains the transactions and monitoring events. You can customize the table headings so that they are more descriptive and relevant to a business analyst that might be viewing this data.

To change the name of each column, double-click the column name, and then enter another name. These changes are stored in the Integration Registry, and are retained uniquely for each data capture store.

You can also limit the data that is displayed in the Data viewer by using the Filter function.

WebSphere Education
IBM

## Replaying messages

1. Select the Data Capture Store that contains the data to replay.
2. Select the messages to replay by clicking the check box next to each message.
3. Click **Mark for replay** to add messages to the replay list.
4. Click the **Replay list** tab.
5. Select the destination for the messages.
6. To replay all messages, click **Replay all**.  
To replay a single message, click the **Replay** icon in the row.

The screenshot shows the 'Data Capture Store' interface. It has two tabs: 'Data viewer' and 'Replay list'. The 'Data viewer' tab is active, showing a table of messages. A 'Mark for replay' button is highlighted with a red box. The table has columns: Event timeX, Customer number, Trade number, Stock / Trade amount, Data, and Error. The first two rows are selected with checkboxes.

Event timeX	Customer number	Trade number	Stock / Trade amount	Data	Error
2013-01-02 11:16:52.598	CG123456	BNY347290			
2013-01-02 11:16:52.598	CG123456	BNY347290			
2013-01-02 10:53:36.920	GU123456	BNY348475	MSFT \$5000		

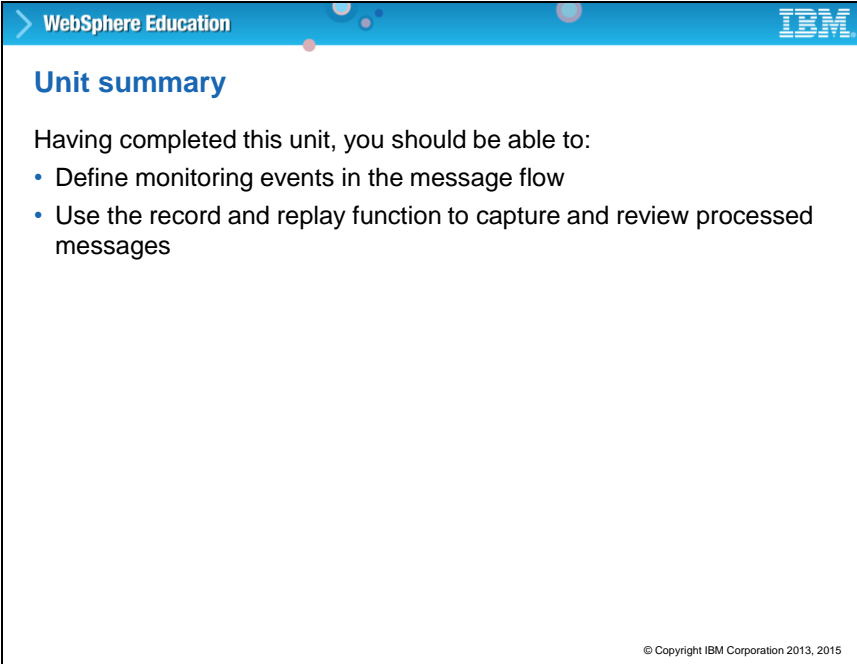
© Copyright IBM Corporation 2013, 2015


## Replaying messages

The slide lists the steps for replaying messages by using the Integration web user interface.

Each row that is displayed represents a recorded message. To sort these rows into a particular order, click a column heading.

You can also use a filter to help find rows in which you are interested. Filtering is case-sensitive and allows the use of wildcards. To search for an exact match, enclose the search string in quotation marks. You can also search for a substring, and you can use an asterisk to match zero or more characters.

The slide features a blue header bar with the text 'WebSphere Education' on the left and the IBM logo on the right. The main content area is white with a blue title 'Unit summary'. Below the title, it states 'Having completed this unit, you should be able to:' followed by a bulleted list of two items. At the bottom right, there is a small copyright notice.

> WebSphere Education 

### Unit summary

Having completed this unit, you should be able to:

- Define monitoring events in the message flow
- Use the record and replay function to capture and review processed messages

© Copyright IBM Corporation 2013, 2015

### Unit summary

Integration nodes use a publish/subscribe broker to publish business events. For audit or problem determination purposes, you can record events to a database, view them, and replay them. This unit described the IBM Integration Bus tools that are available for monitoring message flow events and analyzing message data.

Having completed this unit, you should be able to:

- Define monitoring events in the message flow
- Use the record and replay function to capture and review processed messages