

IBM WEBSPHERE BUSINESS MONITOR 6.2 – LAB EXERCISE

# WebSphere Business Monitor V6.2 - Clips and Tacks

## Business activity monitoring using XSD style events

What this exercise is about .....	2
Changes from the previous version .....	2
Lab requirements .....	2
What you should be able to do .....	2
Introduction .....	3
Part 1: Create the monitor model.....	5
Part 2: Publish the model to the server.....	56
Part 3: Configure action manager for business situation events .....	58
Part 4: Run events to exercise the model.....	64
Part 5: Create a dashboard.....	65
What you did in this exercise .....	68
Appendix 1 – Import the solution into the monitor model editor .....	69

---

## What this exercise is about

The objective of this lab is to show you how to build a monitor model using XSD event definitions in Rational Application Developer or WebSphere Integration Developer, deploy it to WebSphere Business Monitor and then view your monitored data on the Monitor dashboards.

This lab will show you business activity monitoring (BAM) which involves event based monitoring. With BAM, your monitored application can run anywhere, and submit events to the Monitor server so that you can view monitored data in the dashboard. Typically, you identify the events that will be created by the application, and then create a monitor model that represents the monitored data that you want to collect from the events.

## Changes from the previous version

This lab has been modified for the 6.2 release. Here are the major changes:

- Updated the dashboard section to show how to use business space.

This lab has been modified significantly for the 6.1 release. Here are the major changes:

- In the monitor model, you can reference other KPIs when defining the calculation for a given KPI.
- KPIs are now based on a metric and aggregation function, and not based on cube measures, so the dimensional model is now simplified.
- Deploying the model has been dramatically simplified, so this is a very short process now.
- LDAP is not required, so now you can test alerts without installing LDAP.
- Portal is no longer a requirement, so configuration of a dashboard is simplified.

## Lab requirements

List of system and software required for the student to complete the lab:

- Rational Application Developer V7.5 or WebSphere Integration Developer V6.2.
- WebSphere Business Monitor V6.2 – Toolkit Installation including the monitor model editor and embedded monitor server

## What you should be able to do

At the end of this lab you should be able to:

- Use Rational Application Developer or WebSphere Integration Developer to create the monitor model, and deploy it to the server.
- Use a sample program to create events that represent events from your process.
- Define a dashboard to view monitored data.

---

## Introduction

In this lab you will learn how to develop and deploy a model from end to end using Rational Application Developer or WebSphere Integration Developer and WebSphere Business Monitor. This lab will show you the basic procedure for creating and implementing a simple model.

The model used in this lab is **ClipsAndTacks** model which is also used in IBM Redbook **SG247148 – Business Process Management: Modeling through Monitoring Using WebSphere V6.0.2 Products**. In the Redbook, there are several models that are referenced, but in this lab you will be using the Future 1 Process. In the Redbook, the process is implemented as a BPEL process running in WebSphere Process Server. However this lab will demonstrate **Business Activity Monitoring** which can monitor events from any source. So, the lab will simulate the running of the process, by creating events that could have been sourced from anywhere, a J2EE application or BPEL, or any other source. These events are in the form of Common Base Events, which is an OASIS standard for common event format, and these are used by the Common Event Infrastructure (CEI) in WebSphere Process Server.

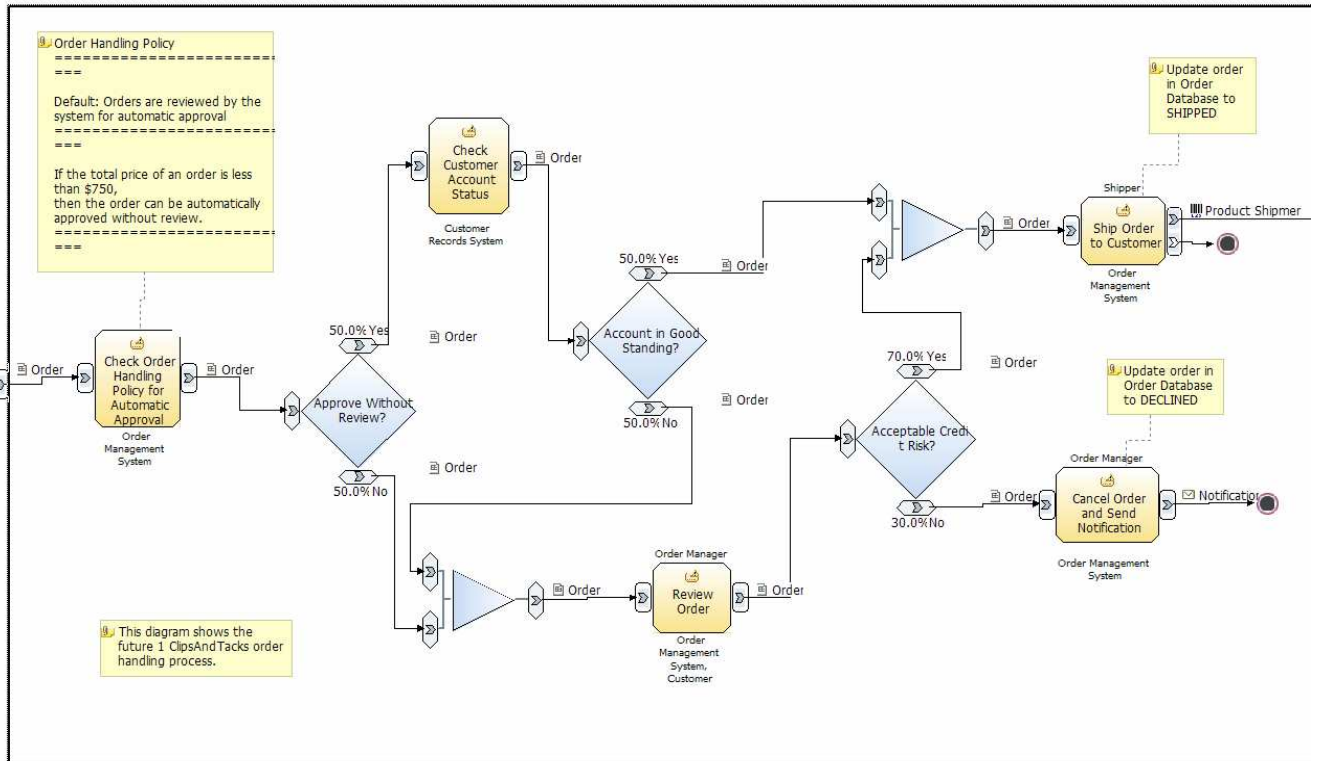
In this lab you will build a monitor model that has the same basic elements as the one that is defined in the Redbook. Namely you will create monitoring constructs in support of two KPIs:

- Average order fulfillment is three days or less
- Percentage of shipped orders is greater than 90%

You will also create situation events if declined orders are greater than or equal to three and if the order fulfillment time is greater than three days.

For dimensional analysis, you will create a Location dimension which allows you to drill down on country and city. You will also add an Order Status metric, which shows orders as 'New', 'Cancelled' or 'Shipped'. And you will add measures for average order price, sum of order price for all orders, and order count.

Here is a diagram of the ClipsAndTacks process model:



This is the ordering process for the Clips And Tacks Company. In this process, orders are received and optionally submitted to a review process. Orders which pass all checks are shipped to the customer. Other orders may be cancelled.

In the monitor model, you will create one monitoring context for each customer order. You will identify the inbound event that represents the receipt of the order, and this is marked so that it will create a new monitoring context instance for the order. You will also identify the inbound event that represents shipping the order, and this is marked so that it will trigger the termination of the monitoring context instance for the order. Since an order may also be cancelled, you will identify the inbound event that represents cancellation of the order, and this is also marked so that it will trigger the termination of the monitoring context instance.

**Useful URLs**

The following URLs may be helpful to you as you exercise this lab. Note that the port numbers in the URL of your installation may be different depending on your configuration.

- Server administrative console  
<https://localhost:9044/ibm/console>
- Business space dashboard  
<https://localhost:9444/BusinessSpace>

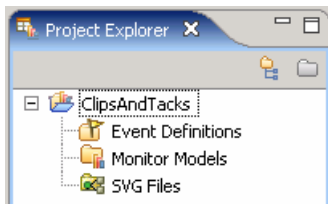
## Part 1: Create the monitor model

To create the monitor model, you will use the monitor model editor in Rational Application Developer or WebSphere Integration Developer. Note that the screen captures in this lab are based on WebSphere Integration Developer, so they may differ slightly from yours if you are using Rational Application Developer.

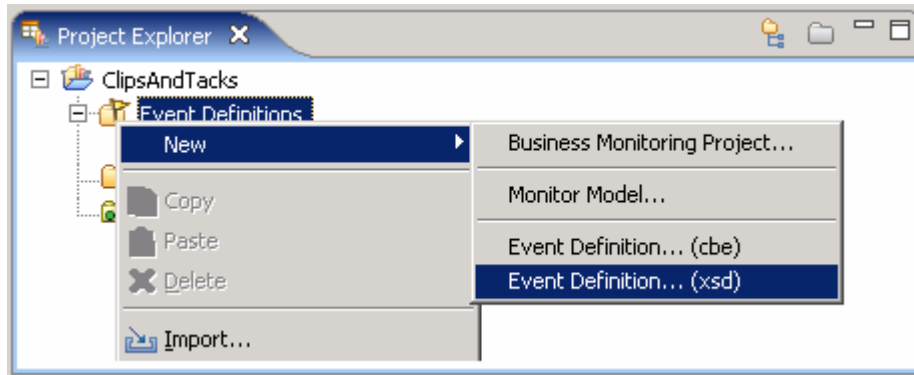
Note that there is a 'Problems' tab that displays errors concerning the model. Periodically, you should check this view to see if you have any problems that need to be addressed. Normally, warnings and informational messages are not a problem, but errors should be addressed.

If you want to skip this section, then a solution has been provided. You can import the supplied monitor model into Rational Application Developer or WebSphere Integration Developer, and then proceed to the next section. Refer to Appendix 1, and then proceed to the next section.

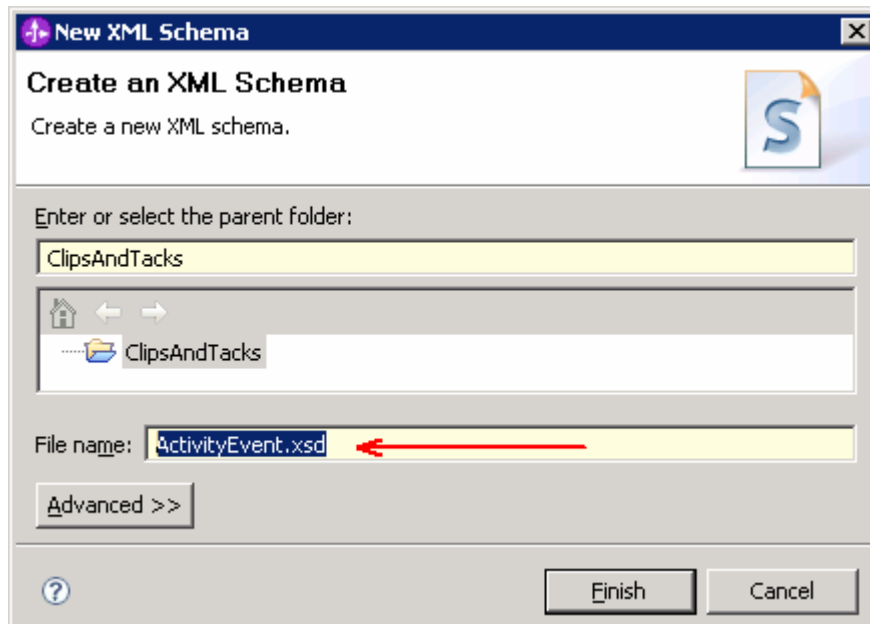
- \_\_\_ 1. Start Rational Application Developer or WebSphere Integration Developer and setup the environment.
  - \_\_\_ a. Start Rational Application Developer or WebSphere Integration Developer, and when prompted point to a new workspace such as C:\workspaces\ClipsAndTacksXSD
  - \_\_\_ b. Close the Welcome tab
  - \_\_\_ c. By default, you are in the Business Integration perspective. But you need to open the Business Monitoring perspective. From the main menu, select **Window → Open Perspective → Other**. The 'Open Perspective' dialog opens
  - \_\_\_ d. Select '**Business Monitoring**' from the 'Open Perspective' dialog and click **OK**
- \_\_\_ 2. Create a new monitoring project. A project is a container for your monitor models and event definitions.
  - \_\_\_ a. Right-click inside **Business Monitoring** project explorer (top left view in the Business Monitoring Perspective) and select **New → Business Monitoring Project...** from the pop-up menu. The 'New Business Monitoring Project' panel opens
  - \_\_\_ b. In the 'New Business Monitoring Project', enter the 'Project name:' as **ClipsAndTacks**
  - \_\_\_ c. Click the **Finish** button.
  - \_\_\_ d. A new project named **ClipsAndTacks** is created as shown below:



- \_\_\_ 3. Create a new XSD event definition named, **ActivityEvent**
  - \_\_\_ a. In the Business Monitoring project explorer view, expand **ClipsAndTacks**, right-click **Event Definitions** and select **New → Event Definition... (xsd)** from the pop-up menu



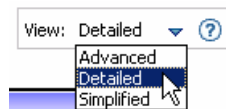
\_\_\_ b. The 'New XSD Schema' panel opens as shown below:

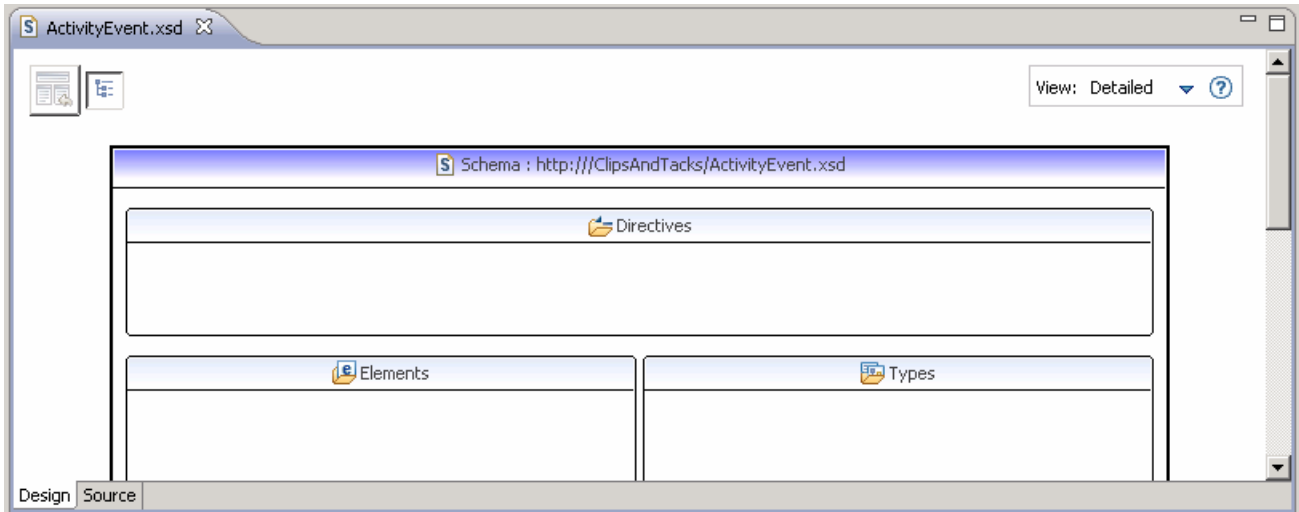


\_\_\_ c. Enter 'ActivityEvent.xsd' as 'File name:' and click **Finish**

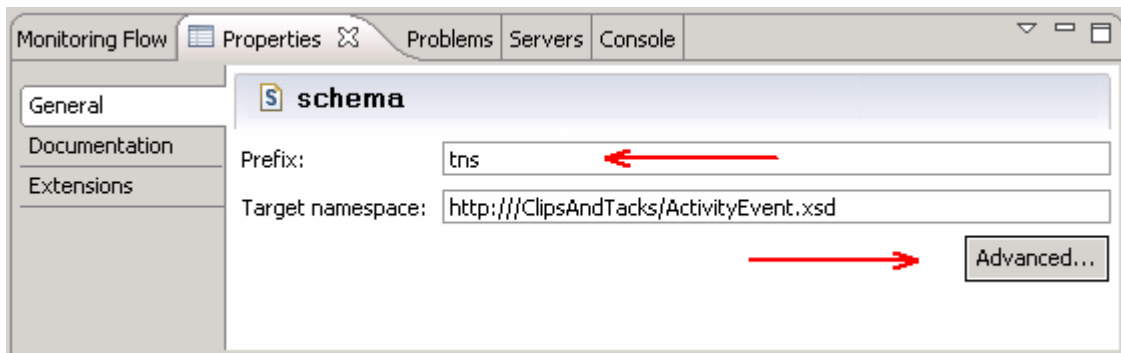
\_\_\_ d. The schema you created is listed under the event definitions category and the schema is opened in the XSD editor.

\_\_\_ e. There are several view modes that you can use on the XSD editor. Change the XSD editor view to **Detailed** by clicking on the pull down menu (▼) located in the top right corner.

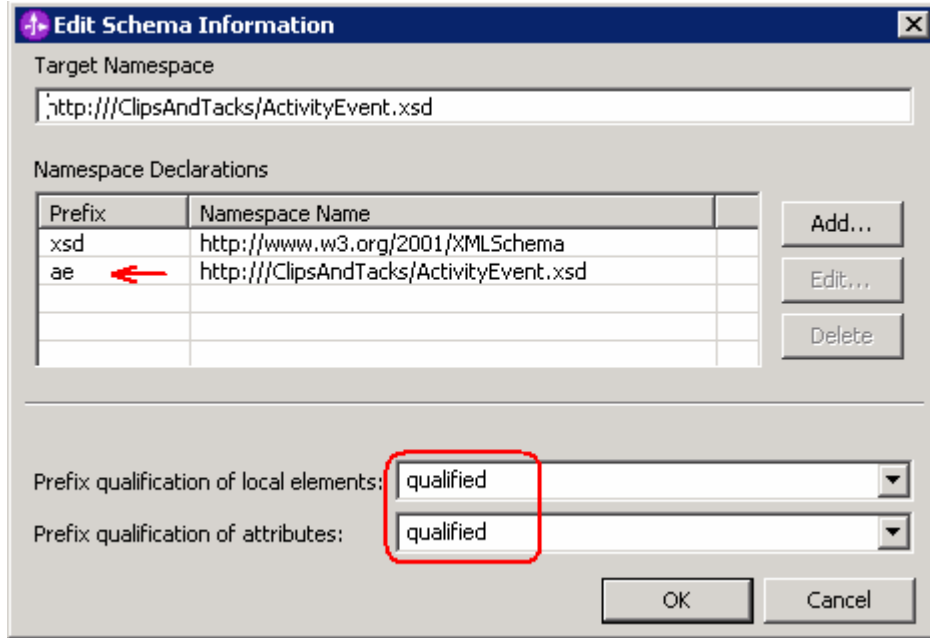




- \_\_\_ 4. Review and update the <schema> element, which is the root element of the schema
  - \_\_\_ a. In the XSD editor, select the schema element and then select the 'Details' tab of the 'Properties' view



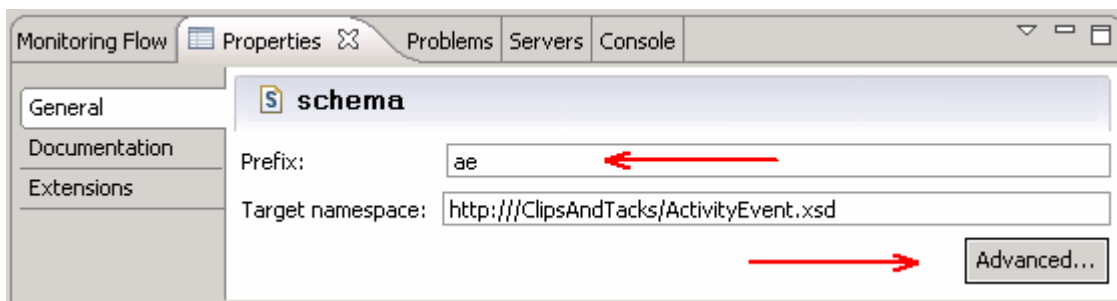
- \_\_\_ b. Update the 'Prefix:' from **tns** to **ae** (where **ae** stands for activity event and the prefix resembles the name of the schema)
- \_\_\_ c. Click the **Advanced** button to edit the schema information (attributes)



\_\_\_ d. Ensure that the prefix qualification of local elements and the attributes is set to **'qualified'**

**Note:** Prefix qualification indicates whether locally declared elements or attributes must be qualified by the target namespace in an instance document. If the value of this attribute is 'unqualified', then locally declared elements should not be qualified by the target namespace. If the value of this attribute is 'qualified', then locally declared elements must be qualified by the target namespace. Using 'qualified' helps to ensure that expressions written to access pieces of information within a type are unambiguous.

\_\_\_ e. Click **OK**. The **'General'** properties view of the 'ActivityEvent' schema should look like the picture below:



\_\_\_ f. Save the changes. **File → Save** or **Ctrl + S**

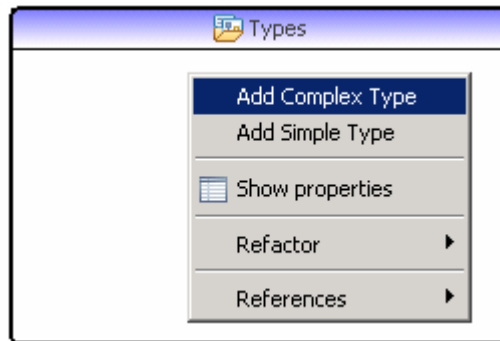
\_\_\_ 5. Add a complex element type named, **ActivityEventData** and the following sequence of elements to the complex type

The following are the elements including the data types you will be adding:

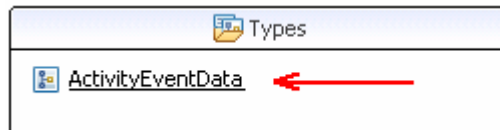


Name of the Element	Type
businessUnit	string
processName	string
mcID	string
activityName	string
eventType	string
startTime	dateTime
endTime	dateTime

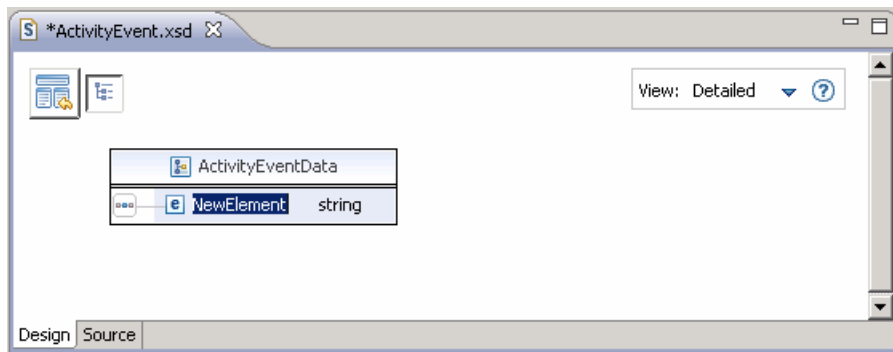
- \_\_\_ a. Right click in the **Types** section of the XSD editor and select '**Add Complex Type**' from the pop-up menu



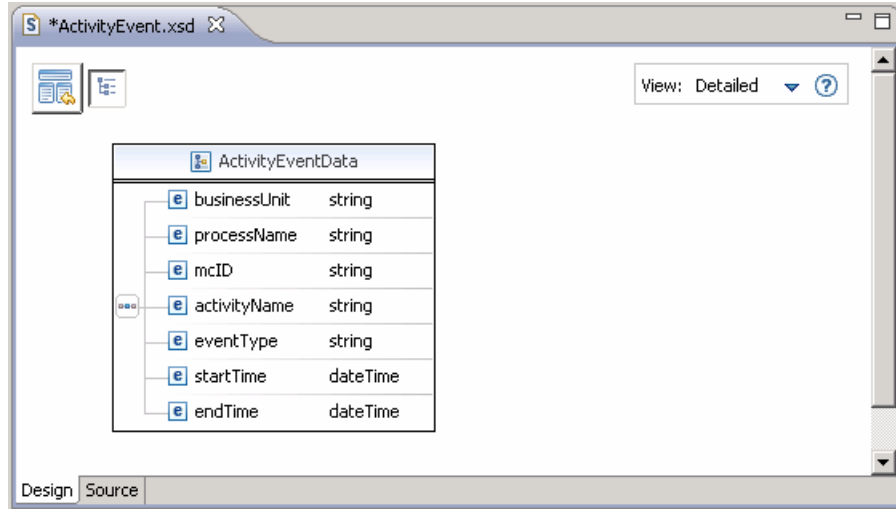
- \_\_\_ b. Name the complex type as **ActivityEventData** and save the changes. You should see the new complex type element added, as shown below:




- \_\_\_ c. Right click the complex type '**ActivityEventData**' again and select '**Add Element**' from the pop-up menu. By default the new element is created with a default name '**NewElement**' and xsd type '**string**' as shown below:



- \_\_\_ d. Modify the element name from 'NewElement' to '**businessUnit**' and accept the default xsd type as '**string**'
- \_\_\_ e. Repeat the above instructions to add the remaining elements mentioned in the table. The final '**ActivityEventData**' complex type should look like the picture below:



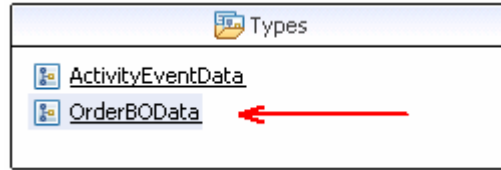
- \_\_\_ f. Now click the icon (  ) located in the upper left corner. This leads you back to the full schema editor view.
- \_\_\_ g. Save the changes. **File → Save** or **Ctrl + S**

\_\_\_ 6. Add another complex element type named **OrderBOData** and the following sequence of elements to the complex type

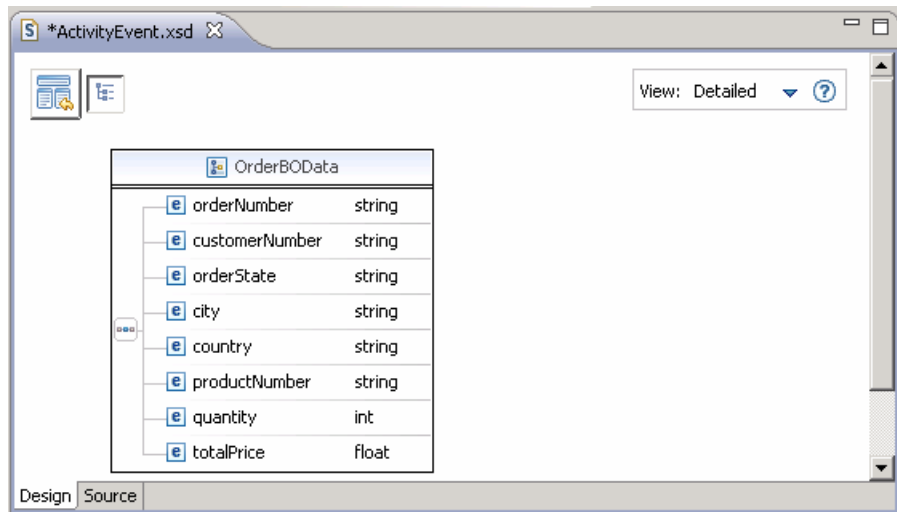
The following are the elements including the data types you will be adding:


Name of the Element	Type
orderNumber	string
customerNumber	string
orderState	string
city	string
country	string
productNumber	string
quantity	int
totalPrice	float

- \_\_\_ a. Right click in the **Types** section of the XSD editor and select '**Add Complex Type**' from the pop-up menu and name it OrderBOData



- \_\_\_ b. Right click the complex type '**OrderBOData**' again and use '**Add Element**' as required to create all the elements required for this complex type. The final '**OrderBOData**' complex type should look like the picture below:



- \_\_\_ c. Now click the icon (  ) located in the upper left corner. This leads you back to the full schema editor view.

- \_\_\_ d. Save the changes. **File → Save** or **Ctrl + S**

\_\_\_ 7. You can click the source tab to view the source XSD of the ActivityEvent schema you created.

\_\_\_ 8. WebSphere Business Monitor 6.2 has support for concurrent versions of monitor models. In order to take advantage of the Monitor server's version handling capabilities, each monitoring context should have a unique event definition that is the single inbound event that has the setting 'If no instances are found: Create new instance'. One can think of this as one inbound event carrying 'create monitor context' semantics, and all others as 'update monitor context' semantics. Clips And Tacks is a simple sample and could use a single event definition for the process. However, to satisfy the requirement of a unique event definition for version handling, you will create a new event definition that will create new instances of the monitor model.

- \_\_\_ a. In the project explorer, right click **Event Definitions** and select **New → Event Definition... (xsd)** from the pop-up menu. The 'New' event definition panel opens.

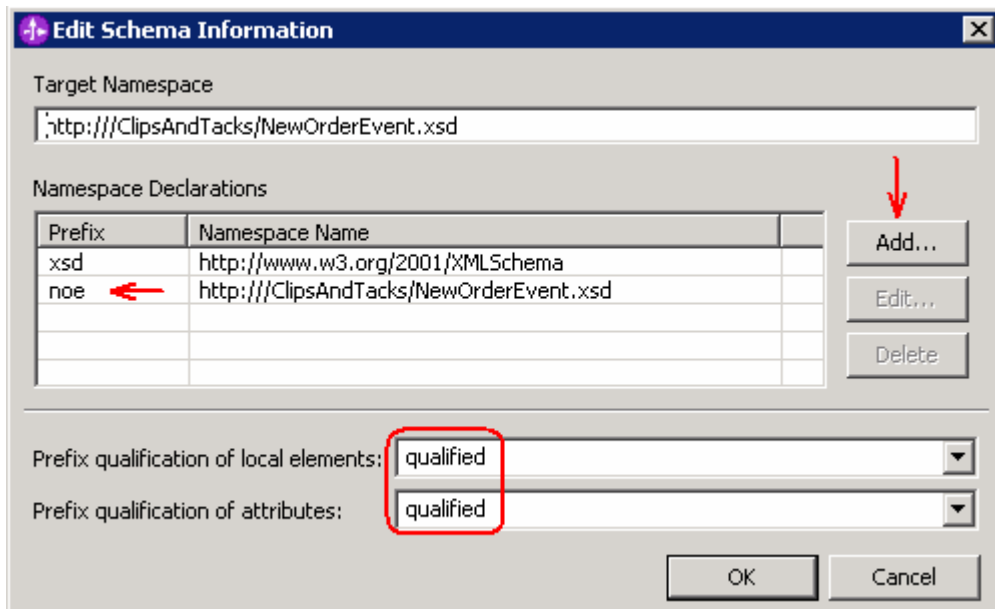
- \_\_\_ b. In the **New** event definition panel, select **ClipsAndTacks** as the parent folder and enter the 'File name:' as **NewOrderEvent.xsd** and click **Finish**.

\_\_\_ 9. Review and update the <schema> element, which is the root element of the schema

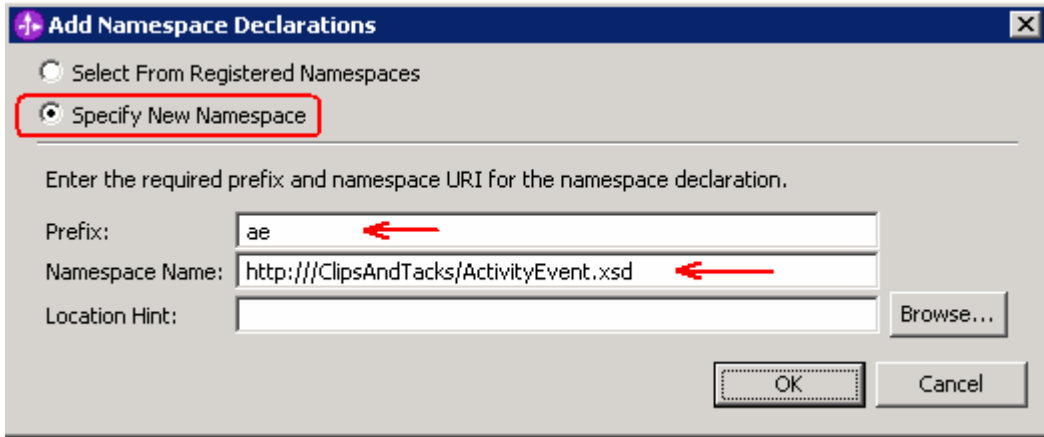
- \_\_\_ a. In the XSD editor, select the schema element and then select the '**General**' tab of the '**Properties**' view



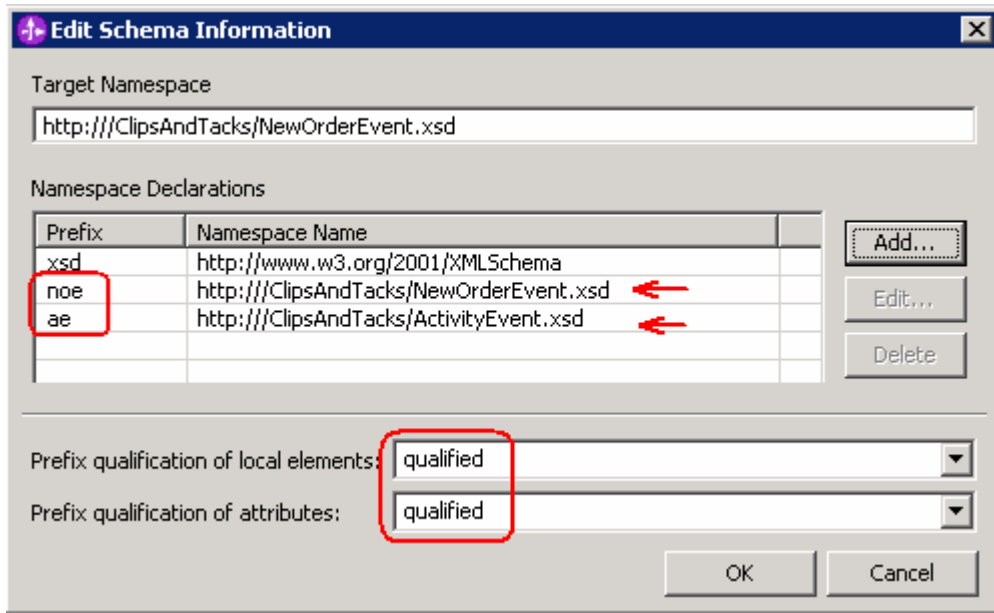
- \_\_\_ b. Update the 'Prefix:' from **tns** to **noe** (short for NewOrderEvent)
- \_\_\_ c. Click the **Advanced** button to edit the schema information (attributes)



- \_\_\_ d. Click the **Add** button to add the **ActivityEvent** namespace so the 'ActivityEvent' schema can be imported to the current 'NewOrderEvent' schema. The 'Add Namespace Declarations' panel opens
- \_\_\_ e. Enter the following parameters:
  - Select the radio button for 'Specify New Namespace'
  - Prefix : **ae**
  - Namespace Name : **http://ClipsAndTacks/ActivityEvent.xsd**

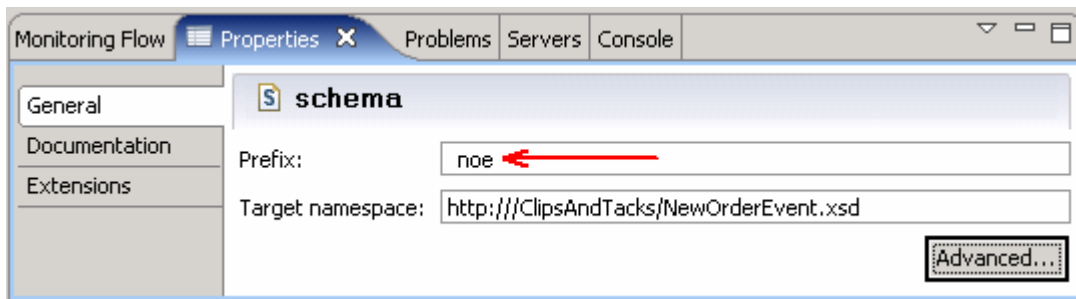


\_\_ f. Click **OK**



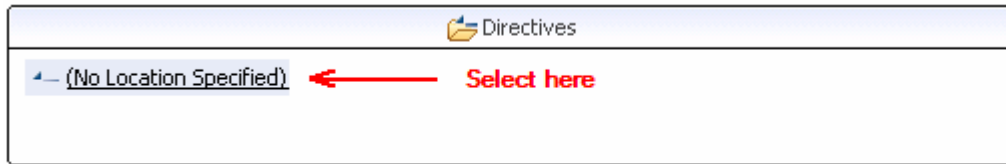
\_\_ g. Ensure that the prefix qualification of local elements and the attributes is set to **'qualified'**

\_\_ h. Click **OK**. The **'General'** properties view of the 'ActivityEvent' schema should look like the picture below:

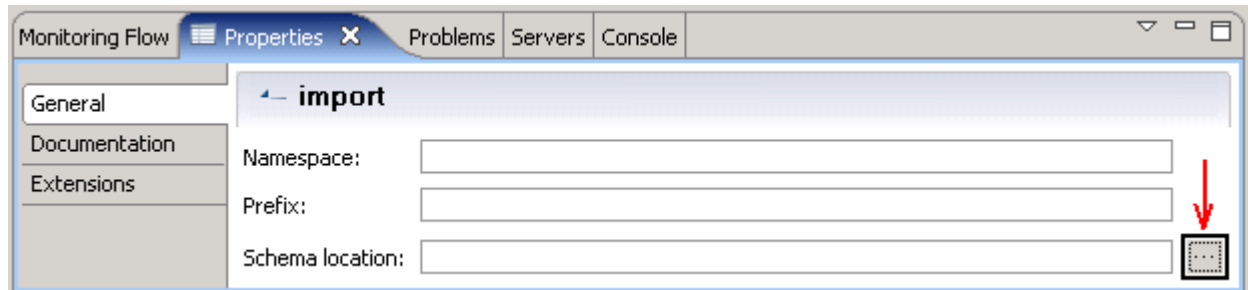


\_\_ i. Save the changes. **File** → **Save** or **Ctrl + S**

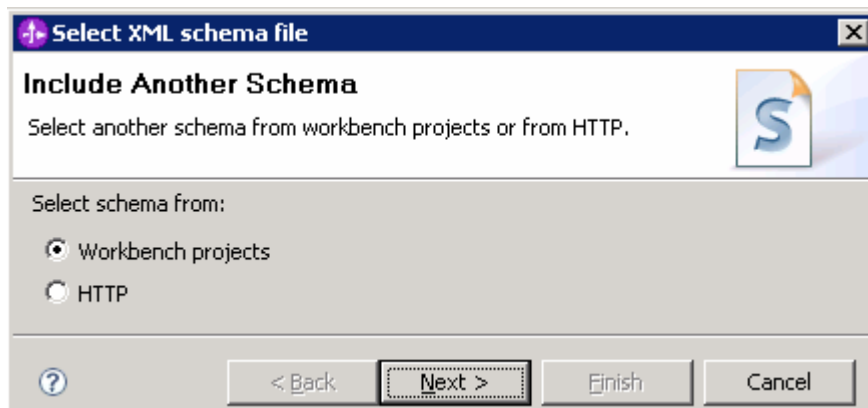
- \_\_\_ j. Right click any where in the **'Directives'** section of the XSD editor and select **'Add Import'** from the pop-up menu to import the 'ActivityEvent' schema. An empty XSD <import> element is added



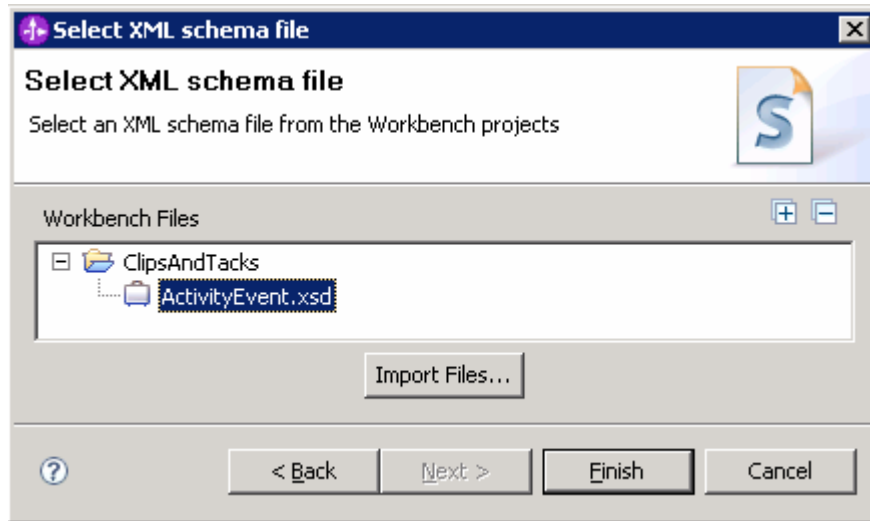
- \_\_\_ k. In the XSD editor, select the empty XSD <import> element and then select the 'Properties' General panel



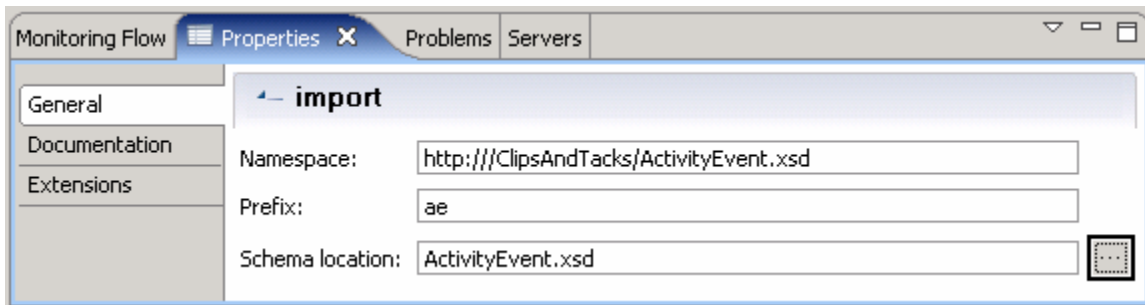
- 1) Click the button for **'Schema Location'**. The **'Select XML Schema file'** panel opens



- 2) Ensure the radio button next to **'Workbench projects'** is selected and click **Next**
- 3) In the following panel, expand the 'ClipsAndTacks' folder and then select 'ActivityEvent.xsd'

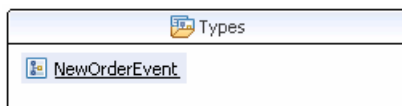


4) Click **Finish**. The Properties **General** panel should look the picture below:



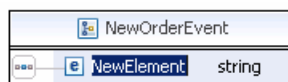
5) Save the configuration. **File → Save** or **Ctrl + S**

- \_\_\_ l. Now right click in the **'Types'** section and select the **'Add Complex Type'** from the pop-up menu. An XSD <complexType> element is added with a default name, which you should change to **'NewOrderEvent'**.



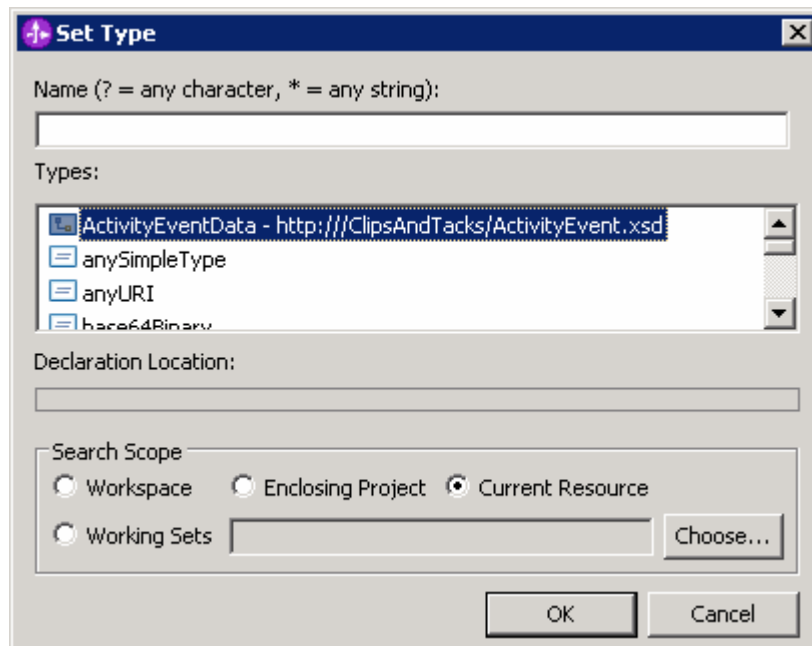
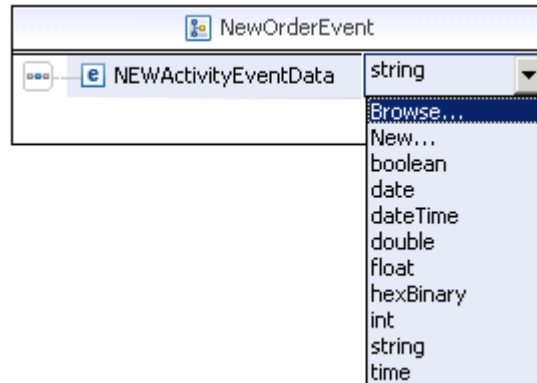
\_\_\_ m. Save the configuration. **File → Save** or **Ctrl + S**

- \_\_\_ n. Right click the **'NewOrderEvent'** complex type element again and select **'Add Element'** from the pop-up menu. A new element of type string is added with a default name as shown below:



\_\_\_ o. Rename the name of the element to **'NEWActivityEventData'**

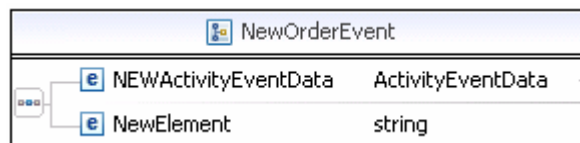
\_\_\_ p. For the **'Type:'** select **'Browse'** from the drop down list. The **'Set Type'** panel opens



- \_\_\_ q. Select 'ActivityEventData' from the 'Types' text area and click **OK**. The **NEWActivityEventData** element should look like the picture below. You can click the + next to ActivityEventData to see the details of the element.

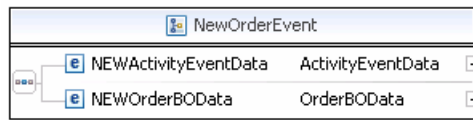


- \_\_\_ r. Save the configuration. **File → Save** or **Ctrl + S**
- \_\_\_ s. Now right click the '**NewOrderEvent**' complex type element and select '**Add Element**' from the pop-up menu. A new element of type string is added with a default name as shown below:

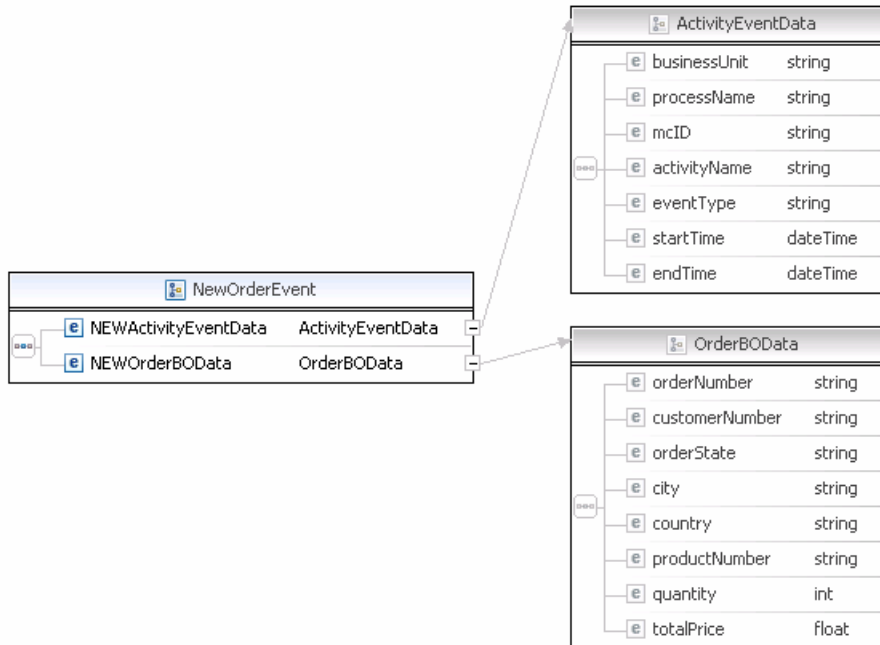




- \_\_\_ t. Rename the name of the element to **NEWOrderBOData**. For the **'Type:'** select **'Browse'** from the drop down list. The **'Set Type'** panel opens. Select **'OrderBOData'** from the **'Types'** text area and click **OK**. The **NEWOrderBOData** element should look like the picture below.



- \_\_\_ u. Click the + icons to see the details.



- \_\_\_ v. Save the configuration. **File → Save** or **Ctrl + S**

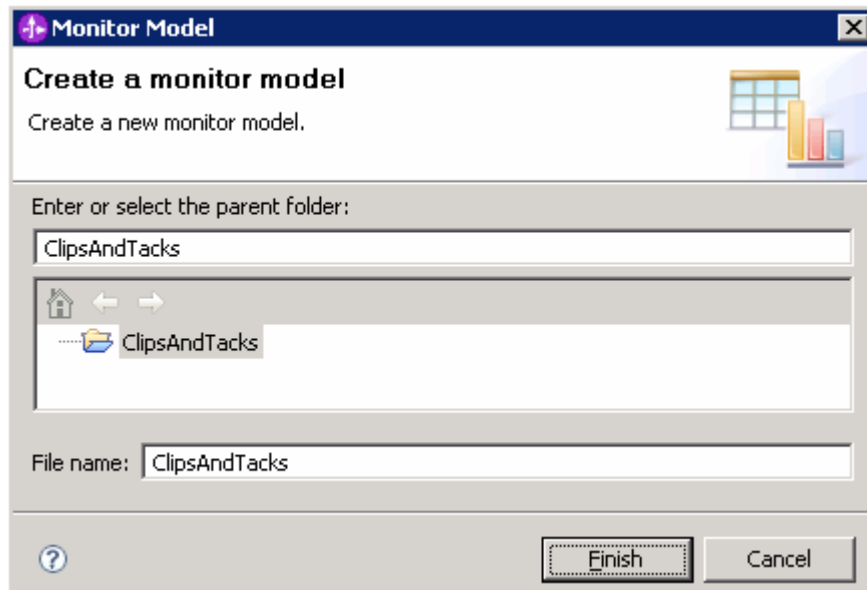
- \_\_\_ w. Click the source tab to see the XSD source.

- \_\_\_ x. Close the XSD schema editor

- \_\_\_ 10. Create a new monitor model ClipsAndTacks. The monitor model contains the metrics and KPIs that you want to monitor and it will be the source for creating the deployment code for the monitor server.

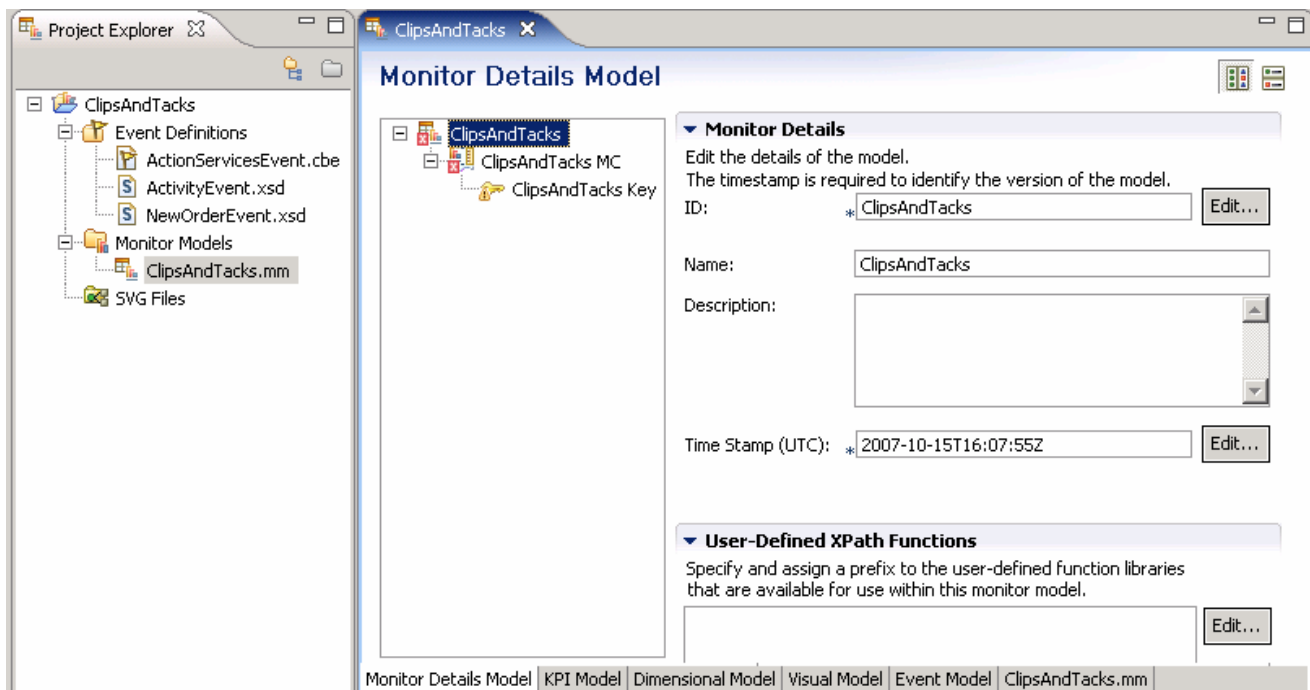
- \_\_\_ a. Right click in the **'Business Monitoring'** project explorer view and select **New → Monitor Model...** from the pop-up menu. The Monitor Model panel opens

- 1) For the **'File name:'**, enter **ClipsAndTacks**



2) Click **Finish**. The model editor is opened as follows:

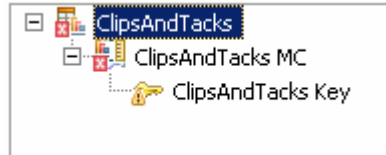
**Note:** When the model editor first opens you will have errors in the model. This is normal and is signaling that required elements are missing or incomplete. Next you will complete these required elements.



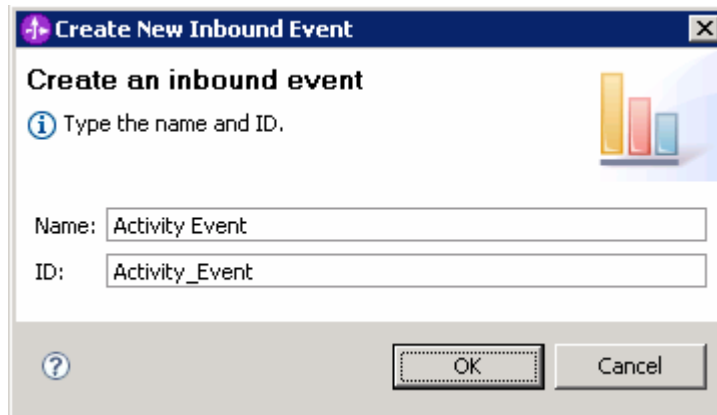
11. Notice in the Monitor Details Model navigator that a new model is created (ClipsAndTacks) along with a monitoring context (ClipsAndTacks MC). Now you need to specify the inbound events that will be processed by this model. You need to indicate which event will create a new monitoring context instance and which event will terminate the monitoring context instance. You also need to identify the correlation information for the monitoring context so that Monitor knows which monitoring context instance should receive the events. The ClipsAndTacks model uses one event definition, ActivityEvent, to represent the layout of all of the events to be processed. NewOrderEvent derives

from ActivityEvent but does not add additional event elements. In ActivityEvent, ActivityName is a field used to identify specific activities in the ClipsAndTacks process. ActivityName is used to determine when to terminate the monitoring context. You will create correlation that is based on orderNumber, which is another field defined in ActivityEvent.

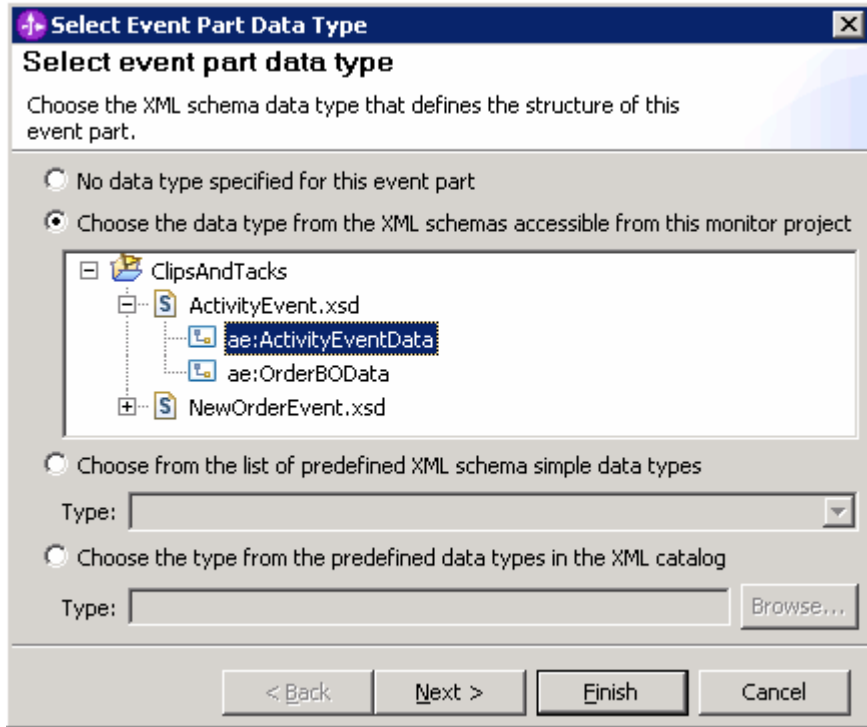
- \_\_\_ 12. In the Monitor Details Model, expand ClipsAndTacks in the navigation view. Then expand ClipsAndTacks MC. You will see that a key has been created for you automatically when you created the model.



- \_\_\_ 13. In the model navigation view, right click ClipsAndTacks MC, then select **New → Inbound Event** from the pop-up menu
- \_\_\_ a. For the name, type **Activity Event**. Notice that the ID will automatically be set for you to the same name with underscores (**Activity\_Event**)



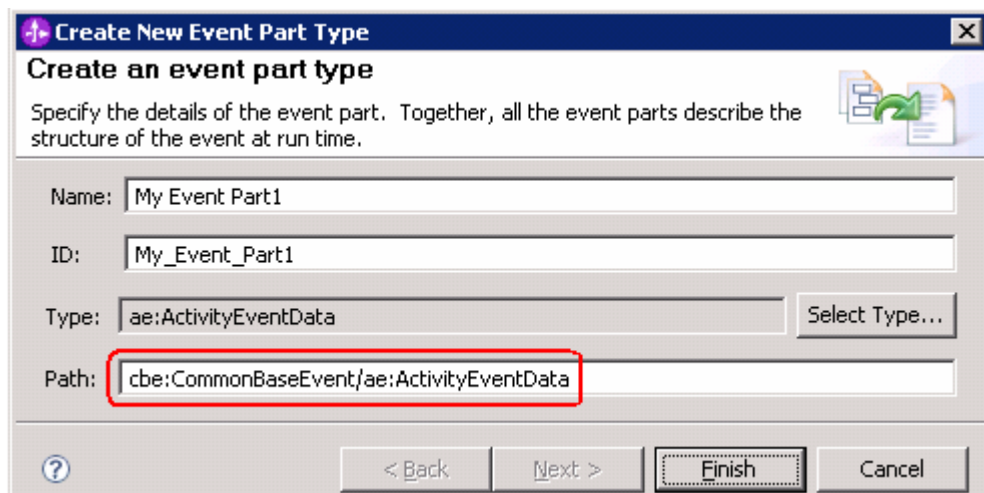
- \_\_\_ b. Click **OK**. The inbound 'ActivityEvent' opens in an editor
- \_\_\_ c. In the inbound 'ActivityEvent' editor, click the **'Add'** button for **'Event Parts'** to create a new event part type in the 'Event Type Details' section. The 'Create New Event part Type' panel opens
- \_\_\_ d. In the 'Create New Event part Type' panel, enter the following:
- 1) Name : **My Event Part1**
  - 2) ID : **My\_Event\_Part1**
  - 3) Click the **'Select Type'** button for the 'Type' field. The 'Select Event Part Data Type' panel opens
    - a) Select the check box for 'Choose the data type from XML schemas accessible from this monitor project'
    - b) Expand **ClipsAndTacks → ActivityEventData** and select **'ae:ActivityEventData'**



c) Click **Finish**

4) Path : **cbe:CommonBaseEvent/ae:ActivityEventData**

**Note:** The path is an XML Path Language (XPath) expression that identifies the location in the event definition of the structure defined by the event part type. The expression is based on the structure of the actual event instance to be received at run time. It always starts with `cbe:CommonBaseEvent` and navigates into content that is placed in the `xs:any` slot of the Common Base Event. In general, if an XML Schema Definition (XSD) used to define an event structure contains an `<xsd:any>` or `<xsd:anyType>` slot, you should use an event part to specify the actual structure of the content that will fill that slot at run time. In this event part, you use **`cbe:CommonBaseEvent/ae:ActivityEventData`** as the path. Refer to **'allXSDevents.xml'** located in the **Labfiles62.zip** file, to see how the XSD based events are being sent during runtime.



5) Click **Finish**

14. In the inbound 'ActivityEvent' editor, click the 'Add' button 'Event Parts' again to create another event part type in the 'Event Type Details' section. The 'Create New Event part Type' panel opens

a. In the 'Create New Event part Type' panel, enter the following:

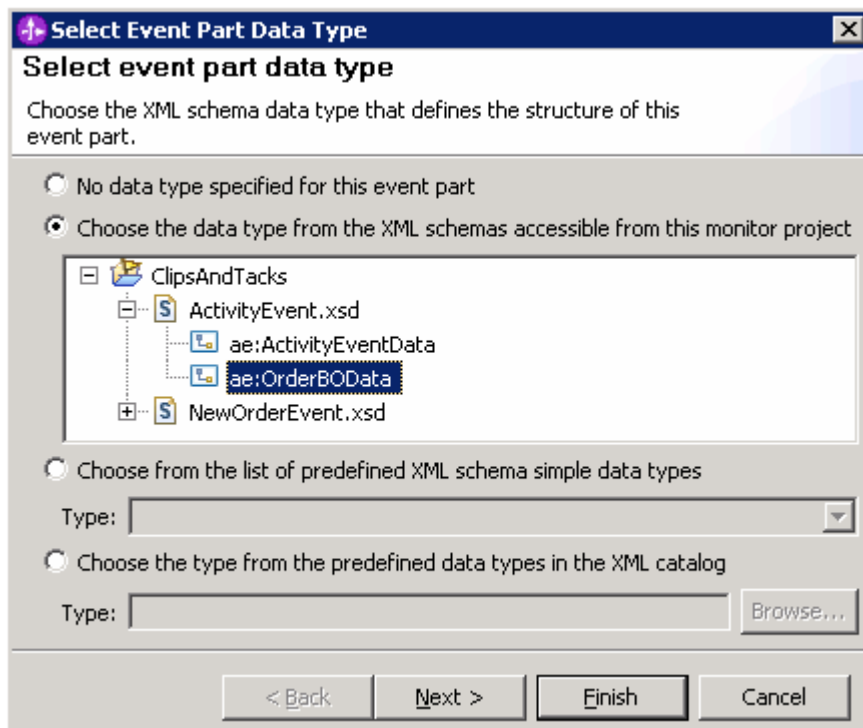
1) Name : **My Event Part2**

2) ID : **My\_Event\_Part2**

3) Click the 'Select Type' button for the 'Type' field. The 'Select Event Part Data Type' panel opens

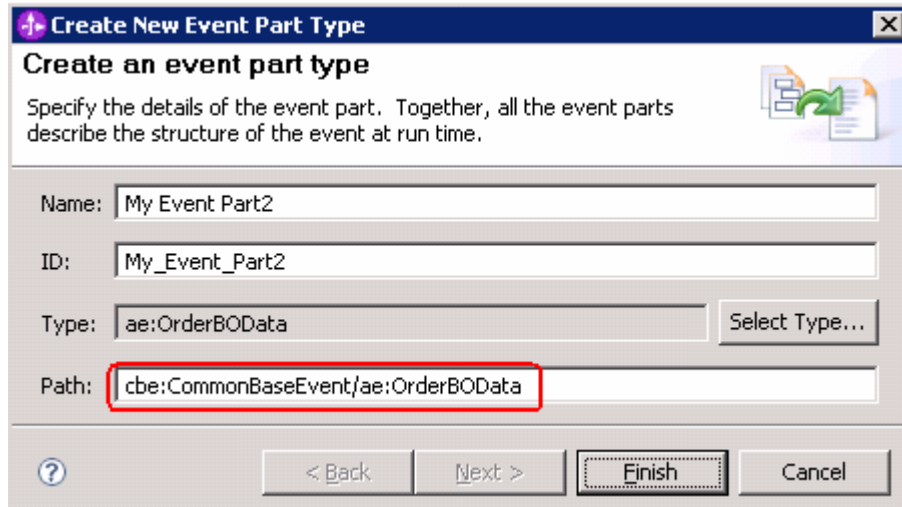
a) Select the check box for 'Choose the data type from XML accessible from this monitor project'

b) Expand **ClipsAndTacks** → **ActivityEventData.xsd** and select '**ae:OrderBOData**'



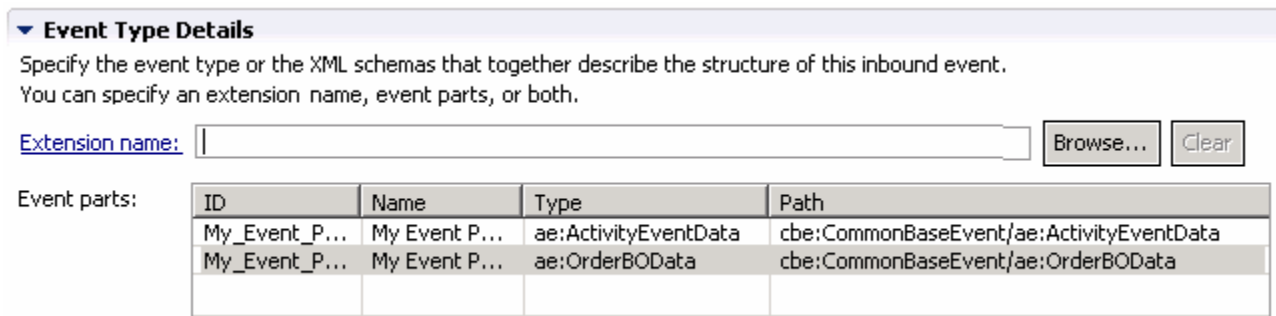
c) Click **Finish**

4) Path : **cbe:CommonBaseEvent/ae:OrderBOData**



5) Click **Finish**

\_\_ b. The final 'Event Type Details' section with the two 'Event parts' should look like the picture below:



\_\_ c. Save the configuration. **File → Save** or **Ctrl + S**

\_\_ d. To enter the **Filter Condition**, you can use Content Assist with the keys Ctrl-Space, or you can type the expression directly into the expression window. When completed the expression should be as follows. Note the single quotation marks used for string handling.

Activity\_Event/My\_Event\_Part1/ae:businessUnit = 'Clips And Tacks' and  
 Activity\_Event/My\_Event\_Part1/ae:processName = 'Order Handling'

\_\_ e. To enter the **Correlation Expression**, you can use Content Assist with the keys Ctrl-Space, or you can type the expression directly into the expression window. When complete the expression should be

ClipsAndTacks\_Key =Activity\_Event/My\_Event\_Part2/ae:orderNumber

\_\_ f. For 'If no instances are found', select 'Treat as error'

\_\_ g. For 'If one instance is found', select 'Deliver to the instance'

\_\_ h. For 'If multiple instances are found', select 'Treat as error'

\_\_ i. Here is a sample of a portion of the definition:

**▼ Filter Condition**  
Define a condition based on the event attributes to identify whether to accept an event of this type.

Activity\_Event/My\_Event\_Part1/ae:businessUnit = 'Clips And Tacks' and  
Activity\_Event/My\_Event\_Part1/ae:processName = 'Order Handling'

**▼ Correlation Expression**  
Define an expression to identify the monitoring context instance or instances that receive the event at runtime.

ClipsAndTacks\_Key = Activity\_Event/My\_Event\_Part2/ae:orderNumber

If no instances are found

If one instance is found

If multiple instances are found

\_\_\_ j. Save the configuration. **File → Save** or **Ctrl + S**

\_\_\_ 15. In the model navigation view, right click ClipsAndTacks MC, then select **New → Inbound Event** from the pop-up menu

\_\_\_ a. For the name, type **New Order Event**. Notice that the ID will automatically be set for you to the same name with underscores (**New\_Order\_Event**)

\_\_\_ b. Click **OK**. The inbound 'NewOrderEvent' opens in an editor

\_\_\_ c. In the inbound 'NewOrderEvent' editor, click the '**Add**' button for '**Event Parts**' to create a new event part type in the 'Event Type Details' section. The 'Create New Event part Type' panel opens

\_\_\_ d. In the 'Create New Event part Type' panel, enter the following:

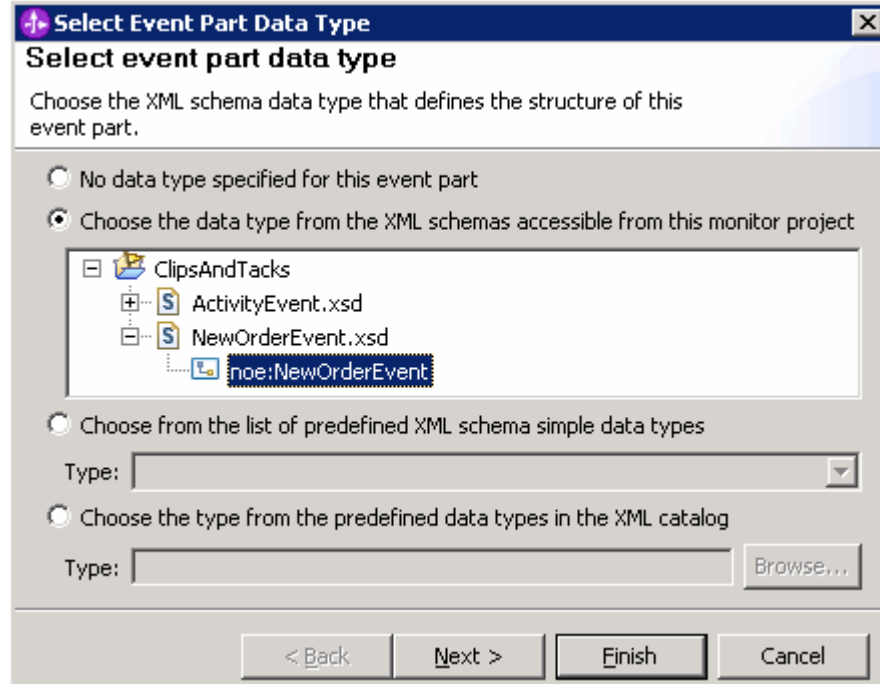
1) Name : **My Event Part**

2) ID : **My\_Event\_Part**

3) Click the '**Select Type**' button for the 'Type' field. The 'Select Event Part Data Type' panel opens

a) Select the check box for 'Choose the data type from XML accessible from this monitor project'

b) Expand **ClipsAndTacks → NewOrderEvent.xsd** and select '**noe:NewOrderEvent**'



c) Click **Finish**

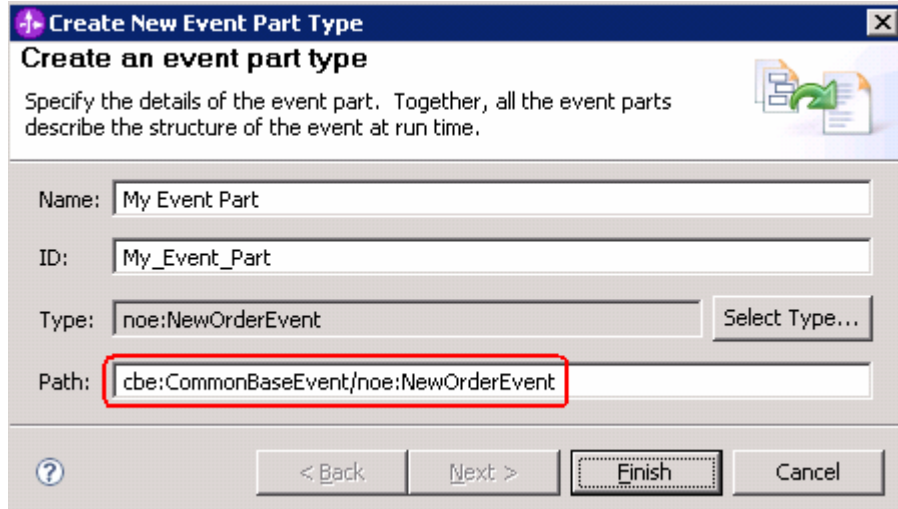
4) Path : **cbe:CommonBaseEvent/noe:NewOrderEvent**

---

**Note:** The path is an XML Path Language (XPath) expression that identifies the location in the event definition of the structure defined by the event part type. The expression is based on the structure of the actual event instance to be received at run time. It always starts with `cbe:CommonBaseEvent` and navigates into content that is placed in the `xs:any` slot of the Common Base Event. In general, if an XML Schema Definition (XSD) used to define an event structure contains an `<xsd:any>` or `<xsd:anyType>` slot, you should use an event part to specify the actual structure of the content that will fill that slot at run time. In this event part, you use **`cbe:CommonBaseEvent/ noe:NewOrderEvent`** as the path. Refer to 'allXSDevents.xml' part of the **Labfiles62.zip** to see how the XSD based events are being sent during runtime.

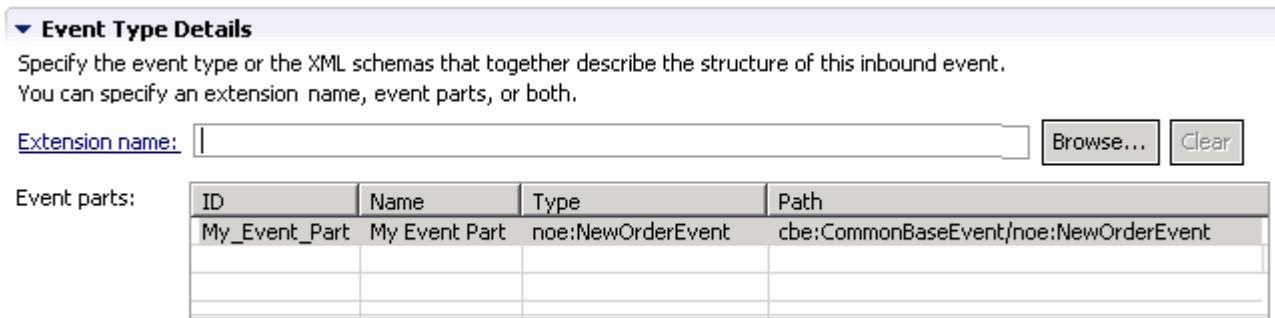
---





5) Click **Finish**

\_\_ e. The final 'Event Type Details' section with the 'Event part' should look like the picture below:



\_\_ f. Save the configuration. **File → Save** or **Ctrl + S**

\_\_ g. To enter the Filter Condition, you can use Content Assist with the keys Ctrl-Space, or you can type the expression directly into the expression window. When complete, the expression should be as follows. Note the single quotation marks used for string handling.

New\_Order\_Event/My\_Event\_Part/noe:NEWActivityEventData/ae:businessUnit = 'Clips And Tacks' and New\_Order\_Event/My\_Event\_Part/noe:NEWActivityEventData/ae:processName = 'Order Handling'

\_\_ h. To enter the Correlation Expression, you can use Content Assist or you can type the expression directly into the expression window. When complete the expression should be

ClipsAndTacks\_Key =  
New\_Order\_Event/My\_Event\_Part/noe:NEWOrderBODData/ae:orderNumber

\_\_ i. For 'If no instances are found', select 'Create new instance'

\_\_ j. For 'If one instance is found', select 'Treat as error'

\_\_ k. For 'If multiple instances are found', select 'Treat as error'

\_\_ l. Here is a sample of a portion of the definition:

**▼ Filter Condition**  
 Define a condition based on the event attributes to identify whether to accept an event of this type.

New\_Order\_Event/My\_Event\_Part/noe:NEWActivityEventData/ae:businessUnit = 'Clips And Tacks' and  
 New\_Order\_Event/My\_Event\_Part/noe:NEWActivityEventData/ae:processName = 'Order Handling'

**▼ Correlation Expression**  
 Define an expression to identify the monitoring context instance or instances that receive the event at runtime.

ClipsAndTacks\_Key = New\_Order\_Event/My\_Event\_Part/noe:NEWOrderBOData/ae:orderNumber

If no instances are found

If one instance is found

If multiple instances are found

\_\_\_ m. Save the configuration. **File → Save** or **Ctrl + S**

\_\_\_ 16. In the model navigation view, double click **ClipsAndTacks Key**, to open the key in the editor

\_\_\_ a. For Key Value Expressions, click '**Add...**' Note that double clicking in the table will produce the same results.

\_\_\_ b. A row is added to the table and in the expression cell it shows << No expression specified >>. Click this cell of the table, then a button is displayed. Click this button and the expression editor is displayed. You can use Content Assist, or type this expression directly into the window:

New\_Order\_Event/My\_Event\_Part/noe:NEWOrderBOData/ae:orderNumber

\_\_\_ c. Click **OK**.

\_\_\_ d. Here is a sample of the key:

ID: \* ClipsAndTacks\_Key Edit...

Name: ClipsAndTacks Key

Description:

Type: \* String

Maximum String Length: 256

Allocate additional space in database to accommodate Unicode string for globalization

A value is required for this metric

Default Value: Edit...

This key can be used for sorting

**Key Value Expressions**

Specify the expressions that set the value of the key.

Expression
New_Order_Event/My_Event_Part/ noe:NEWOrderBOData/ae:orderNumber

Add Remove

\_\_ e. Save the configuration. **File → Save** or **Ctrl + S**

---

**Note:** At this point there should not be any errors remaining in the model. The errors shown initially were the result of the monitoring context key not being complete.

---

\_\_\_\_ 17. Create triggers that indicate the end of the monitoring context. The first trigger is based on an order being shipped, and the second trigger is based on an order being cancelled.

\_\_ a. In the model navigation view, right click **ClipsAndTacks MC**, then select **New → Trigger**. Enter the following values:

- 1) Name : **Ship Order to Customer Trigger**
- 2) ID : **Ship\_Order\_to\_Customer\_Trigger**
- 3) Select the check box next to '**Terminate monitoring context**'
- 4) For Trigger Sources, click Add, then select **Other source type → Activity Event** and then click OK
- 5) For Trigger Condition, enter  
 Activity\_Event/My\_Event\_Part1/ae:activityName = 'Ship Order to Customer' and  
 Activity\_Event/My\_Event\_Part1/ae:eventType = 'completed'

\_\_ b. Save the configuration. **File → Save** or **Ctrl + S**


\_\_ c. Here is a sample of this trigger:

**▼ Trigger Details**  
Edit the details of the trigger, which detects an occurrence and initiates an action in response.


ID: \*Ship\_Order\_to\_Customer\_Trigger

Name: Ship Order to Customer Trigger

Description:

Trigger is repeatable  
 Terminate monitoring context 

**▼ Trigger Sources**  
Specify the source of this trigger.

Source Type	Source
Event	 Activity Event

**▼ Trigger Condition**  
Specify the condition that determines whether the trigger will fire.

Activity\_Event/My\_Event\_Part1/ae:activityName = 'Ship Order to Customer' and  
Activity\_Event/My\_Event\_Part1/ae:eventType = 'completed'

\_\_\_ d. In the model navigation view, right click ClipsAndTacks MC, then select **New > Trigger**. Enter these values:

- 1) Name : **Cancel Trigger**
- 2) ID : **Cancel\_Trigger**
- 3) Select the check box next to '**Terminate monitoring context**'
- 4) For Trigger Sources, click Add, then select Other source type > Activity Event, then click OK.
- 5) For Trigger Condition, enter  
Activity\_Event/My\_Event\_Part1/ae:activityName = 'Cancel Order and Send Notification' and  
Activity\_Event/My\_Event\_Part1/ae:eventType = 'completed'

\_\_\_ e. Save the configuration. **File → Save** or **Ctrl + S**

\_\_\_ 18. Create a KPI for Average fulfillment three days or less. First you will create a trigger which fires when a new order is started. Metrics are created to hold the order start time and order end time. The values of the metrics are set when the new order trigger fires. You create a duration metric which is calculated based on subtracting the order end time metric from the order start time metric. You will create a KPI context which is a container to hold the KPIs. Then you create a KPI which is based on the duration metric and you apply an average function to it.

\_\_\_ a. In the model navigation view, right click ClipsAndTacks MC, then select **New → Trigger**. Enter these values:

- 1) Name : **New Order Trigger**
- 2) ID : **New\_Order\_Trigger**
- 3) For Trigger Sources, click Add, then select Other source type > New Order Event, then click OK.
- 4) For Trigger condition, enter  
  
New\_Order\_Event/My\_Event\_Part/noe:NEWActivityEventData/ae:activityName = 'Check Order Handling Policy for Automatic Approval' and  
New\_Order\_Event/My\_Event\_Part/noe:NEWActivityEventData/ae:eventType = 'started'
- 5) Save the configuration. **File → Save** or **Ctrl + S**

\_\_\_ b. In the model navigation view, right click ClipsAndTacks MC, then select **New → Metric**. Enter these values:

- 1) Name : **Order Start Time**
- 2) ID : **Order\_Start\_Time**
- 3) Type : **DateTime**
- 4) For Metric Value Expressions, click Add
- 5) In the first row of the Metric Value Expressions table, select the cell under column Trigger, then a button is shown, then click the button and select Trigger type > ClipsAndTacks MC > New Order Trigger, click OK
- 6) In the first row of the Metric Value Expressions table, select the cell under column Expression, then a button is shown, then click the button and enter this text:  
  
New\_Order\_Event/My\_Event\_Part/noe:NEWActivityEventData/ae:startTime
- 7) Click OK.
- 8) Save the configuration. **File → Save** or **Ctrl + S**

**▼ Metric Details**

Edit the details of the metric, which is a holding spot for information used in other calculations.

ID:

Name:

Description:

Type:

A value is required for this metric

Default Value:

This metric can be used for sorting

**▼ Metric Value Expressions**

Specify the expressions that set the value of the metric. If a trigger is specified, the map is evaluated when the trigger fires.

Trigger	Expression
<input type="button" value="New Order Trigger"/>	<input type="text" value="New_Order_Event/My_Event_Part/ noe:NEWActivityEventData/ae:startTime"/>

\_\_\_ c. In the model navigation view, right click ClipsAndTacks MC, then select **New → Metric**. Enter these values:

- 1) Name : **Order End Time**
- 2) ID : **Order\_End\_Time**
- 3) Type : **DateTime**
- 4) For Metric Value Expressions, click Add
- 5) In the first row of the Metric Value Expressions table, select the cell under column Trigger, then a button is shown, then click the button and select Trigger type > ClipsAndTacks MC > Ship Order to Customer Trigger, click OK
- 6) In the first row of the Metric Value Expressions table, select the cell under column Expression, then a button is shown, then click the button and enter this text:
 

Activity\_Event/My\_Event\_Part1/ae:endTime
- 7) Click OK.
- 8) For Metric Value Expressions, click Add
- 9) In the second row of the Metric Value Expressions table, select the cell under column Trigger, then a button is shown, then click the button and select Trigger type > ClipsAndTacks MC > Cancel Trigger, click OK
- 10) In the second row of the Metric Value Expressions table, select the cell under column Expression, then a button is shown, then click the button and enter this text:

Activity\_Event/My\_Event\_Part1/ae:endTime

11) Click OK.

\_\_\_ d. Save the configuration. **File → Save** or **Ctrl + S**

**▼ Metric Details**

Edit the details of the metric, which is a holding spot for information used in other calculations.

ID: \* Order\_End\_Time Edit...

Name: Order End Time

Description:

Type: \* DateTime ▼

A value is required for this metric

Default Value: Edit...

This metric can be used for sorting

---

**▼ Metric Value Expressions**

Specify the expressions that set the value of the metric. If a trigger is specified, the map is evaluated when the trigger fires.

Trigger	Expression
Ship Order to Customer Trigger	Activity_Event/My_Event_Part1/ae:endTime
Cancel Trigger	Activity_Event/My_Event_Part1/ae:endTime

\_\_\_ e. In the model navigation view, right click ClipsAndTacks MC, then select **New > Metric**. Enter these values:

- 1) Name : **Order Fulfillment Duration**
- 2) ID : **Order\_Fulfillment\_Duration**
- 3) Type : **Duration**
- 4) For Default Value, click Edit...
- 5) On the Select Duration dialog, click OK, and this will set the default duration to zero
- 6) For Metric Value Expressions, click Add
- 7) In the first row of the Metric Value Expressions table, select the cell under column Trigger, then a button is shown, then click the button and select Trigger type > ClipsAndTacks MC > Ship Order to Customer Trigger, click OK
- 8) In the first row of the Metric Value Expressions table, select the cell under column Expression, then a button is shown, then click the button and enter this text:

```
xs:dayTimeDuration(Order_End_Time - Order_Start_Time)
```

- 9) Click OK

- 10) For Metric Value Expressions, click Add
- 11) In the second row of the Metric Value Expressions table, select the cell under column Trigger, then a button is shown, then click the button and select Trigger type > ClipsAndTacks MC > Cancel Trigger, click OK
- 12) In the second row of the Metric Value Expressions table, select the cell under column Expression, then a button is shown, then click the button and enter this text:
 

```
xs:dayTimeDuration (Order_End_Time - Order_Start_Time)
```
- 13) Click OK.

\_\_ f. Press **Ctrl-S** to save your work.

**▼ Metric Details**  
 Edit the details of the metric, which is a holding spot for information used in other calculations.

ID:  Edit...

Name:

Description:

Type:  ▼

A value is required for this metric

Default Value:  Edit...

This metric can be used for sorting

**▼ Metric Value Expressions**  
 Specify the expressions that set the value of the metric. If a trigger is specified, the map is evaluated when the trigger fires.

Trigger	Expression
Ship Order to Customer Trigger	xs:dayTimeDuration (Order_End_Time - Order_Start_Time)
Cancel Trigger	xs:dayTimeDuration (Order_End_Time - Order_Start_Time)

Add Remove

\_\_ g. Click the **KPI Model** tab.

\_\_ h. Create a KPI context to store the KPIs.

- 1) In the navigation view of the KPI Model, right click ClipsAndTacks, then select **New → KPI Context**
  - a) Name :- **My KPI Context**
  - b) ID :- **My\_KPI\_Context**

\_\_ i. Create the average order fulfillment KPI.



---

Note that you will create the KPI based on events for August 2006. This will work for you no matter what month or year that you run this lab because the timestamps in the test data are based on August 2006, so your results will be the same as in these lab instructions.

---

- 1) In the navigation view of the KPI Model, right click **My KPI Context**, then select **New → KPI**
  - a) Name :- **Average Order Fulfillment KPI August 2006**
  - b) ID :- **Average\_Order\_Fulfillment\_KPI\_August\_2006**
- 2) Enter these values:
  - a) Type – Duration
  - b) Select 'Keep track of historical values for this KPI'
  - c) In KPI Target and Ranges, for Target click **Details...**, then change the value to 3 Days, then click OK
  - d) In KPI Target and Ranges, for Ranges select Actual value
  - e) In the Range table,
    - (1) Click Add, or double click anywhere in the table
    - (2) Change the Name to Day 1, and the ID to Day\_1
    - (3) A row is created in the Range table with a range name Day 1.
    - (4) For this row, select the cell in the Start value column and a button is displayed. Click this button, then change all values to 0, then click OK.
    - (5) For this row, select the cell in the End value column and a button is displayed. Click this button, then change Days to 1 and the other values to 0, then click OK.
  - f) Now you have created one range called Day 1. Repeat the above step to create four more ranges:
    - (1) Day 2 with start values 1 Days and end value 2 Days
    - (2) Day 3 with start values 2 Days and end value 3 Days
    - (3) Day 4 with start values 3 Days and end value 4 Days
    - (4) Day 5 with start values 4 Days and end value 5 Days
  - g) KPI Value – select '**Base this KPI on a metric and an aggregation function**'
  - h) Under KPI Details, for Monitoring context, click Browse..., then select '**ClipsAndTacks MC**', then click OK
  - i) Metric – click Browse..., then select **Order Fulfillment Duration**, then click OK
  - j) Aggregation function – select **Average**.
  - k) Under Time Filter, for Metric, click Browse..., then select **Order Start Time**, then click OK

l) Specify time period – **Fixed**

m) Start date – click Edit... then select date **2006-08-01**, time **00:00:00**, click OK

n) End date – click Edit..., then select **2006-08-31**, time **23:59:59**, click OK

3) Press **Ctrl-S** to save your work.

**▼ KPI Details**  
 Edit the details of the KPI, which is a performance measurement used to track business objectives.

ID:

Name:

Description:

Type:

Keep track of historical values for this KPI

**▼ KPI Target and Ranges**  
 Specify a target, which is an exact value for the KPI to achieve, or ranges against which to track the KPI, or both.

Target:

Ranges:

Range name	Start value	End value	Color
Day 1	0 Milliseconds	< 1 Days	
Day 2	1 Days	< 2 Days	
Day 3	2 Days	< 3 Days	
Day 4	3 Days	< 4 Days	
Day 5	4 Days	< 5 Days	

**▼ KPI Definition**  
Specify how the value of the KPI is set.

**KPI Value**

Choose how the KPI will get its value:

Base this KPI on a metric and an aggregation function.

Write an expression to calculate this KPI based on existing KPIs

**KPI Details**

Monitoring context: \* ClipsAndTacks MC

Metric: \* Order Fulfillment Duration

Aggregation function: \* Average

Use values from:  All model versions  Only this version of the model

**Time Filter**

Select a time period over which the KPI should be calculated.

Metric: Order Start Time

Time period:

None  Repeating  Rolling  Fixed

Start date: 2006-08-01T00:00:00  End date: 2006-08-31T23:59:59

Time zone: \* GMT-06:00  Location (daylight saving):

- \_\_\_ j. Note that this KPI as it stands averages order fulfillment time for shipped orders and cancelled orders, but you probably are not interested in the cancelled orders to be included in this KPI. So update the average order fulfillment KPI to show the values for shipped orders only. First you will create a new metric to hold the order status. Then you will add this metric as a data filter on the KPI.
- \_\_\_ k. Click the **Monitor Details Model** tab.
- \_\_\_ l. In the Monitor Details Model navigation view, right click ClipsAndTacks MC, then select **New** → **Metric**. Enter these values:
- 1) Name :- **Order Status**
  - 2) ID :- **Order\_Status**
  - 3) Type :- **String**
  - 4) Select the check box for '**A value is required for this metric**'
  - 5) Default Value (enter this text with single quotation marks) – '**New**'
  - 6) For Metric Value Expressions, click **Add**
  - 7) In the first row of the Metric Value Expressions table, select the cell under column Trigger, then a button is shown, then click the button and select Trigger type > ClipsAndTacks MC > Cancel Trigger, click OK
  - 8) In the first row of the Metric Value Expressions table, select the cell under column Expression, then a button is shown, then click the button and enter this text with the quotation marks:

'Cancelled'

- 9) Click OK.
- 10) For Metric Value Expressions, click Add
- 11) In the second row of the Metric Value Expressions table, select the cell under column Trigger, then a button is shown, then click the button and select Trigger type > ClipsAndTacks MC > Ship Order to Customer Trigger, click OK
- 12) In the second row of the Metric Value Expressions table, select the cell under column Expression, then a button is shown, then click the button and enter this text with the quotation marks:

'Shipped'

- 13) Press **Ctrl-S** to save your work

**▼ Metric Details**

Edit the details of the metric, which is a holding spot for information used in other calculations.

ID:  Edit...


Name:

Description:

Type:  ▼

Maximum String Length:

Allocate additional space in database to accommodate Unicode string for globalization

A value is required for this metric 

Default Value:  Edit...

This metric can be used for sorting

**▼ Metric Value Expressions**

Specify the expressions that set the value of the metric. If a trigger is specified, the map is evaluated when the trigger fires.

Trigger	Expression
Cancel Trigger	'Cancelled'
Ship Order to Customer Trigger	'Shipped'

\_\_\_ m. Update the order fulfillment KPI to use the order status as a filter.

- 1) Click the **KPI Model tab**.
- 2) In the KPI Model navigation view, navigate to ClipsAndTacks > My KPI Context > Average Order Fulfillment KPI August 2006
- 3) In the Data Filter section for this KPI, click Add
- 4) In the dialog, navigate to **ClipsAndTacks MC → Order Status**, then click OK

- 5) In the row containing metric Order Status, make sure the operation defaults to 'equals'. Also, click the cell in the column Values, then enter this text (with quotation marks):

'Shipped'

- 6) Press **Ctrl-S** to save your work.

Metric	Operator	Values	Case-sensitive
Order Status	equals	'Shipped'	

19. Create a KPI for number of approved orders greater than 90%. First you will create a KPI to count the number of total orders. Then you will create a KPI to count the number of shipped orders. Finally, you will create a KPI that calculates the approval percentage as this formula: (ship count / order count) \* 100.

- a. Click the **KPI Model** tab then create the order count KPI.

- 1) In the navigation view of the KPI Model, right click My KPI Context, then select **New > KPI**

- a) Name :- **Order Count KPI**
- b) ID :- **Order\_Count\_KPI**

- 2) Enter these values:

- a) Type :- **Decimal**
- b) Select 'Keep track of historical values for this KPI'
- c) Do not specify targets nor ranges
- d) KPI Value – select '**Base this KPI on a metric and an aggregation function**'
- e) Under KPI Details, for Monitoring context, click Browse..., then select '**ClipsAndTacks MC**', then click OK
- f) Metric – click Browse..., then select **ClipsAndTacks Key**, then click OK
- g) Aggregation function – select **Count**

- 3) Press **Ctrl-S** to save your work.

- b. Create the ship count KPI.

- 1) In the navigation view of the KPI Model, right click My KPI Context, then select **New > KPI**

- a) Name :- **Ship Count KPI**
- b) ID :- **Ship\_Count\_KPI**

- 2) Enter these values:

- a) Type :- **Decimal**
- b) Do not specify targets nor ranges
- c) KPI Value – select '**Base this KPI on a metric and an aggregation function**'

- d) Under KPI Details, for Monitoring context, click Browse..., then select **'ClipsAndTacks MC'**, then click OK
- e) Metric – click Browse..., then select **ClipsAndTacks Key**, then click OK
- f) Aggregation function – select **Count**
- g) In the Data Filter section for this KPI, click **Add**
- h) In the dialog, navigate to **ClipsAndTacks MC → Order Status**, then click OK
- i) In the row containing metric Order Status, make sure the operation defaults to 'equals'. Also, click the cell in the column Values, then enter this text (with quotation marks):

**'Shipped'**

3) Press **Ctrl-S** to save your work.

\_\_ c. Create the percent of orders approved KPI.

1) In the navigation view of the KPI Model, right click My KPI Context, then select **New > KPI**

- a) Name :- **Percent of Orders Approved KPI**
- b) ID :- **Percent\_of\_Orders\_Approved\_KPI**

2) Enter these values:

- a) Type :- **Decimal**
- b) Select 'Keep track of historical values for this KPI'
- c) In KPI Target and Ranges, for Target click Details..., then change the value to **90**, then click OK
- d) In KPI Target and Ranges, for Ranges select **Actual value**
- e) In the Range table,
  - (1) Click **Add**
  - (2) Change the Name to **Low Range**, and the ID to **Low\_Range**
  - (3) A row is created in the Range table with a range name Low Range.
  - (4) For this row, select the cell in the Start value column and a button is displayed. Click this button, then change the value to 0 and then click OK.
  - (5) For this row, select the cell in the End value column and a button is displayed. Click this button, then change the value to 90 and then click OK.

f) Repeat the above step to create one more range:

- (1) **High Range** with start value 90 and end value 100

g) KPI Value – select **'Write an expression to calculate this KPI based on existing KPIs'**

h) For KPI Calculation, enter this text:

`fn:round((Ship_Count_KPI div Order_Count_KPI) * 100)`

3) Press **Ctrl-S** to save your work.

**▼ KPI Target and Ranges**

Specify a target, which is an exact value for the KPI to achieve, or ranges against which to track the KPI, or both.

Target:  Details...

Ranges:  ▼

Range name	Start value	End value
Low Range	0	< 90
High Range	90	< 100

Add Remove Sort

**▼ KPI Definition**

Specify how the value of the KPI is set.

**KPI Value**

Choose how the KPI will get its value:

Base this KPI on a metric and an aggregation function.

Write an expression to calculate this KPI based on existing KPIs

KPI Calculation

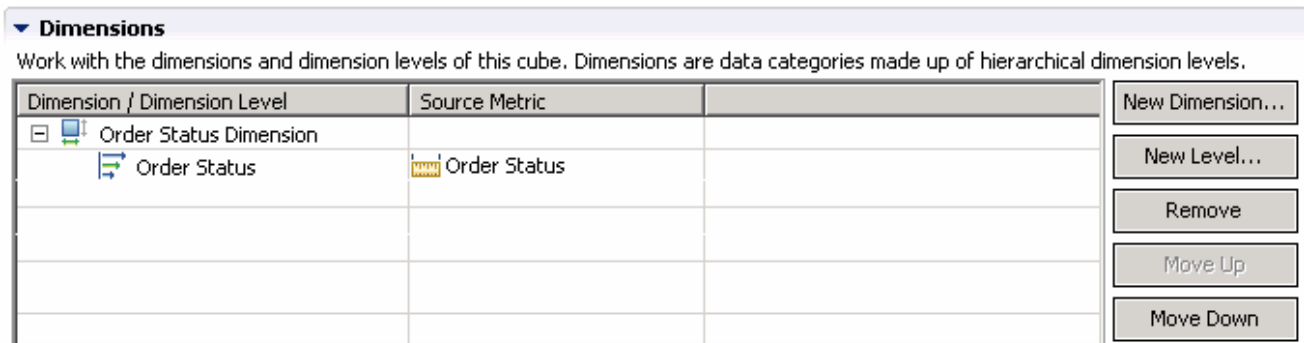
For example, you could have a Total Profit KPI that subtracts the Total Cost KPI from the Total Revenue KPI.

`fn:round((Ship_Count_KPI div Order_Count_KPI) * 100)`

\_\_\_ 20. In the dashboards, you might like to see the total order price, the average order price and the total number of orders. And you might be interested in seeing this information by country and city, and by order status. So now you will create dimensions and measures that allow you to see this information. For aggregated numeric information like total order price, you will create a measure in the dimensional model. For textual attributes like order status, you will create a dimension in the dimensional model. In order to see the country and city information, you will create a multi-level location dimension containing country as the first level and city as the second level. Each measure and dimension needs a metric as a source, so you will also need to create metrics for country, city and total price.

- \_\_\_ a. Click the **Dimensional Model** tab.
- \_\_\_ b. Add order status as a dimension to the Dimensional Model.
  - 1) In the navigation view, click ClipsAndTacks MC Cube
  - 2) In the dimensions table, click **New Dimension...**
    - a) Name :- **Order Status Dimension**
    - b) ID :- **Order\_Status\_Dimension**

- c) Click **OK**
- 3) Click **New Level...**
  - a) Name :- **Order Status**
  - b) ID :- **Order\_Status**
  - c) Source metric – Click Browse..., then select **ClipsAndTacks MC → Order Status**. Then click OK.
  - d) Click OK.
- 4) Press **Ctrl-S** to save your work.



- \_\_ c. Click the **Monitor Details Model tab**.
- \_\_ d. In the Monitor Details Model navigation view, right click ClipsAndTacks MC, then select **New > Metric**. Enter these values:
  - 1) Name :- **country**
  - 2) ID :- **country**
  - 3) Type :- **String**
  - 4) Select the check box for '**A value is required for this metric**'
  - 5) Default value : " (**Note:** Empty single quotation marks)
  - 6) For Metric Value Expressions, click Add
  - 7) In the first row of the Metric Value Expressions table, select the cell under column Trigger, then a button is shown, then click the button and select Trigger type > ClipsAndTacks MC > New Order Trigger, click OK
  - 8) In the first row of the Metric Value Expressions table, select the cell under column Expression, then a button is shown, then click the button and enter this text:  
 New\_Order\_Event/My\_Event\_Part/noe:NEWOrderBODData/ae:country
  - 9) Click OK.
  - 10) Press **Ctrl-S** to save your work



\_\_\_ e. In the Monitor Details Model navigation view, right click ClipsAndTacks MC, then select **New > Metric**. Enter these values:

- 1) Name :- **city**
- 2) ID :- **city**
- 3) Type :- **String**
- 4) Select the check box for '**A value is required for this metric**'
- 5) Default value : "" (**Note:** Empty single quotation marks)
- 6) For Metric Value Expressions, click Add
- 7) In the first row of the Metric Value Expressions table, select the cell under column Trigger, then a button is shown, then click the button and select Trigger type > ClipsAndTacks MC > New Order Trigger, click OK
- 8) In the first row of the Metric Value Expressions table, select the cell under column Expression, then a button is shown, then click the button and enter this text:  
  
New\_Order\_Event/My\_Event\_Part/noe:NEWOrderBOData/ae:city
- 9) Click OK.
- 10) Press **Ctrl-S** to save your work

\_\_\_ f. In the Monitor Details Model navigation view, right click ClipsAndTacks MC, then select **New > Metric**. Enter these values:

- 1) Name :- **totalPrice**
- 2) ID :- **totalPrice**
- 3) Type :- **Decimal**
- 4) For Metric Value Expressions, click Add
- 5) In the first row of the Metric Value Expressions table, select the cell under column Trigger, then a button is shown, then click the button and select Trigger type > ClipsAndTacks MC > New Order Trigger, click OK
- 6) In the first row of the Metric Value Expressions table, select the cell under column Expression, then a button is shown, then click the button and enter this text:  
  
New\_Order\_Event/My\_Event\_Part/noe:NEWOrderBOData/ae:totalPrice
- 7) Click OK.
- 8) Press **Ctrl-S** to save your work

\_\_\_ g. Click the **Dimensional Model tab**.

\_\_\_ h. Add location as a dimension to the Dimensional Model.

- 1) In the navigation view, click ClipsAndTacks MC Cube
- 2) In the dimensions table, click **New Dimension...**

- a) Name :- **Location**
  - b) ID :- **Location**
  - c) Click **OK**
- 3) Click **New Level...**
- a) Name :- **Country**
  - b) ID :- **Country**
  - c) Source metric – Click Browse..., then select **ClipsAndTacks MC → country**. Then click OK.
  - d) Click OK.
- 4) Click **New Level...**
- a) Name :- **City**
  - b) ID :- **City**
  - c) Source metric – Click Browse..., then select **ClipsAndTacks MC → city**. Then click OK.
  - d) Click OK.
- 5) Press **Ctrl-S** to save your work.

▼ Dimensions		
Work with the dimensions and dimension levels of this cube. Dimensions are data categories made up of hierarchical dimension levels.		
Dimension / Dimension Level	Source Metric	
[-] [Icon] Order Status Dimension		New Dimension... New Level... Remove Move Up Move Down
[Icon] Order Status	[Icon] Order Status	
[Icon] Location		
[Icon] Country	[Icon] country	
[Icon] City	[Icon] city	

- \_\_\_ i. Ensure that the **Dimensional Model tab** is selected.
- \_\_\_ j. In the Dimensional Model, add a measure for average order price.
  - 1) In the navigation view, click ClipsAndTacks MC Cube
  - 2) In the Measures table, click **New ...**
    - a) Name :- **Average Order Price**
    - b) ID :- **Average\_Order\_Price**
    - c) Source metric - Click Browse..., then select **ClipsAndTacks MC → totalPrice**.
    - d) Aggregation function: - select **Average**.

e) Then click OK.

3) Press **Ctrl-S** to save your work.

**Measures**  
Work with the measures for this cube. Measures are calculations based on a metric, key, counter, or stopwatch.

Measure	Source Metric	Aggregation Function	
Average Order Price	totalPrice	Average	

\_\_ k. In the Dimensional Model, add a measure for sum of order price.

1) In the navigation view, click ClipsAndTacks MC Cube

2) In the Measures table, click **New ...**

a) Name :- **Sum Order Price**

b) ID :- **Sum\_Order\_Price**

c) Source metric - Click Browse..., then select **ClipsAndTacks MC → totalPrice.**

d) Aggregation function – select **Sum.**

e) Then click OK.

3) Press **Ctrl-S** to save your work.

\_\_ l. In the Dimensional Model, add a measure for order count.

1) In the navigation view, click ClipsAndTacks MC Cube

2) In the Measures table, click **New ...**

a) Name :- **Order Count**

b) ID :- **Order\_Count**

c) Source metric - Click Browse..., then select **ClipsAndTacks MC → ClipsAndTacks Key.**

d) Aggregation function – select **Count.**

e) Then click OK.

3) Press **Ctrl-S** to save your work.

**Measures**  
Work with the measures for this cube. Measures are calculations based on a metric, key, counter, or stopwatch.

Measure	Source Metric	Aggregation Function	
Average Order Price	totalPrice	Average	
Sum Order Price	totalPrice	Sum	
Order Count	ClipsAndTacks Key	Count	

\_\_\_ 21. In the dashboards you might like to see the average order price and track that relative to a target.

\_\_ a. Create a KPI for the average order price.

1) Click the **KPI Model tab**.

2) In the navigation view of the KPI Model, right click My KPI Context, then select **New → KPI**

a) Name :- **Average Order Price KPI (Dollars)**

b) ID :- **Average\_Order\_Price\_KPI\_\_x0028\_Dollars\_x0029\_**

3) Enter these values:

a) Type :- **Decimal**

b) Select 'Keep track of historical values for this KPI'

c) In KPI Target and Ranges, for Target click Details..., then change the value to **300**, then click OK

d) In KPI Target and Ranges, for Ranges select **Actual value**

e) In the Range table,

(1) Click **Add**

(2) Change the Name to **Low Range**, and the ID to **Low\_Range**

(3) A row is created in the Range table with a range name Low Range.

(4) For this row, select the cell in the Start value column and a button is displayed. Click this button, then change the value to 0, and then click OK.

(5) For this row, select the cell in the End value column and a button is displayed. Click this button, then change the value to 300, and then click OK.

f) Now you have created one range called Low Range. Repeat the above step to create one more range:

(1) High Range with start value 300 and end value 1000

g) KPI Value – select '**Base this KPI on a metric and an aggregation function**'

h) Under KPI Details, for Monitoring context, click Browse..., then select '**ClipsAndTacks MC**', then click OK

i) Metric – click Browse..., then select **totalPrice**, then click OK

j) Aggregation function – select **Average**.

4) Press **Ctrl-S** to save your work.

**▼ KPI Target and Ranges**

Specify a target, which is an exact value for the KPI to achieve, or ranges against which to track the KPI, or both.

Target:

Ranges:  ▼

Range name	Start value	End value	
Low Range	0	< 300	
High Range	300	< 1000	

**▼ KPI Definition**

Specify how the value of the KPI is set.

**KPI Value**

Choose how the KPI will get its value:

Base this KPI on a metric and an aggregation function.

Write an expression to calculate this KPI based on existing KPIs

**KPI Details**

Monitoring context:

Metric:

Aggregation function:  ▼

\_\_\_ 22. **(Optional)** To demonstrate the use of a counter, you can optionally use this section to create a counter which will be incremented if you ship the order (versus canceling the order). For every monitoring context, this counter will either be zero or one depending on whether the order is cancelled or shipped. Counters would be more useful in other scenarios – perhaps to count the number of times an iterated activity is executed.

\_\_\_ a. In the Monitor Details Model navigation view, right click ClipsAndTacks MC, then select **New > Counter**. Enter these values:

- 1) Name :- **Ship Counter**
- 2) ID :- **Ship\_Counter**
- 3) For Counter Controls, click Add, then select **ClipsAndTacks MC → Ship Order to Customer Trigger**, click OK
- 4) Note that the Resulting Action defaults to **Add One**, which is correct.

\_\_\_ b. Press **Ctrl-S** to save your work

**▼ Counter Details**  
 Edit the details of the counter, which counts the number of occurrences of some situation or event.


ID: \*

Name:

Description:

This counter can be used for sorting

**▼ Counter Controls**  
 Specify what causes the counter to change and what action is taken.

Trigger / Inbound Event	Resulting Action
 Ship Order to Customer Trigger	Add One

- \_\_\_ 23. **(Optional)** ) To demonstrate the use of a stopwatch, you can optionally use this section to create a timer which will measure the duration of the monitoring context. You might find this useful in the dashboard instances view to quickly see which monitoring contexts have not been terminated.
- \_\_\_ a. In the model navigation view, right click ClipsAndTacks MC, then select **New → Stopwatch**. Enter these values:
- 1) Name :- **Monitoring Context Timer**
  - 2) ID :- **Monitoring\_Context\_Timer**
  - 3) For Stopwatch Controls, click Add, then select **ClipsAndTacks MC → New Order Trigger** and then click OK.
  - 4) Note that the default Resulting Action is **Start**
  - 5) For Stopwatch Controls, click Add, then select **ClipsAndTacks MC → Ship Order to Customer Trigger** and then click OK.
  - 6) Note that the default Resulting Action is Start, but you need to change it to **Stop** for this trigger. So click the second row in column Resulting Action. A drop down box is shown, so pick **Stop** in the list.
  - 7) For Stopwatch Controls, click Add, then select **ClipsAndTacks MC → Cancel Trigger** and then click OK.
  - 8) Note that the default Resulting Action is Start, but you need to change it to **Stop** for this trigger. So click the second row in column Resulting Action. A drop down box is shown, so pick **Stop** in the list.
- \_\_\_ b. Press **Ctrl-S** to save your work.

**▼ Stopwatch Details**

Edit the details of the stopwatch, which keeps track of elapsed time. If an accumulating stopwatch is given two or more start and stop times, the durations are added together to produce an accumulated time.

ID: \*Monitoring\_Context\_Timer Edit...

Name: Monitoring Context Timer




Description: ▲  
▼


This stopwatch is an accumulating stopwatch

This stopwatch can be used for sorting

**▼ Stopwatch Controls**

Specify what causes the stopwatch to change and what action is taken.

Trigger / Inbound Event	Resulting Action
 New Order Trigger	Start
 Ship Order to Customer Trigger	Stop
 Cancel Trigger	Stop

- \_\_\_ 24. Create a business situation event when the number of declined orders is greater than or equal to 3. First you will create an event definition for the outbound situation event. You will create a field called BusinessSituationName which is a required field for the outbound situation event. You will create a KPI that will track the total number of declined orders. Then you will create a trigger which is fired when the number of declined orders is greater than or equal to 3. Then you will define an outbound event which is fired by the trigger. In this outbound event, you will provide a value for the field BusinessSituationName which will be used later when configuring action manager in the server administrative console.
- \_\_\_ a. Create an event definition for declined orders.
- 1) In Project Explorer, right click ClipsAndTacks and select **New > Event Definition...(cbe)**
  - 2) For file name, enter DeclinedOrderEvent.cbe
  - 3) Click Finish
  - 4) The Event Definition Editor for DeclinedOrderEvent opens.
  - 5) Double click the hierarchical icon next to the label **Parent**.
  - 6) On the **Select Event Definition** Dialog double click **event** under Matching events.
  - 7) **Event** is now shown as the parent of **DeclinedOrderEvent**
  - 8) On the event definition, click the icon for Add Extended Data . This adds a row to the Extended Data table.
  - 9) For the new data element, change the Extended Data name from 'data1' to 'BusinessSituationName' by clicking on data1 and typing the new name.
  - 10) For the new element, change the type to string. Do this by clicking on 'noValue' and then you are given a drop down box.

11) Press **Ctrl-S** to save your work.

Name*	DeclinedOrderEvent	
Parent*	event	
	Property	
	Extended Data	Type
	BusinessSituationName	string

\_\_ b. Create a KPI for declined orders.

- 1) Click the **KPI Model tab** for the ClipsAndTacks model.
- 2) In the navigation view of the KPI Model, right click My KPI Context, then select **New > KPI**
  - a) Name – Declined Order KPI
  - b) ID - Declined\_Order\_KPI
- 3) Enter these values:
  - a) Type – Decimal
  - b) Select 'Keep track of historical values for this KPI'
  - c) In KPI Target and Ranges, for Target click Details..., then change the value to 3, then click OK
  - d) In KPI Target and Ranges, for Ranges select Actual value
  - e) In the Range table,
    - (1) Click Add
    - (2) Change the Name to Low Range, and the ID to Low\_Range
    - (3) A row is created in the Range table with a range name Low Range.
    - (4) For this row, select the cell in the Start value column and a button is displayed. Click this button, then change the value to 0, and then click OK.
    - (5) For this row, select the cell in the End value column and a button is displayed. Click this button, then change the value to 3, and then click OK.
  - f) Now you have created one range called Low Range. Repeat the above step to create one more range:
    - (1) High Range with start value 3 and end value 10
  - g) KPI Value – select 'Base this KPI on a metric and an aggregation function.'
  - h) Under KPI Details, for Monitoring context, click Browse..., then select 'ClipsAndTacks MC', then click OK
  - i) Metric – click Browse..., then select 'ClipsAndTacks Key', then click OK



- j) Aggregation function – select Count.
- k) In the Data Filter section for this KPI, click Add
- l) In the dialog, navigate to ClipsAndTacks MC > Order Status, then click OK
- m) In the row containing metric Order Status, make sure the operation defaults to 'equals'. Also, click the cell in the column Values, then enter this text (with quotation marks):

'Cancelled'

- 4) Press **Ctrl-S** to save your work.

**KPI Definition**

Specify how the value of the KPI is set.

**KPI Value**

Choose how the KPI will get its value:

Base this KPI on a metric and an aggregation function.

Write an expression to calculate this KPI based on existing KPIs

---

**KPI Details**

Monitoring context:

Metric:

Aggregation function:

Use values from:  All model versions  Only this version of the model

---

**Time Filter**

Select a time period over which the KPI should be calculated.

Metric:

Time period:  None  Repeating  Rolling  Fixed

---

**Data Filter**

Select the metrics that you want to use to determine what values to use in the calculation. For example, if you have a KPI called Average Price in London, you only want to use monitoring contexts where the value of the City metric is London.

Metric	Operator	Values	Case-sensitive
<input type="button" value="A"/> Order Status	equals	<input type="text" value="'Cancelled'"/>	<input type="checkbox"/>

\_\_\_ c. Create a trigger to fire the outbound situation event.

- 1) In the navigation view of the KPI Model, right click My KPI Context, then select **New > Trigger**
  - a) Name – Declined Order Trigger
  - b) ID - Declined\_Order\_Trigger
- 2) Enter these values:
  - a) Under Trigger Sources, click Add.
  - b) Select Recurring wait time, then click OK

c) In the Source column you see the default wait time is 1 minute.

d) In Trigger Condition, enter this text:

Declined\_Order\_KPI >= 3

3) Press **Ctrl-S** to save your work.

**▼ Trigger Details**  
 Edit the details of the trigger, which detects an occurrence and initiates an action in response.


ID:

Name:

Description:

Trigger is repeatable

**▼ Trigger Sources**  
 Specify the source of this trigger.

Source Type	Source
Recurring wait time	 0 days 0 hours 1 minutes

**▼ Trigger Condition**  
 Specify the condition that determines whether the trigger will fire.

\_\_\_ d. Create an outbound situation event for declined orders.

1) In the navigation view of the KPI Model, right click My KPI Context, then select **New > Outbound Event**

- a) Name – Declined Order Outbound Event
- b) ID - Declined\_Order\_Outbound\_Event
- c) Select 'Configure this event to generate an alert in the dashboards'
- d) For trigger, click Browse..., then select My KPI Context > Declined Order Trigger, then click OK

2) Enter these values:

- a) For extension name, click Browse..., then select ClipsAndTacks > DeclinedOrderEvent.cbe > DeclinedOrderEvent, then click OK
- b) Under Event Attributes Details, click Add
- c) Select My KPI Context > Declined Order Trigger, then click OK
- d) In the table, expand Extended Data, then on the same row as BusinessSituationName, set Expression to (include the quotation marks):  
'Too many orders have been declined'

3) Press **Ctrl-S** to save your work.

**▼ Outbound Event Details**  
 Edit the details of the outbound event, which is sent by the monitoring context. The type must be an event definition.

ID: \*Declined\_Order\_Outbound\_Event Edit...

Name: Declined Order Outbound Event

Description:

---

**▼ Event Type Details**  
 Specify the event type or the XML schemas that together describe the structure of this outbound event. You can specify an extension name, event parts, or both.

Extension name: DeclinedOrderEvent Browse... Clear

Event parts:

ID	Name	Type	Path

Add Remove

---


**▼ Event Attributes Details**  
 Specify the triggers that cause the event to be sent. Use the Expression column to specify the value for each event attribute when the event is sent.



Name	Type	Expression
Declined Order Trigger		
Property Data		
Extended Data		
BusinessSituationName	string	'Too many orders have been declined'

25. Create a business situation event when the order fulfillment duration is greater than 3 days. First you will create an event definition for the outbound situation event. You will create a field called BusinessSituationName which is a required field for the outbound situation event, and you will create another field for the average order processing time which will eventually be placed into the alert body. You do not need to create a KPI since there is already a KPI containing the average processing time. You will create a trigger which is fired when the average processing time is greater than 3 days. Then you will define an outbound event which is fired by the trigger. In this outbound event, you will provide a value for the field BusinessSituationName which will be used later when

configuring action manager in the server administrative console, and you will populate the field on the event containing the average order processing time.

\_\_\_ a. Create an event definition for average order processing time.

- 1) In Project Explorer, right click ClipsAndTacks and select **New > Event Definition...(cbe)**
- 2) For file name, enter LateAverageOrderShippedEvent.cbe
- 3) Click Finish
- 4) The Event Definition Editor for LateAverageOrderShippedEvent opens.
- 5) Double click the hierarchical icon next to the label **Parent**.
- 6) On the **Select Event Definition** Dialog double click **event**.
- 7) **Event** is now shown as the parent of **LateAverageOrderShippedEvent**
- 8) On the event definition, click the icon for Add Extended Data . This adds a row to the Extended Data table.
- 9) For the new data element, change the Extended Data name from 'data1' to 'BusinessSituationName' by clicking on data1 and typing the new name.
- 10) For the new element, change the type to string. Do this by clicking on 'noValue' and then you are given a drop down box.
- 11) On the event definition, click the icon for Add Extended Data. This adds a second row to the Extended Data table.
- 12) For the second data element, change the Extended Data name to AverageOrderProcessingTime
- 13) For the second element, change the type to string.
- 14) Press **Ctrl-S** to save your work.

Name*	LateAverageOrderShippedEvent	
Parent*	 event	
	Property	
	Extended Data	Type
	BusinessSituationName	string
	AverageOrderProcessingTime	string

\_\_\_ b. Create a trigger to fire the outbound situation event.

- 1) In the navigation view of the KPI Model, right click My KPI Context, then select **New > Trigger**
  - a) Name – Order Fulfillment Timer Trigger
  - b) ID - Order\_Fulfillment\_Timer\_Trigger
- 2) Enter these values:

- a) Under Trigger Sources, click Add.
- b) Select Recurring wait time, then click OK
- c) In the Source column you see the default wait time is one minute.
- d) In Trigger Condition, enter this text:

Average\_Order\_Fulfillment\_KPI\_August\_2006 ge dayTimeDuration  
(‘P3DT1H’)

3) Press **Ctrl-S** to save your work.

**▼ Trigger Details**  
 Edit the details of the trigger, which detects an occurrence and initiates an action in response.


ID: \* Order\_Fulfillment\_Timer\_Trigger

Name: Order Fulfillment Timer Trigger

Description:

Trigger is repeatable

**▼ Trigger Sources**  
 Specify the source of this trigger.

Source Type	Source
Recurring wait time	 0 days 0 hours 1 minutes

**▼ Trigger Condition**  
 Specify the condition that determines whether the trigger will fire.

Average\_Order\_Fulfillment\_KPI\_August\_2006 ge dayTimeDuration(‘P3DT1H’)

\_\_\_ c. Create an outbound situation event for average order processing time.

1) In the navigation view of the KPI Model, right click My KPI Context, then select **New > Outbound Event**

- a) Name – Order Fulfillment Outbound Event
- b) ID - Order\_Fulfillment\_Outbound\_Event
- c) Select 'Configure this event to generate an alert in the dashboards'
- d) For trigger, click Browse..., then select My KPI Context > Order Fulfillment Timer Trigger, then click OK

2) Enter these values:

- a) For extension name, click Browse..., then select ClipsAndTacks > LateAverageOrderShippedEvent.cbe > LateAverageOrderShippedEvent, then click OK
- b) Under Event Attributes Details, click Add
- c) Select My KPI Context > Order Fulfillment Timer Trigger, then click OK
- d) In the table, expand Extended Data, then on the same row as BusinessSituationName, set Expression to (include the quotation marks):  
  
'Average shipment is too late'
- e) In the table, expand Extended Data, then on the same row as AverageOrderProcessingTime, set Expression to:  
  
`xs:string(xs:decimal(Average_Order_Fulfillment_KPI_August_2006) div 86400)`

---

Note: Here the decimal function converts the duration to seconds, so then it can be divided by 86400 which will give you the number of days.

---

3) Press **Ctrl-S** to save your work.

**▼ Outbound Event Details**

Edit the details of the outbound event, which is sent by the monitoring context. The type must be an event definition.

ID: \*Order\_Fulfillment\_Outbound\_Event Edit...

Name: Order Fulfillment Outbound Event

Description:

**▼ Event Type Details**

Specify the event type or the XML schemas that together describe the structure of this outbound event. You can specify an extension name, event parts, or both.

Extension name: LateAverageOrderShippedEvent Browse... Clear

Event parts:

ID	Name	Type	Path

Add Remove

**▼ Event Attributes Details**

Specify the triggers that cause the event to be sent. Use the Expression column to specify the value for each event attribute when the event is sent.

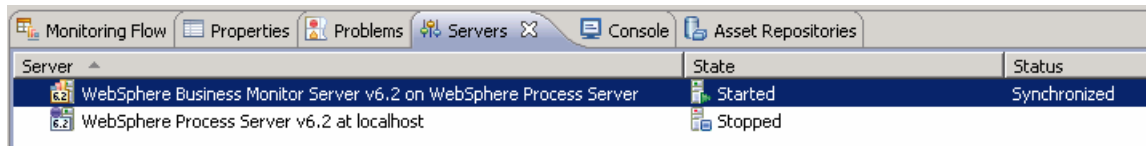
Name	Type	Expression
Order Fulfillment Timer Trigger		
Property Data		
Extended Data		
BusinessSituationName	string	'Average shipment is too late'
AverageOrderProcessingTime	string	string(decimal(Average_Order_Fulfillment_KPI_August_2006) div 86400)

- \_\_\_\_ 26. Check for any errors in the Problems view. You should resolve any errors before continuing. Warnings and informational messages may be present but these will not be a problem.

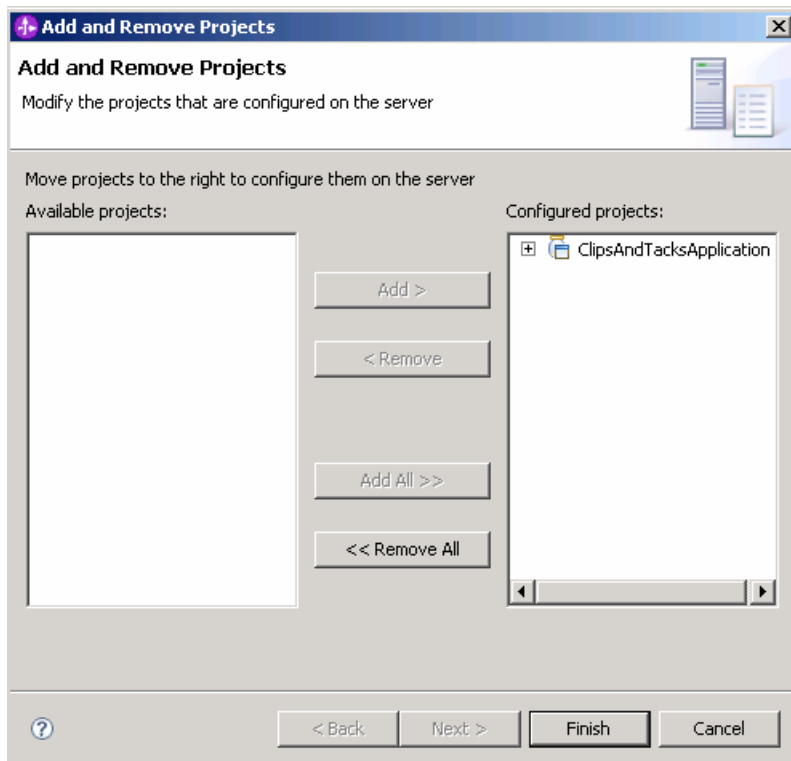
## Part 2: Publish the model to the server

In this section you will use Rational Application Developer or WebSphere Integration Developer to publish the monitor model to the monitor server.

- \_\_\_ 27. In Project Explorer, expand **ClipsAndTacks** → **Monitor models** → **ClipsAndTacks.mm**. Right click over **ClipsAndTacks.mm** and then select **Generate Monitor J2EE Projects** from the pop-up menu
- \_\_\_ 28. Select **Finish**.
- \_\_\_ 29. A progress dialog shows the status of the operation and it closes when the operation is complete. Check for errors in the Problems view. There may be warnings, but there should not be any errors. If you see errors, then try to perform a clean to see if the errors can be removed: Project > Clean... > select 'Clean all projects', then click OK.
- \_\_\_ 30. Click the Servers tab, then right click and select the **Start** option to start the server WebSphere Business Monitor V6.2. This may take a few minutes to complete.

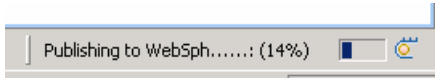


- \_\_\_ 31. Right click in the servers view, then select **Add and Remove Projects....**
- \_\_\_ 32. Click **Add** to move the ClipsAndTacksApplication from the list of available projects to the list of configured projects.





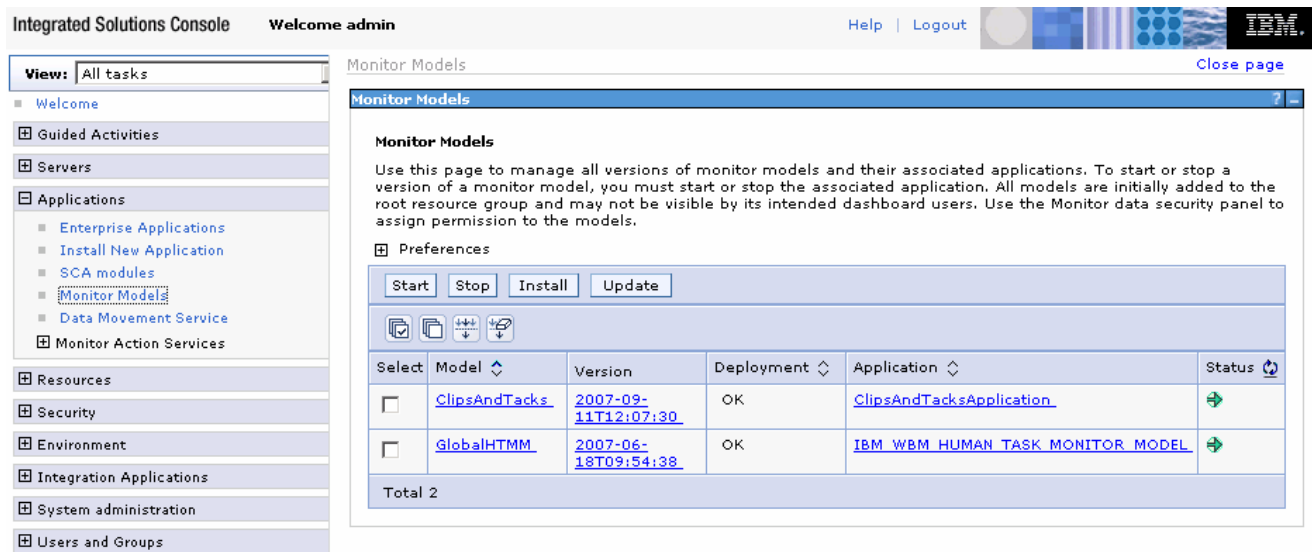
- \_\_\_ 33. Click **Finish**.
- \_\_\_ 34. A progress message is displayed in the lower right corner of the window.



- \_\_\_ 35. Check the messages in the console view. You should see the following message when the application has been started:

Application started: ClipsAndTacksApplication

- \_\_\_ 36. In the servers view, right click, then select **Administration > Run administrative console**. You should see it open in a separate tab.
- \_\_\_ 37. It will prompt you for user ID and password. Enter 'admin' without the quotation marks for user ID, then enter 'admin' without the quotation marks for password. Click **Log in**
- \_\_\_ 38. Click **Applications > Monitor Models**. The application should show green status if it started successfully.



- \_\_\_ 39. If the model shows red (stopped), then wait a moment, then refresh by clicking on the icon to the right of Status in the last column of the table. You should see green (started) for the model. If it does not show green, be patient and keep refreshing until it does show green.
- \_\_\_ 40. Check the server log to ensure there are no problems. You can check this in the console view.
- \_\_\_ 41. If you are using the integrated server within WebSphere Integration Developer or Rational Application Developer, then you do not need to setup Monitor data security, since the administrative user is automatically authorized to all models. If you are using a different server, then you should open the administrative console, navigate to **Security > Monitor Data Security**, then add the model, role and user information to a resource group.

---

## Part 3: Configure action manager for business situation events

This section shows you how to configure action manager to create alerts based on business situation events defined in the monitor model. The alerts will be visible in the Alert view on your dashboard.

You do not need an LDAP server to test the situation events. When you configure the notification template, you can specify a dashboard login rather than a user registry.

In this section, you will use the administrative console to configure the alerts. However, you could also create business user alerts using the Alert Manager widget in the dashboard. In a production environment, an end user could use the alert manager widget to manage the business user alerts; however, an IT administrator would manage the modeled alerts in the administrative console.

- \_\_\_ 1. Add a template for declined order alerts.
  - \_\_\_ a. In the administrative console, navigate to **Applications > Monitor Action Services > Template Definitions > Notifications**
  - \_\_\_ b. Click **New**.
  - \_\_\_ c. Enter **AlertDeclined** as Template name and a description.
  - \_\_\_ d. Select **Dashboard Alert**.
  - \_\_\_ e. Select **User id**.
  - \_\_\_ f. For the **To** field, enter **admin**. This 'User id' is the one that will receive the alert, so you should log into the dashboard with this 'User id'.

**Note:** The user **admin** is the default username configured for Monitor Test Environment profile
  - \_\_\_ g. Enter a subject.
  - \_\_\_ h. Enter the body.

**General Properties**

\* Template name  
AlertDeclined

Description  
Declined Orders

**Default action service type**

Dashboard Alert  
 Cell phone  
 Email  
 Pager

**'To' query type**

Federated repositories query  
 LDAP query  
 Email address  
 User id

To  
admin ←

Query base

Subject  
Declined Orders

Body  
Too many orders have been declined.

\_\_\_ i. Click **OK**.

\_\_\_ 2. Add a template for order processing time alerts.

\_\_\_ a. In the admin console, navigate to **Applications → Monitor Action Services → Template Definitions → Notifications**

\_\_\_ b. Click **New**.

\_\_\_ c. Enter **AlertLate** as Template name and a description.

\_\_\_ d. Select **Dashboard Alert**

\_\_\_ e. Select **User id**

\_\_\_ f. For the **To** field, enter **admin**. This 'User id' is the one that will receive the alert, so you should log into the dashboard with this 'User id'.

**Note:** The user **admin** is the default username configured for Monitor Test Environment profile

\_\_\_ g. Enter a subject.

- \_\_\_ h. Enter the body, and you may optionally enter a substitution variable as shown below. For the actual alert that is sent, the average order processing time from the event payload will be substituted for %AverageOrderProcessingTime% in the body. For example:

The average order processing time is %AverageOrderProcessingTime% days.

**General Properties**

\* Template name  
AlertLate

Description  
Order Processing time

**Default action service type**

Dashboard Alert  
 Cell phone  
 Email  
 Pager

**'To' query type**

Federated repositories query  
 LDAP query  
 Email address  
 User id

To  
admin ←

Query base

Subject  
Order processing time

Body  
The average order processing time is % AverageOrderProcessingTime% days.

- \_\_\_ i. Click **OK**.

- \_\_\_ 3. Add the binding from the situation event to the action type for declined orders.

- \_\_\_ a. In the admin console, navigate to **Applications → Monitor Action Services → Installed Situation Event Bindings**

- \_\_\_ b. Click **New**.

- \_\_\_ c. Enter the situation event name that you defined in the model. In this lab, you created a business situation name as follows:

**Too many orders have been declined**

New Situation Event Binding

**General Properties**

\* Situation event name

Description

\_\_\_ d. This should match exactly the value in the BusinessSituationName field in the outbound event. You may want to copy/paste from the model. Here is a screen capture from the model showing the Declined Order Outbound Event:

**Outbound Event Details**  
 Edit the details of the outbound event, which is sent by the monitoring context. The type must be an event definition.

ID: \* Declined\_Order\_Outbound\_Event

Name: Declined Order Outbound Event

Description:

**Event Type Details**  
 Specify the event type or the XML schemas that together describe the structure of this inbound event.

Extension name: DeclinedOrderEvent

Event parts:

ID	Name	Type	Path

**Event Attributes Details**  
 Specify the triggers that cause the event to be sent. Use the Expression column to specify the value for each event attrit

Name	Type	Expression
Declined Order Trigger		
Property Data		
Extended Data		
BusinessSituationName	string	'Too many orders have been declined'

\_\_\_ e. Enter a description, then click **Apply**.

\_\_\_ f. Click **Add**.

\_\_\_ g. Enter a binding name, then select the template **AlertDeclined**

Add template to situation event binding

**General Properties**

\* Binding name

Description

Category

Template name

Apply  Reset Cancel

\_\_\_ h. Click **OK**.

\_\_\_ i. Notice that you now have one action defined for this situation event. If you had other action templates defined then you could add more actions to this event. So then you could send a notification for this situation to multiple destinations including e-mail, alerts and Web services.

**General Properties**

\* Situation event name

Description

Apply  Reset Cancel

**Preferences**

Add Remove				
Select	Binding Name	Category Name	Template Name	Action Service Type
<input type="checkbox"/>	Declined		AlertDeclined	AlertHandler
Total 1				

\_\_\_ j. Click **OK**.

\_\_\_ 4. Add the binding from the situation event to the action type for order processing time.

\_\_\_ a. In the admin console, navigate to **Applications > Monitor Action Services > Installed Situation Event Bindings**

\_\_\_ b. Click **New**.

\_\_\_ c. Enter the situation event name that you defined in the model. In this lab, you created a business situation name as follows:

**Average shipment is too late**

\_\_\_ d. . This should match exactly the value in the BusinessSituationName field in the outbound event. You may want to copy/paste from the model

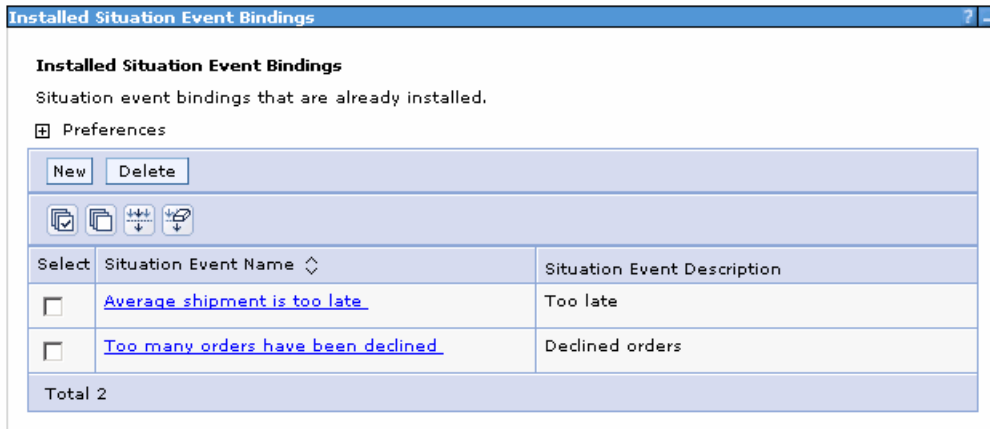
\_\_\_ e. Enter a description, then click **Apply**.

\_\_\_ f. Click **Add**.

\_\_\_ g. Enter a binding name, then select the template **AlertLate**, then click **OK**.

\_\_\_ h. Click **OK**.

\_\_\_ 5. In the admin console, navigate to **Applications → Monitor Action Services → Installed Situation Event Bindings**. You should see two bindings as follows:



## Part 4: Run events to exercise the model

Rather than installing a J2EE application to actually create the events that you want to monitor, you are going to use a program to simulate the submission of events from the application. The supplied program is 'BatchCBEWriter61' and it will submit the events to the Common Event Infrastructure. Look for the program in \Labfiles62\ClipsAndTacks\BatchCBEWriter. This program reads XML files which represent the Common Base Events for the model.

- \_\_\_ 1. Update BatchCBEWriter61.bat to point WAS\_HOME to the monitor server home, for example '**set WAS\_HOME=C:\IBM\WebSphere\MonServer**'. If you are using the integrated monitor server in WebSphere Integration Developer, the path would be **<WID\_HOME>\runtimes\bi\_v62**. If you are using the integrated monitor server in Rational Application Developer, the path would be **<RAD\_HOME>\runtimes\base\_v62**.
- \_\_\_ 2. Update **config.properties**, setting the **serverName** and **portNumber**. You can find the port number by browsing the server log and finding '**bootstrap port**'. For example, check for the log at C:\IBM\WebSphere\MonServer\profiles\WBMon01\logs\server1\SystemOut.log. For the integrated server in WebSphere Integration Developer, the path would be **<WID\_HOME>\pf\WBMonSrv\_wps\logs\server1\SystemOut.log** and the portNumber should be 2810. For the integrated server in Rational Application Developer, the path would be **<RAD\_HOME>\runtimes\base\_v62\profiles\WBMonSrv\logs\server1\SystemOut.log** and the portNumber should be 2810. Here is an example of the config.properties settings:
  - \_\_\_ a. connect.serverName = localhost
  - \_\_\_ b. connect.portNumber = 2810
- \_\_\_ 3. Open a command window, then change directory to the folder containing BatchCBEWriter61, for example, type this command
  - \_\_\_ a. cd \Labfiles62\ClipsAndTacksXSD\BatchCBEWriter
- \_\_\_ 4. Run commands to load the common base events to the server.
  - \_\_\_ a. batchcbewriter61 -Dsource.filename="C:/Labfiles62/ClipsAndTacksXSD/cbe/allXSDevents.xml"
- \_\_\_ 5. When you run BatchCBEWriter61, you should see results such as:

```

C:\WINDOWS\system32\cmd.exe
C:\Labfiles61\ClipsAndTacksXSD\BatchCBEWriter>batchcbewriter61 -Dsource.filename
="C:/Labfiles61/ClipsAndTacksXSD/cbe/allXSDevents.xml"
Getting CBEs.
Getting Emitter.
Removing GlobalInstanceIds.
Setting missing values.
Changing Instance Ids.
Updating timestamps.
Validating CBEs.
Sending CBEs.
START=16:31:43.812
Sending cbe[10].
Sending cbe[20].
Sending cbe[30].
Sending cbe[40].
Sending cbe[50].
Sending cbe[60].
END=16:31:58.203
TotalTime=14391 milliseconds.
C:\Labfiles61\ClipsAndTacksXSD\BatchCBEWriter>_

```



## Part 5: Create a dashboard

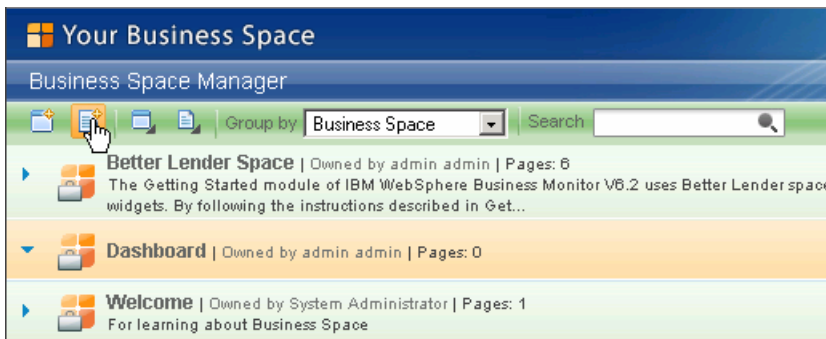
In this section you will build a dashboard in business space. You will add widgets to the dashboard and configure them.

\_\_\_ 6. Create the dashboard.

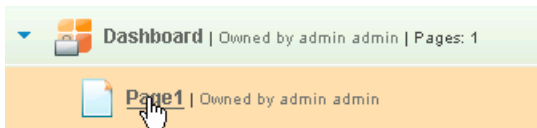
- \_\_\_ a. The default browser is 'Internal Web Browser', but you should not use this one since some standard functions are not provided that you may need. Select Window > Preferences > General > Web browser > select 'Use external Web browser'. Select an external browser. Click OK.
- \_\_\_ b. In WebSphere Integration Developer, in the servers view, right click and select **Business Space**
- \_\_\_ c. When prompted, enter **admin** for the user ID and enter **admin** for the password. You must log in with 'admin' so that you can view the alerts which were setup in action manager to be sent to this particular user ID. Also, in the toolkit environment, this is the user that is automatically defined on the secured server.
- \_\_\_ d. Click the icon to create a new space.



- \_\_\_ e. Enter a name such as Dashboard, select Empty, then click **OK**.
- \_\_\_ f. Click the icon to create a new page in the Dashboard space.



- \_\_\_ g. Enter a name such as Page1, select Empty, then click **OK**.
- \_\_\_ h. Click Page1 to open it.

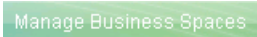


\_\_\_ i. Now you can add widgets to the page by clicking on 'Add widgets' in the upper right corner



, then dragging a widget to the page. If you want to rename the page, change the layout or remove the page from display, use the icons to the right of the page name on the tab




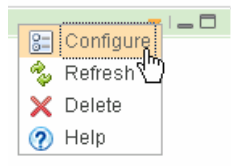
. To return to the business space list, click 'Manage Business Spaces' in the upper right corner .

\_\_\_ 7. Add the instances widget to see monitored instances.

\_\_\_ a. Click **Add widgets**, then select **Instances** and drag it to the page.

\_\_\_ b. Click **Configure** on the widget menu.

1) To Access the menu, hover the mouse over the right corner of the widget  and then click the down arrow. On this menu you can configure the widget, refresh the widget and delete the widget.



\_\_\_ c. Click the **Show/Hide** tab, and select monitoring context 'ClipsAndTacks MC'.



Click button >> to move all the fields to the Selected box. Also, click button 'Set as default' which will make this the default monitoring context to display.

\_\_\_ d. Then click **OK**. You should see a list of monitoring context instances for the events that you just processed.

Instances										
city	ClipsAndTacks Key	COMPLETED	country	CreationTime	Monitoring Context Timer	Order End Time	Order Fulfillment Duration	Order Start Time	Order Status	Sh
Raleigh	o1	★	USA	January 5, 2009 1:44:41 PM	0 s	August 9, 2006 3:20:05 AM	2 d, 0 h, 0 m, 0 s	August 7, 2006 3:20:05 AM	Shipped	1
Toronto	o2	★	Canada	January 5, 2009 1:44:41 PM	0 s	August 11, 2006 4:14:55 AM	4 d, 0 h, 0 m, 0 s	August 7, 2006 4:14:55 AM	Shipped	1
Mexico City	o3		Mexico	January 5, 2009 1:44:41 PM	17 m, 24 s		0 s	August 7, 2006 5:01:22 AM	New	0
Raleigh	o4		USA	January 5, 2009 1:44:41 PM	17 m, 24 s		0 s	August 10, 2006 6:01:34 AM	New	0
Toronto	o5		Canada	January 5, 2009 1:44:41 PM	17 m, 24 s		0 s	August 12, 2006 7:12:22 AM	New	0
Raleigh	o6	★	USA	January 5, 2009 1:44:41 PM	0 s	August 26, 2006 3:20:05 AM	4 d, 0 h, 0 m, 0 s	August 22, 2006 3:20:05 AM	Cancelled	0
Toronto	o7	★	Canada	January 5, 2009 1:44:41 PM	0 s	August 27, 2006 4:14:55 AM	5 d, 0 h, 0 m, 0 s	August 22, 2006 4:14:55 AM	Cancelled	0
Mexico City	o8		Mexico	January 5, 2009 1:44:41 PM	17 m, 24 s		0 s	August 22, 2006 5:01:22 AM	New	0
Raleigh	o9		USA	January 5, 2009 1:44:41 PM	17 m, 24 s		0 s	August 22, 2006 6:01:34 AM	New	0
Toronto	o10		Canada	January 5, 2009 1:44:41 PM	17 m, 24 s		0 s	August 22, 2006 7:12:22 AM	New	0

\_\_\_ e. There are two pages of results, so if you browse through these instances, you see that there are 4 cancelled orders. Since you setup an alert if declined orders is greater than or equal to 3, then an alert should be displayed in the alert view.

\_\_\_ f. Add the alerts widget to see business alerts.

- 1) Click **Add widgets**, then select **Alerts** and drag it to the page. The alerts view is added to your dashboard, and you should see alerts listed since the model processed three or more cancelled orders, and the average order processing time is greater than three days.

Alerts	
<input type="button" value="Mark Read"/> <input type="button" value="Mark Unread"/> <input type="button" value="Forward Alert"/> <input type="button" value="Remove"/>	
<input type="checkbox"/>	Subject
<input type="checkbox"/>	<ul style="list-style-type: none"> <li>• Order processing time</li> </ul>
<input type="checkbox"/>	<ul style="list-style-type: none"> <li>• Declined orders</li> </ul>
<input type="text" value="1 - 2"/> 2	

- 2) You can click an alert in the list, then you will get a pop-up showing the details of the alert.

<input type="checkbox"/>	Subject	Date ar
<input type="checkbox"/>	• Order processing time	January
<input type="checkbox"/>	• Declined orders	January

January 5, 2009 2:49:20 PM

**Subject**  
Order processing time

**Business situation name**  
Average shipment is too late

**Comments**  
The average order processing time is 3.750 days.

## What you did in this exercise

In the lab, you created a monitor model using Rational Application Developer or WebSphere Integration Developer.

Then you published the model to the monitor server.

You also configured action manager to create alerts based on business situation events.

You used the supplied program to simulate the submission of events from the monitored application.

You configured dashboards and viewed monitored data.

## Appendix 1 – Import the solution into the monitor model editor

A solution has been provided so that you do not have to build the model from scratch. This section shows you how to import the monitor model into Rational Application Developer or WebSphere Integration Developer. After importing the model then you can proceed to the section to publish the model to the server.

- \_\_\_ 1. Start Rational Application Developer or WebSphere Integration Developer and setup the environment.
  - \_\_\_ a. Start Rational Application Developer or WebSphere Integration Developer, and when prompted point to a new workspace such as **C:\workspaces\ClipsAndTacksXSD**
  - \_\_\_ b. Close the Welcome tab
  - \_\_\_ c. By default, you are in the Business Integration perspective. You want to open the Business Monitoring perspective. Select **Window → Open Perspective → Other → Show all → Business Monitoring**.
  - \_\_\_ d. If it asks you to Confirm Enablement, then click **OK**.
- \_\_\_ 2. Create a new monitoring project.
  - \_\_\_ a. Right click the Project Explorer and then select **New → Business Monitoring Project...**
  - \_\_\_ b. For Project name, enter **ClipsAndTacks**
  - \_\_\_ c. Click Finish. You will see the new project in the Project Explorer view.
- \_\_\_ 3. Import the supplied mm and cbe files.
  - \_\_\_ a. Right click in the Project Explorer view, and then select **Import...**
  - \_\_\_ b. Select **General → File system**, click **Next**.
  - \_\_\_ c. Browse to the location containing the files, for example, **C:\Labfiles62\ClipsAndTacksXSD**
  - \_\_\_ d. Select **ActivityEvent.xsd**
  - \_\_\_ e. Select **ClipsAndTacks.mm**
  - \_\_\_ f. Select **DeclinedOrderEvent.cbe**
  - \_\_\_ g. Select **LateAverageOrderShippedEvent.cbe**
  - \_\_\_ h. Select **NewOrderEvent.xsd**
  - \_\_\_ i. Click **Finish**.
- \_\_\_ 4. Expand the project in the Project Explorer view, then expand the Event Definitions and you will see the new events listed. Expand Monitor Models and you will see the new ClipsAndTacks model listed.
- \_\_\_ 5. Now you may proceed to the section to publish the model to the server.