SAP Integration Workshop 2007

Implement Business Monitor Model using WebSphere Integration Developer 6.0.2

Author: Khirallah Birkler

Version 1.0

Table of Contents

Key steps in this lab

1. **Build the process Monitor Model**

   In this lab you will start in WebSphere Integration Developer with the finished BPEL ***Update Marital Status Process***. Remember, the BPEL process came originally from the Model tool and will now be enhanced to include also business monitoring capabilities. On top of this process you will create the Monitor Model based on Common Base Events that are defined in the BPEL process.

2. **Define the Monitor Detail Model**

   These steps define the low level monitor elements like triggers, stopwatches, and metrics.

3. **Define the Data Mart Model**

   These steps define the data mart with its cubes for the data you want to visualize in the monitor dashboards.

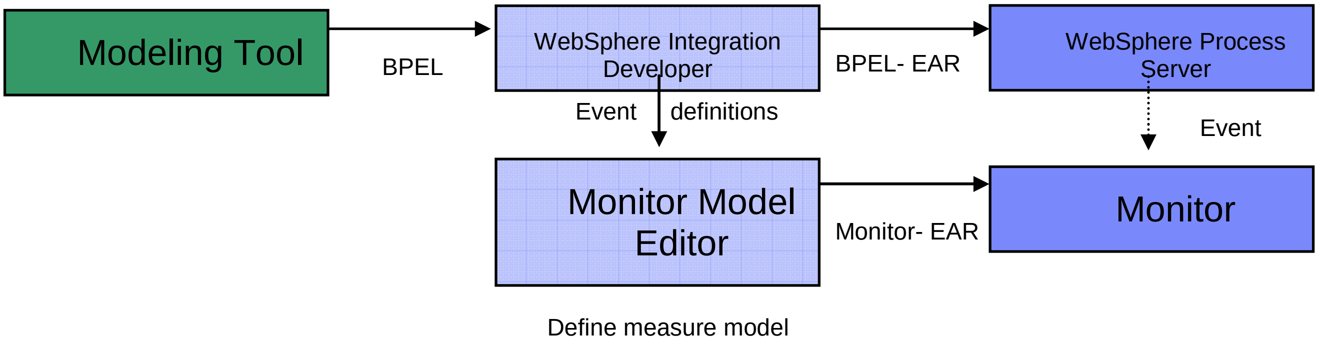
4. **Define the KPI model**

   In this Lab you will create the KPI from scratch and connect them to the Data Mart Model.

5. **Generate the Monitor EAR file**

   The final step is use the Monitor model editor to generate the EAR file for WebSphere Business Monitor.

The following figure shows the overall dependencies of the Monitor Model:



KPIs defined for the business scenario

The following KPIs have been defined by the business analysts. You will implement them in WebSphere Integration Developer with the Monitor Model Editor (MME).

Measurement area	**Customer satisfaction**
Strategic Goal	Increase Customer Satisfaction by 15%
Business Objective	Reduce the amount of time to run the ***Update Marital Status Process***, thereby reducing the amount of time that a employee waits to be notified if their Marital status has been updated in the backend systems. Also, automate the routing of erroneous requests to the correct Supervisor who is able to correct the problems.
KPI 1	**Average Process Duration Indicator**
KPI Definition	Average time to update the Marital status in the SAP HR backend system and to provide a notification to the employee.
Business Measure	Average Process Duration (Account Verification process)
Target/Benchmark/Range	Target = 14 business hours. (**Target according to process simulation**
Derived from (metrics in process Model)	Sum of the Process Duration for all completed ***Update Marital Status*** process instances in a given time period divided by the number of ***Update Marital Status*** process instances in the same time period.

The first KPI is very simple. You want to measure the duration of the ***Update Marital Status*** process. Whether the request was accepted, adjusted or rejected does not matter. What matters is the employee receives a response within a target of 14 hours (which is based on simulation results of the To-Be process). If you are able to achieve the target of 14 hours or less, you will likely increase customer satisfaction by 15%. The previous amount of time was 16.5 hours (based on simulation). If you lower it by 15% it is estimated that you will increase customer satisfaction by 15%. If you lower the business hours by more than 2.5 hours, your customer satisfaction will continue to go up.

Measurement area	**Financial (Profitability, Growth, Value)**
Strategic Goal	Increase Revenue by 30%
Business Objective	Executing more status updates in a shorter time period and increasing in the same procedure the quality of the data entered into the SAP HR backend system. Whenever a backend system reports an error a supervisor is involved to adjust the data entered or to reject the request.
KPI 2	**Percentage of automatic updates Indicator**
KPI Definition	The percentage of Marital status updates that run through the process without the involvement of a supervisor.
Business Measure	**Percentage of automatic updates**
Target/Benchmark/Range	Benchmark = 60%. Target = 80%.
Derived from (metrics in process Model)	

This KPI is very straight forward. You want to know the percentage of new status updates that passed the process without any involvement of a supervisor.

The goal here is to maximize the accounts accepted so you can generate more revenue. Based on estimates, if you can increase the number of customers by 30% you will also increase your revenue by 30%.

Note: This lab is intended to be completed in 60 minutes.

Generate the Monitor Model from the process

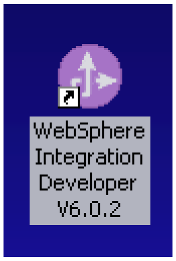
Launch the Lab VMWare Image

Make sure your TLE lab image is up and running. For details see the Lab instructions.

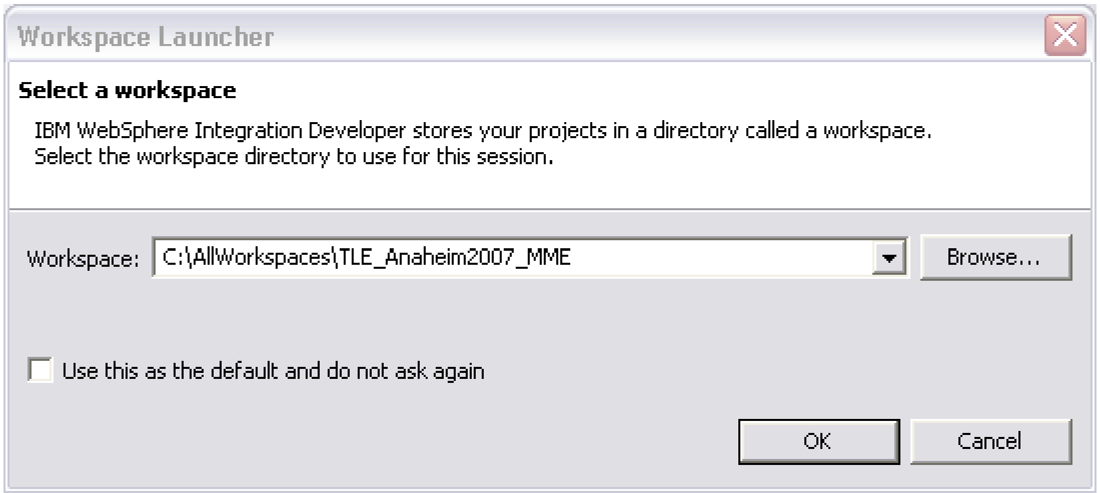
Open WebSphere Integration Developer

Note: In WebSphere Integration Developer, you will start on top of the finalized BPEL process. The workspace you start with contains refined **Update Marital status** process model.

On the Desktop open the WebSphere Integration Developer using the icon on the desktop.



Choose to open the prepared workspace for the TLE Anaheim lab you are going to complete.


Workspace Launcher

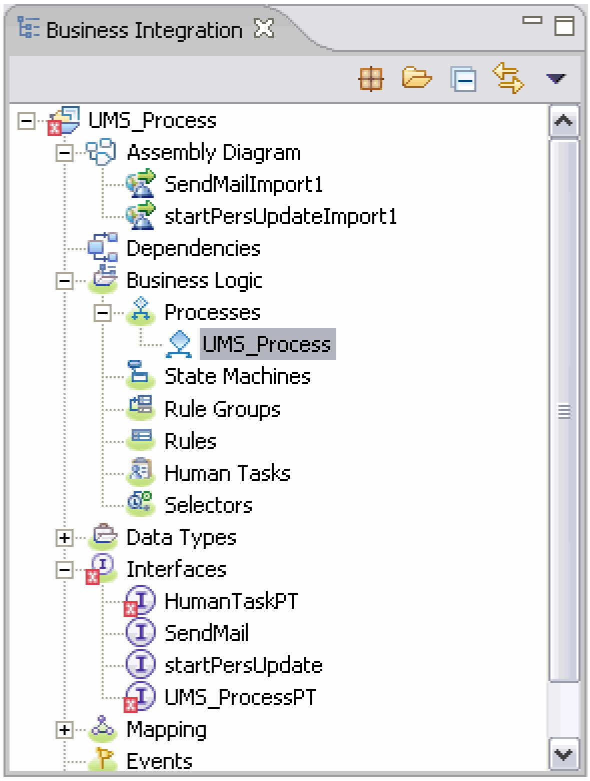
Select a workspace

IBM WebSphere Integration Developer stores your projects in a directory called a workspace.
Select the workspace directory to use for this session.

Workspace: C:\AllWorkspaces\TLE_Anaheim2007_MME

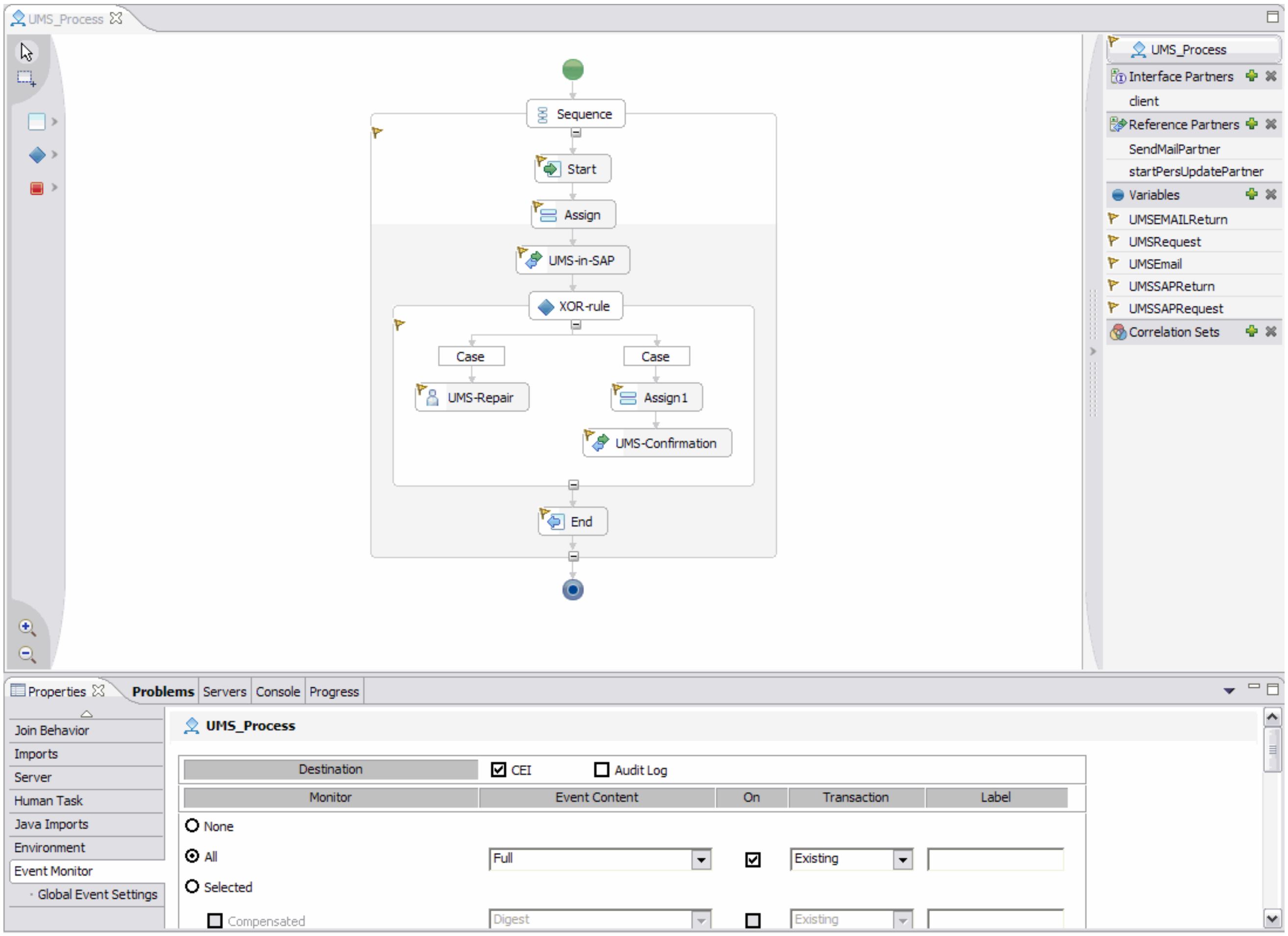
□ Use this as the default and do not ask again

After the WebSphere Integration Developer started you will see the Business Integration perspective containing a module with errors. These errors are caused because the online referenced WSDL files are not accessible – the server instance hosting these files is not started yet. So these errors will not negatively affect the lab. The Business Integration perspective will look like this:



Open the BPEL process named UMS_Process and verify that all common base events are flagged. Check the Event enablement on all levels, global BPEL process, variables, invocations …

A typical screen might look like this:



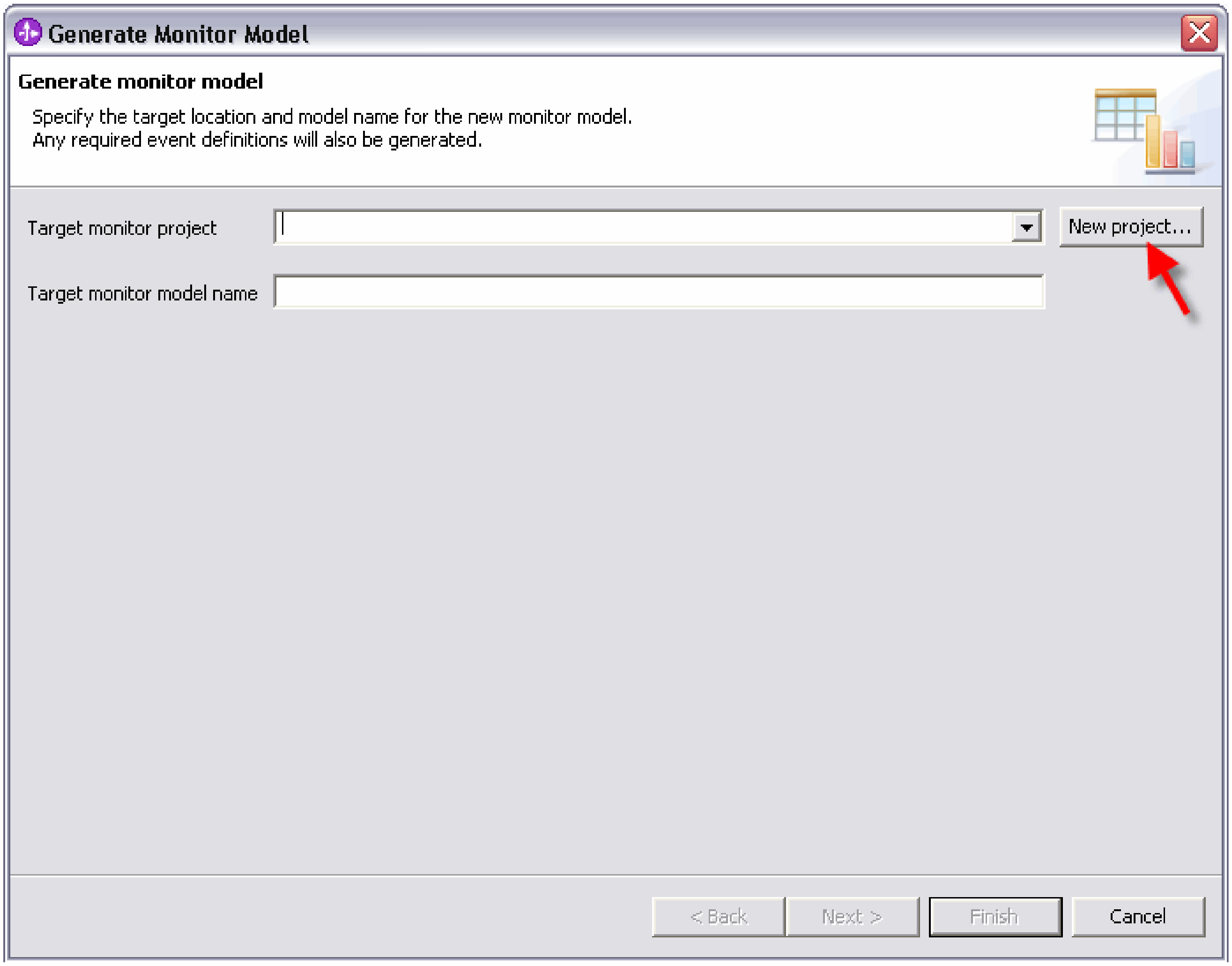
UMS_Process

Destination ☑ CEI □ Audit Log

Monitor	Event Content	On	Transaction	Label
○ None				
● All	Full	☑	Existing	
○ Selected				
□ Compensated	Digest	□	Existing	

Right-Click on the UMS_Process icon in the Business perspective tree and select **Monitor Tools → Generate Event Definitions.** All relevant events are created and stored in the same project in the subfolder /events. This folder is only visible in the Physical Resource view. The only indication that the Event generation ran successfully is to switch to this view and browse to the events folder.

Right-Click on the UMS_Process icon in the Business perspective tree and select **Monitor Tools → Generate Monitor Model.** Since you started with a clean workspace and no Monitor Model Project is available yet, press the New Project button in the upcoming screen:



Generate Monitor Model

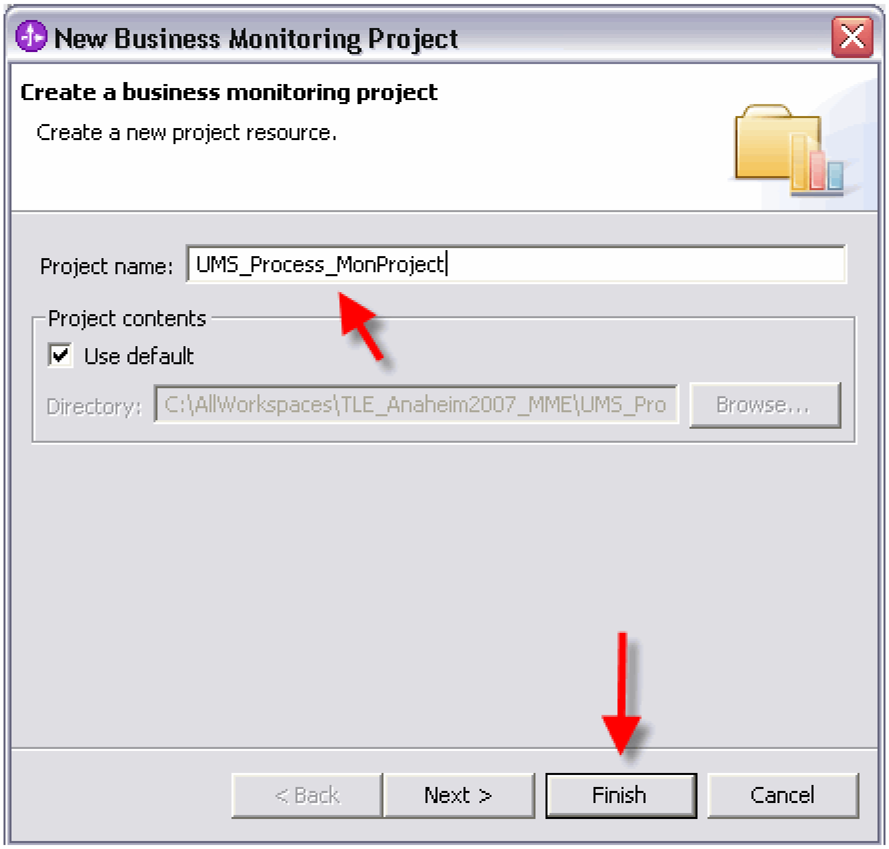
Generate monitor model

Specify the target location and model name for the new monitor model.
Any required event definitions will also be generated.

Target monitor project

Target monitor model name

Enter **UMS_Process_MonProject** as the project name and click on **Finish**.



New Business Monitoring Project

Create a business monitoring project

Create a new project resource.

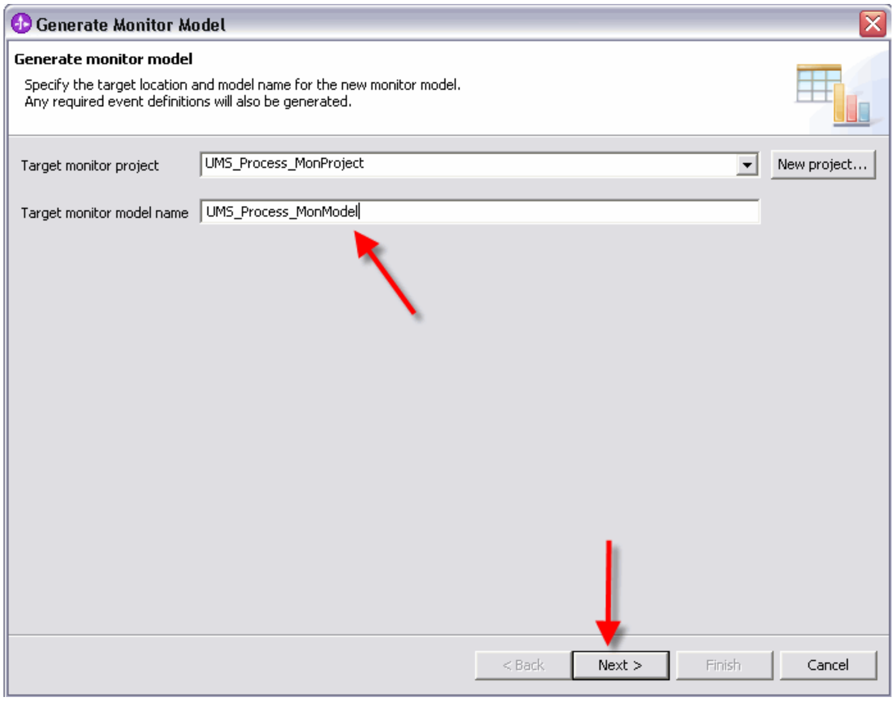
Project name: UMS_Process_MonProject

Project contents

☑ Use default

Directory: C:\AllWorkspaces\TLE_Anaheim2007_MME\UMS_Pro

Enter **UMS_Process_MonModel** as the Target monitor model name and click on **Next**.



Generate Monitor Model

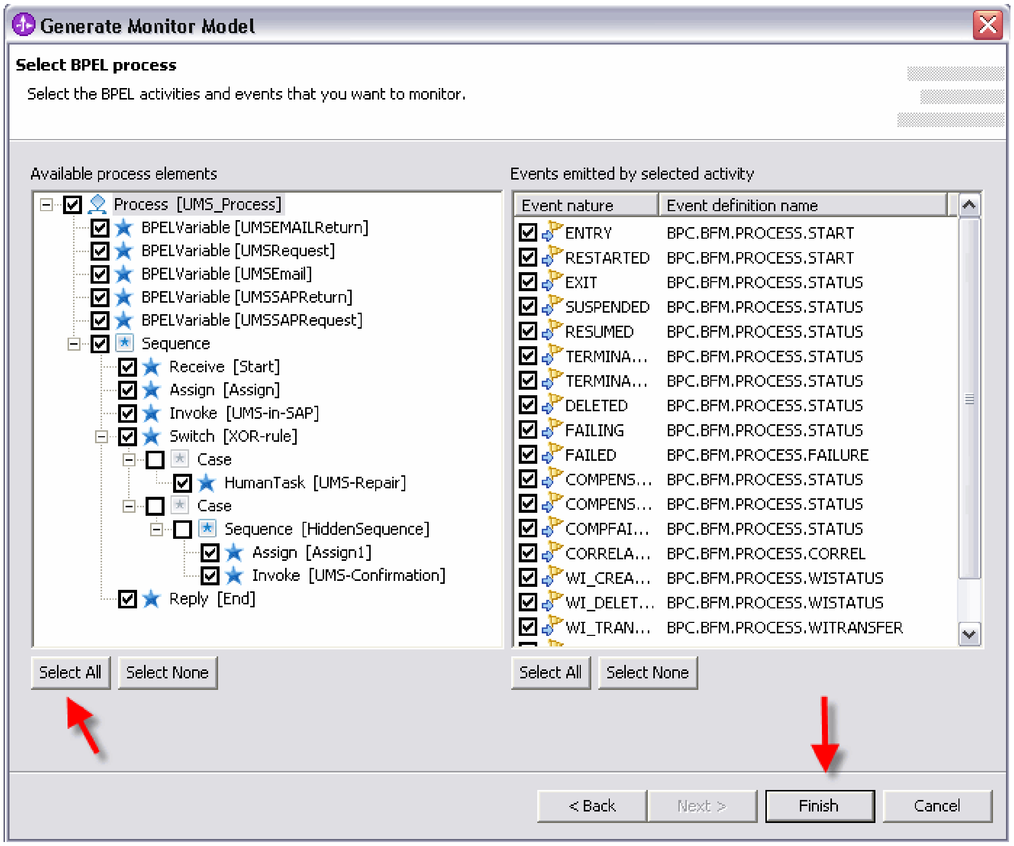
Generate monitor model

Specify the target location and model name for the new monitor model.
Any required event definitions will also be generated.

Target monitor project: UMS_Process_MonProject

Target monitor model name: UMS_Process_MonModel

Click on the **Select All** button to mark all events and click on Finish.



Generate Monitor Model

Select BPEL process

Select the BPEL activities and events that you want to monitor.

Available process elements

- ☑ Process [UMS_Process]
  - ☑ BPELVariable [UMSEMAILReturn]
  - ☑ BPELVariable [UMSRequest]
  - ☑ BPELVariable [UMSEmail]
  - ☑ BPELVariable [UMSSAPReturn]
  - ☑ BPELVariable [UMSSAPRequest]
  - ☑ Sequence
    - ☑ Receive [Start]
    - ☑ Assign [Assign]
    - ☑ Invoke [UMS-in-SAP]
    - ☑ Switch [XOR-rule]
      - □ Case
        - ☑ HumanTask [UMS-Repair]
      - □ Case
        - □ Sequence [HiddenSequence]
          - ☑ Assign [Assign1]
          - ☑ Invoke [UMS-Confirmation]
    - ☑ Reply [End]

Events emitted by selected activity

Event nature	Event definition name
☑ ENTRY	BPC.BFM.PROCESS.START
☑ RESTARTED	BPC.BFM.PROCESS.START
☑ EXIT	BPC.BFM.PROCESS.STATUS
☑ SUSPENDED	BPC.BFM.PROCESS.STATUS
☑ RESUMED	BPC.BFM.PROCESS.STATUS
☑ TERMINA...	BPC.BFM.PROCESS.STATUS
☑ TERMINA...	BPC.BFM.PROCESS.STATUS
☑ DELETED	BPC.BFM.PROCESS.STATUS
☑ FAILING	BPC.BFM.PROCESS.STATUS
☑ FAILED	BPC.BFM.PROCESS.FAILURE
☑ COMPENS...	BPC.BFM.PROCESS.STATUS
☑ COMPENS...	BPC.BFM.PROCESS.STATUS
☑ COMPFAI...	BPC.BFM.PROCESS.STATUS
☑ CORRELA...	BPC.BFM.PROCESS.CORREL
☑ WI_CREA...	BPC.BFM.PROCESS.WISTATUS
☑ WI_DELET...	BPC.BFM.PROCESS.WISTATUS
☑ WI_TRAN...	BPC.BFM.PROCESS.WITRANSFER

Note: At this point of the lab all Common Base Event definitions for your BPEL process have been generated automatically by the WebSphere Integration Developer tools. The created business monitoring project includes also a measure module that knows about all available events. It is now up to you to refine this measure model to use these events to fill the measure metrics, data cubes and KPIs accordingly.

Switch to the Business Monitoring perspective and open the created measure model.

The Monitor Model editor opens up with all the defaults generated for the Update Marital status process.

Open the **UMS_Process** tree (1) to view all the generated elements like keys, triggers and events. Make yourself familiar.

Click on the **Data Mart Model** (2). Notice all the default Cubes are generated.

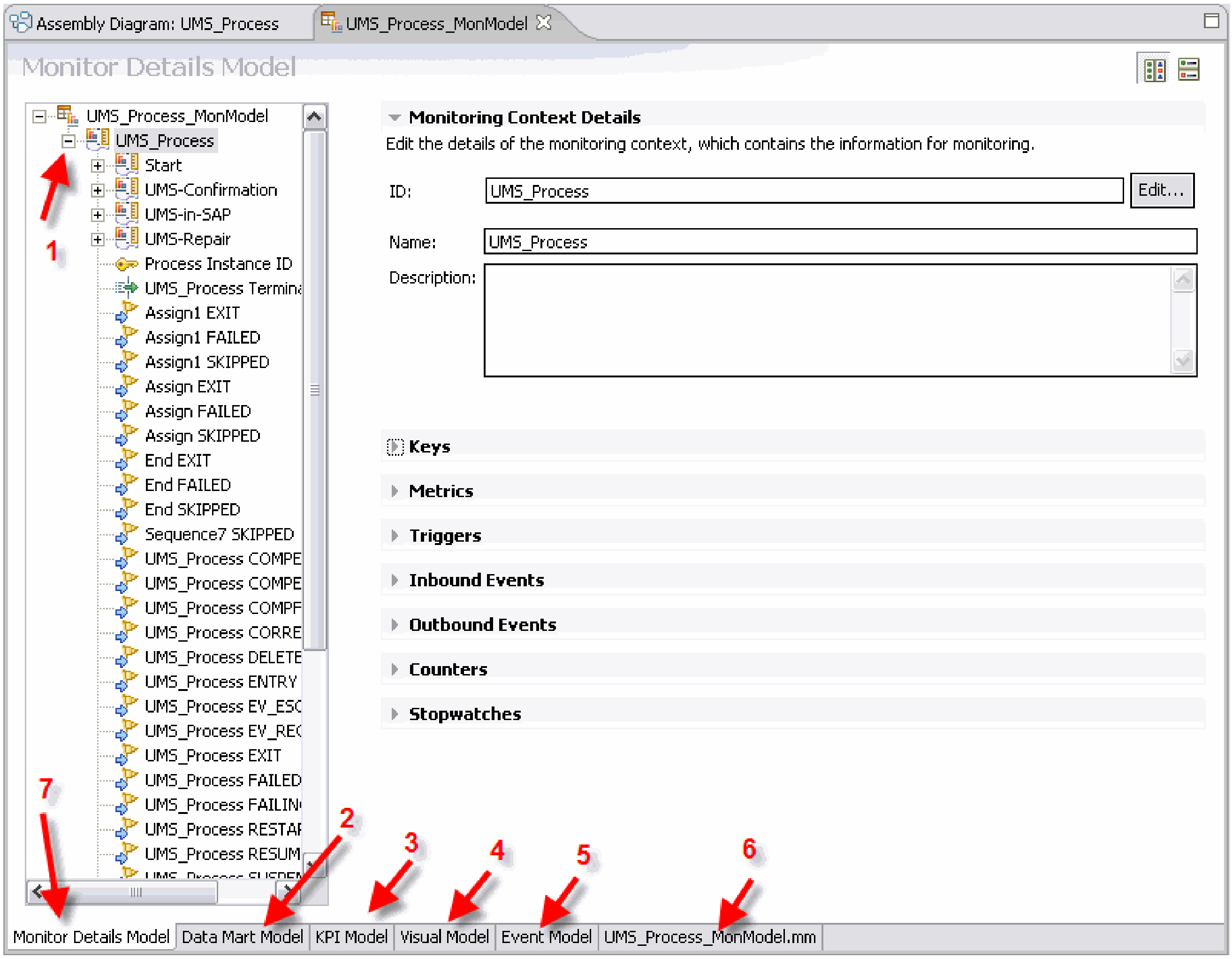
Click on the **KPI Model** tab (3). Notice by default there are no KPIs generated.

Click on the **Visual Model** tab (4). Notice here you can add a .SVG icon file for each activity.

Click on the **Event Model** tab (5). Notice all the default CEI events generated.

Click on the **UMS_Process_MonModel.mm** tab (6) to open the source code model in XML.

Click on **Monitor Detail Model** tab (7) to get back to the default model.



Monitoring Context Details

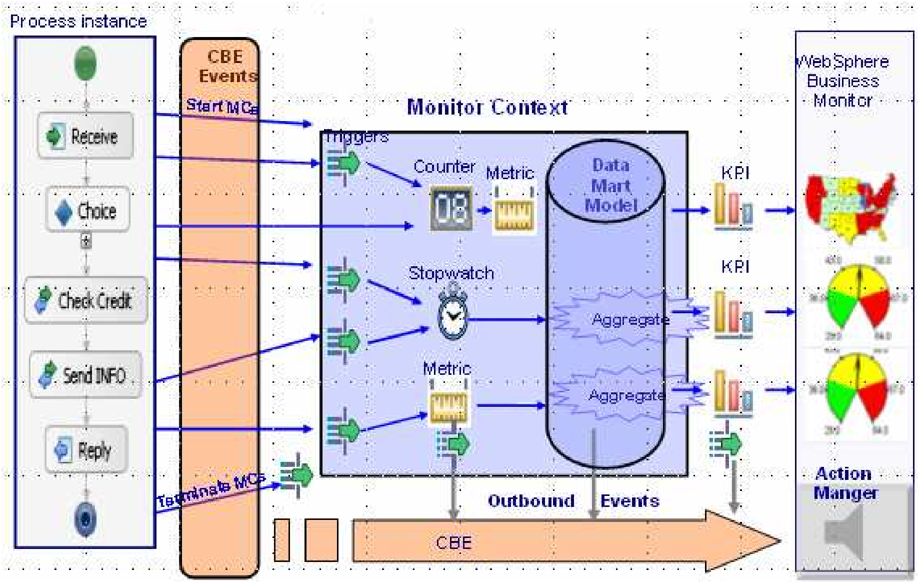
Edit the details of the monitoring context, which contains the information for monitoring.

ID: UMS_Process

Name: UMS_Process

Description:

Note: The **Generate Monitor Model** builds a default Monitor Context and default Data Mart Model. In the rest of this Lab you will add all other elements to finalize the monitor model as shown in the chart below.


Process instance
Receive
Choice
Check Credit
Send INFO
Reply
CBE Events
Start MCs
Terminate MCs
Monitor Context
Triggers
Counter
Metric
08
Stopwatch
Metric
Data Mart Model
Aggregate
Aggregate
KPI
KPI
WebSphere Business Monitor
Action Manger
Outbound Events
CBE

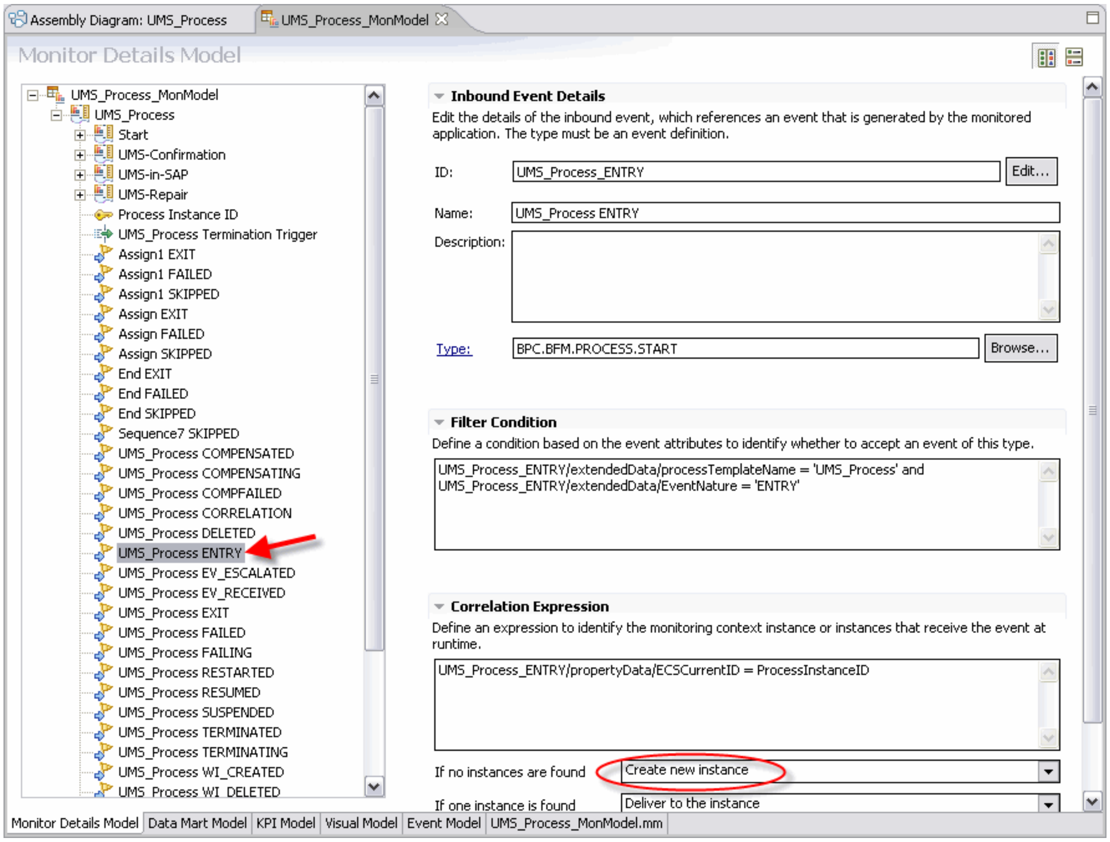
Define the Monitor Detail Model

The next steps define the low level monitor elements needed to gather data from the BPEL process CB events.

Check the Monitor Context (MC) generated by default

Note: First, check the *monitoring context definitions (MC)* to make sure all needed data is collected in monitoring context instances. By default, the generated model contains a MC for each activity in the process model and a MC for the overall process instance. Each monitoring context requires a key to make it unique. By default, the key for the overall process instance MC gets assigned the WebSphere Process Server unique process instance identifier and the process activity MCs gets assigned the WebSphere Process Server activity ID combined with the Process instance ID. A MC instance needs to be started by an inbound event and terminated by a trigger. By default, the MC instances termination triggers are generated and the …_Entry inbound events are marked to start new MC instances.

In the Monitor Detail model tree click on **UMS_Process ENTRY** event to open the Inbound Event Details. **Notice,** in the Correlation Expression section the option 'If no instances are found' is set to **Create new instances**! It means this event will starts new MC instances on new WebSphere Process Server process instance. In all other events this option is set to 'ignore'.



Inbound Event Details

Edit the details of the inbound event, which references an event that is generated by the monitored application. The type must be an event definition.

ID: UMS_Process_ENTRY

Name: UMS_Process ENTRY

Description:

Type: BPC.BFM.PROCESS.START

Filter Condition

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

```
UMS_Process_ENTRY/extendedData/processTemplateName = 'UMS_Process' and
UMS_Process_ENTRY/extendedData/EventNature = 'ENTRY'
```

Correlation Expression

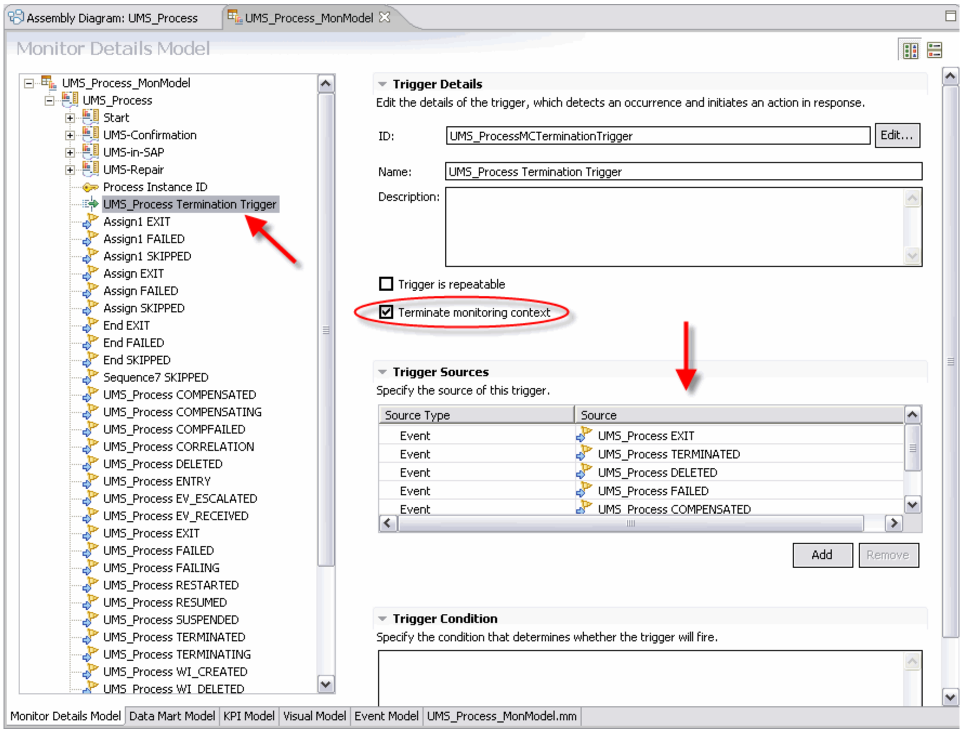
Define an expression to identify the monitoring context instance or instances that receive the event at runtime.

```
UMS_Process_ENTRY/propertyData/ECSCurrentID = ProcessInstanceID
```

If no instances are found: Create new instance

If one instance is found: Deliver to the instance

In the Monitor Detail model tree click on **UMS_Process Termination Trigger** to open the detail section. Notice, in the Trigger Details is marked the **Terminate monitoring context** check box! In the **Trigger Source** list you can see which events will kick off this trigger.



UMS_Process_MonModel
- UMS_Process
  - Start
  - UMS-Confirmation
  - UMS-in-SAP
  - UMS-Repair
  - Process Instance ID
  - UMS_Process Termination Trigger
  - Assign1 EXIT
  - Assign1 FAILED
  - Assign1 SKIPPED
  - Assign EXIT
  - Assign FAILED
  - Assign SKIPPED
  - End EXIT
  - End FAILED
  - End SKIPPED
  - Sequence7 SKIPPED
  - UMS_Process COMPENSATED
  - UMS_Process COMPENSATING
  - UMS_Process COMPFAILED
  - UMS_Process CORRELATION
  - UMS_Process DELETED
  - UMS_Process ENTRY
  - UMS_Process EV_ESCALATED
  - UMS_Process EV_RECEIVED
  - UMS_Process EXIT
  - UMS_Process FAILED
  - UMS_Process FAILING
  - UMS_Process RESTARTED
  - UMS_Process RESUMED
  - UMS_Process SUSPENDED
  - UMS_Process TERMINATED
  - UMS_Process TERMINATING
  - UMS_Process WI_CREATED
  - UMS_Process WI_DELETED

Trigger Details

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

ID: UMS_ProcessMCTerminationTrigger Edit...

Name: UMS_Process Termination Trigger

Description:

☐ Trigger is repeatable

☑ Terminate monitoring context

Trigger Sources

Specify the source of this trigger.

Source Type	Source
Event	UMS_Process EXIT
Event	UMS_Process TERMINATED
Event	UMS_Process DELETED
Event	UMS_Process FAILED
Event	UMS_Process COMPENSATED

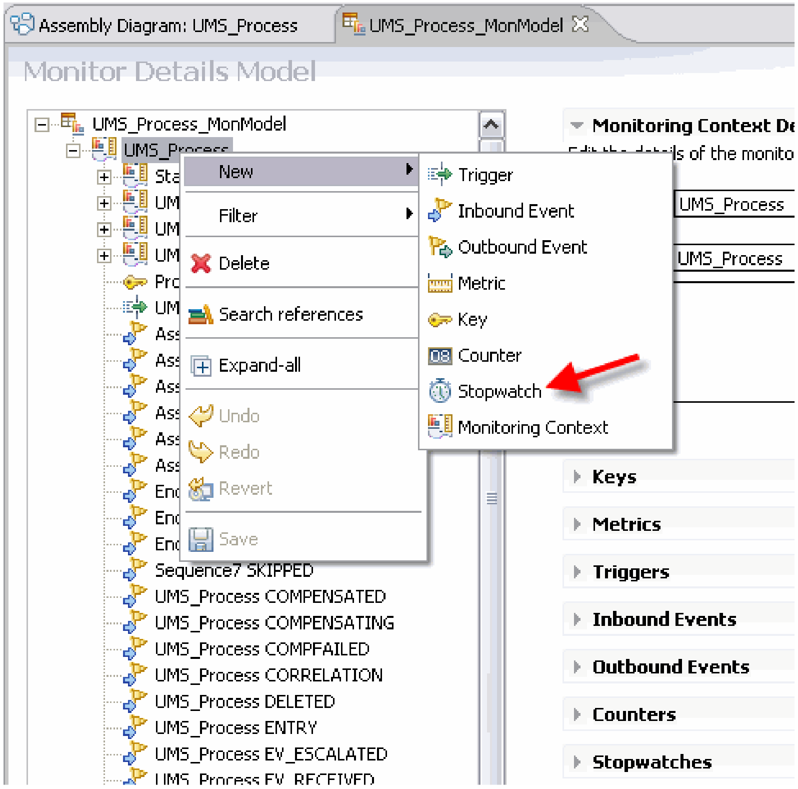
Trigger Condition

Specify the condition that determines whether the trigger will fire.

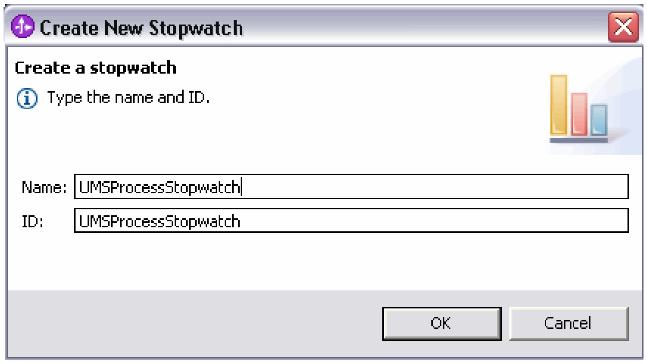
Build the low level elements for KPI1

First KPI: Remember the first KPI measures the process duration. For duration measurements you need to set up a stopwatch that gets triggered when a process instance is started and stopped when the process instance is finished. So, the next steps are to set up a process duration stopwatch.

In the Monitor Detail model tree right click on **UMS_Process** (located near the top of the tree), then select **New-->Stopwatch**.



Enter **UMSProcessStopwatch** as **Name** and click on **OK** to close the window.



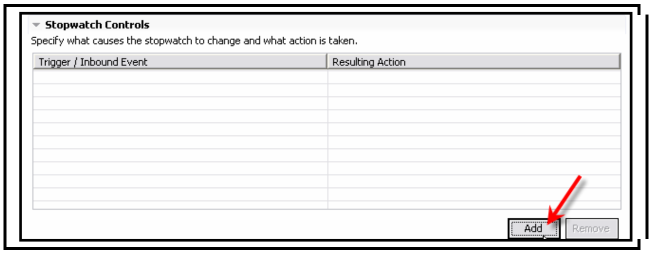
Create a stopwatch

Type the name and ID.

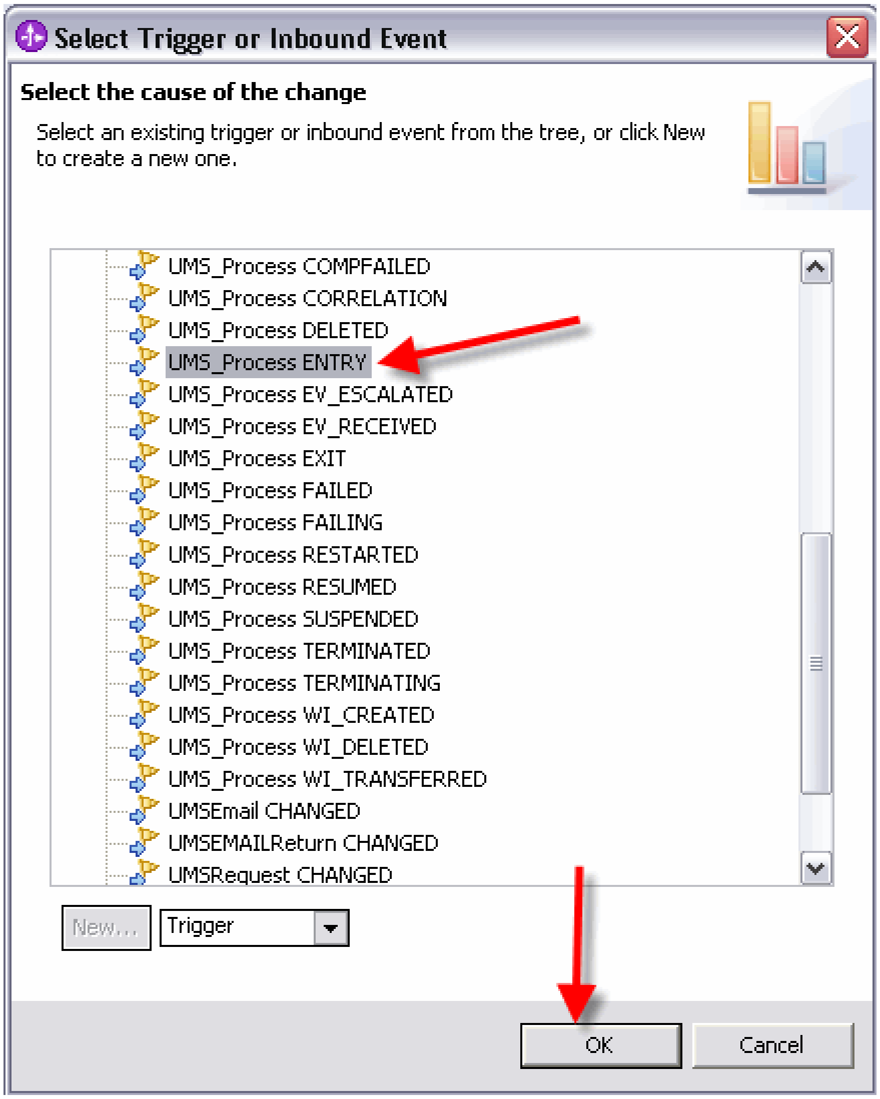
Name: UMSProcessStopwatch

ID: UMSProcessStopwatch

Click on **Add** in the Stopwatch Controls section to create the trigger event which will start the Stopwatch.


Stopwatch Controls
Specify what causes the stopwatch to change and what action is taken.
Trigger / Inbound Event
Resulting Action
Add
Remove

In the Select trigger Source window click on the **UMS_Process ENTRY** event and then click on **OK** to close the window



Select Trigger or Inbound Event

Select the cause of the change

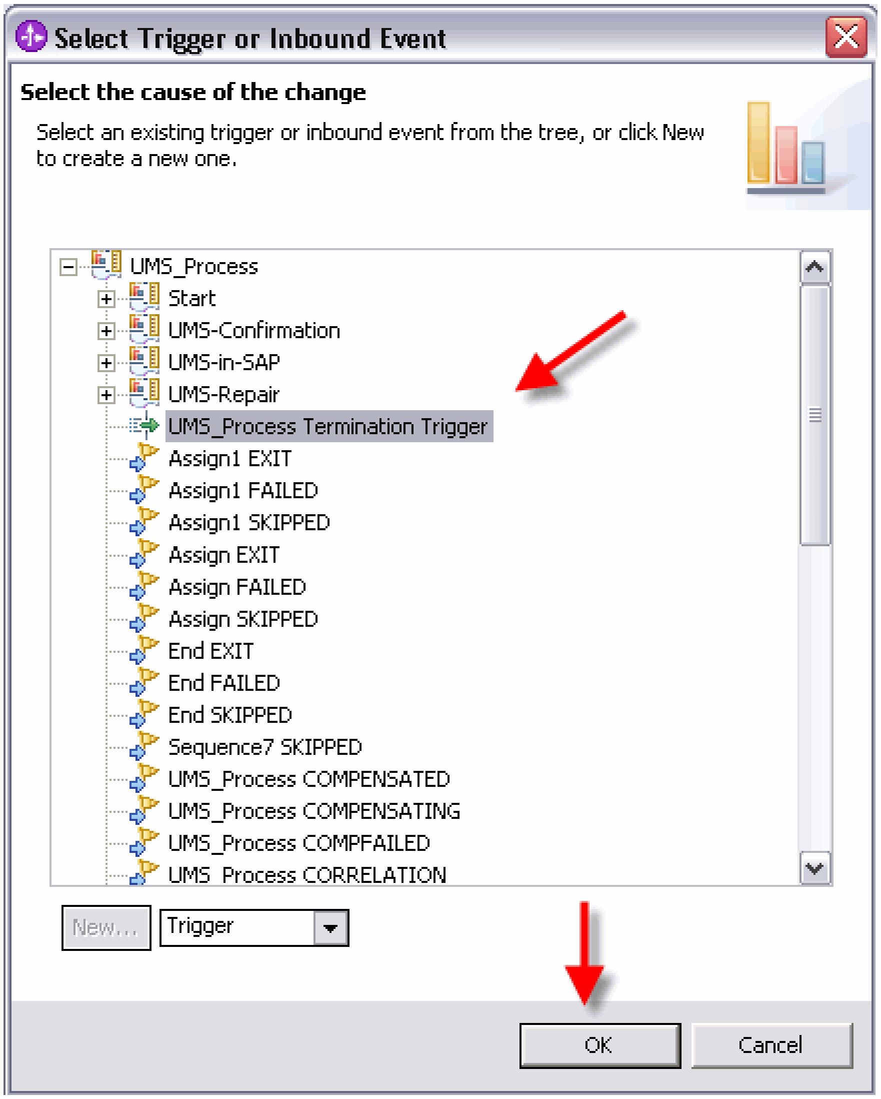
Select an existing trigger or inbound event from the tree, or click New to create a new one.

- UMS_Process COMPFAILED
- UMS_Process CORRELATION
- UMS_Process DELETED
- UMS_Process ENTRY
- UMS_Process EV_ESCALATED
- UMS_Process EV_RECEIVED
- UMS_Process EXIT
- UMS_Process FAILED
- UMS_Process FAILING
- UMS_Process RESTARTED
- UMS_Process RESUMED
- UMS_Process SUSPENDED
- UMS_Process TERMINATED
- UMS_Process TERMINATING
- UMS_Process WI_CREATED
- UMS_Process WI_DELETED
- UMS_Process WI_TRANSFERRED
- UMSEmail CHANGED
- UMSEMAILReturn CHANGED
- UMSRequest CHANGED

Note: The Resulting Action column is set to START by default.

Click on **Add** in the Stopwatch Controls section to create the trigger event which will stop the Stopwatch.

In the Select trigger Source window click on the **UMS_Process Termination Trigger** and then click on **OK** to close the window**.**



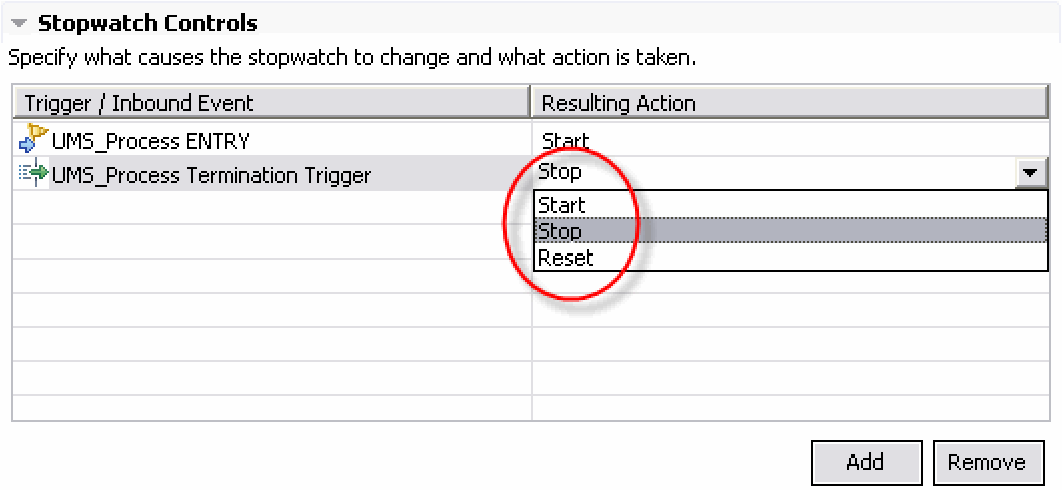
Select Trigger or Inbound Event

Select the cause of the change

Select an existing trigger or inbound event from the tree, or click New to create a new one.

- UMS_Process
  - Start
  - UMS-Confirmation
  - UMS-in-SAP
  - UMS-Repair
  - UMS_Process Termination Trigger
  - Assign1 EXIT
  - Assign1 FAILED
  - Assign1 SKIPPED
  - Assign EXIT
  - Assign FAILED
  - Assign SKIPPED
  - End EXIT
  - End FAILED
  - End SKIPPED
  - Sequence7 SKIPPED
  - UMS_Process COMPENSATED
  - UMS_Process COMPENSATING
  - UMS_Process COMPFAILED
  - UMS_Process CORRELATION

In the Resulting Action column click on the **Start** entry, open up the pull down menu and select **Stop.**



Stopwatch Controls

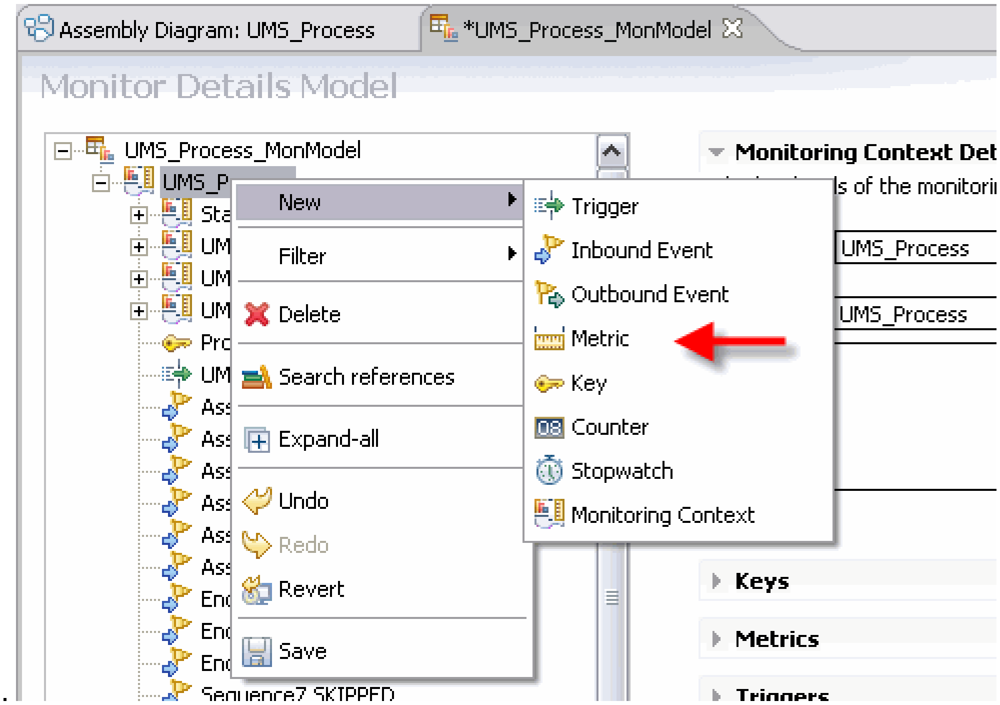
Specify what causes the stopwatch to change and what action is taken.

Trigger / Inbound Event	Resulting Action
UMS_Process ENTRY	Start
UMS_Process Termination Trigger	Stop

Define KPI 2

KPI 2: The third KPI is a percentage of update requests successfully processed without the involvement of a supervisor. You need to add a metric that tells each process instance if the supervisor was incorporated. Building the percentage by aggregation of all instances will be done later on the KPI Model definitions.

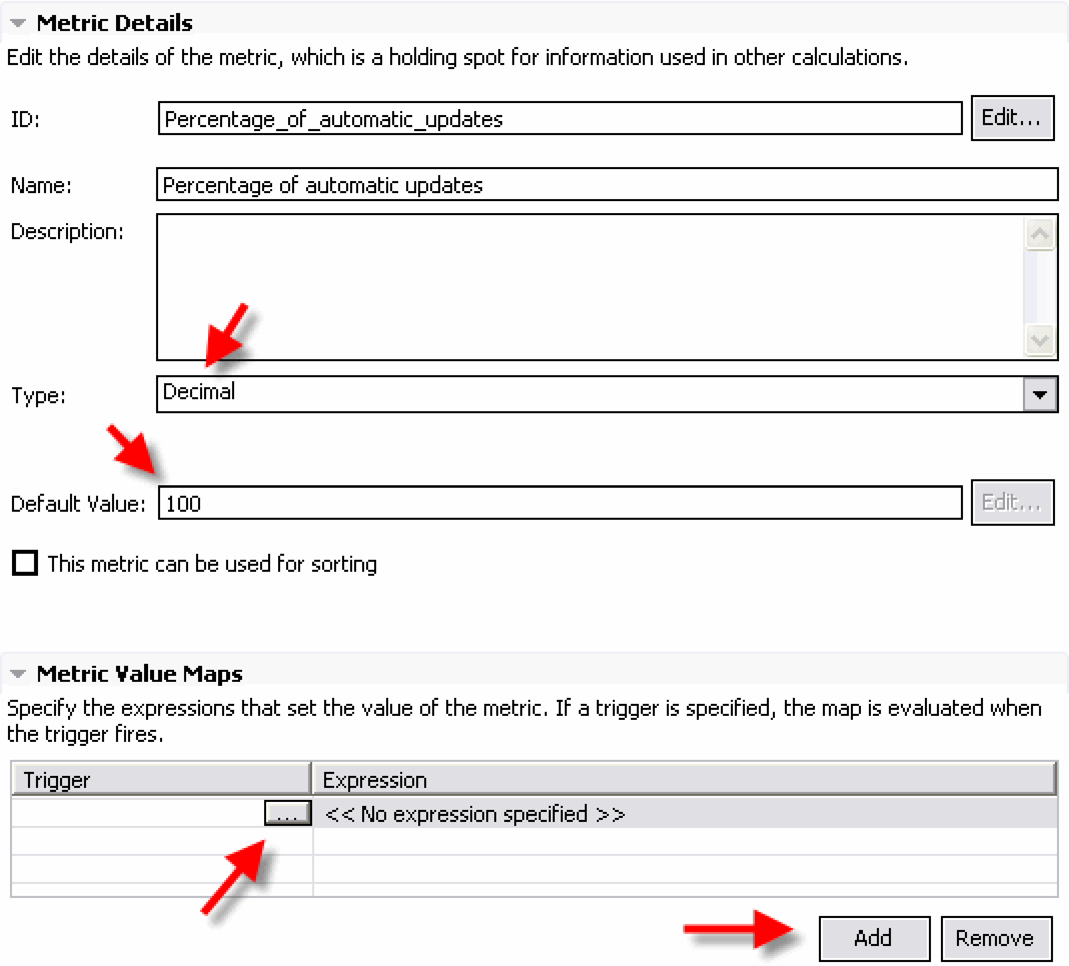
In the Monitor Detail model tree right click on **UMS_Process** to open **New → Metric**. Click on **Metric** to create a new metric that will contain the percentage for each instance.



Enter **Percentage of automatic updates** as **Name** and click on **OK** to close the window.

Define the metric type default vale and calculations:

- Select as Type **Decimal** in the drop down box.
- Enter as Default Value **100.**
- Click on **Add** in the Metric Value Maps section to define a new entry in the table.
- Click under Trigger in the first entry of the table to open up the trigger definition window.



Metric Details

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

ID: Percentage_of_automatic_updates

Name: Percentage of automatic updates

Description:

Type: Decimal

Default Value: 100

□ This metric can be used for sorting

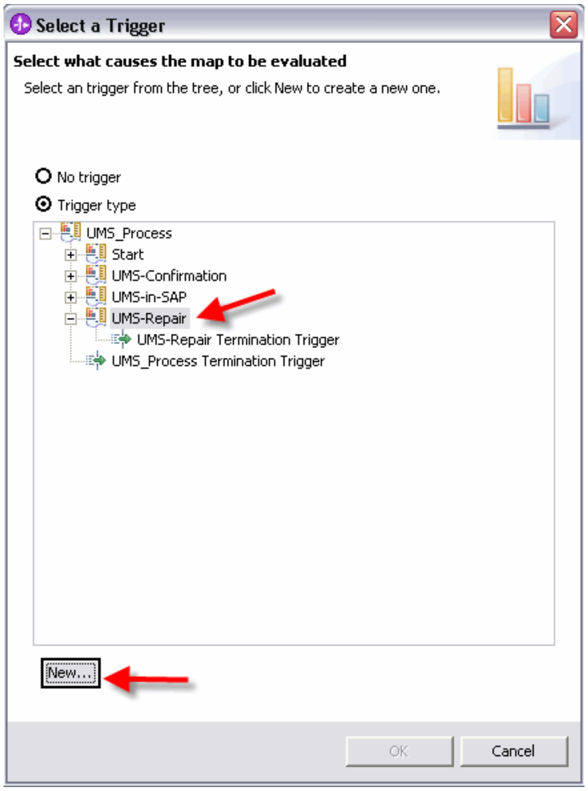
Metric Value Maps

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
...	<< No expression specified >>

Note: In most cases, the process instance ends without human invocation. Therefore initialize the metric with 100. This means the process instance is 100% automatic. In the next steps, you define to set the metric percentage to 0 for all cases wherever the process instance incorporates the supervisor.

In the **Select a Trigger** window there is no trigger available yet which notifies you when the supervisor is incorporated. The supervisor is represented by the Human Task activity **UMS_Repair**. So select this activity and click on the **New…** button.



Select a Trigger

Select what causes the map to be evaluated

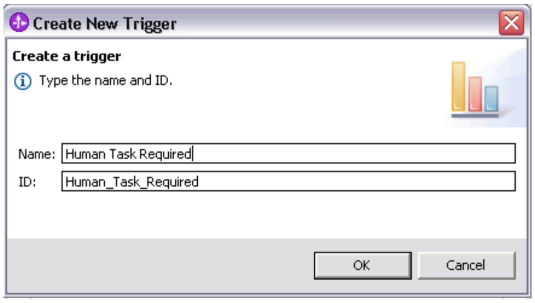
Select an trigger from the tree, or click New to create a new one.

○ No trigger

● Trigger type

- UMS_Process
  - Start
  - UMS-Confirmation
  - UMS-in-SAP
  - UMS-Repair
    - UMS-Repair Termination Trigger
  - UMS_Process Termination Trigger

For the trigger name enter **Human Task Required** and click on **OK**.



Create New Trigger

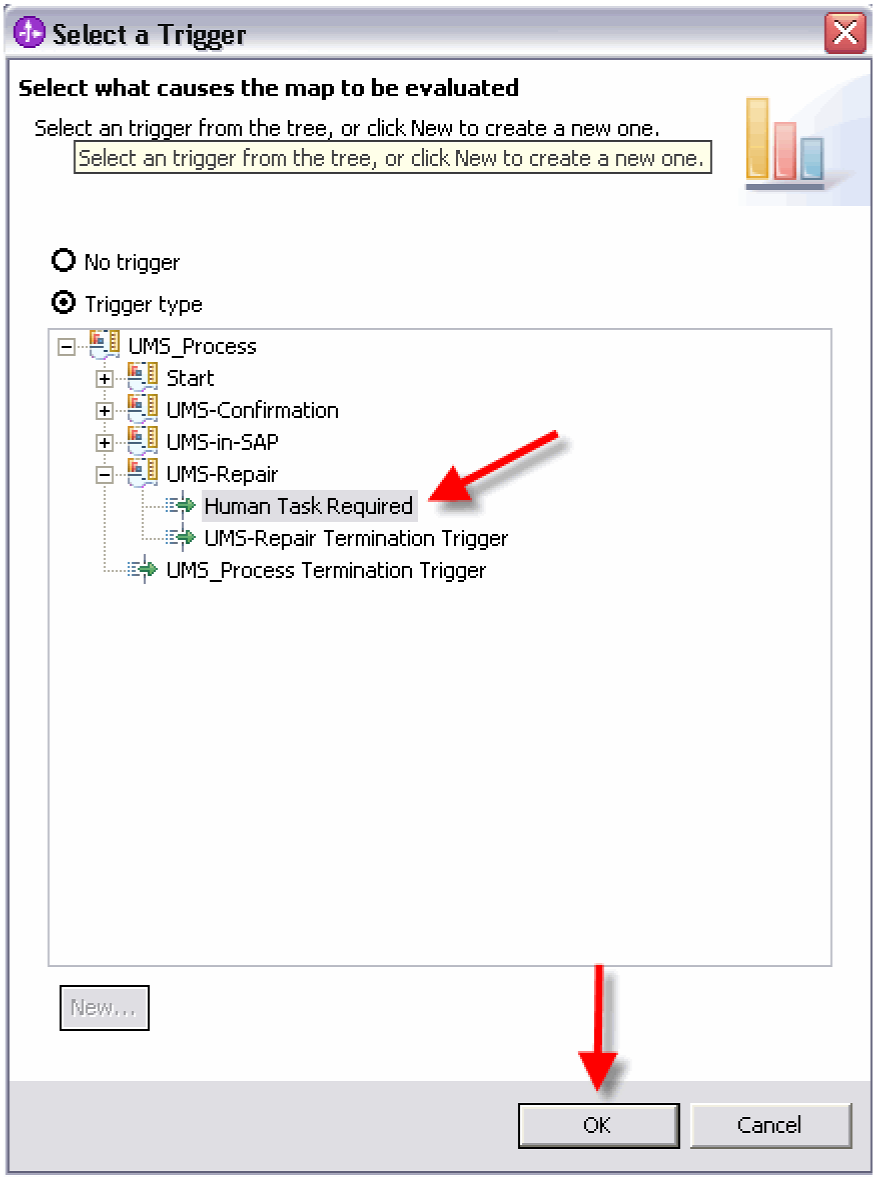
Create a trigger

Type the name and ID.

Name: Human Task Required

ID: Human_Task_Required

Select the newly created trigger and click on **OK**.



Select what causes the map to be evaluated

Select an trigger from the tree, or click New to create a new one.

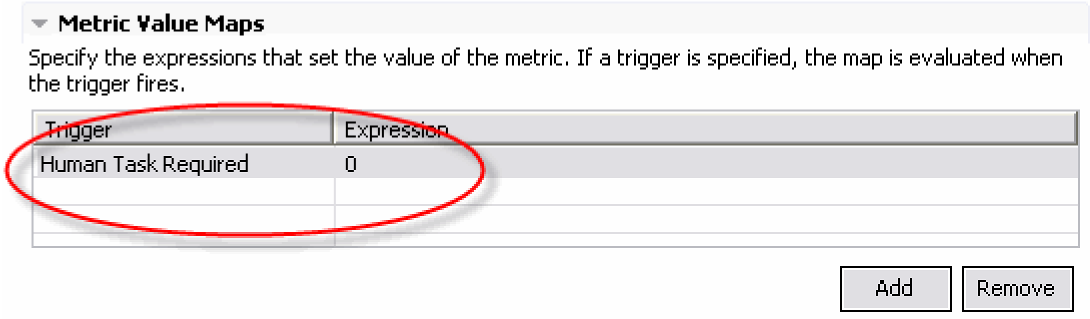
Select an trigger from the tree, or click New to create a new one.

○ No trigger

● Trigger type

- UMS_Process
  - Start
  - UMS-Confirmation
  - UMS-in-SAP
  - UMS-Repair
    - Human Task Required
    - UMS-Repair Termination Trigger
  - UMS_Process Termination Trigger

In the expression field of the new trigger enter the value 0.

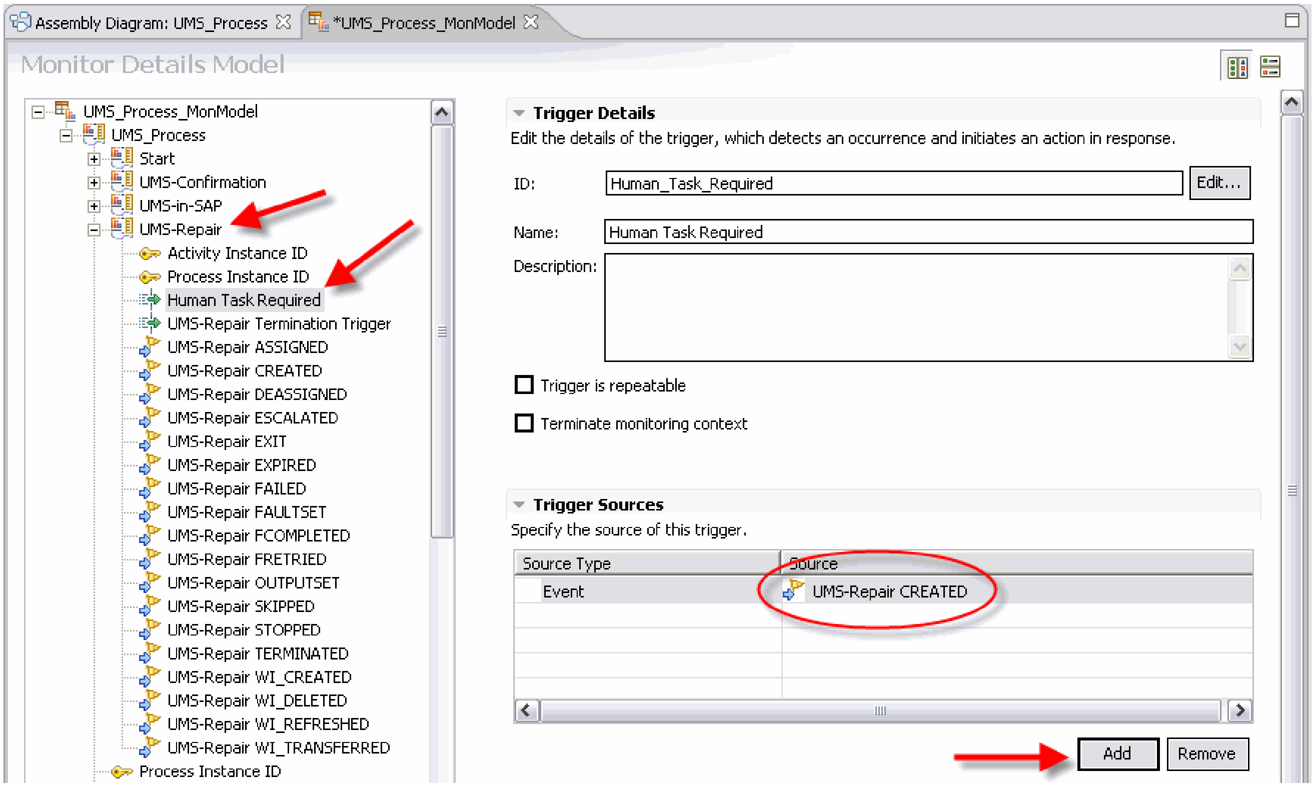


Metric Value Maps

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
Human Task Required	0

At this point the have to define which common base event is kicking off this newly created trigger. To do so click on the trigger in the left tree and click on the **Add** button in the **Trigger Sources** are. Then choose the event **UMS_Repair CREATED**.



- UMS_Process_MonModel
  - UMS_Process
    - Start
    - UMS-Confirmation
    - UMS-in-SAP
    - UMS-Repair
      - Activity Instance ID
      - Process Instance ID
      - Human Task Required
      - UMS-Repair Termination Trigger
      - UMS-Repair ASSIGNED
      - UMS-Repair CREATED
      - UMS-Repair DEASSIGNED
      - UMS-Repair ESCALATED
      - UMS-Repair EXIT
      - UMS-Repair EXPIRED
      - UMS-Repair FAILED
      - UMS-Repair FAULTSET
      - UMS-Repair FCOMPLETED
      - UMS-Repair FRETRIED
      - UMS-Repair OUTPUTSET
      - UMS-Repair SKIPPED
      - UMS-Repair STOPPED
      - UMS-Repair TERMINATED
      - UMS-Repair WI_CREATED
      - UMS-Repair WI_DELETED
      - UMS-Repair WI_REFRESHED
      - UMS-Repair WI_TRANSFERRED
    - Process Instance ID

Trigger Details

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

ID: Human_Task_Required

Name: Human Task Required

Description:

□ Trigger is repeatable

□ Terminate monitoring context

Trigger Sources

Specify the source of this trigger.

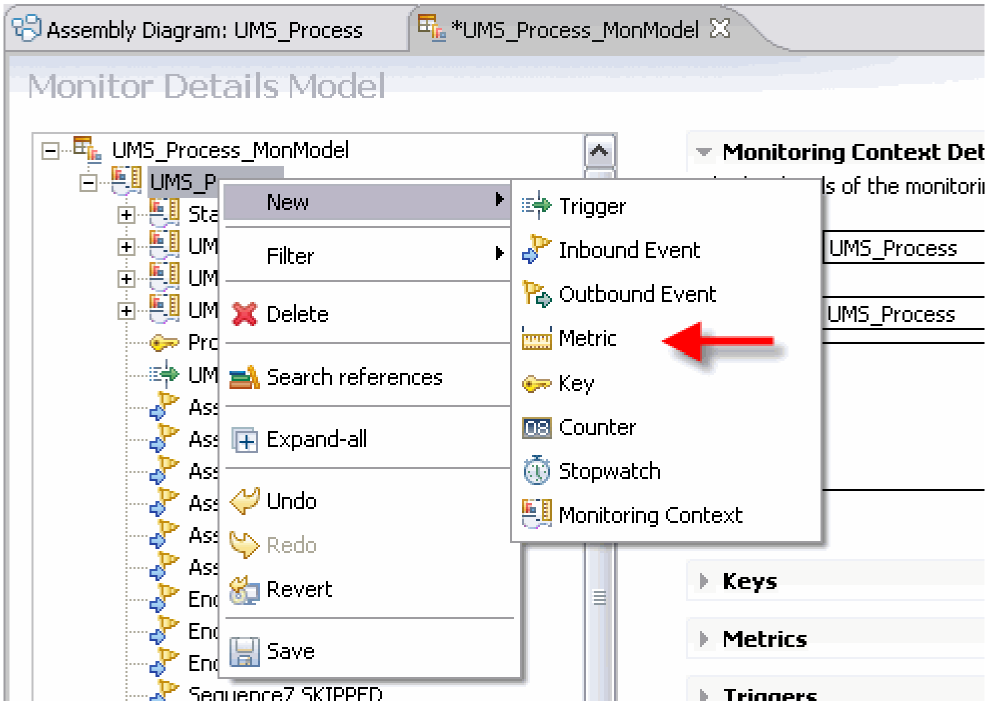
Source Type	Source
Event	UMS-Repair CREATED

Press the **Ctl+S** key to save the changes.

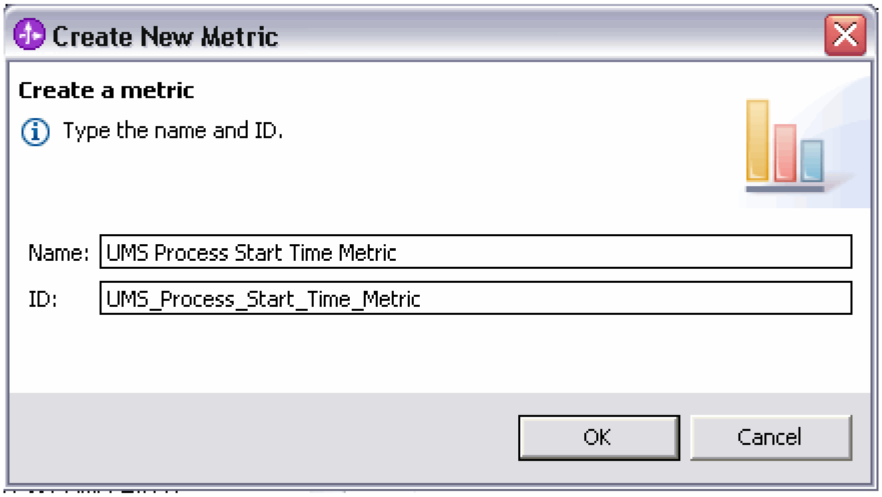
Define the details elements for monitor dashboard drilldowns (dimensions)

In the next steps, you will define the metrics needed for drilling down through dimensions in the monitor dashboards. The best practice is to define a dimension on the process instance start date. This allows you to drill down on timeframes. The business analyst defined in addition in the KPIs customer location for Monitor dashboard dimensions. The Data Mart Model needs a metric for each dimension you want to drill down in the dashboard. Therefore you will define a metric for process instance start time.

In the Monitor Detail model tree right click on **UMS_Process** to open **New → Metric**. Click on **Metric** to create a new metric that gets the start time assigned.



Enter **UMS Process Start Time Metric**. Click on **OK** to close the window and bring up the trigger detail window.


Create New Metric

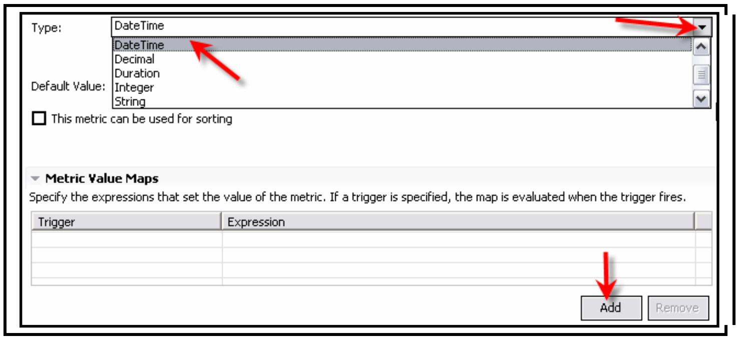
Create a metric

Type the name and ID.

Name: UMS Process Start Time Metric

ID: UMS_Process_Start_Time_Metric

Select in the drop down box **Date Time** as Type of the metric (you need to scroll up). Click on **Add** in the Metric Value Maps to add a new entry in the table.

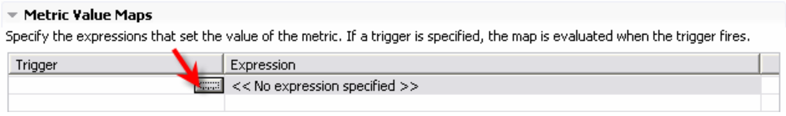


□ This metric can be used for sorting

Metric Value Maps

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

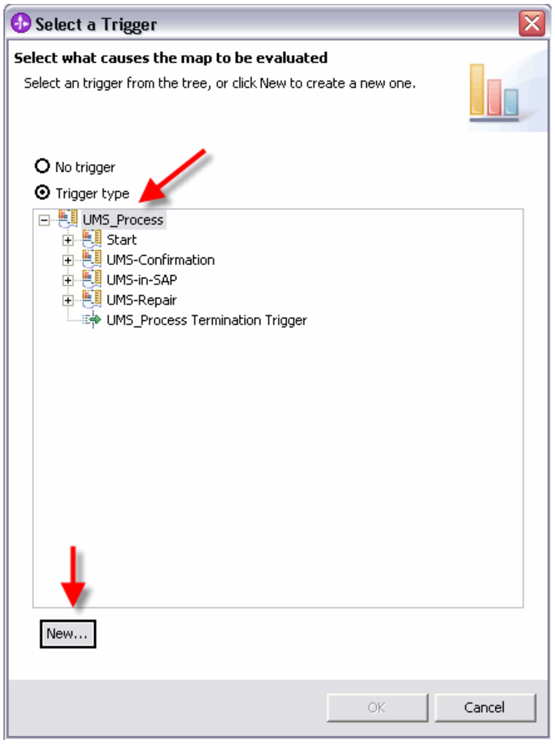
Click in the Trigger row on the first entry to open up the trigger window.



Metric Value Maps

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

Close the **UMS_Process** tree. Click on **New...** to open the naming window.



Select a Trigger

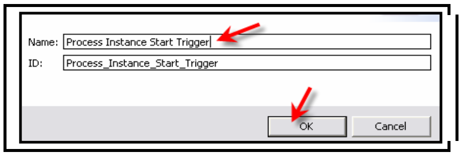
Select what causes the map to be evaluated

Select an trigger from the tree, or click New to create a new one.

○ No trigger

● Trigger type

Enter **Process Instance Start Trigger** as Name. Click on **OK** to close the window.



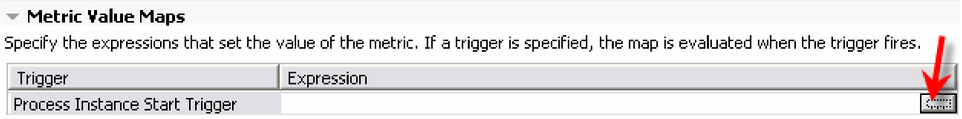
Name: Process Instance Start Trigger

ID: Process_Instance_Start_Trigger

Click on **OK** to close the Select a Trigger window.

You just defined a new trigger that needs to fire on new process instances. The event **UMS_Process Entry** is the event you want to take the process instance start time from. The trigger will be finalized in one of the next steps.

Click in the Expression raw on the first entry to open up the Expression Dialog window.

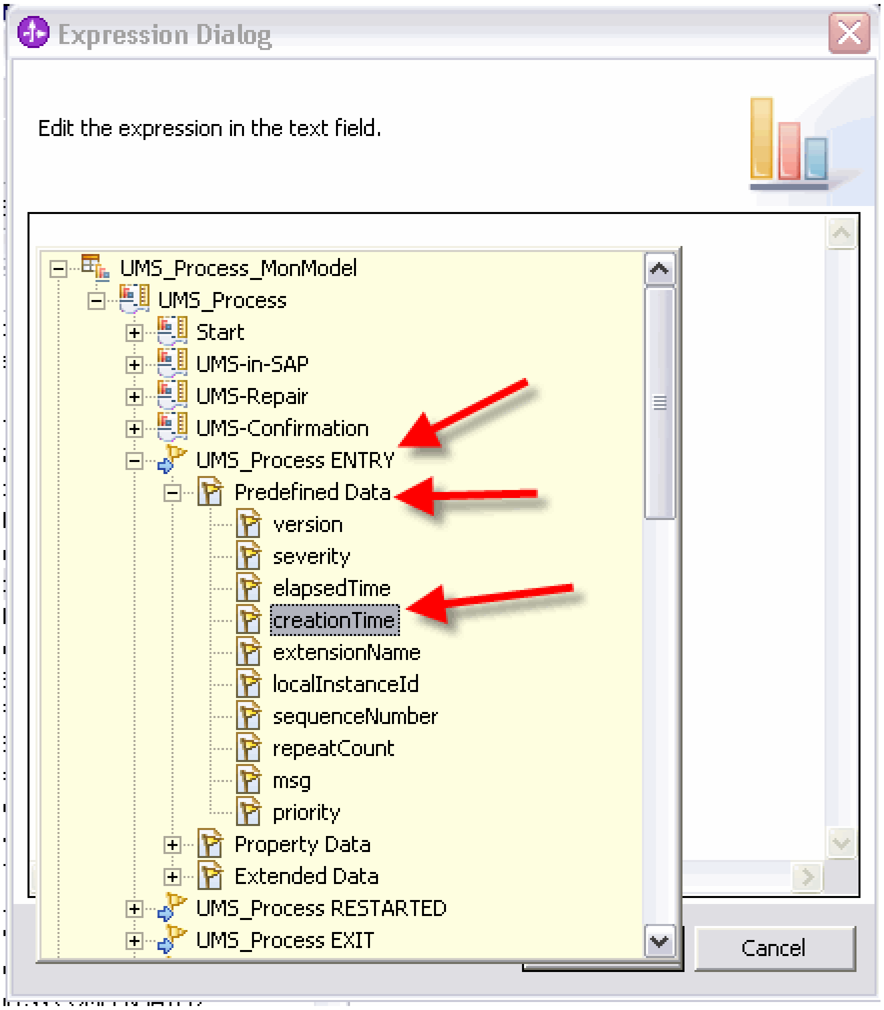


Metric Value Maps

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
Process Instance Start Trigger	

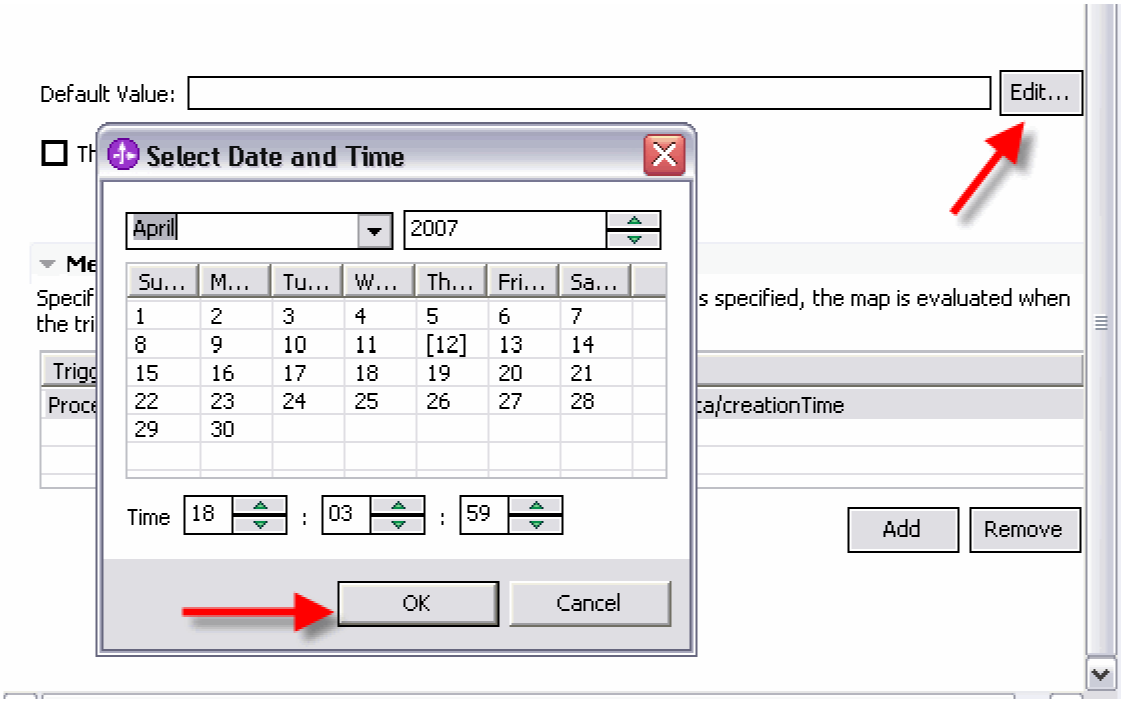
Press **Crtl+Space** keys to open the Control Assist window. Expand the tree **UMS_ProcessMonModel → UMS_Process → UMS_Process ENTRY→ Predefined Data** and double click on **creationTime**.



Edit the expression in the text field.

Click on **OK** to close the Expression Dialog window.

To specify the Default Value click on the **Edit...** button. The Select Date und Time window opens up. Click on **OK** to accept the current date - time as default. You need to define a formal default value for this metric so that it initializes. You will use the current date as default, but due to the fact that for each instance you override it with the real start time the value is unimportant.



Default Value:

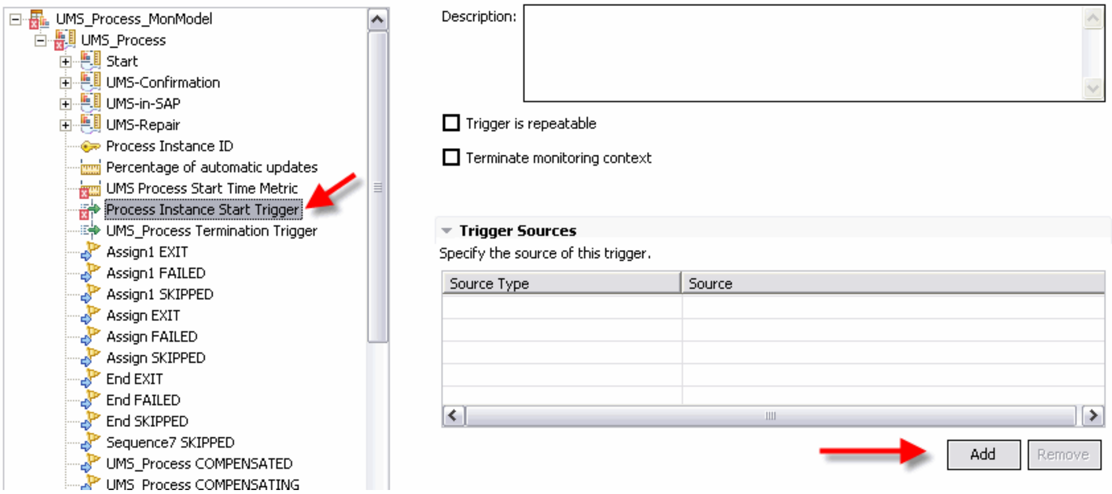
Select Date and Time

April 2007

Su...	M...	Tu...	W...	Th...	Fri...	Sa...
1	2	3	4	5	6	7
8	9	10	11	[12]	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

Time 18 : 03 : 59

Now you will define the specifics for the trigger you just created. Click on the **Process Instance Start Trigger** in the tree window. Click on **Add** in the Trigger Source section.



Description:

☐ Trigger is repeatable

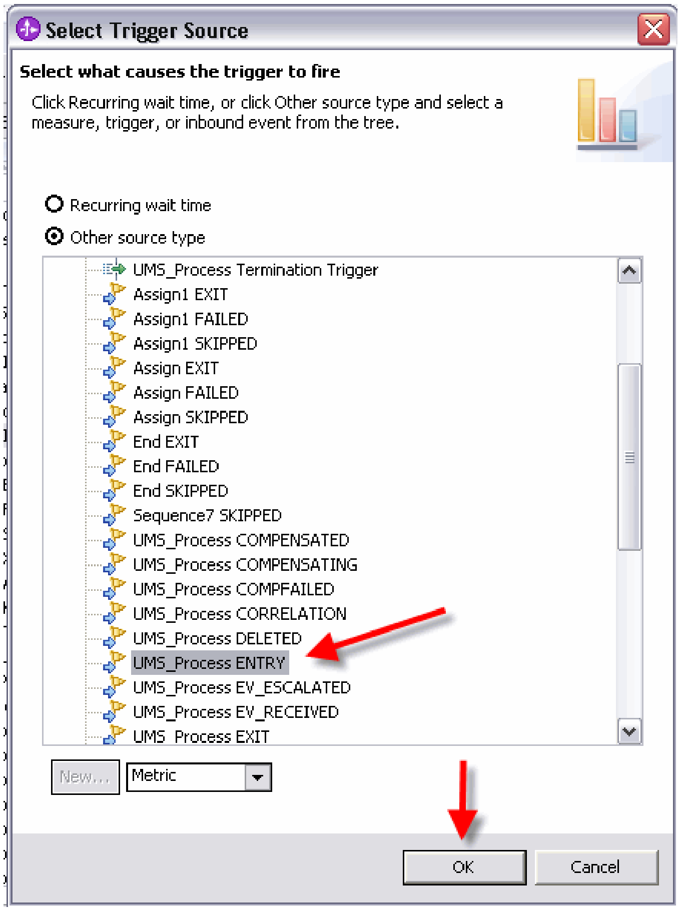
☐ Terminate monitoring context

Trigger Sources

Specify the source of this trigger.

Source Type	Source

Click on **UMS_Process ENTRY** event to highlight it. Click on **OK** to close the window.



Select Trigger Source

Select what causes the trigger to fire

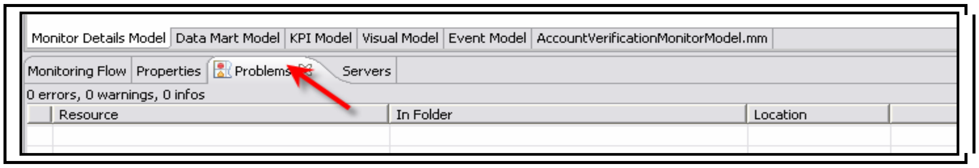
Click Recurring wait time, or click Other source type and select a measure, trigger, or inbound event from the tree.

○ Recurring wait time

● Other source type

- UMS_Process Termination Trigger
- Assign1 EXIT
- Assign1 FAILED
- Assign1 SKIPPED
- Assign EXIT
- Assign FAILED
- Assign SKIPPED
- End EXIT
- End FAILED
- End SKIPPED
- Sequence7 SKIPPED
- UMS_Process COMPENSATED
- UMS_Process COMPENSATING
- UMS_Process COMPFAILED
- UMS_Process CORRELATION
- UMS_Process DELETED
- UMS_Process ENTRY
- UMS_Process EV_ESCALATED
- UMS_Process EV_RECEIVED
- UMS_Process EXIT

Press the **Crtl+S** key to save the changes. Click on the **Problem** tab. There should be no problems left which are related to the measure model you created.



0 errors, 0 warnings, 0 infos

	Resource	In Folder	Location

Define the Data Mart Model

A data mart is a database or collection of databases containing data that is tailored and optimized for the specific reporting needs of a department or team. The data mart model is the part of the monitor model that defines the cubes that are used for storing, retrieving, and analyzing the data that is gathered over time.

Cubes
A data mart model contains one cube for each monitoring context definition. Each cube keeps track of all the information that the monitoring context collects over time. The cube can combine all the instances of that monitoring context so that you can query aggregate information.

Dimensions
Dimensions are data categories used to organize and select instances for reporting and analysis. Some examples of dimensions are time, accounts, products, and markets. Dimensions are composed of one or more attributes, which are hierarchical. For example, some Location attributes are City, Region, and Country.

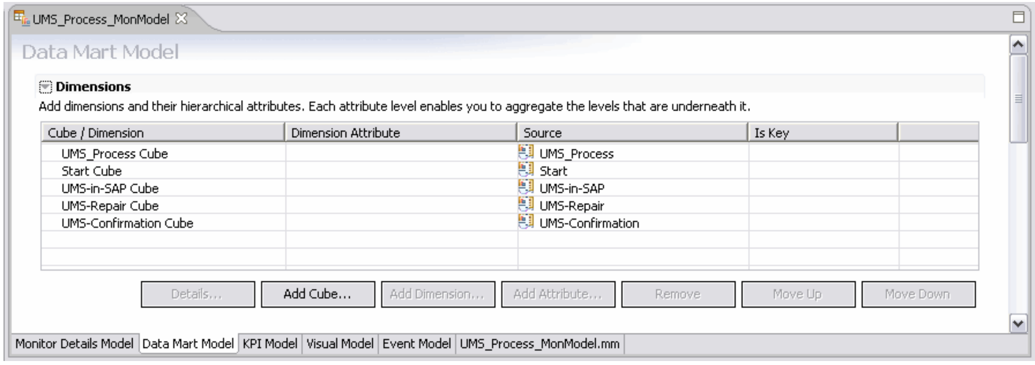
Facts and measures
Facts are numeric data that can be examined and analyzed. A fact defines the cells in a cube data structure and associates them with metrics, counters, and stopwatches that supply the values. Measures are calculations based on facts. A measure points to a fact (such as order amount) and specifies an aggregation function (such as average or sum).

Generate Monitor Model generates in WebSphere Integration Developer on processes by default a cube defined on process level and in addition a cube for each activity and BPEL element you selected as monitor relevant.

Finalize the Data Mart model

Note: In the next steps you will extend the generated Data Mart model default with custom specific dimensions. Remember, you want to drill down on process start timeframes.

Click on the **Data Mart Model** tab to open the dimension details view.



Data Mart Model

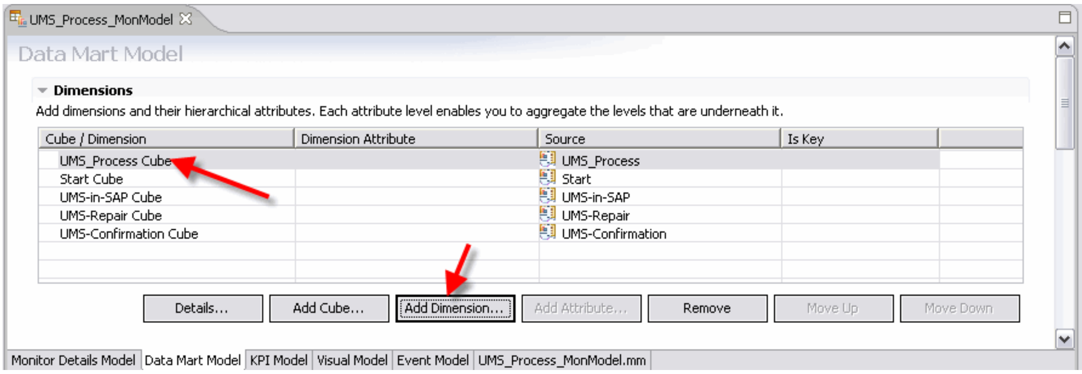
Dimensions

Add dimensions and their hierarchical attributes. Each attribute level enables you to aggregate the levels that are underneath it.

Cube / Dimension	Dimension Attribute	Source	Is Key	
UMS_Process Cube		UMS_Process		
Start Cube		Start		
UMS-in-SAP Cube		UMS-in-SAP		
UMS-Repair Cube		UMS-Repair		
UMS-Confirmation Cube		UMS-Confirmation		

Note: In the next steps you will add a dimension for the process instance start time. This allows drilling down by timeframes in the monitor dashboards.

Click on the **UMS_Process_Cube** entry in the Dimension table to highlight it. Click on **Add Dimension...** to add a new dimension.

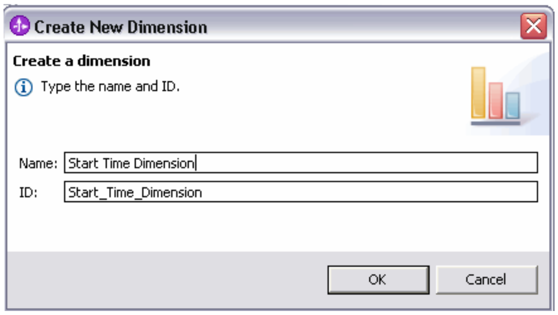

UMS_Process_MonModel
Data Mart Model
Details... | Add Cube... | Add Dimension... | Add Attribute... | Remove | Move Up | Move Down
Monitor Details Model | Data Mart Model | KPI Model | Visual Model | Event Model | UMS_Process_MonModel.mm

Dimensions

Add dimensions and their hierarchical attributes. Each attribute level enables you to aggregate the levels that are underneath it.

Cube / Dimension	Dimension Attribute	Source	Is Key
UMS_Process Cube		UMS_Process	
Start Cube		Start	
UMS-in-SAP Cube		UMS-in-SAP	
UMS-Repair Cube		UMS-Repair	
UMS-Confirmation Cube		UMS-Confirmation	

Enter **Start Time Dimension** as Name. Click on **Ok** to close the window.



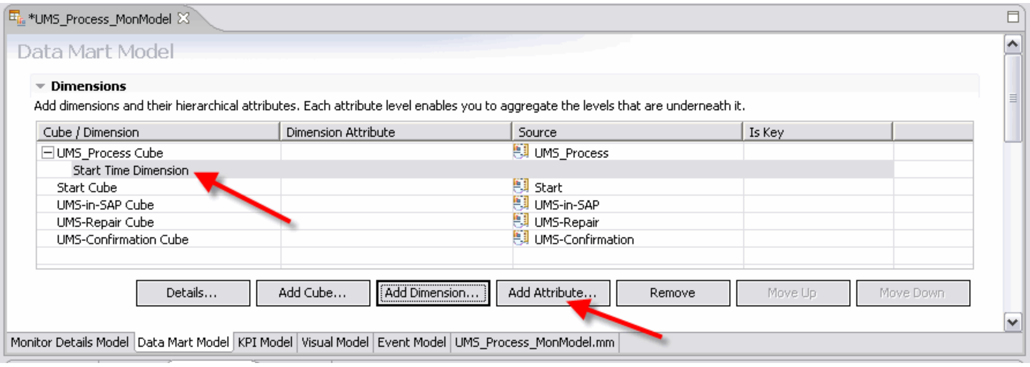
Create a dimension

Type the name and ID.

Name: Start Time Dimension

ID: Start_Time_Dimension

Click on the **Add Attribute** button.

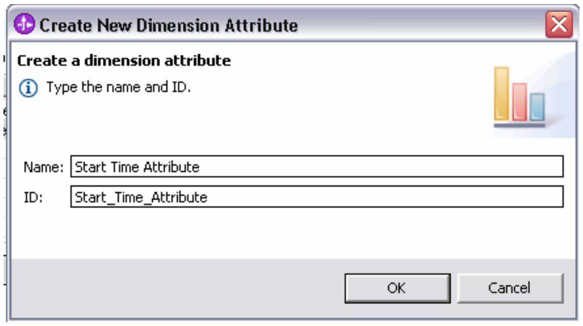

*UMS_Process_MonModel
Data Mart Model
Details... | Add Cube... | Add Dimension... | Add Attribute... | Remove | Move Up | Move Down
Monitor Details Model | Data Mart Model | KPI Model | Visual Model | Event Model | UMS_Process_MonModel.mm

Dimensions

Add dimensions and their hierarchical attributes. Each attribute level enables you to aggregate the levels that are underneath it.

Cube / Dimension	Dimension Attribute	Source	Is Key
UMS_Process Cube		UMS_Process	
Start Time Dimension			
Start Cube		Start	
UMS-in-SAP Cube		UMS-in-SAP	
UMS-Repair Cube		UMS-Repair	
UMS-Confirmation Cube		UMS-Confirmation	

Enter **Start Time Attribute** as Name. Click on **Ok** to close the window



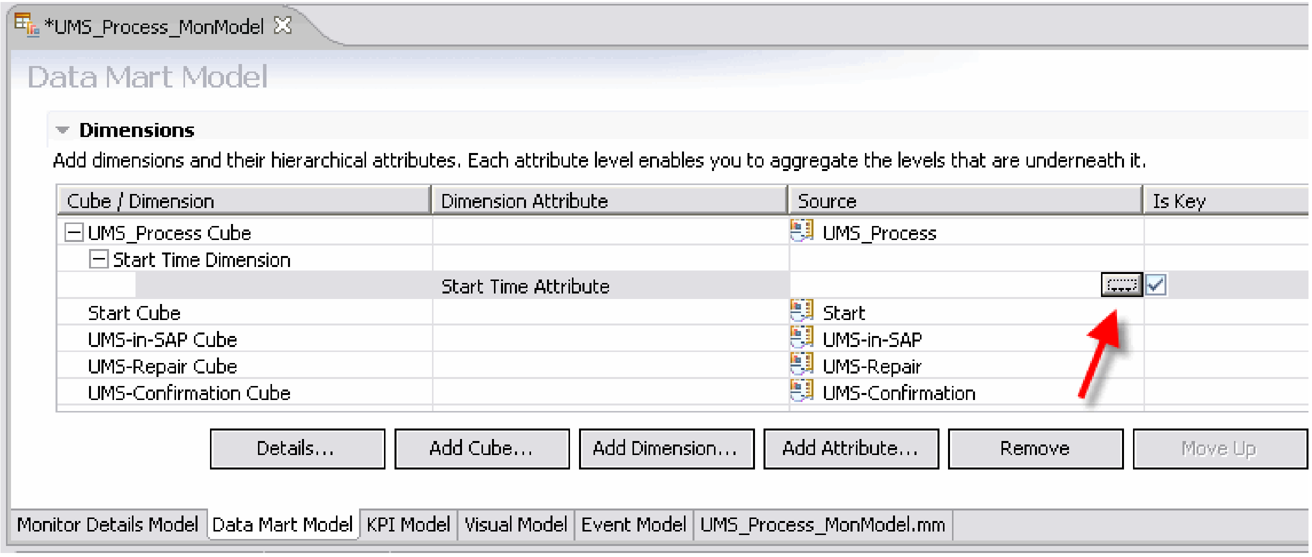
Create a dimension attribute

Type the name and ID.

Name: Start Time Attribute

ID: Start_Time_Attribute

Click under the Source column in the Start Time Attribute entry row to open up the source window



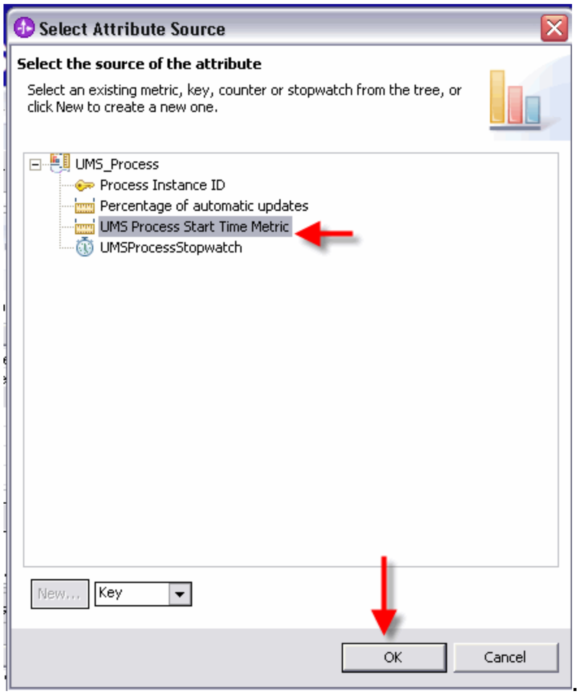
Data Mart Model

Dimensions

Add dimensions and their hierarchical attributes. Each attribute level enables you to aggregate the levels that are underneath it.

Cube / Dimension	Dimension Attribute	Source	Is Key
⊟ UMS_Process Cube		UMS_Process	
⊟ Start Time Dimension			
	Start Time Attribute		☑
Start Cube		Start	
UMS-in-SAP Cube		UMS-in-SAP	
UMS-Repair Cube		UMS-Repair	
UMS-Confirmation Cube		UMS-Confirmation	

Click on **UMS_Process Start Time Metric** to highlight it. Click on **OK** to close the window.



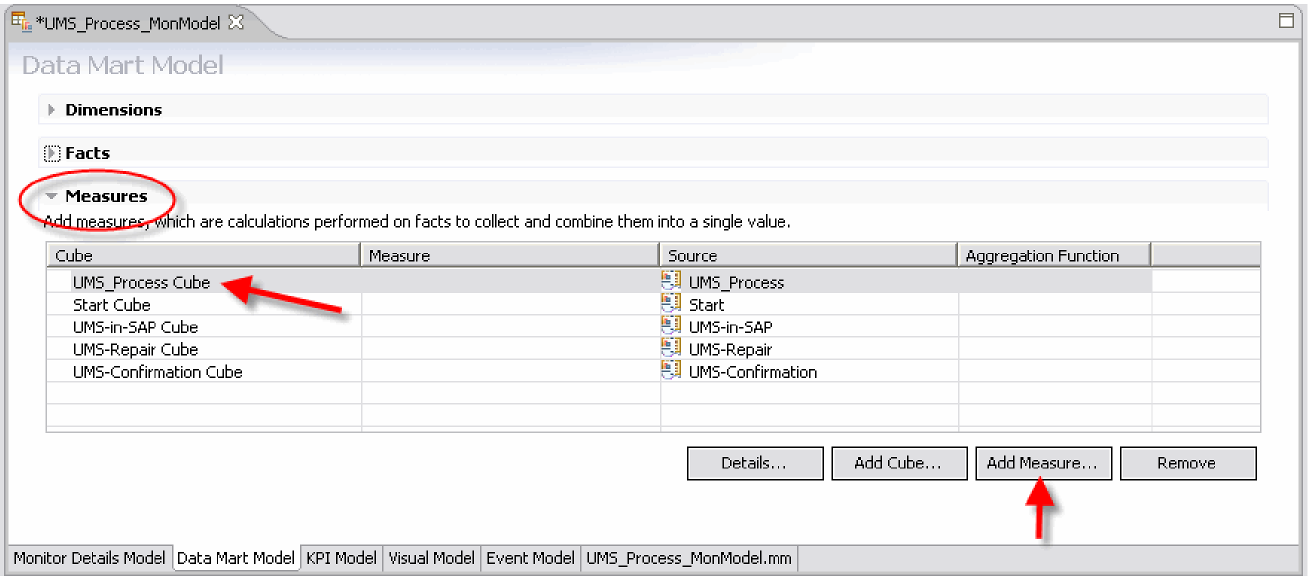
Select the source of the attribute

Select an existing metric, key, counter or stopwatch from the tree, or click New to create a new one.

⊟ UMS_Process
Process Instance ID
Percentage of automatic updates
UMS Process Start Time Metric
UMSProcessStopwatch

Note: In the next steps you will add **Measures** to the **Data Mart Model**. For each **KPI** you have to add one entry and specify the fact in the table above used for calculation. The facts have been created automatically when you added a metric or stopwatch to the Monitor Detail Model.

In the Measures section click on **UMS_Process Cube** to highlight the entry. Click on **Add Measure...** to add a new measure. You may need to scroll down for the Measure table.



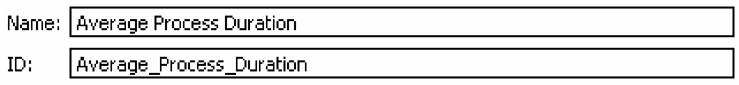
Data Mart Model

Measures

Add measures, which are calculations performed on facts to collect and combine them into a single value.

Cube	Measure	Source	Aggregation Function
UMS_Process Cube		UMS_Process	
Start Cube		Start	
UMS-in-SAP Cube		UMS-in-SAP	
UMS-Repair Cube		UMS-Repair	
UMS-Confirmation Cube		UMS-Confirmation	

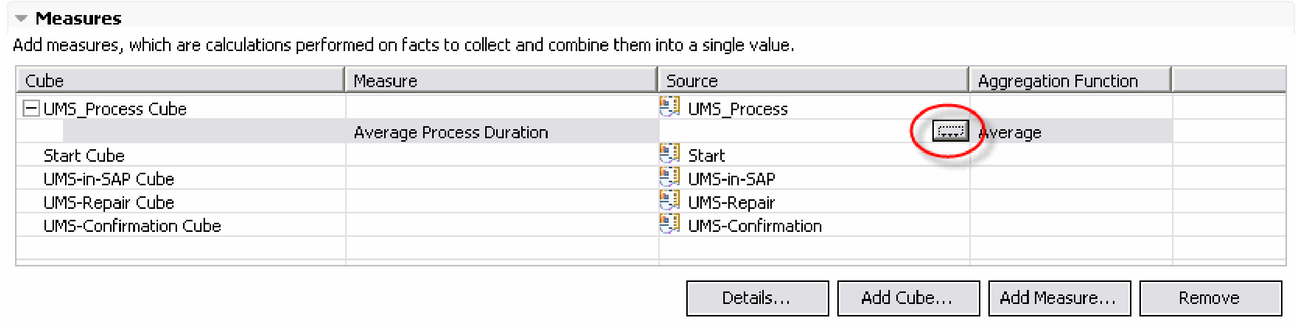
Enter **Average Process Duration** as Name. Click on **OK** to close the window**.**



Name: Average Process Duration

ID: Average_Process_Duration

Click under the **Source** column in **the Average Process Duration** raw to open up the **Source** window

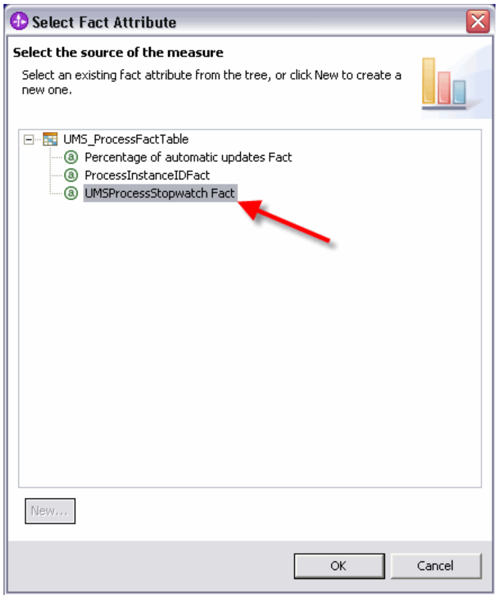


Measures

Add measures, which are calculations performed on facts to collect and combine them into a single value.

Cube	Measure	Source	Aggregation Function
UMS_Process Cube		UMS_Process	
	Average Process Duration		Average
Start Cube		Start	
UMS-in-SAP Cube		UMS-in-SAP	
UMS-Repair Cube		UMS-Repair	
UMS-Confirmation Cube		UMS-Confirmation	

Click on **UMSProcessStopwatch Fact** to highlight it. Click on **OK** to close he window.



Select the source of the measure

Select an existing fact attribute from the tree, or click New to create a new one.

- UMS_ProcessFactTable
  - Percentage of automatic updates Fact
  - ProcessInstanceIDFact
  - UMSProcessStopwatch Fact

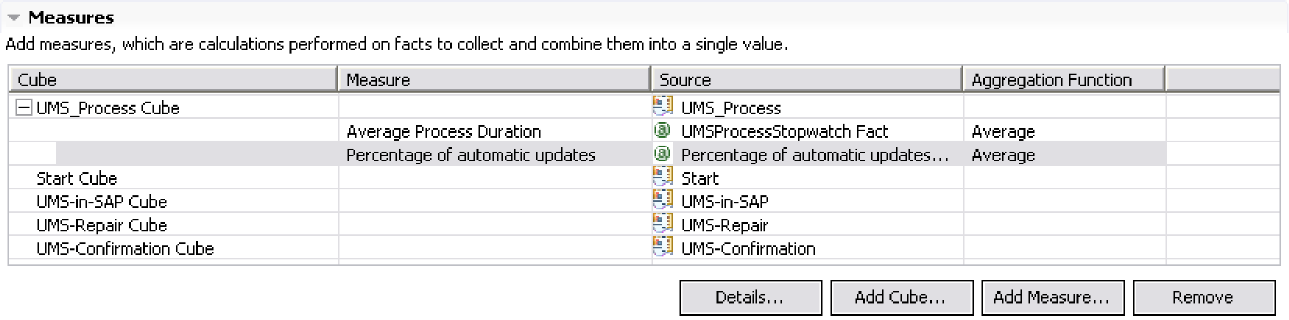
Note: by default the Aggregation Function is set to Average. This is correct for the **Average Process Duration**, since you want to have the average duration needed.

To finalize cube measure definitions you need to add the above steps to add one more measure.

The following table contains the details. Add this measures to the **UMS_Process Cube** too.

Measure Name	**Source**	**Aggregation Function**
Percentage of automatic updates	Percentage of automatic updates Fact	Average

The final Measures section looks as follow:

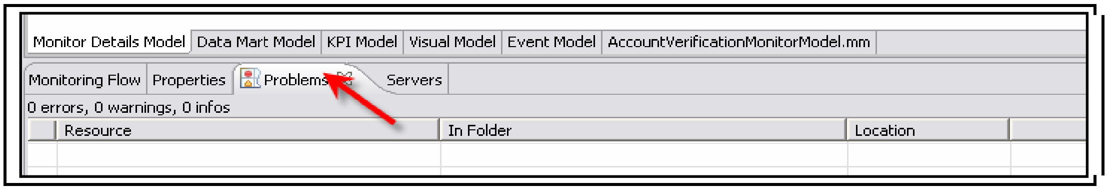


Measures

Add measures, which are calculations performed on facts to collect and combine them into a single value.

Cube	Measure	Source	Aggregation Function
⊟ UMS_Process Cube		UMS_Process	
	Average Process Duration	UMSProcessStopwatch Fact	Average
	Percentage of automatic updates	Percentage of automatic updates...	Average
Start Cube		Start	
UMS-in-SAP Cube		UMS-in-SAP	
UMS-Repair Cube		UMS-Repair	
UMS-Confirmation Cube		UMS-Confirmation	

Press the **Crtl+S** key to save the changes. Click on the **Problem** tab. There should be no problems left which are related to the measure model you created.



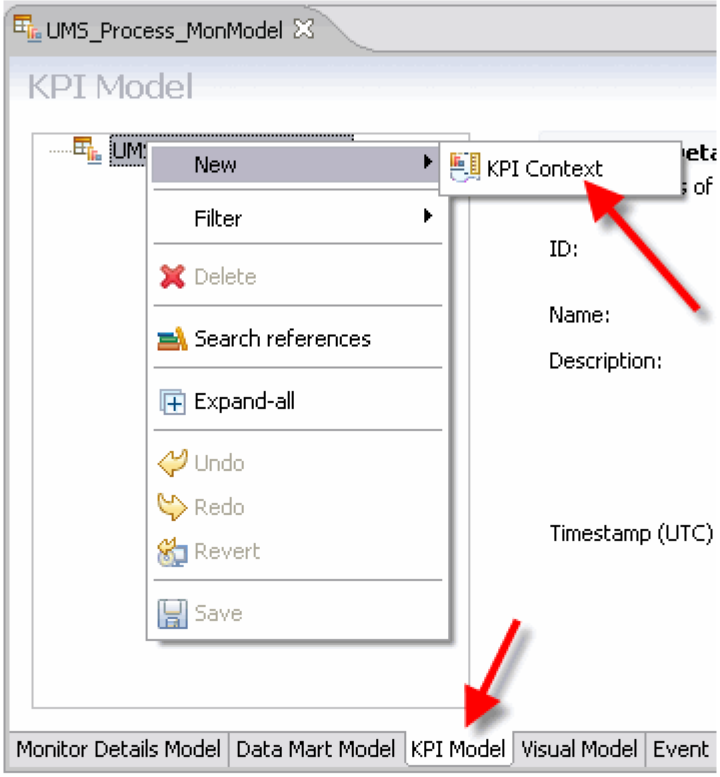
0 errors, 0 warnings, 0 infos

Resource	In Folder	Location

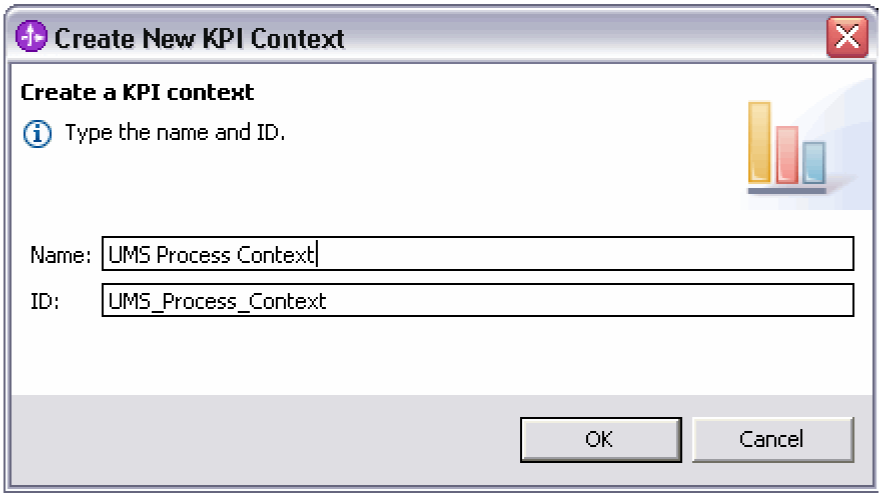
KPI Model and model merge

The key performance indicator (KPI) model is the part of the monitor model that contains the KPI contexts, which in turn contain KPIs and their associated triggers and events. KPIs are the detailed specifications used to track business objectives.

Switch to the KPI Model tab of the measure model and select **New → KPI Context**



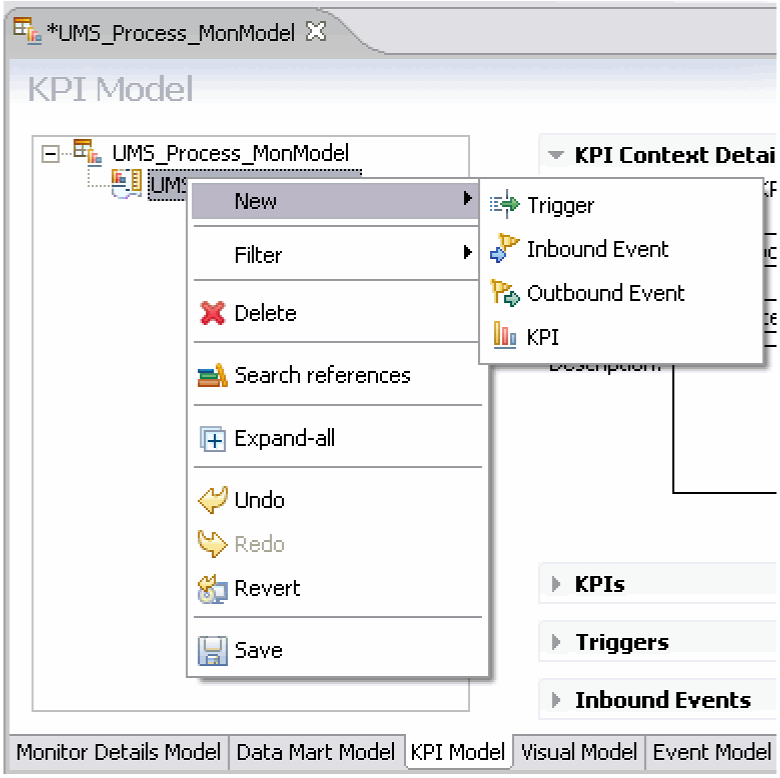
Name the KPI Context **UMS Process Context**.



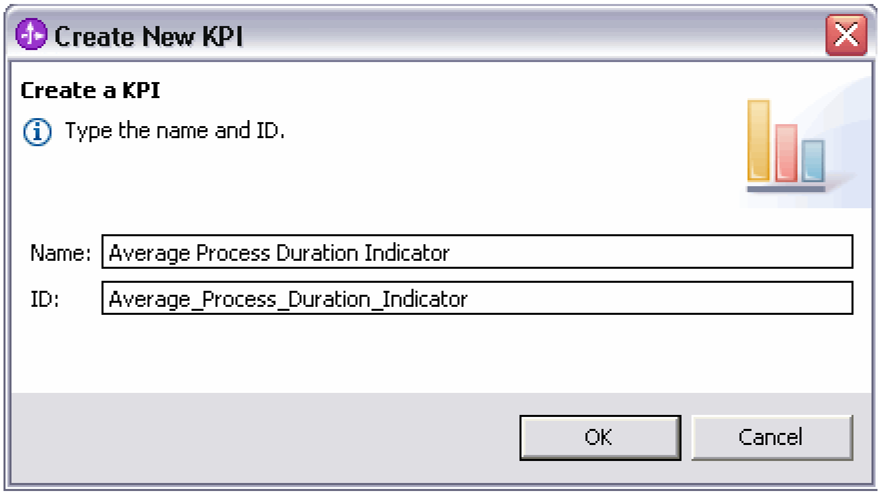
Name: UMS Process Context

ID: UMS_Process_Context

Right-Click on the created context in the measure model tree and select **New → KPI**



Name the new KPI **Average Process Duration Indicator**.



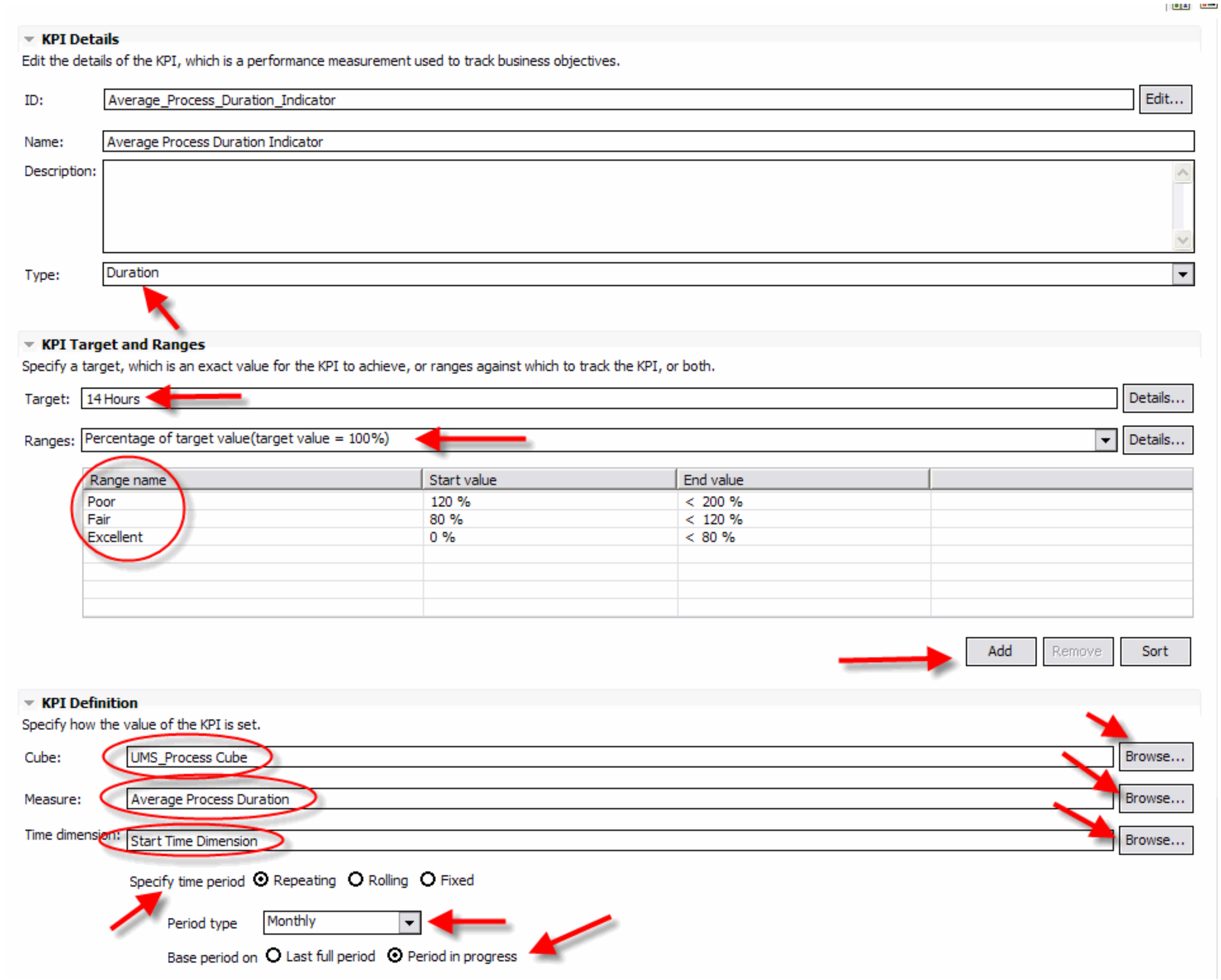
Create a KPI

Type the name and ID.

Name: Average Process Duration Indicator

ID: Average_Process_Duration_Indicator

Now enter the business measures for your KPI – use the values from the image below:



KPI Details

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

ID: Average_Process_Duration_Indicator

Name: Average Process Duration Indicator

Description:

Type: Duration

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: 14 Hours

Ranges: Percentage of target value(target value = 100%)

Range name	Start value	End value
Poor	120 %	< 200 %
Fair	80 %	< 120 %
Excellent	0 %	< 80 %

KPI Definition

Specify how the value of the KPI is set.

Cube: UMS_Process Cube

Measure: Average Process Duration

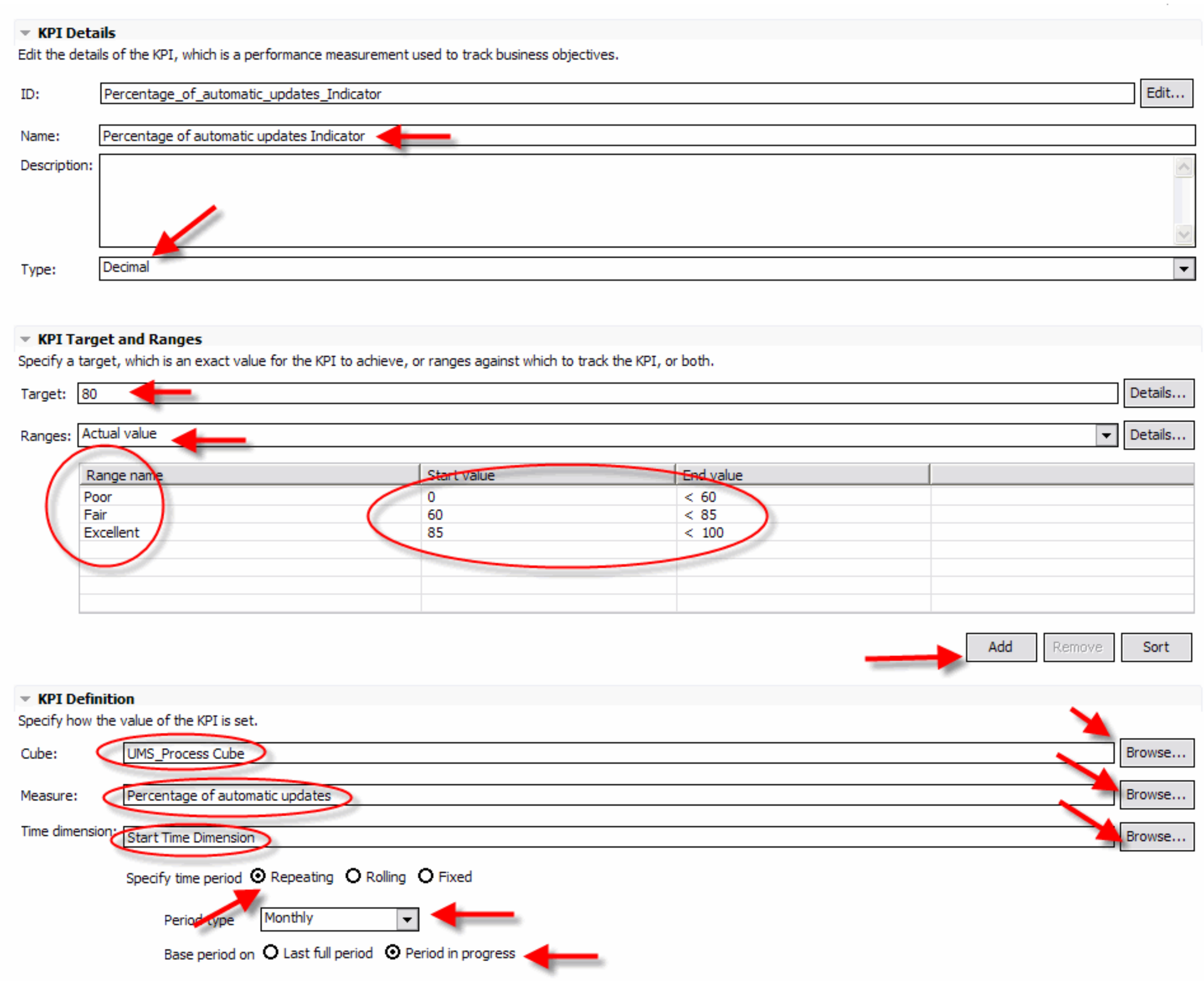
Time dimension: Start Time Dimension

Specify time period ● Repeating ○ Rolling ○ Fixed

Period type Monthly

Base period on ○ Last full period ● Period in progress

Create another **KPI** with the name **Percentage of automatic updates Indicator** using the same approach. Fill in the values displayed in the image below:



KPI Details

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

ID: Percentage_of_automatic_updates_Indicator [Edit...]

Name: Percentage of automatic updates Indicator

Description:

Type: Decimal

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: 80 [Details...]

Ranges: Actual value [Details...]

Range name	Start value	End value
Poor	0	< 60
Fair	60	< 85
Excellent	85	< 100

[Add] [Remove] [Sort]

KPI Definition

Specify how the value of the KPI is set.

Cube: UMS_Process Cube [Browse...]

Measure: Percentage of automatic updates [Browse...]

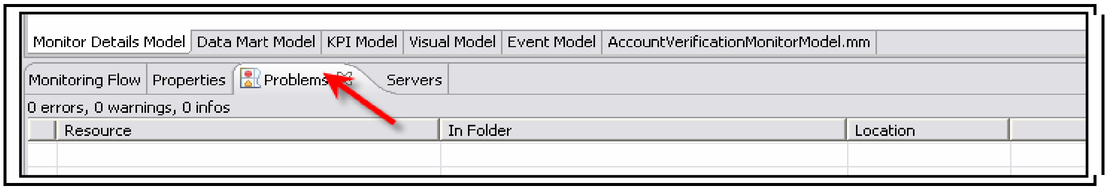
Time dimension: Start Time Dimension [Browse...]

Specify time period ● Repeating ○ Rolling ○ Fixed

Period type Monthly

Base period on ○ Last full period ● Period in progress

Press the **Crtl+S** key to save the changes. Click on the **Problem** tab. There should be no problems left which are related to the measure model you created.



Monitor Details Model | Data Mart Model | KPI Model | Visual Model | Event Model | AccountVerificationMonitorModel.mm

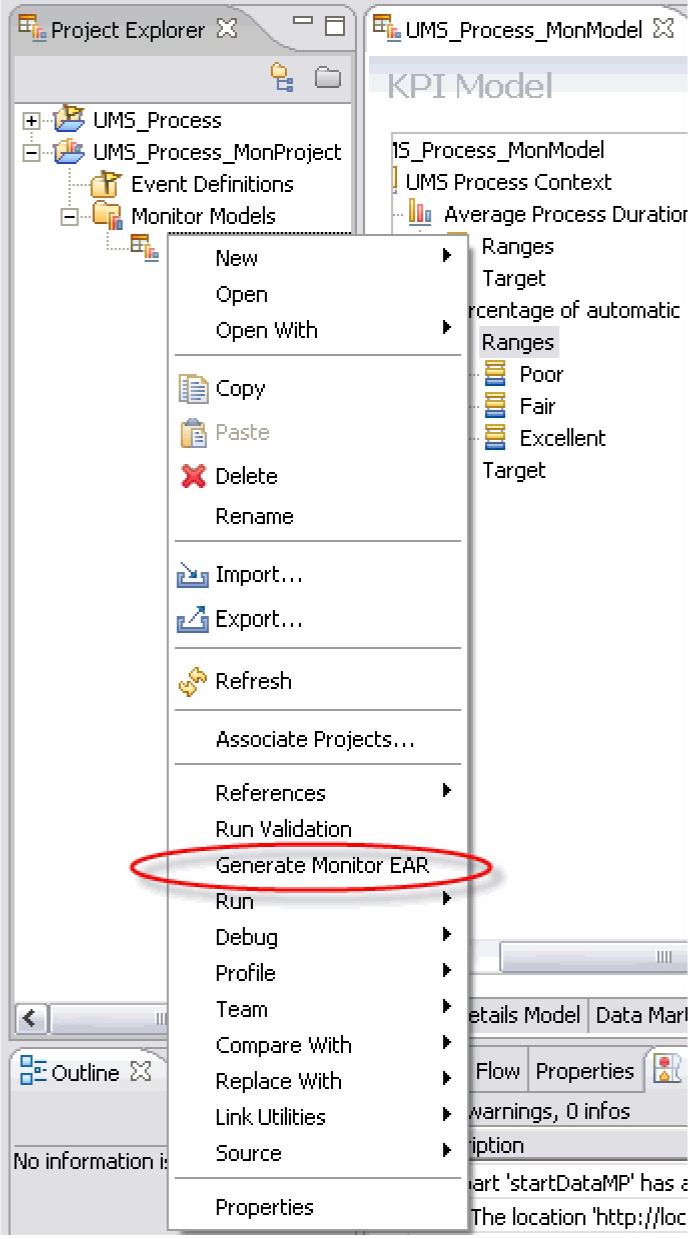
Monitoring Flow | Properties | Problems | Servers

0 errors, 0 warnings, 0 infos

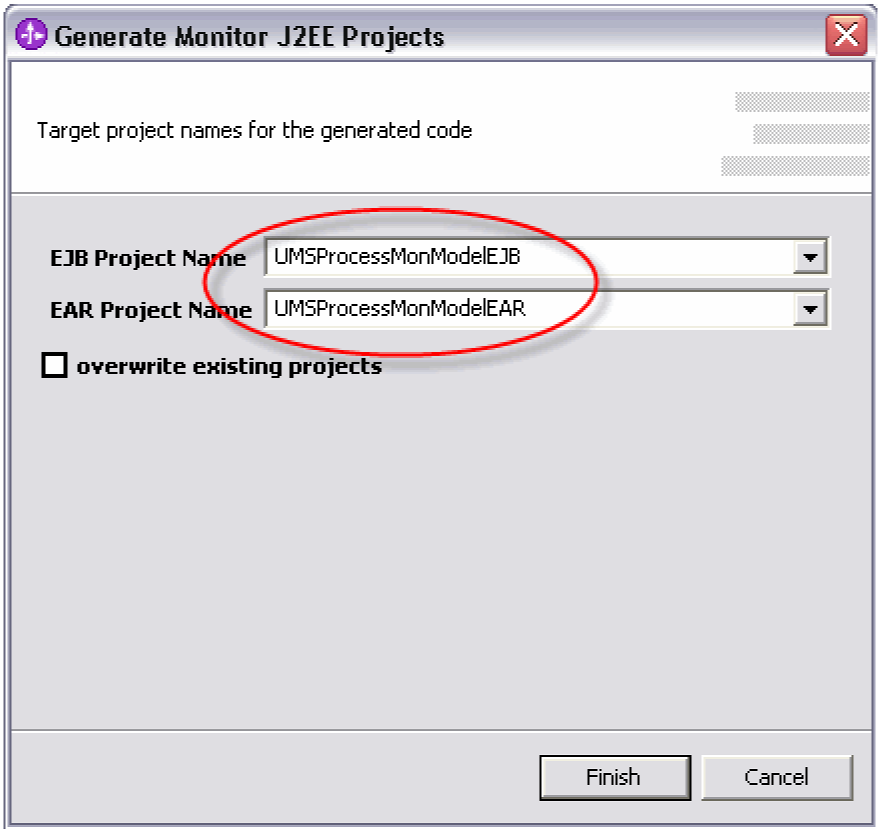
Resource	In Folder	Location

Generate the Monitor EAR file

The Monitor EAR file will be generated in the Monitor Deployment Lab by right clicks in the project tree on **UMS_Process_MonModel** and **Generate Monitor EAR**.



Enter **UMSProcessMonModelEJB** as the *EJB Project Name* and **UMSProcessMonModelEAR** as the *EAR Project Name* and click on **Finish.**



Generate Monitor J2EE Projects

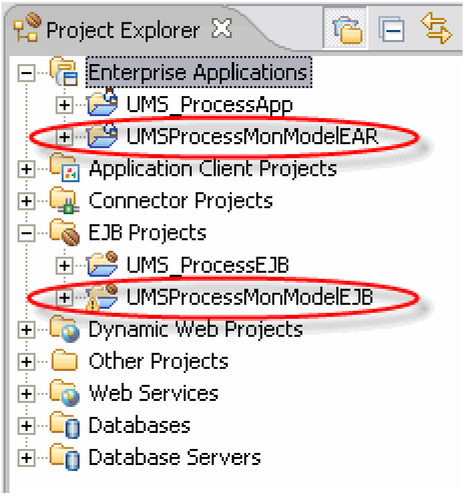
Target project names for the generated code

EJB Project Name: UMSProcessMonModelEJB

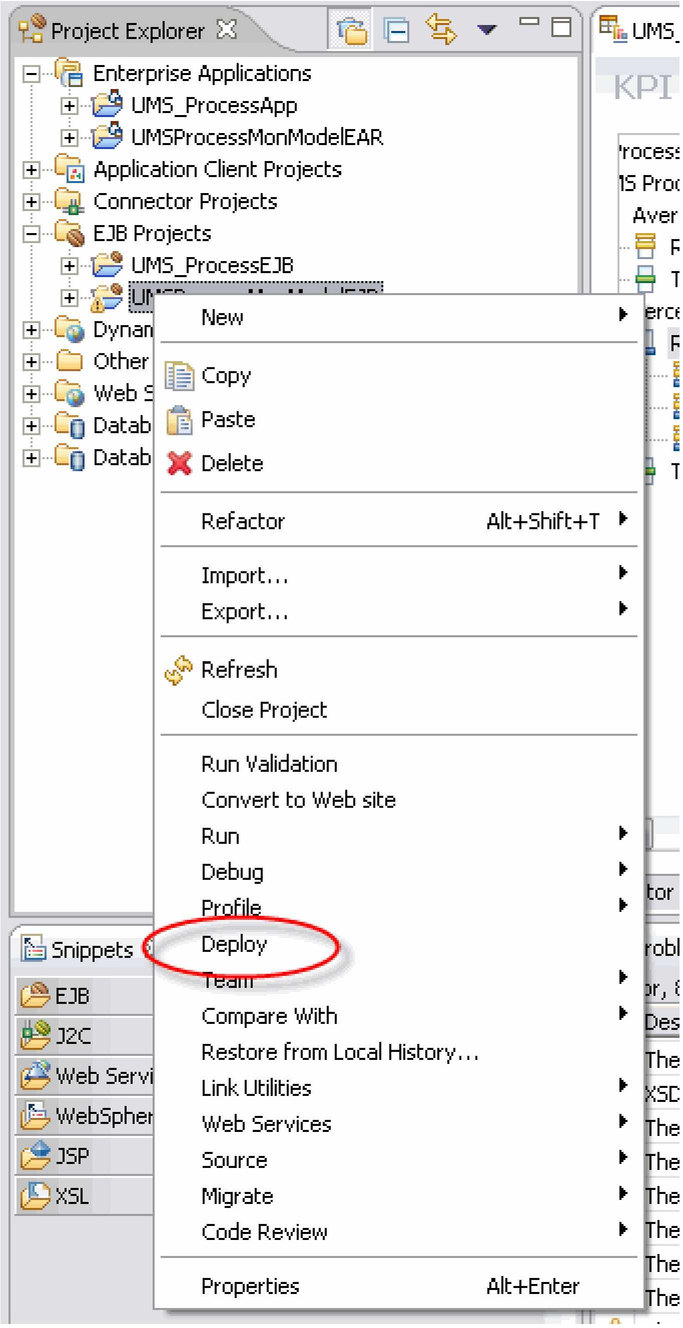
EAR Project Name: UMSProcessMonModelEAR

□ overwrite existing projects

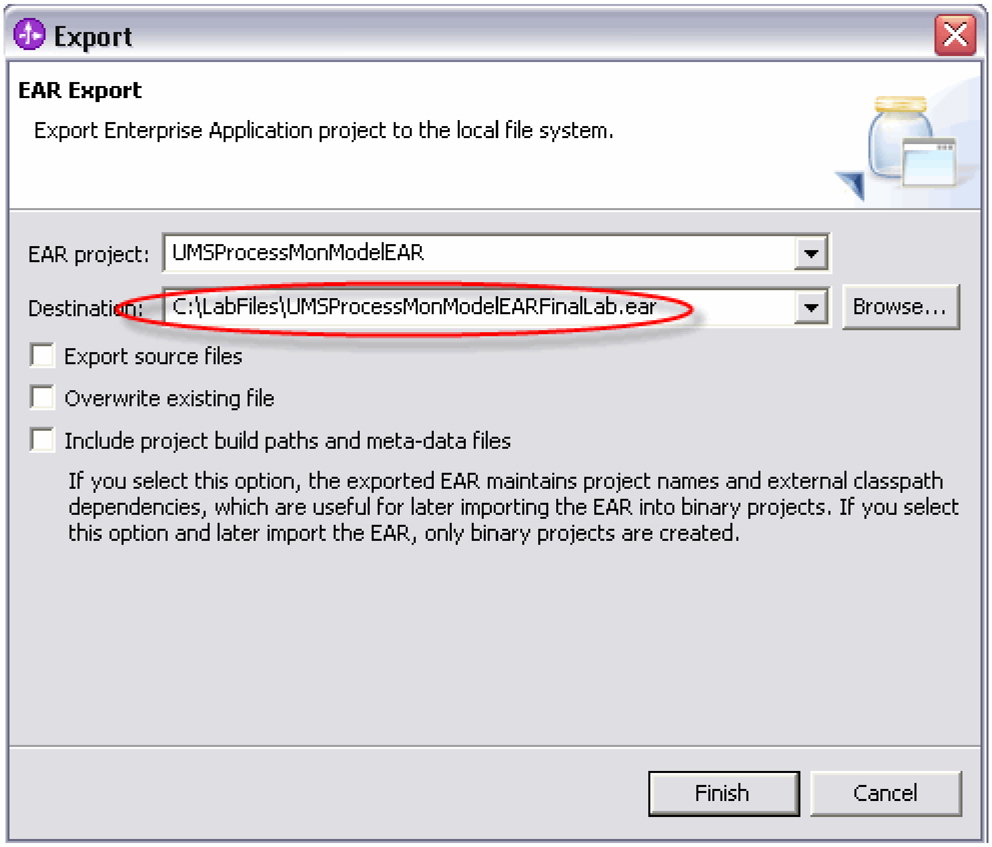
Switch to the **J2EE perspective** and verify if the EAR and the EJB project is listed in the **Project Explorer.**



Right-Click on the **UMSProcessMonModelEJB** project and select Deploy. This will speed up the deployment process later on the WebSphere Business Monitor runtime environment.



Right-Click on the **UMSProcessMonModelEAR** project and select **Export … → EAR File**. Enter in the pop up screen a suitable destination for the EAR file and click on **Finish**.



EAR Export

Export Enterprise Application project to the local file system.

EAR project: UMSProcessMonModelEAR

Destination: C:\LabFiles\UMSProcessMonModelEARFinalLab.ear

□ Export source files

□ Overwrite existing file

□ Include project build paths and meta-data files

If you select this option, the exported EAR maintains project names and external classpath dependencies, which are useful for later importing the EAR into binary projects. If you select this option and later import the EAR, only binary projects are created.

Congratulation!!!

You have successfully finished this lab!!!