



IBM Software Group

WebSphere® Business Monitor V6.0.2

Monitor Model Editor – Monitor Details Model



@business on demand.

© 2007 IBM Corporation
Updated 31 January 2007

This presentation covers the Monitor Details Model in the Monitor Model Editor of WebSphere Business Monitor V6.0.2.

Goals

- Introduce the Monitor Details Model within the Monitor Model Editor (MME)

The goal of this presentation is to cover using the Monitor Model Editor to create various components in the Monitor Details Model.

Agenda

- Monitor details model
 - ▶ Monitoring contexts
 - ▶ Counters
 - ▶ Stopwatches
 - ▶ Metrics
 - ▶ Keys
 - ▶ Triggers
 - ▶ Inbound events
 - ▶ Outbound events



The agenda includes in-depth coverage of the first of five of the Monitor sub-models, the Monitor details model, including the various components of the model such as monitoring contexts, counters, stopwatches, metrics, keys, triggers, and inbound and outbound events.

Monitor Details Model

- Displayed when you create a new model or double click on a model in Project Explorer
- One or more Monitoring Context definitions can be added to the Monitor Details Model
 - ▶ Monitoring context and key automatically created when model is created
- Right click on the model
 - ▶ New > Monitoring Context

The screenshot displays the 'Monitor Model Editor' interface. On the left, a tree view shows the structure of the 'ClipsAndTacks' model, including a 'ClipsAndTacks MC' (Monitoring Context) and a 'ClipsAndTacks Key'. The right pane is titled 'Monitor Details' and contains the following fields:

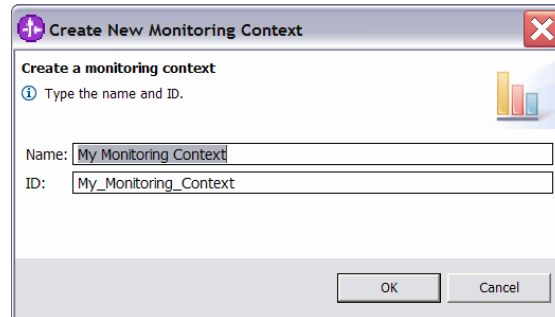
- ID: ClipsAndTacks
- Name: ClipsAndTacks
- Description: (empty text area)
- Timestamp (UTC): 2006-11-02T09:13:58

At the bottom, a navigation bar shows several tabs: 'Monitor Details Model', 'Data Mart Model', 'KPI Model', 'Visual Model', 'Event Model', and 'ClipsAndTacks.mm'. The 'Monitor Details Model' tab is highlighted and circled in red.

Shown here on the right pane is the Monitor Details Model tab, which is displayed when you create a new model or when you double click on a model in the project explorer. Details for the monitor model are displayed in the right pane and the navigation tree is displayed in the left pane. You can create one or more monitoring contexts in this model, and when you create a new model, a monitoring context and key are automatically created for you. To create a new monitoring context, right click in the tree and select New, followed by Monitoring Context.

Model naming dialog

- Displayed when creating new elements in the model
- ID is entered automatically based on the Name that you type, but spaces are replaced with underscores



5

This is the model naming dialog, which is presented when you create new elements in the monitor model. The name is a display name and the ID is the unique identifier for the element. When you enter the name in the dialog, the ID is automatically completed based on the name that you enter, with underscores substituted for spaces.

Monitoring Context

- You can view the details of components such as keys, and metrics on this page (no edit)
- You can double click on an item in the table to open the editor for that item
- Single click on the item in the tree to open the editor

Monitoring Context Details
Edit the details of the monitoring context, which contains the

ID: ClipsAndTacks_MC
Name: ClipsAndTacks MC
Description:

Keys

ID	Name	Type
ClipsAndTacks_Key	ClipsAndTacks Key	String

Metrics
Triggers
Inbound Events
Outbound Events
Counters
Stopwatches

On the Monitoring context page, you can view the details of the monitoring context but you cannot update them. To open the editor for an item in the monitoring context, double click on the item on the right side of the monitoring context page or single click on the item in the tree.

Monitoring Context

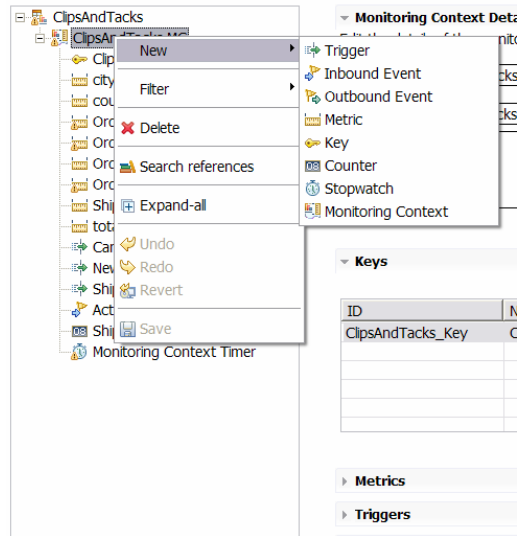


- An item's ID must be unique within the context
- A Monitoring Context must contain at least one Key
- An item defined within a context cannot be directly referenced by another peer context
- A Context is the definition of what an Instance of this Context would contain
- An instance of a Context must be started (or created), and it must terminate.
- Once a context is terminated, all metrics defined within the context are gone. Aggregations to this Context must be explicitly accomplished prior to termination of the instance.

Within any given monitoring context, an item must have a unique identifier and the monitoring context must have at least one key. You cannot reference items in one monitoring context from a peer monitoring context. At runtime, monitoring context instances are created based on the MC creation criteria that you define in the model, so the MC definition are those items that are contained in each MC instance. You are responsible for defining creation and termination criteria for every MC. Once a MC instance is terminated, all metrics in the instance are deleted, so you should aggregate the data into aggregated measures prior to termination of the instance.

Monitoring Context

- Right click on the monitoring context to create new monitoring elements
 - ▶ Trigger
 - ▶ Inbound Event
 - ▶ Outbound Event
 - ▶ Metric
 - ▶ Key
 - ▶ Counter
 - ▶ Stopwatch
 - ▶ Monitoring Context



A monitoring context can contain monitoring elements such as triggers, inbound events, outbound events, metrics, keys, counters, stopwatches and other child MCs. To create them, right click on the MC in the tree then select New.

Counters



- A Counter is a specialized metric for tracking the number of times something happens
- Provides add one (+1), subtract one (-1) and reset-to-zero functions

Counters provide a pre-built mechanism for counting how many times things happen within your model. When triggers are fired, you can add one to a counter, subtract one from the counter or reset the counter to zero.

Counters



- Use sorting to create an index on the schema for dashboard performance sorting in instances view
- Specify trigger or inbound event
- Specify resulting action
 - ▶ Add one
 - ▶ Subtract one
 - ▶ Reset to zero

▼ **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
<input checked="" type="checkbox"/> Ship Order to Customer Trigger	Add One
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	



Shown here is an example of a counter. You can select the checkbox so that the counter will be used for sorting, which means an index will be created on the schema for improved performance.

In Counter Controls, you define the trigger or inbound event that triggers the resulting action, which can be to add one, subtract one or reset the counter.

Stopwatches



- A Stopwatch is a specialized metric for measuring duration of time between two events
- Provides start, stop and reset functions

Stopwatches provide a pre-built mechanism for timing durations between events. You specify triggers that cause the stopwatch to start, stop or reset.

Stopwatches



- Use accumulating to sum multiple pairs of start/stop
- Use sorting to create an index on the schema
- Specify trigger or inbound event
- Specify resulting action – start/stop use CBE creation time not Monitor Server system time
 - ▶ Start
 - ▶ Stop
 - ▶ Reset

Stopwatch Details

Edit the details of the stopwatch, which keeps track of elapsed time. If an accumulating stopwatch is used, the durations are added together to produce a total time.

ID:

Name:

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
* Ship Order to Customer Trigger	Stop
* Cancel Trigger	Stop



Shown here is a screen capture with an example of a stopwatch. An accumulating stopwatch indicates that the specified trigger actions result in accumulation of duration times, meaning that if you have two pairs of 'start' and 'stop' actions, the duration times between both pairs will be added together to produce an 'accumulated' time.

In Stopwatch Controls, you specify the trigger or inbound events that cause the resulting actions, which can be start, stop or reset. The start and stop functions use the creation time on the inbound event, not the server system time.

Metrics



- A Metric collects some piece of data from your monitored environment
 - ▶ For example the value of a specific extended data element on the CBE

A metric is used to store payload information from the inbound events, and also as a source for aggregation of other business measures.

Metrics



- Type – boolean, date, dateTime, decimal, duration, integer, string, time
- Default value
 - ▶ Literal or function set during MC initialization
- Use sorting to create an index on the schema
- Metric Value Maps
 - ▶ Trigger
 - ▶ Expression – event, key, metric, stopwatch, counter

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

ID:

Name:

Description:

Type:

Maximum String Length:

Allocate additional space in database to accommodate Unicode string for glc

Default Value:

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 eval the trigger fires.

Trigger	Expression
New Order Trigger	Activity_Event/extendedData/OrderBOData/city



Shown here is a screen capture of a metric in the MME.

A metric can be of type boolean, date, dateTime, decimal, duration, integer, string or time. You can specify a default value that is set when the monitoring context instance is created and can be a literal or a function. The Metric Value Maps identify the triggers that set the value of the metric based on the expression that you specify.

A map that has no trigger executes whenever any of its inputs changes.

Keys



- Used to correlate runtime events with the instance of the Monitoring Context
- The key value of a Monitoring Context Instance can only be set once per lifetime of the instance
 - value must be unique amongst child MC instances of same parent
 - global key is aggregation of "key metrics" in MC and ancestors

Keys are used to correlate the events coming into the model with the correct monitoring context instances and the key value can be set only once during the lifetime of the MC instance and must be unique among peer MC instances. A global key is the aggregation of all the key metrics in a MC and its parents.

IBM Software Group IBM

Keys

- Type – boolean, date, dateTime, decimal, duration, integer, string, time
- Default value
 - ▶ Literal or function
- Use sorting to create an index on the schema
- Key Value Maps
 - ▶ Expression – event data

Key Details
 Edit the details of the key. Each monitoring context requires at least one key.

ID:

Name:

Description:

Type:

Maximum String Length:

Allocate additional space in database to accommodate Unicode string for globalization

Default Value:

This key can be used for sorting

Key Value Maps
 Specify the expressions that set the value of the key.

Expression

WebSphere Business Monitor V6.0.2 – Monitor Model Editor © 2007 IBM Corporation 16

Shown here is a screen capture of a key. The type can be boolean, date, dateTime, decimal, duration, integer, string or time and you can specify default values or sorting capability. In the Key Value Maps you specify the expression that typically sets the value of the key from some payload information on an inbound event.

Triggers



- A trigger is used to cause a calculation to be made
 - ▶ Trigger defines when that happens
- Metrics, Stopwatches and Counters all use triggers

Triggers are used to tell the Monitor when to capture and calculate values and are used by metrics, stopwatches and counters.

Triggers

- Repeatabe means trigger it every time condition evaluates to true
- Terminate an MC
- Trigger source
 - ▶ Recurring wait time
 - ▶ Trigger
 - ▶ Value change on key, metric or counter
 - ▶ Inbound event
- Trigger condition

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

ID:

Name:

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.



Shown here is a screen capture of a trigger. Repeatabe means that the rigger is fired beginning when the condition first resolves to 'true' and every time after that when the Trigger Source is received and the condition remains 'true'. Non-repeatabe means that the trigger is fired beginning when the condition first resolves to 'true', and will not fire again unless the condition first resolves to 'false' and then to 'true' again.

You can specify that the trigger causes the monitoring context to terminate.

The trigger source can be a recurring wait time, another trigger, an inbound event, or a value change on a key, metric or counter. You can also specify a condition to further refine when the trigger source will cause the trigger to fire.

Inbound event



- Associate with a previously defined event type
- Restrict which events are applicable to this Monitoring Context via the **Filter**
- Define how the event is delivered to a Monitoring Context using the **correlation criteria**
- Describe what to do when a Monitoring Context match is or is not found or if multiple matches are found

An inbound event is used to associate an event type with an event that Monitor subscribes to. You can use filters to restrict the set of events which the MC will process and you can also specify correlation information to determine which MC instances are affected by the event. In addition, you can define the behaviour when correlation occurs.

Inbound event

- Type is a pre-existing event definition (CBE)
- Optional filter condition to restrict event set
- Optional correlation expression based on the specified type
- Instance matching behavior

ID:

Name:

Description:

Type:

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

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

If no instances are found

If one instance is found

If multiple instances are found

Shown here is a screen capture of an inbound event, which is a CBE event type that has already been imported into the project. Filter Condition is optional and specifies criteria used to restrict the event set. The Correlation Expression is optional and specifies an expression that identifies the MC instances to match the event to based on the MC key. You also specify what to do if no instances are found, or if one instance or multiple instances are found.

Outbound event



- Associate with a previously defined event type
- Restrict the conditions under which the event is emitted using the **Filter**
- Modify the event attribute values when the event is triggered
- Are used for outbound situation events – both monitoring contexts and KPI contexts

An outbound event is used to associate an event type with an event that Monitor will emit. You can use filters to restrict the set of events which the MC will process and you can specify the event attribute values for the outbound event. You can use outbound events for outgoing business situation events for both monitoring contexts and KPI contexts.

Outbound event

- Type is a pre-existing event definition (CBE)
- Event attribute details
 - ▶ Specify trigger
 - ▶ Assign values to event attributes – literal, event data, key, metric, counter, stopwatch, functions
 - ▶ Use BusinessSituationName for business situation events
- Optional filter

Name:

Description:

Type:

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

Name	Type	Expression
Order Fulfillment Timer Trig...		
Property Data		
Extended Data		
AverageOrderProcess	string	'Too many orders have been declined'
BusinessSituationName	string	'Average shipment is too late'

Filter Condition
Define a condition to determine whether to send the event.

Shown here is a screen capture of an outbound event in the MME. Type is a pre-existing event definition that has been created in the project. In event attribute details, you specify the trigger that determines when this event is emitted, and you also specify expressions to set the values on the event attributes. For business situation events, you will need to use an extended data element called BusinessSituationName in the CBE definition for the event type. Optionally, you can specify a filter to determine whether to send the event.

Summary

- You have seen how to use the Monitor Model Editor to create various components in the Monitor details model



In summary, this presentation provided a review of the components that comprise the Monitor details model, and described how to maintain them using the Monitor Model Editor.

Trademarks, Copyrights, and Disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	CICS	IMS	MQSeries	Tivoli
IBM (logo)	Cloudscape	Informix	OS/390	WebSphere
eIogo business	DB2	Series	OS/400	xSeries
AIX	DB2 Universal Database	Lotus	pSeries	zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

Other company, product and service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprocessing in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2006. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.

