Deploying HP OMi in an HP BSM Solution

Technical white paper

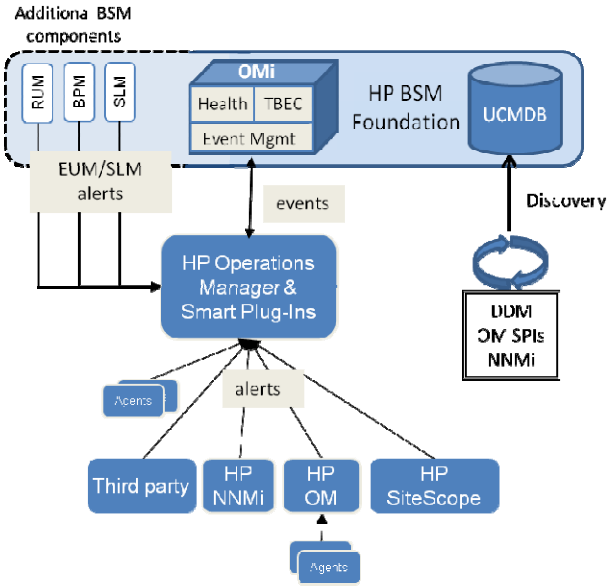
Table of Contents

Introduction

This white paper describes how HP Operations Manager i (HP OMi), the new operations management product introduced by HP in 2008, can be deployed as part of a larger HP Business Service Management (BSM) solution.

HP OMi links the operations management world with the business service management world and can be seen as *the* event management foundation for the complete HP BSM solution. HP OMi provides a new, enhanced event console and is able to process events that originate from various HP BSM components, primarily from HP Operations Manager (HPOM) for Windows or Unix, but also from applications such as HP SiteScope, HP Business Process Monitors, HP Real User Monitors, HP Network Node Manager i and even from third party applications.

Figure 1: HP OMi architecture



The first chapter of this white paper, "How to Design the HP BSM Solution Architecture around HP OMi", helps you to design the role that HP OMi plays in the HP BSM solution. It talks about a typical deployment scenario and explains how the different components of the HP BSM solution work together. HP OMi high-availability and security aspects are discussed as well.
A section about best practices and guidelines helps you to decide how to integrate data into the overall HP BSM solution. The last section of the first chapter outlines what to do if the out-of-the-box content (provided by the UCMDB, HP OMi and the HP BSM solution overall) needs to be extended.

The second chapter "Combining HP OMi Data and other HP BSM Data" explains what you need to do to show HP OMi data and HP BSM data side-by-side in the HP BSM Dashboard and in the HP OMi health perspective.

The third chapter "How to Configure Event Integrations to Obtain Maximum Benefits from HP OMi" concentrates on the Health Perspective and Topology Based Event Correlation (TBEC) of HP OMi, and

describes how you need to configure the various integrations so that events can be mapped to Configuration Items (CIs), set with a meaningful health indicator value in HP OMi, and be correlated.

The last chapter of this white paper contains some information about possible use cases for HP Service Manager, HP Operations Orchestration and HP Dependency Mapping Automation.

How to Design the HP BSM Solution Architecture around HP OMi

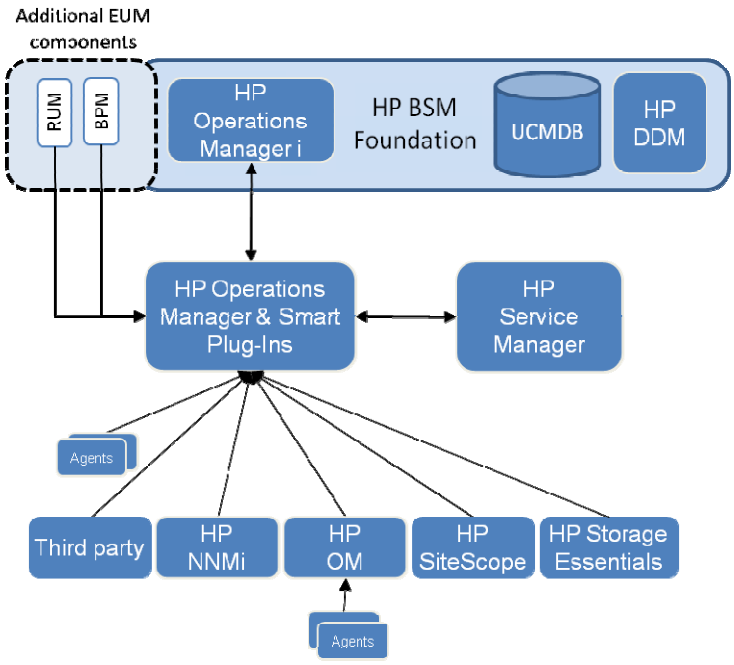
A Typical HP BSM Solution Deployment with HP OMi

HP OMi offers the most value in solutions that aim to provide an "Operations Bridge" (an ITIL term) which consolidates performance and event management from technology based silos, such as servers, networks, applications and storage, into a single event stream that is handled by a unified team.

This consolidation is the basis for understanding relationships between events, and allows operators to use consistent and repeatable processes when working on all the events.

HP OMi adds additonal value by mapping the events to UCMDB Configuration Items (CIs), by showing fine-grained health of CIs based on received events, and by automatically correlating events based on the discovered topology.

Figure 2: A typical HP BSM solution deployment with HP OMi



It is therefore quite typical that a HP BSM solution with HP OMi contains various other management products, such as HP Network Node Manager i (NNMi,for network management), HP Storage Essentials (for storage management) and HPOM and HP SiteScope (for agent-based and agent-less system and application management). We also assume that many customers looking for such a solution will be interested in HP Discovery and Dependency Mapping (DDM) for automated discovery.

A solution with such an Operations Bridge typically also includes a link to a help desk or incident management system, where forwarded events are tracked as incidents as part of an incident management process.

The solution is complemented by End User Management (EUM) products such as HP Business Process Monitors or HP Real User Monitors that provide the *customer view* on the IT environment.

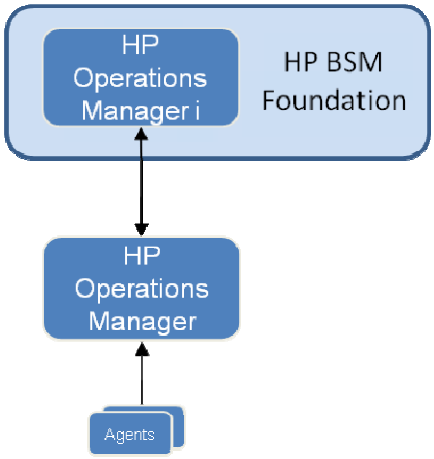
Such a HP BSM solution might even include third party management software, which is typically used to monitor certain environments or niches not monitored by the other solution components.

Note that although we believe that HP OMi offers the most value in such an HP BSM solution, you do not have to implement the complete solution as outlined above all at once, taking an incremental approach if you prefer:

- You can start your HP OMi Operations Bridge using the discovery provided by the HP Operations Smart Plug-Ins (and NNMi if it is part of the solution) and add HP DDM later (if necessary).
- The integration with an incident management process is optional.
- End user monitoring can be integrated at any time into the Operations Bridge.
- Each monitoring solution integrated into HPOM is optional.

However, the basis for every HP BSM solution with HP OMi is at least one HP Operations Manager (can be either HPOM for Windows or for UNIX). Figure 3 shows such a minimal HP OMi solution, targeted on operations management.

Figure 3: Minimal HP OMi solution



Deployment Architecture

Once it has been decided what functionality should be provided by the HP BSM solution (and which components should be deployed), the next logical step is to outline the system deployment architecture. What and how many servers will be required for the solution deployment? How can a high availability of the solution be achieved? Where will the different components be installed? How will they communicate with each other? What are the sizing limitations?

Multiple HPOM Server Architecture

The scalable architecture of HPOM enables one or more management systems to be combined into a single management solution.

Agent-based flexible management allows you to configure managed nodes to send messages to management servers other than the primary management server. Nodes can redirect messages based on the message attributes and the time.

Server-based flexible management allows you to configure management servers to forward messages to other management servers. Management servers can also forward message operations, so that the status of each forwarded message is synchronized on all management servers.

Agent-based and server-based flexible management can be combined to suit customer requirements and can be used to achieve a high availability of the management solution.

Multiple HPOM Servers with HP OMi

HP OMi can work with such a multi-server HPOM environment, but keep the following in mind:

1. HP OMi connects to one HPOM server at a time.
2. HP OMi does not yet offer a function to automatically connect to another HPOM server in case of a failover situation (see also the section "High Availability and Load Balancing" on the next page.)
3. HP OMi 8.10 has been tested connecting to a single HPOM server. Some tests have been done connecting to different HPOM servers at different times. There is one known issue when connecting to a different HPOM server:
    - Topology synchronization: When run after the switch, all CIs created by a previous topology snchronization are removed and new CIs are created based on the data on the new HPOM server.

Therefore, if you want to receive all messages from a hierarchical HPOM environment, and map them to corresponding CIs, you must connect to the top-level HPOM server and, if you want to use topology-synchronization to populate the CMDB, then you have to make sure that the top-level HPOM server contains the nodes and services of all underlying servers.
(HPOM allows you to download services and nodes from one HPOM server and to upload these on another HPOM server. This works even between HPOM for Windows and HPOM for UNIX servers. See the HPOM documentation for details.)

Note: A node configuration in HPOM for Windows does not require an IP address. However, IP addresses are important to identify a host in the UCMDB (see Integrating CI Data / Discovery Options on page 13). If your top-level HPOM server is a Windows server, you should therefore make sure that the *Resolve IPs During Synchronization* setting is enabled and that all node names can be resolved on the processing server.

High Availability and Load Balancing

A typical HP BSM solution as outlined above should use the **multi-server** HP BSM deployment with **load-balancing** for the Gateway servers and **high-availability** for Operations Manager, Gateway and Processing servers. How many Gateway servers are required depends on the number of monitors and events that have to be processed and on the number of users that will connect. For details, see the HP OMi *Performance Guide*.
For high-availability and load balancing details, see the *HP Business Availability Center Deployment Guide* and the HP Operations Manager manuals and High Availability white papers.
HP OMi supports all high-availability and load balancing features provided by the HP BSM platform.

If you plan to implement a high-availability concept for the HPOM server, please see the OMi *Installation and Deployment Guide* from August 2009, which contains details about additional steps necessary for backup server/failover scenarios.

Hardening

The HP BSM platform is designed so that it can be part of a secure architecture which minimizes potential attack points. HP OMi uses the HP BSM platform and therefore supports the same hardening functionality as the platform.

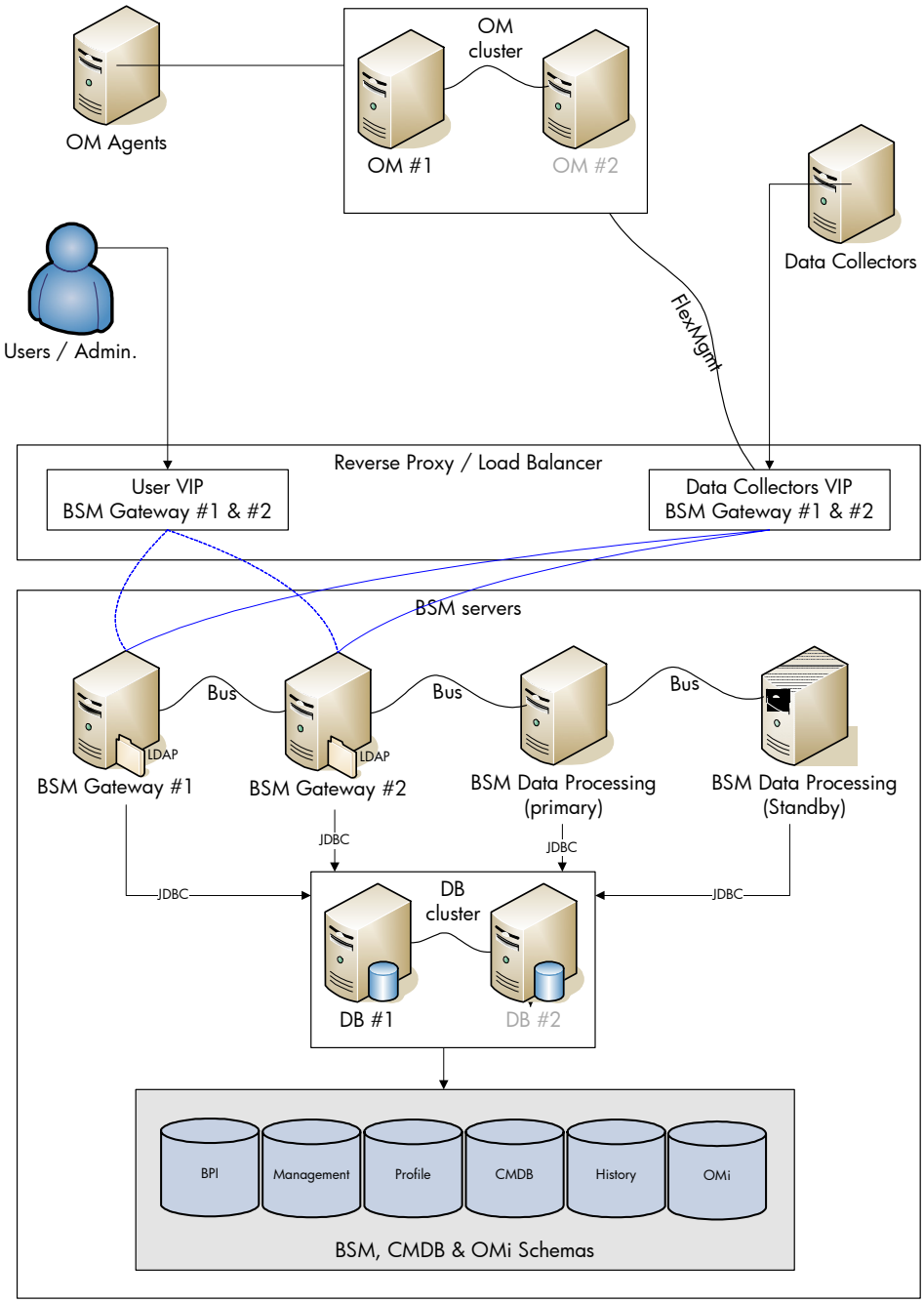
Key hardening features are:

- **DMZ architecture using a firewall**
  A secure architecture is typically a DMZ architecture using a device as a firewall. The basic concept of such an architecture is to create a complete separation, and to avoid direct access between the HP BSM clients and the HP BSM servers.
- **Reverse proxy architecture**
  HP BSM fully supports a reverse proxy architecture as well as a secure reverse proxy architecture. This is the recommended solution to deploy HP BSM and HP OMi.
- **SSL communication protocol**
  The Secure Sockets Layer protocol secures the connection between the clients and the servers.

For details, see the *HP Business Availability Center Hardening Guide*.

An example of a deployment architecture incorporating high-availability, load balancing and hardening is shown in Figure 4.

Figure 4: High-available, secure, scalable HP BSM solution deployment with HP OMi



Best Practices and Guidelines

Integrating Event Data

Event data should be integrated in OM. HPOM can be seen as the first event consolidation layer that already provides useful features such as deduplication and good-bad message correlation. From HPOM, the HPOM message is forwarded to HP OMi.

HP Operations Manager offers various ways to integrate event-based data:

- HPOM policies - integrate SNMP traps and WMI events or read event data from log files
- opcmsg command line interface - create and modify HPOM messages from the command line
- opcmsg C, COM and Java APIs - create and modify HPOM messages via API
- HPOM Incident Web Service interface - create and modify HPOM messages via a standards-based web service interface

For details, see the Operation Manager online help and manuals.

There are several integration packages already on the market that make use of these interfaces.

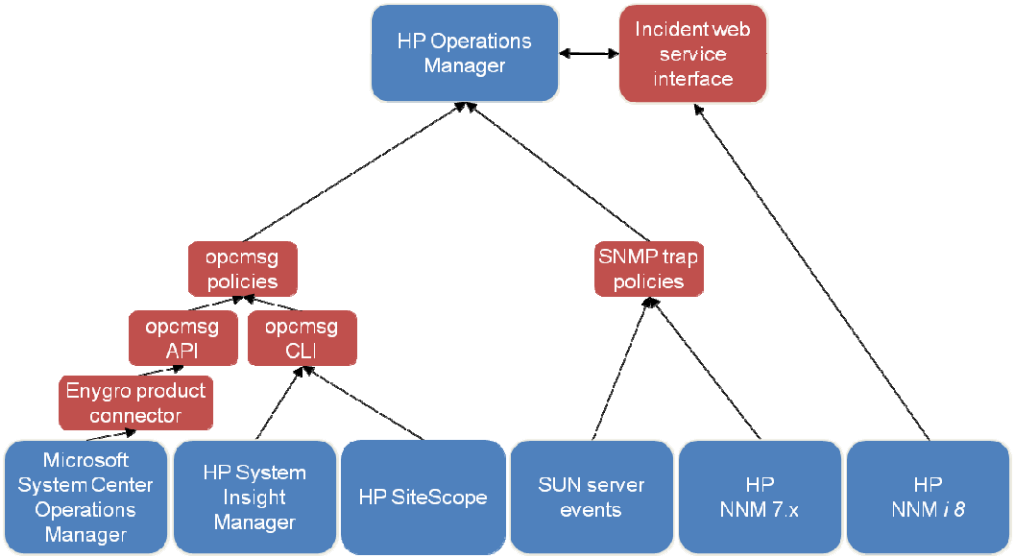
HP provides the following integration packages that integrate events into OM:

- HP Operations Smart Plug-in for HP Systems Insight Manager
  Integrates HP SIM events via SNMP trap policies
- HP NNM/NNMi Adapter for HP Operations Manager
  Integrates NNM alerts via SNMP trap policies and NNMi events via the HPOM Incident Web Service interface
- HP SiteScope Adapter for HP Operations Manager
  Integrates SiteScope alerts via opcmsg policies and the opcmsg API

Examples of third party integration packages are:

- Engyro Product Connector for Microsoft System Center Operations Manager to HP OpenView
  Integrates Microsoft System Center Operations Manager messages via opcmsg C API with HPOM for Windows
- Sun Integration for HP OpenView Operations (for Windows and Unix)
  Integrates Sun server hardware events via SNMP trap policies

Figure 5: Event integration technology used by some of the existing integration packages for HP Operations Manager



Once events have been integrated into HPOM, they can also be forwarded to HP OMi as well.

In an HP OMi solution, it is not advisable to use the Event Management System (EMS) HPOM integration to get the HPOM events into the HP BSM platform via SiteScope. HP OMi can be seen as the successor of that integration, as it offers a more flexible way of setting KPIs based on event data and as it provides direct access to the event data in the event and health perspectives of the HP BSM console.

Integrating Performance Data

Performance data for HPOM can be collected by the embedded performance component (EPC) of the HPOM agent or by the HP Performance Agent.

Data collected by these agents can be used in HPOM measurement threshold policies to create alarm messages, and historical data can be displayed using HP Performance Manager or the Performance Grapher of HP OMi, helping operators to analyze and fix problems.

Integrating Performance Data Using Measurement Threshold Policies

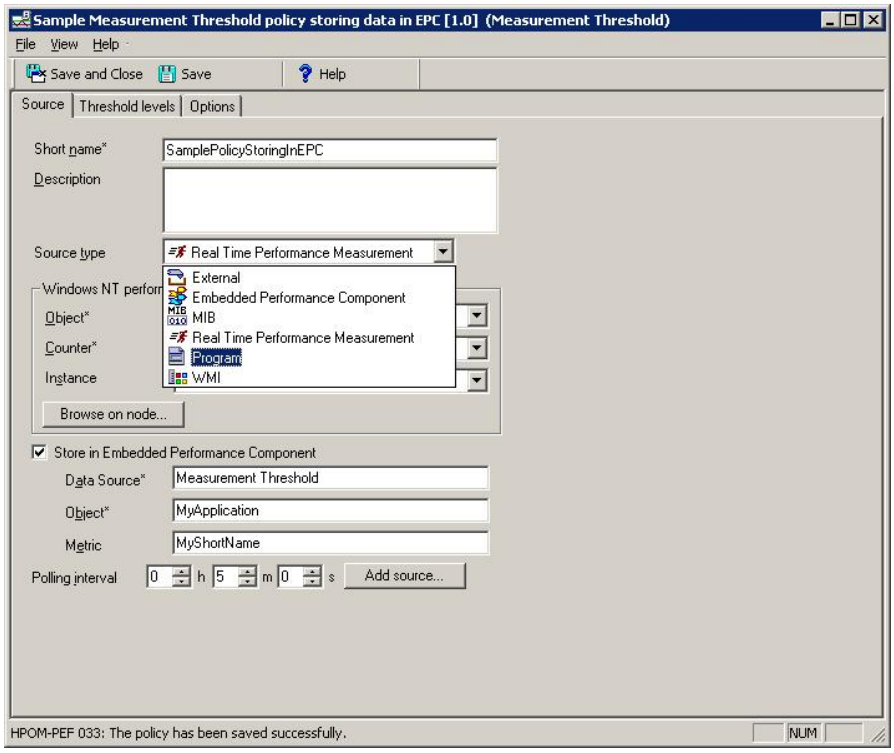
An easy way to integrate performance data is to use the HPOM measurement threshold policy type.

The HPOM measurement threshold policy enables you to collect performance data from several sources:

- External / Program - sent data from an external program (using the opcmon command line interface or API)
- MIB – collect metrics from a SNMP Management Information Base (MIB)
- Real Time Performance Measurement – collect metrics from Windows Performance Counters
- WMI – collect metrics from Windows Management Instrumentation

The metrics collected from these sources can be easily stored in the Embedded Performance Component using the "Store in Embedded Performance Component" option (an example is shown in Figure 6).

Figure 6: HPOM Measurement Threshold policy storing metric in EPC



Integrating CI Data / Discovery Options

The health perspective and topology-based event correlation features of HP OMi depend on an up-to-date CMDB containing configuration items for all monitored objects *and* their relationship to other configuration items.

In large, dynamic and constantly changing IT environments, automatic discovery and reconciliation techniques are essential for maintaining an up-to-date CMDB.

For HP OMi, you have a choice of two automatic discovery and reconciliation techniques:

1. Create the CIs and relationships using discovery features of the UCMDB and HP Discovery and Dependency Mapping.
2. Create the CIs and relationships using the topology synchronization feature of HP OMi, which reuses the service model native to HPOM to create corresponding CIs and relationships. This requires a suitable service model in HPOM.

Creating CIs Using UCMDB and HP DDM Features

HP DDM automatically discovers and maps logical application assets in Layers 2 to 7 of the Open System Interconnection (OSI) model. The basis of the discovery technology are discovery patterns.

HP DDM Standard discovers hosts and hosts relationships. This is not sufficient to discover all CI types that are monitored by HP OMi.

HP DDM Advanced discovers resources such as applications, databases, network devices, servers, and so on. **HP DDM Advanced ships discovery patterns for all CI types that are part of the HP OMi content packs**, and even more.

HP DDM Advanced also allows users to extend the UCMDB model and write their own patterns.

Users can also query external data sources:

- Comma Separated Value (CSV) Files
- Properties Files
- Databases

For details, see the *HP Discovery and Dependency Mapping* manual, especially chapter 8 -*Discovery and Dependency Mapping Content* and chapter 9 - *Importing Data from External Sources*.

The UCMDB can also be populated using special integration packages. See the *HP Universal CMDB–HP Network Node Manager* i *(NNMi) Integration Guide* and *HP Universal CMDB–Storage Essentials (SE) Integration Guide*. Be aware that these integration packages only create certain CI types (like host, IP), and not all CI types that are part of HP OMi content packs.

Creating CIs Using HPOM Service Model and Topology Synchronization

Many Operations Manager customers are already familiar with the HPOM service views or Service Navigator to visualize dependencies between IT resources and IT services to their operators. These customers have either used the service discovery features of the Operations Smart Plug-Ins or their own discovery mechanisms to create such a service tree.

If this is the case, then they can use HP OMi's topology synchronization to create CIs and CI relationships, based on the already existing service model and topology synchronization mapping rules. HP OMi ships mapping rules that are able to map service models created by the following Operations Manager Smart Plug-Ins (called *OMi-enabled SPIs* in the following):

- HP Operations Smart Plug-in for Databases (Oracle and MS SQL Server only)
- HP Operations Smart Plug-in for IBM WebSphere Application Server
- HP Operations Smart Plug-in for BEA WebLogic Application Server

- HP Operations Smart Plug-in for Microsoft Active Directory
- HP Operations Smart Plug-in for Microsoft Exchange Server
- HP Operations Smart Plug-in for Virtualization Infrastructure

If customers have invested a lot of effort in creating their own service model (and in a corresponding manual or automatic service model creation process), then they can reuse that model and create corresponding UCMDB configuration items automatically using custom topology synchronization mapping rules. However, UCMDB knowledge is required to build these mapping rules. Additionally UCMDB enrichment rules might be required to create the correct model in the UCMDB. (HP provides consulting services for creating custom mapping rules.)

Useful Questions when Evaluating Discovery Options

Do you use or plan to use HP DDM to populate your CMDB?

If yes: Use HP DDM to populate the CMDB for HP OMi.

If no: **Do you have a service model in OM?**

If no: We recommend using HP DDM to populate the CMDB for HP OMi. (It does not make a lot of sense to invest in home-grown discovery and creating a service model in HPOM as the possibilities to do this in the UCMDB/HP DDM are much better).

If yes: **Does the service model contain services discovered by *OMi-enabled* HPOM Smart Plug-Ins?**

If yes: Use the out of the box topology synchronization rules provided by the HP OMi content packs.

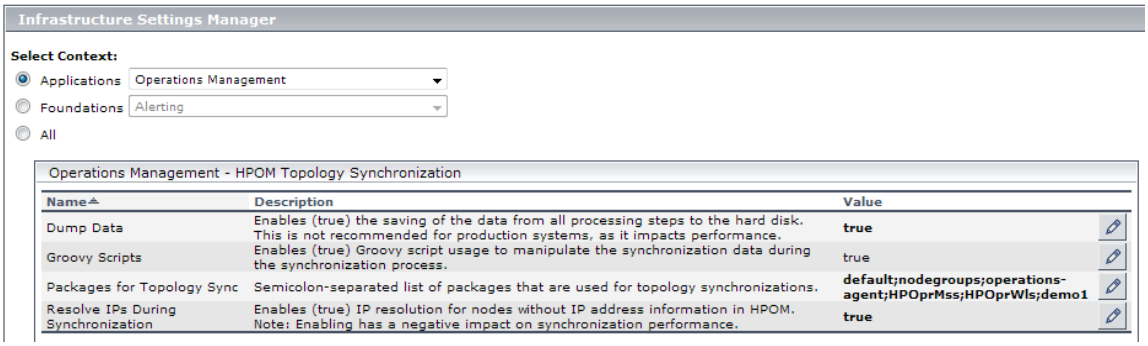
Does the service model contain other services that should be converted into CIs?

If yes: *Either* create your own topology synchronization rules to map these to CIs.

Or use HP DDM to discover those CIs

Or create them manually using the UCMDB Modeling Studio

Best practice: Before launching the topology synchronization tool, make sure that the *Resolve IPs During Synchronization* setting is enabled and that all node names can be resolved on the processing server. This will assure that host CIs are created with a correct IP address relationship, which is especially important when HP DDM or other sources like HP SiteScope also create host CIs. If the IP address relationships are missing, then you might end up having duplicate host CIs in the UCMDB.



Designing Enhancements of the Out-of-the-Box Content

HP OMi contains several content packs. These content packs provide out of the box value for the following key IT resources:

- System infrastructure resources such as virtual and physical hosts
- Microsoft Active Directory Services
- Microsoft Exchange Server
- Oracle and Microsoft SQL Server databases
- BEA WebLogic and IBM WebSphere application servers.

HP DDM provides discovery for additional IT resources such as:

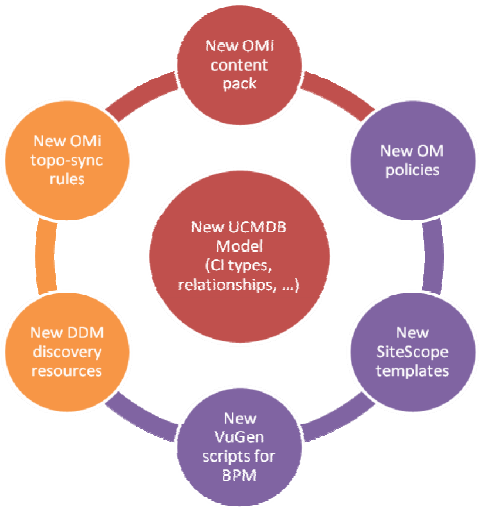
- Host resources
- Software elements
- Other J2EE application servers
- Siebel
- SAP

Additionally, the UCMDB and BAC provide pre-defined views for many of these applications.

If this content is not sufficient, then additional content can be developed to address specific customer needs.

UCMDB / Model Enhancements

Additional CI Types and relationships can be defined using the UCMDB CI Type manager. A wizard guides you through the necessary steps and asks for key attributes, icons, and so forth. After the new application has been modeled in the CI Type manager, you can design new views using the UCMDB View Manager.



See the *Model Management* manual for details.

Discovery Enhancements

Once the model and views are in place, corresponding CIs must be created in the UCMDB. Possible discovery options were already described in the previous chapter.

For details, see the *HP Discovery and Dependency Mapping* manual.

HP OMi Enhancements

HP OMi enhancements can range from small adjustments of existing indicator values to full-blown new content packs for applications not yet covered.

Typical content development steps to manage a new application with HP OMi are:

1. Start by defining the key health indicators that should be monitored for each CI type.
2. Create mapping rules or adjust the underlying monitoring to set indicators.
3. Define view mappings for the health perspective view.
4. Define HP OMi tools that help operators to manage the application. (Check if tools have been defined on the HPOM side already, and if so, these tools should be provided to HP OMi operators as well, if possible.)
5. Define application-specific graphs and assign them to CI types (you might have to extend the underlying monitoring first to collect necessary metrics).

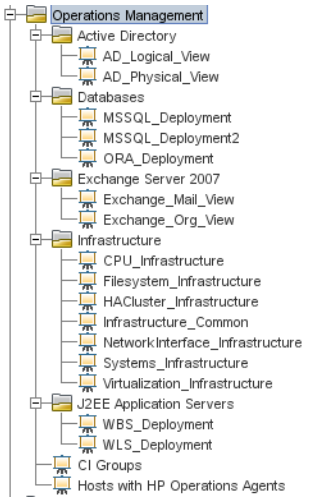
6. Monitor the incoming events for a while. Analyze the event stream and check if certain problems always create multiple events. Check whether it makes sense to create corresponding correlation rules.
7. Define an Operator user group with permissions to the tools, graphs and views required to monitor the application.
8. Make sure that events can be automatically assigned to this operator group during runtime (you might have to adjust existing events slightly by adding a suitable message group).

The "HP OMi Content" chapter of the HP OMi *Concepts Guide* provides a good overview of the content management workflow, the content of content packs, and the tools available to manage content packs. Further details can be found in the HP OMi online help.

Monitoring Enhancements

The customized HP BSM solution might require the development of new HPOM policies and instrumentation files, HP SiteScope monitor templates or VuGen scripts for BPM. The combined functionality should enable the monitoring of any customer application. For details, refer to the corresponding product documentation.

Combining HP OMi Data and other HP BSM Data



UCMDB views display a defined subset of the UCMDB's overall IT model. HP OMi ships pre-configured views that can be used in the CI tree view of the event and health perspective to focus on a specific area of interest monitored by HP OMi, such as:

- "Microsoft SQL Server deployment" (MSSQL_Deployment)
- "WebLogic deployment" (WLS_Deployment)

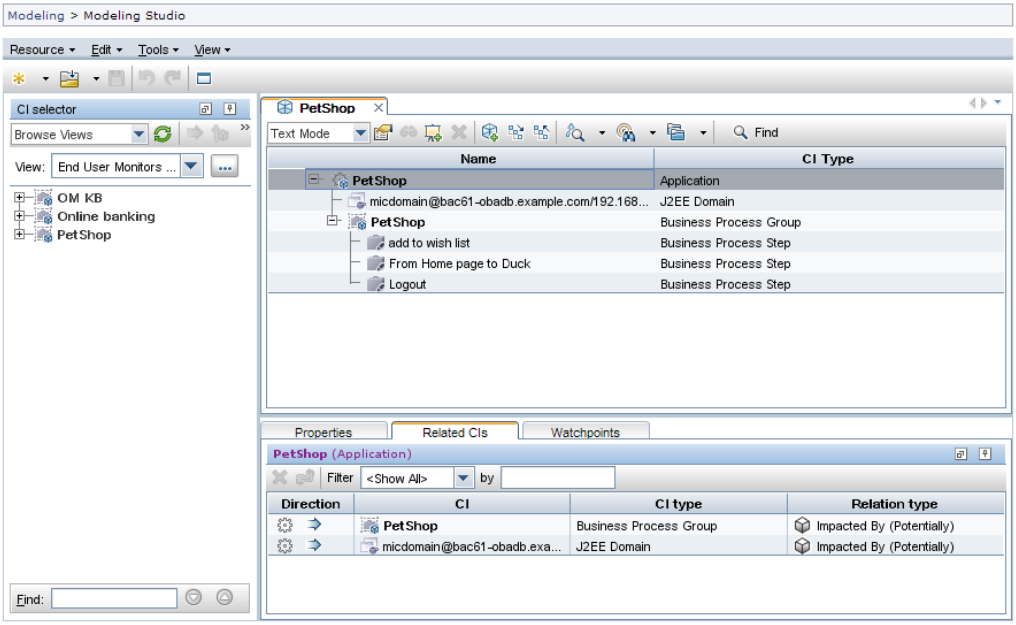
Furthermore, these views are used in the top view of HP OMi's health perspective to show the health of the CI related to an event and of its neighbors.
Additionally, views are used in topology-based event correlation rules as a starting point to define the necessary topology of a correlation rule.

However, out-of-the-box there are no views that show the relationship between a discovered application system, application resource or host (all these are CI types monitored by HP OMi) and a business service, business process or logical application they belong to. These relationships cannot be discovered – they need to be created manually.

Creating a Model that Links HP OMi Data with other HP BSM Data

An easy way to create such a model of business services, business processes, applications and underlying IT infrastructure is using the UCMDB Modeling Studio. The New Model wizard allows you to create a new (logical) application and relate other CIs using drag and drop.

Make sure that you also drag and drop the CIs to the *Related CIs* tab, where they are automatically related using a *depends* relationship. After you save and reopen the model, the Modeling Studio shows the *Impacted By (Potentially)* relation type. This is normal, and shows the calculated relationship that was added based on the *depends* relationship.

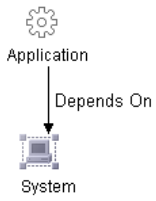


Best Practice: Although you have some flexibility in regarding what model you create, it is recommended that you create a (logical) application and then relate to it the IT infrastructure on which it depends. Business Process Groups can be linked to applications as well, either through the Modeling Studio or when setting up the Business Process Profile.

You can also create the relationships using the IT Universe Manager. Just make sure that you create *depends on* relationships *from* the application *to* the CIs it depends on.

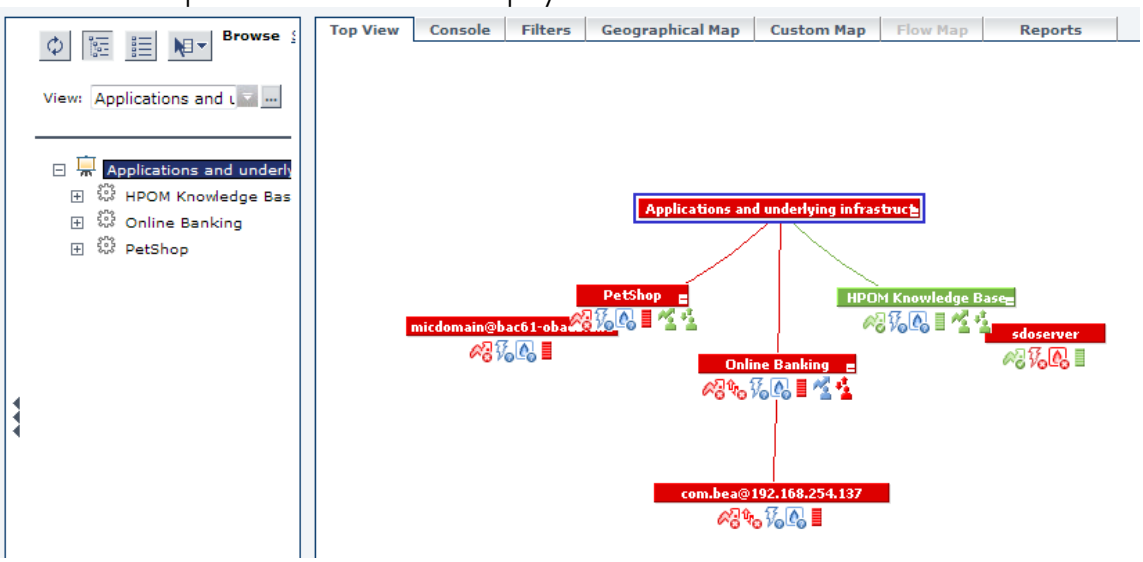
Building Views for the HP BSM Dashboard and HP OMI's Top View

The next step after you have created the model in the UCMDB is to create a view so that you can look at the corresponding CIs and their status in the HP BSM Dashboard as well as in the HP OMi CI tree browser and top view. The UCMDB Model Studio allows you to create a view from the model you just created.



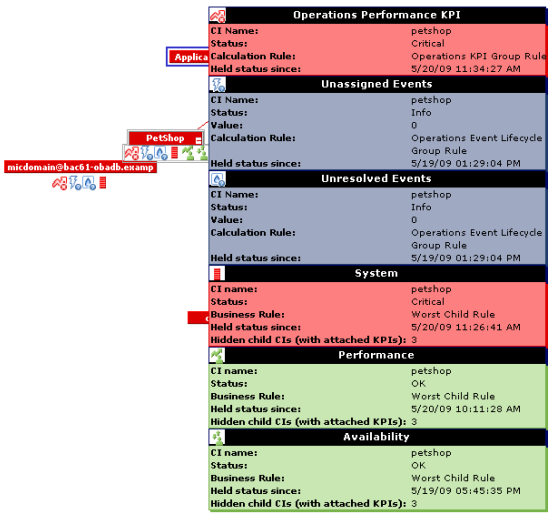
You can also create a pattern-based view using the UCMDB view manager. It should contain the application CIT and the relationships to all possible underlying infrastructure CITs. You see one possible view definition on the left.

Here is an example of what that view will display in the HP BSM Dashboard:



The individual KPIs and the status of the application CI on the top level depends on the underlying CIs (even if they are not shown in the view). If the application depends on a Business Process Group, then you see the Availability and Performance KPIs added by the Business Process monitors. If the application depends on IT infrastructure CIs monitored by HP OMi or HP SiteScope, then you will see Operations Availability and Performance KPIs (based on HPOM events) and the System KPIs (based on HP SiteScope monitors).

All KPIs together show the health from an end-user perspective as well as an operational perspective.



Note: You can fine-tune which KPIs are displayed and which KPIs should be used to set the CI status in the Dashboard Administration UI.

If required, you can additionally link the application CIs to business service CIs and create corresponding views that show business services, applications and the underlying infrastructure.

Building Views as a Basis for TBEC Correlation Rules

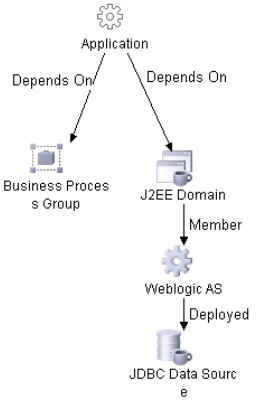
You may be wondering why this is a separate section, and why you cannot just use the views created for the HP BSM dashboard as basis for correlation rules.

There are two reasons for this:

- Firstly, TBEC correlation rules are defined on CI types and their relationships. The views created by the UCMDB Modeling Studio are instance views based on perspectives. Therefore these views cannot be used in TBEC correlation rules.

- Secondly, even if you define pattern views for the HP BSM Dashboard, you might define those views using abstract configuration item types (as above), as this simplifies the view and still allows you to look at various sub-types. For example, you can build a pattern view that shows hosts. As a result, you will see CIs of all sub-types like computers, routers, and so forth, and their sub-types like Windows, UNIX, and so forth.

However, in correlation rules you refer to Event Type Indicators of a certain CI type, and the deeper you go down the inheritance hierarchy, the more Event Type Indicators might be defined. For example, some Event Type Indicators (ETIs) are defined for *Computer*, some more on the sub-type *Windows*, but none on the more abstract type *host*. To be able to see the *Windows* ETIs in HP OMi's correlation manager, the corresponding sub-type must be contained in the view. If the view just contains the abstract CI type (like *host*), then you will not be able to select ETIs for a correlation rule.



Therefore, if you want to create correlation rules for infrastructure CI events and events for other "BSM" CIs like Business Process Group, then you first need to create detailed views that contain the specific CI types and the topology that links them together (you typically anyhow want to make the correlation rule very specific as it otherwise might incorrectly correlate events). An example is shown on the right. Once this view exists, you can use it to define the correlation rule. (An example correlation rule is shown in the section "Example: BPM integration" on page 25ff).

(After the correlation rule has been created, you can theoretically delete the underlying view.)

How to Configure Event Integrations to Obtain Maximum Benefits from HP OMi

Concepts

Mapping Events to Configuration Items

The basis for all advanced HP OMi features is that events are mapped to the correct CIs in the UCMDB. If that is not achieved, then setting health or event type indicators in HPOM messages will have no affect, correlation rules will never trigger and the health perspective view will show either incorrect health and KPI data, or none at all.

To obtain the maximum benefits of HP OMi it is therefore essential that:

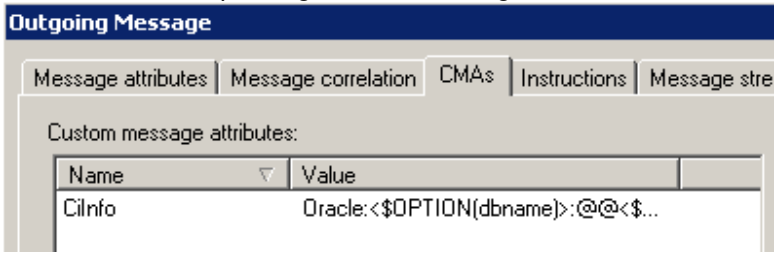
- CIs exist in the UCMDB.
- Adjustments are made to the event integrations (if necessary) in such a way that events can be mapped to these CIs.

HP OMi uses a smart mapping technology to map events to CIs. The system looks for hints in certain event attributes, compares these hints with CI attributes and then maps an event to the best matching CI. Most events contain at least the DNS name of the affected host (in the HPOM message node name field) and as HP OMi uses this as one possible hint, the smart mapper will almost always be able to map an incoming event to at least the host CI (assuming it exists in the CMDB).
But to make use of the complex IT topology model, database events should be mapped to corresponding database CIs, and events related to other applications to their respective CIs.

For this, HP OMi evaluates further event attributes, namely the *application*, *object*, and the HPOM *service ID* or the *CI Resolution hint* attributes (if the *CI Resolution hint* attribute is set, the HPOM *service ID* attribute is ignored). The more identifying hints are provided in these attributes the higher the likelihood that the event will be mapped to the correct CI.

Hints must be specified using a certain format to allow the smart mapper to identify what belongs to one attribute. The default format is <hint 1>:<hint 2>:...:<hint n> for the Application and Object attribute and <hint 1>:<hint 2>:...:<hint n>@@<hostname> for the HPOM service ID and CI Resolution hint attributes. (The separator : can be configured in the Infrastructure settings.)
All the hints are then evaluated to find a matching CI in the UCMDB.

The hints in the application, object and HPOM service ID attributes are evaluated because of backward compatibility reasons. In the past, many HPOM SPIs already used these fields to transport information about what object (in UCMDB terms, the configuration item) the event is related to. If this information is already sufficient to identify a single CI - and the correct CI – then there is no need to change anything. However, if you find out that an event is related to an incorrect CI, then you should set the CI Resolution hint attribute and provide the necessary hints in this attribute.
This can be done by setting a custom message attribute (CMA) called CiInfo in the HPOM message:



The CMA CiInfo is mapped to the CI Resolution hint event attribute (and does not appear in the custom attributes list of the event).

(Note: some SPIs set a custom message attribute called OPR_CI_INFO. This also sets the CI Resolution hint attribute, but you should no longer use the OPR_CI_INFO custom message attribute. Use CiInfo instead.)

How Integrated Products (Including Third Party Products) can Set HP OMi Health

After mapping an event to a CI, it is straightforward to set a health indicator value with it, if the event happens to be a good indicator for the health of the CI.

Note that not all events have to set health indicators. Events that do not set health are still important events that affect the unassigned and unresolved event KPIs and which operators need to analyze.

Defining Health Indicators should be done using a top-down approach, where you consider what the five to ten most important attributes are of an application or service (or of any CI type) that indicate the current health of the CI. These attributes should be modeled as health indicators with specific values that provide meaningful information about the health of the CI to operators.

Examples of health indicators are **Servlet Performance** or **JVM Memory Utilization** for **J2EE Servers**, or **Replication Status** and **SQL Query Performance** for **Databases**.

The value of a health indicator should be self-explanatory and should always be meaningful to an operator, like **up**, **down** or **near capacity**, and so forth.

Setting a Health Indicator Value

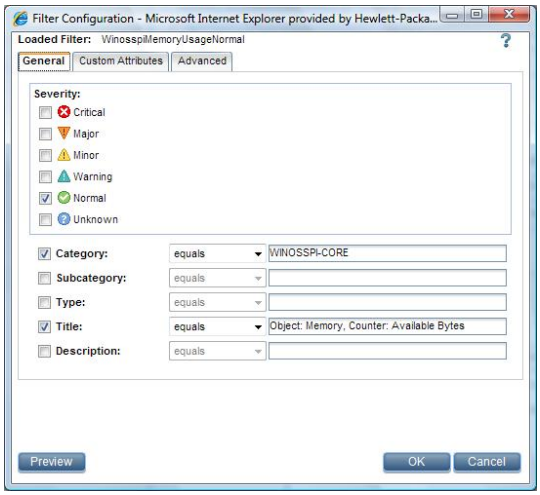
After you have defined a health indicator and its values in the HP OMi Indicator Manager, you have two options for setting health indicator values:

- Add the health indicator value to the event itself.
- Set the health indicator value on the HP OMi server using a mapping rule.

Mapping Rules

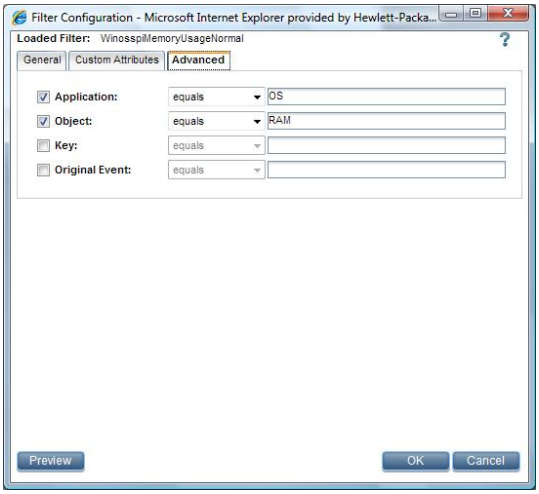
You can use the information in HP OMi events to set a health indicator to a particular value. For example, you can use the attributes of a critical event reporting a lack of storage space on a logical volume to set a health indicator value **Near Capacity** with the severity **Critical** assigned to the configuration item type **Logical Disk**.

Mapping rules allow you to set a health indicator based on detected event attributes.



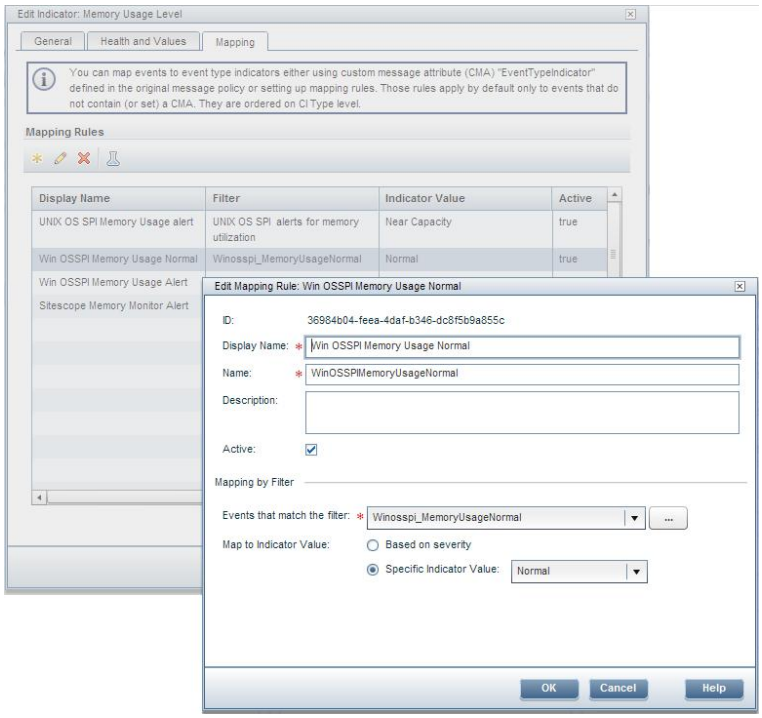
This example shows a filter used in a mapping rule of the infrastructure content pack to map an event coming from an Operations Smart Plug-In for Windows to the **Memory Usage Level** health indicator.

The filter looks for a special **Title**, **Category** and **Severity**, as well as for certain **Application** and **Object** values.



The mapping rule then defines that, if the filter matches, the value of the health indicator is set to a specific value, in this case **Normal**.

It is also possible to set health indicator values based on the event severity, which can be quite handy as it allows you to set different health indicator values with just one rule.

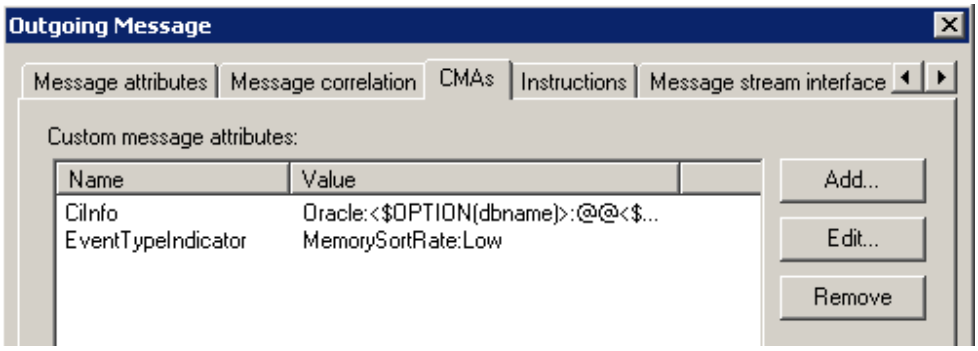


Custom Message Attributes

Health indicator values can also be set using the Custom Message Attribute *EventTypeIndicator*. A Health Indicator is a special event type indicator that includes health information.

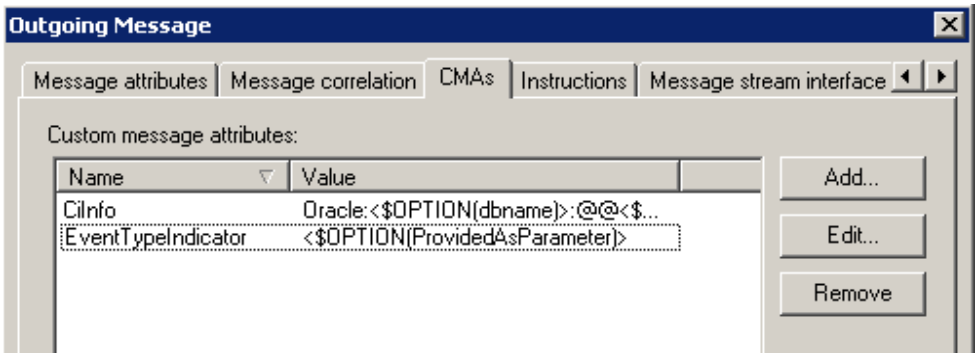
The CMA can be easily added to a message in any HPOM policy.

This message sets the value of EventTypeIndicator to **MemorySortRate:Low**

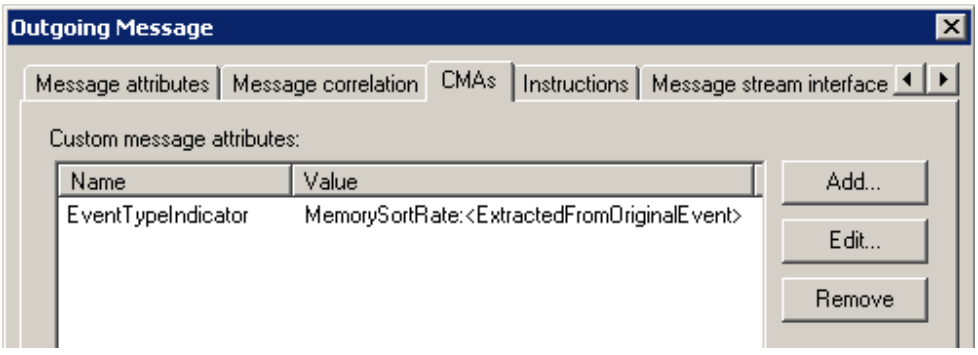


(CMAs can also be added easily to policies on an HPOM for UNIX server.)

It is also possible to set the CMA using data contained in an -option parameter (for details, see the HPOM opcmsg and opcmon online help):



If the ETI name and value (or part of it) is already included in the original event (for example, the SNMP trap or WMI event), then HPOM can extract that data using pattern matching and store it in a variable which can then be referred to in the CMA.



When to Use Mapping Rules and CMAs

It is appropriate to use CMAs if you are developing HPOM policies from scratch, and you can set the ETI value in the source event. If this is the case, we recommend that you set the ETI value inside the HPOM policy. When developing policies, which will eventually create events, it is important to consider the semantics of the events you are creating. Can the events be classified somehow? Are the events always associated with a certain meaning? Then that meaning (the semantic or classification) of the event can be expressed through the Event Type Indicator. For example, if you monitor certain performance metrics with a measurement threshold policy, then you can categorize the resulting events as **EJB Performance** events (which separates them from other metric-related **CPU Usage level** or **Memory usage level** events). Other events created by log file policies might be categorized as **Unexpected reboot** events.

If the event also stands for a certain state of the monitored object, then that can be shown via the Event Type Indicator value, like **EJB Performance:low** (which maps to an equivalent health indicator value).

It is appropriate to set the ETI/HI value using mapping rules on the HP OMi server if:

- HPOM policies should not be changed (because of the maintenance effort).
- Events are received by HPOM by other means (for example, using the incident web service interface).
- It is not possible for you to set the ETI value in the source event (because you have no control over the source of the events).

Example: BPM integration

In this section, we take a closer look at one of the existing integrations and how it can be configured to obtain the maximum benefits from HP OMi. The BPM alert integration is a good example of an integration where it is not possible for a customer to set the ETI value in the source event. Therefore the ETI value has to be set using an HP OMi mapping rule.

Linking BPM Data and HP OMi Data

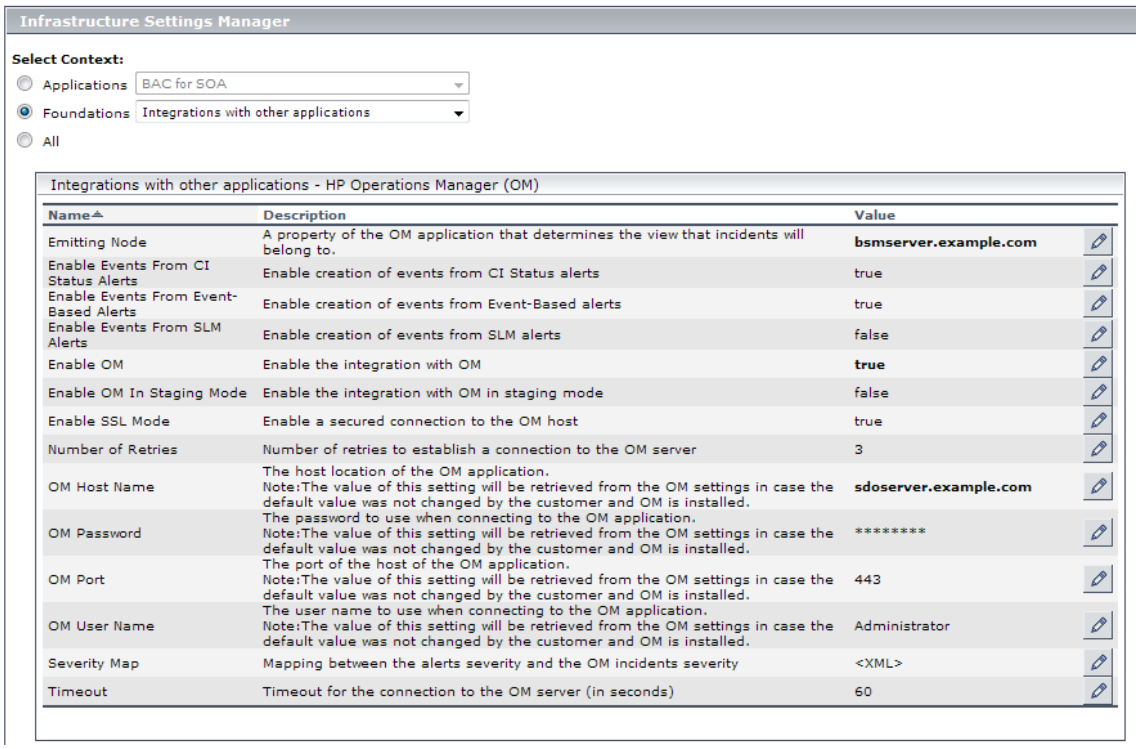
The first goal of the integration is to bring together BPM Key Performance Indicator (KPI) data and HP OMi KPI data, so that operators have an overview of the EUM status for a CI as well as the operations management status. To achieve that, you first have to model where both parts come together, as outlined in "Creating a Model that Links HP OMi Data with other HP BSM Data" on page 17.

The *Link to CI* function in the BPM profile wizard allows you to link the business profile group to an application CI. However, you first have to create the application CI manually, or using the Modeling Studio.
In the Modeling Studio, you can also link the infrastructure CIs (like a WebLogic domain or SQL Server) to this (logical) application. Alternatively, use the IT Universe Manager to create the corresponding *depends on* relationships.

Alarm HP OMi Operators when BPM Detects a Problem

A second goal of the integration is typically to alarm HP OMi operators when the business process monitor detects a problem, so that the operator can be assigned to work on and solve the problem.

To achieve that, you first have to configure alerts to be sent to HPOM (from where they will be forwarded to HP OMi) and then setup alerts in the HP BSM platform.
Go to **Infrastructure settings** -> **Foundations** -> **Integrations with other applications** -> **HP Operations Manager (OM)**.



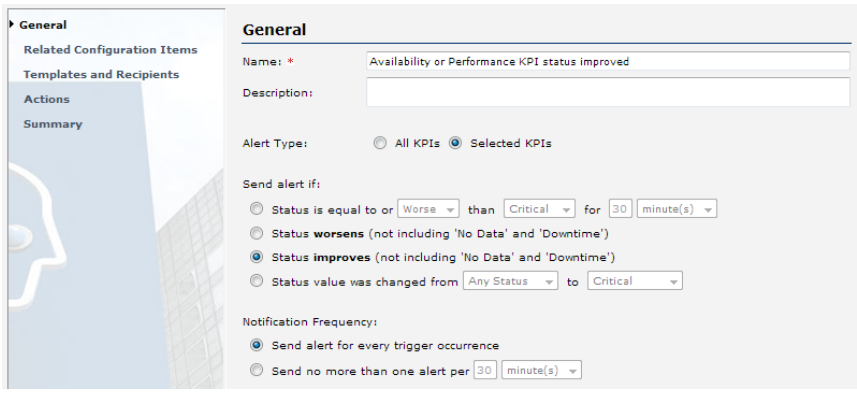
Configure the HPOM host name, user name and password and then set the value for *Enable OM* to **true**. As soon as this is done, all alerts (existing alerts that currently page someone or send an e-mail, as well as alerts that will be created in the future) will also create HPOM messages. You can also setup alerts that will only send HPOM messages (do not specify any alert action in the alert wizard). These HPOM messages are created on the HPOM system using the HPOM incident web service. The message text and all attributes are set using the information in the alert, but cannot be edited by the customer (except if the server message stream interface is used). From the HPOM server, the HPOM messages are then forwarded to HP OMi and available in the HP OMi event browser.

Now the question is, which alerts have to be setup? You have two options for this:

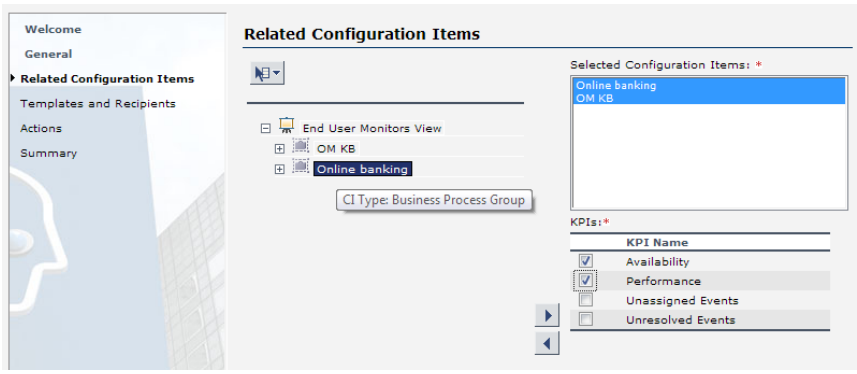
- Create event-based alerts (for example, when the transaction response time is greater than a certain threshold or when a transaction failed).
- Create CI status alerts (triggered when a KPI status changes).

At first glance, event-based alerts look like a good choice. They can report the status of individual transactions and you could even define that the alert is triggered before the affected KPI changes to critical. However, there is one limitation of event-based alerts: the trigger condition only allows you to use "greater than" or "less than". This means you can only report two states, like "response time is less than 2 secs" and "response time is greater than 5 secs".
It turns out that focusing on the KPIs set by the BPM monitor is a better choice.

It is quite easy to setup two alerts that look for CI/KPI status changes. These alerts then inform HP OMi operators whenever a KPI, calculated using BPM monitor data, changes its status.
When setting up new alerts, choose the "End User Monitors view". Then create a new alert as shown in the screenshots.



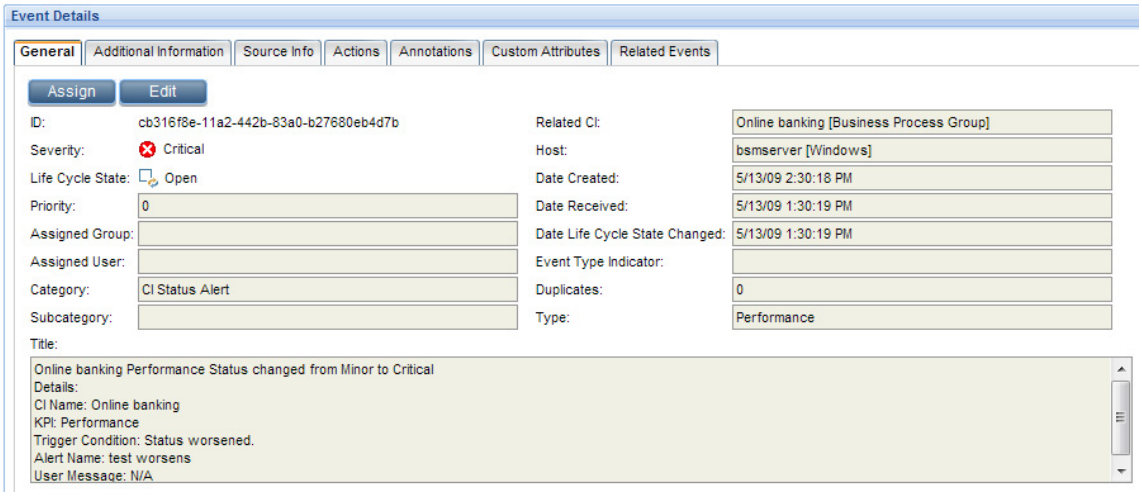
On the Related Configuration Items page, select all Business Process Groups for which you want alerts. Select the Availability and Performance KPIs:



Do not specify anything on the remaining pages.
Note that the summary tells you that this alert opens an event in HP Operations Manager:

Define a similar alert by selecting **send alert if status worsens**.

Here is an example of a corresponding event in HP OMi:



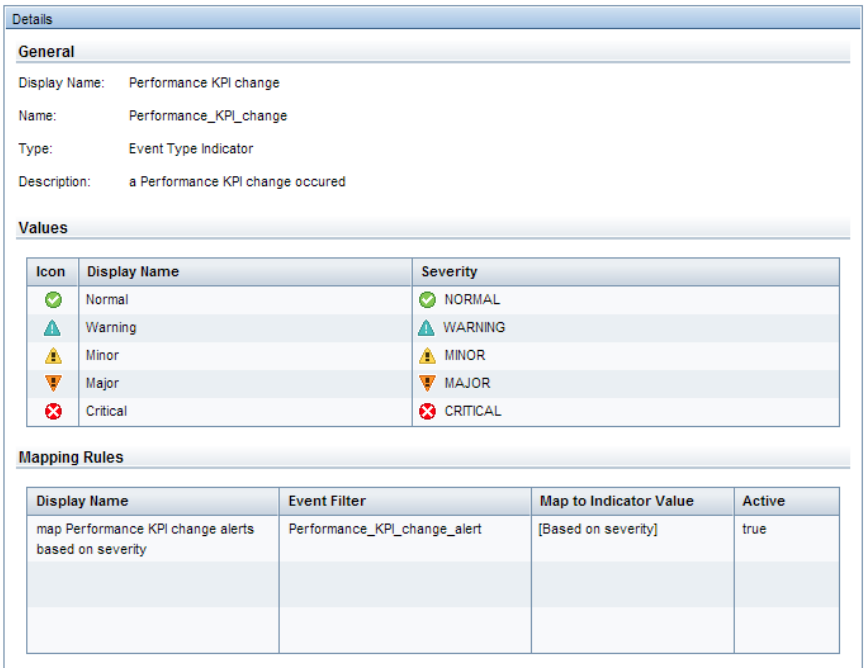
Correlate BPM Events with Infrastructure Events

A third goal of the integration should then be to correlate BPM events with infrastructure events (or at least to allow correlation). For this, you have to define and set ETIs for the business process group CI type (all events created by the alerts above will be related to business process group CIs).

This can be easily done using the Indicator Manager of HP OMi.
Define two ETIs: one for Availability KPI changes, one for Performance KPI changes. For each, define five values from Normal to Critical (as the KPIs can have these values).

To set the event indicator, a mapping rule and filter has to be defined that matches exactly the KPI change events that you created earlier.
You need two filters: one with the condition *Title contains "Availability status changed"*, the other with *Title contains "Performance status changed"*.
In the mapping rule you can use the "map based on severity" option.

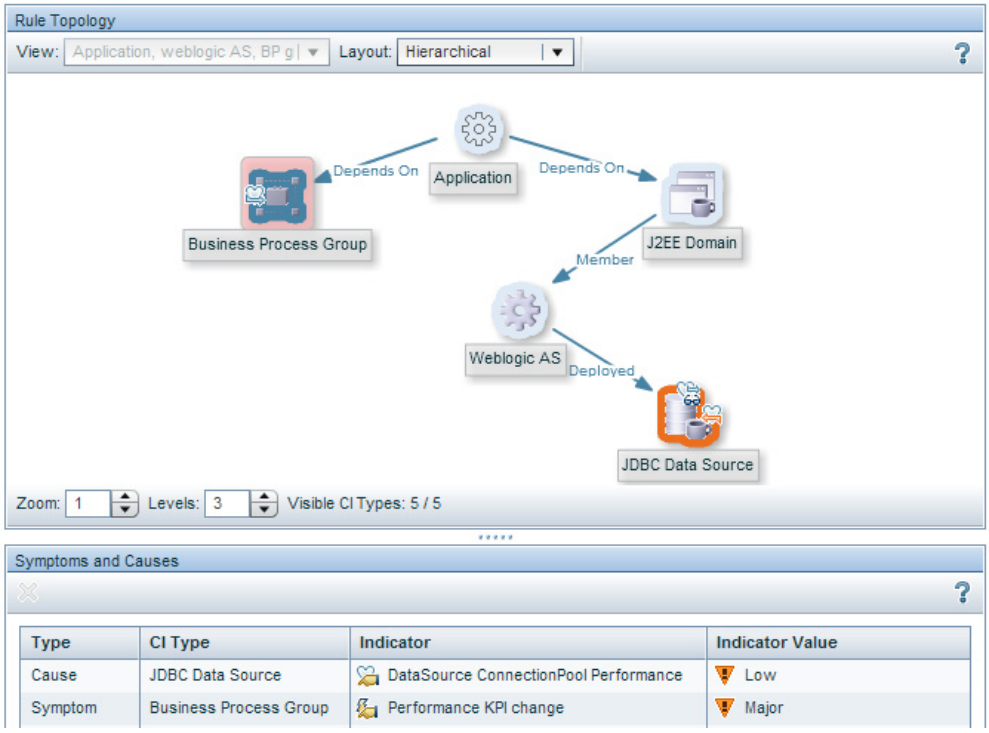
Here are the details of such a Performance KPI change ETI:



It is not recommended to define these ETIs as health indicators, as the alerts currently do not make use of the good/bad message correlation feature of HPOM which makes it difficult to reflect the true health. Furthermore the status based on the BPM metrics is already reflected in the Availability and Performance KPIs.

Now that ETIs have been defined and set via mapping rules, you can create new correlation rules that make use of these ETIs and link BPM events with events from other sources.

Here is an example:



This correlation rule defines that if an event about a Data Source Connection Pool Performance problem of a J2EE server arrives and at the same time BPM reports a Performance problem from an end-user perspective, then the Data Source event will be marked as the cause and the BPM event as a symptom. Note that this correlation will only take place if the Business Process Group is linked to the same application that the J2EE domain is linked to.

Example: HP SiteScope Integration

The HP SiteScope integration is another good example for showing the possibilities that exist to integrate events, to link them to the right CIs and to set ETIs.

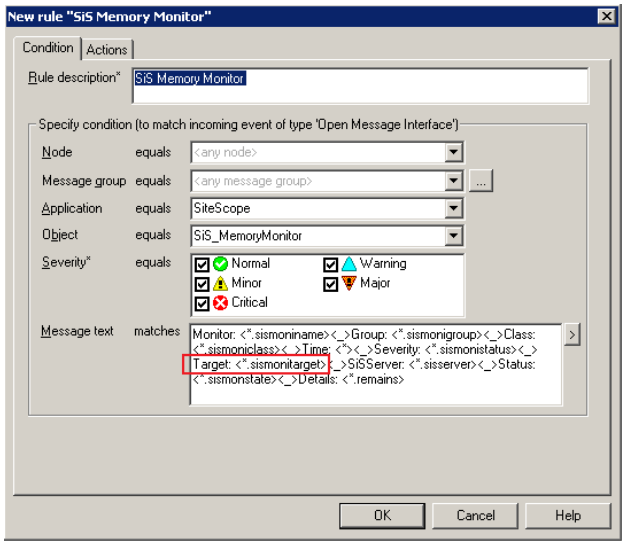
The integration of HP SiteScope into an HP OMi/BSM solution consists of two parts:

- the HP SiteScope adapter for HPOM
- the HP OMi SiteScope content pack

HP SiteScope Adapter

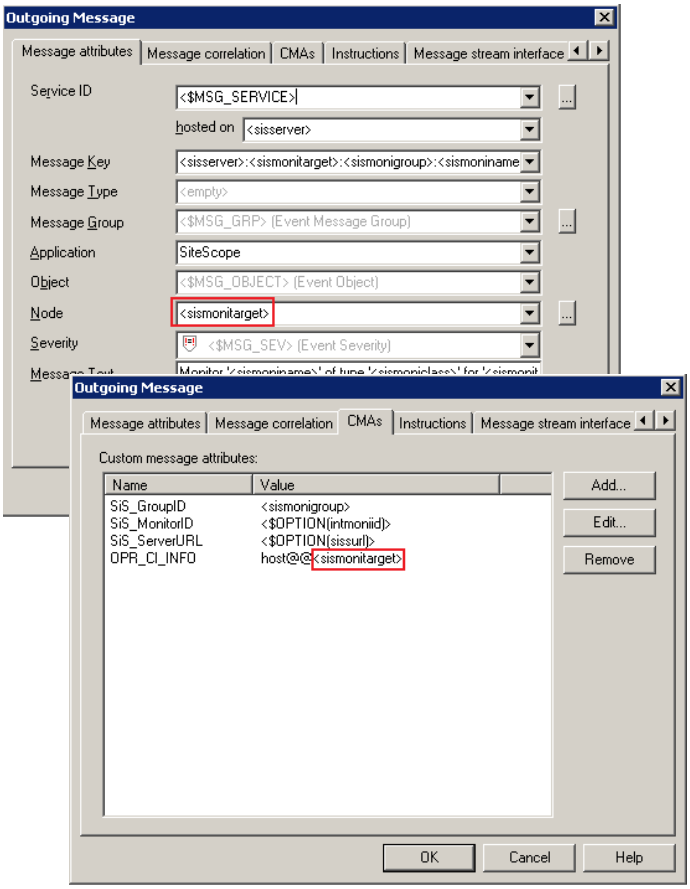
The HP SiteScope Adapter for HPOM allows you to create HPOM messages based on HP SiteScope alerts.

The HP SiteScope Adapter contains predefined HPOM policies for each HP SiteScope monitor class. The policies try to determine the target of an HP SiteScope monitor (like the monitored database or web server system) and set the node name in the message accordingly. Furthermore, where possible, the policies also set the CI resolution hint, which enables HP OMi to relate the event to the correct CI.



The SiS Memory Monitor policy is a good example. Here you can see the policy condition. It extracts the target name (the string behind *Target:*) from the alert and stores it into the variable *sismonitarget*.

In the message, this variable is used to set the node name and the OPR_CI_INFO CMA.



As one policy is provided for each HP SiteScope monitor class, it is very easy to customize the events further. Currently, many events are related to the host CI type. However, if you have a finer, more granular model in the UCMDB and want to relate, for example, FTP events to an FTP service CI, then you just need to edit the corresponding SiS FTP monitor policy and provide the necessary hints in the OPR_CI_INFO CMA.

It is therefore a **best practice** to use the policies in the per monitor class policy group.

The HP SiteScope Adapter also contains a generic policy called *All SiteScope monitor alerts* in the *Advanced mapping for HP OMi* policy group. However, it can only be used if SiteScope is integrated in BAC, it requires that the monitored target CIs are either created by HP SiteScope or created manually (otherwise the events will not be related to the correct CIs), and as it is just a single policy, it does not offer the flexibility that the policies per monitor class provide.

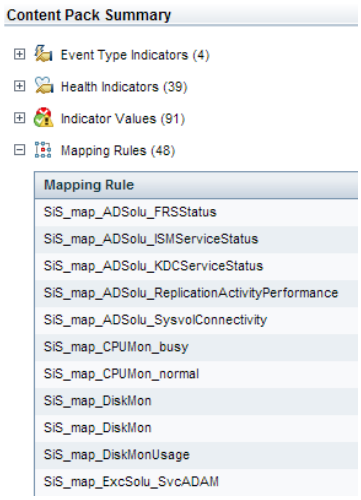
For details about the HP SiteScope Adapter installation, configuration, and how to set up necessary HP SiteScope alerts, see the *HP SiteScope Adapter for HP Operations Manager for UNIX User's Guide* and the *HP Operations Manager for Windows Online Help*.

HP OMi SiteScope Content Pack

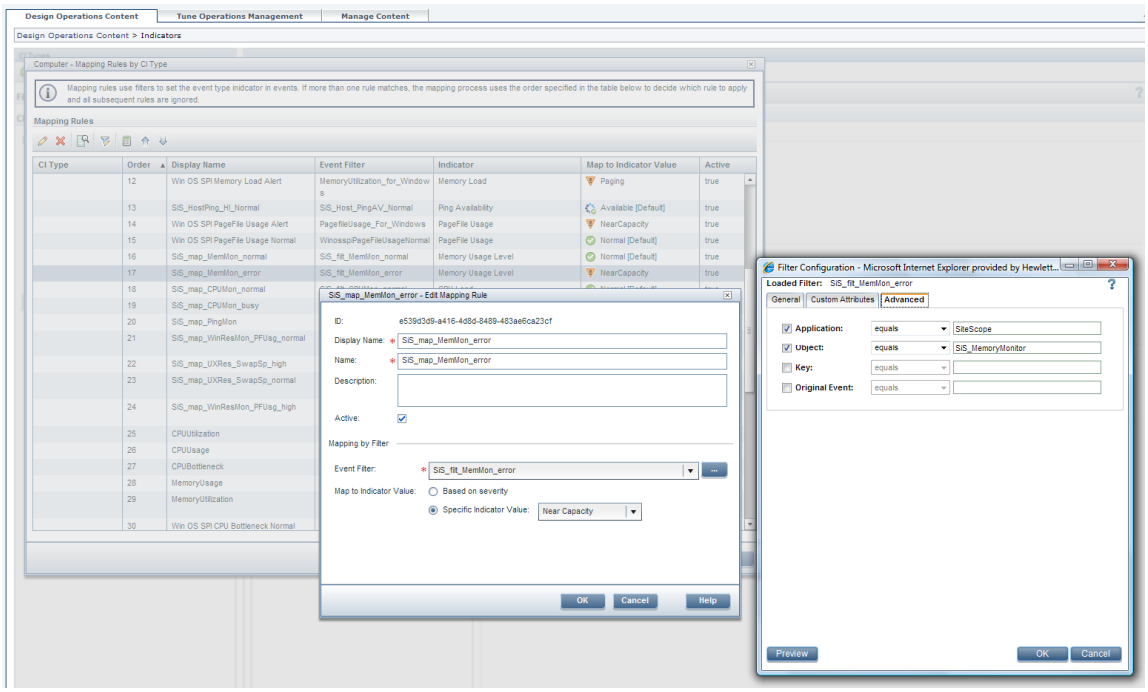
You might have wondered why the HP SiteScope policy shown above does not also set the EventTypeIndicator CMA. This is because in this case ETIs are set using mapping rules. These mapping rules are contained in the HP OMi SiteScope content pack (HPOprSiS). The content pack summary shows 48 such mapping rules.

Furthermore, the content pack provides two tools to start the HP SiteScope UI in context.

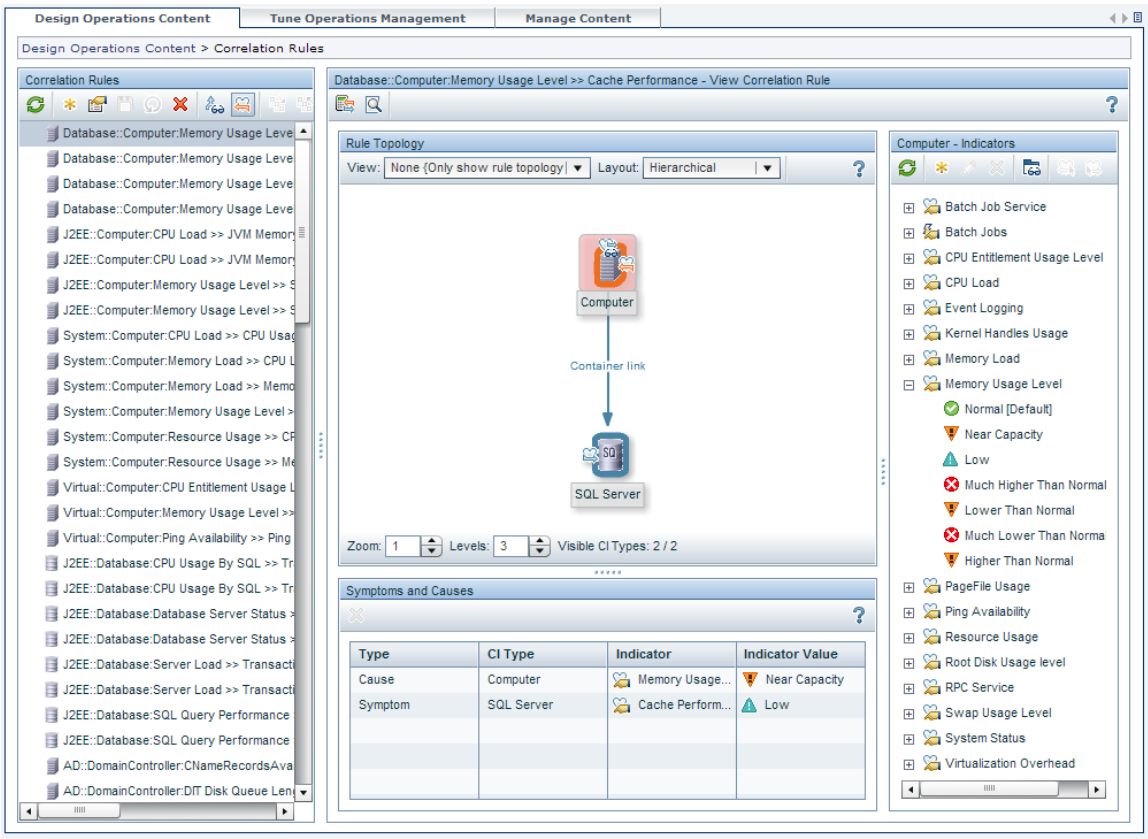
(The Indicators and Indicator values contained in HP OMi SiteScope Content Pack are not really new to that content pack. These Indicators are already defined in other content packs like the Exchange or Infrastructure content pack, but they are referenced in the mapping rules, and therefore part of the HP OMi SiteScope Content Pack as well.)



Most of the mapping rules apply for the Computer CI type or a sub-type like Windows and they are accompanied by corresponding event filters.
Here you see some of the mapping rules in the background, one mapping rule and one tab of the corresponding event filter as an example:



As the HP SiteScope mapping rules can set the same health indicators as SPI events, and as topology-based event correlation rules are based on ETIs, all correlation rules are automatically applied on HP SiteScope events as well. Here you see an example showing a correlation rule. This rule correlates a MemoryUsageLevel:NearCapacity event (from our example above) with a SQL Server Cache Performance event (sent by HP SiteScope or an HPOM SPI).

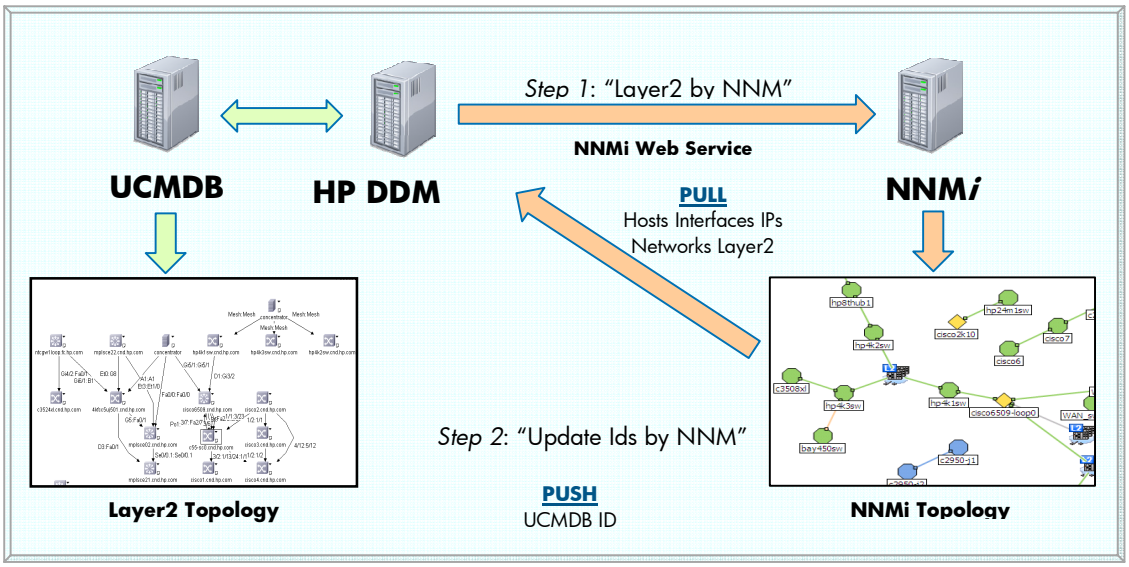


Example: NNMi Integration

HP Network Node Manager i-series (NNMi) features two standard integrations that can easily be leveraged by HP OMi. Through its bi-directional interfaces with both UCMDB and HPOM, NNMi provides us with discovery data (CIs) for network devices, as well as with events describing the network-related status of the monitored network elements.

NNMi–UCMDB Integration

Starting with version 8.11, NNMi features a direct integration with the UCMDB to instantiate network element CIs in the IT Universe Model. Upon activating this integration, CIs of multiple CI Types will be created in UCMDB, including "Networks", "Nodes", "Network Interfaces", and "IP" addresses. A simple "connectivity model" will also be implemented in the UCMDB by instantiating corresponding relationships ("Layer2" connections) between network interfaces that NNMi sees as interconnected.



The above diagram shows how, in the first step of this integration, a HP DDM Probe starts a discovery job ("Layer2 by NNM") to extract network elements and connectivity data from NNMi and creates corresponding CI instances in the UCMDB. Once this has been accomplished, a second discovery job ("Update Ids by NNM") will create a custom attribute named "UCMDB_ID" in NNMi for each node "exported" to UCMDB. The custom attribute stores the node's corresponding UCMDB UUID, so that NNMi and other programs can easily reference the exact CI in UCMDB.

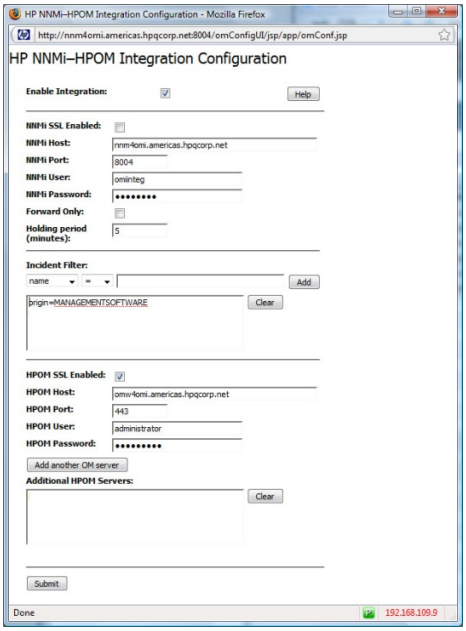
NOTE: While the integration module updates node entries in NNMi with the node's UCMDB ID, the integration does not perform the equivalent for other CI Types that are instantiated from NNMi data. In other words, even though "Network", "Network Interface" and "IP" CI instances will be created in the UCMDB, the corresponding NNMi objects will not be updated to contain their UCMDB ID as a custom attribute.

(For additional information on the NNMi–UCMDB integration, how to install and configure it, and how the UCMDB can utilize the instantiated CI instances for node impact analysis, see the *HP Universal CMDB – HP Network Node Manager i (NNMi) Integration Guide* available at http://support.openview.hp.com/selfsolve/manuals. Select "Universal CMDB (Application Mapping)", and version 8.01 (or later) as your search criteria.)

HP NNMi–HPOM/HP OMi Integration

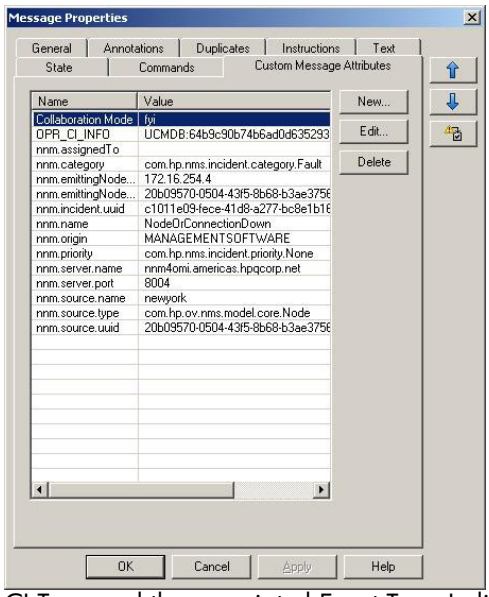
NNMi provides an out-of-the-box integration with HP Operations Manager (HPOM) for Windows or HPOM for Unix (OMU). The integration uses the newly added Incident Web Services (IWS) capabilities of HPOM. IWS is installed with OMW 8.10 (or later) and OMU 9.0. The integration module provides for multiple capabilities: incidents generated by NNMi are sent to one or multiple HPOM servers, and the incident lifecycle is synchronized between NNMi and HPOM. HPOM console users are able to launch incident-specific views into the NNMi web user interface for further information or diagnostic action. A single HPOM server can accept incidents from multiple NNMi installations, consolidating network operational data in an enterprise console. When launching an HP OMi tool from an incident that originated in NNMi, users are automatically connected to the correct NNMi instance.

In order to provide the full integration functionality described in this section, the minimum requirement for NNMi is version 8.12 (or, optionally, 8.11, with patch 3 and hotfix QCCR38986 installed). Earlier versions of NNMi did not add the OPR_CI_INFO custom message attribute (CMA) to the event sent to OM. This CMA is used by HP OMi to resolve the event to a UCMDB CI, and link any optionally sent Event Type Indicators to the network element CI so that HP OMi correlation rules can be triggered.



While the integration is installed automatically, it does require some configuration, both in NNMi and HPOM. In NNMi, the application administrator uses the "HP HPOM …" configuration dialog available in the "Integration Module Configuration" workspace to define the receiving HPOM server's name and port number, as well as the credentials required to use the HPOM incident web services. The administrator configures similar attributes for the NNMi server, as well as filter definitions to specify which types of incidents will get forwarded. An often-overlooked configuration setting is the "Holding period (minutes)" field. This setting specifies the time frame that will need to pass before NNMi invokes the HPOM incident web services to create an event in HPOM. The delay does allow for NNMi to perform additional correlations on the incident, and is meant to assure that only root cause incidents are being forwarded to HPOM. Shortening or eliminating this holding period causes incidents to appear in HPOM sooner, but may of course lead to a larger number of non-root cause events being created.

For HPOM to receive NNMi events and display them in the message browser, in most installations additional node definitions need to be created. Typically, an HPOM administrator creates definitions for "Nodes for External Events" (HPOM for UNIX) or "External Nodes" (HPOM for Windows). The node definitions can be configured using pattern-matching wildcards, e.g. "<*>.example.com", to accept events for all possible hostnames in the example.com domain. While OMW will then be able to receive events from (or about) these nodes, an additional configuration step is necessary in OMU: a node (including a "node for external events") must be a member of a Node Group before events are accepted from this node.



Once these prerequisites are met and the integration is enabled in NNMi, incidents created in NNMi are automatically sent to OM. Beyond a descriptive event text, each event contains a long list of CMAs that NNMi inserts to provide much additional detail about the incident, including its category ("nnm.category", i.e. "Fault"), event name ("nnm.name", i.e. "NodeDown"), the sending NNMi server's name ("nnm.server.name") and port number ("nnm.server.port"). For events pertaining to network nodes, their interfaces or IP addresses, as well as a CMA ("OPR_CI_INFO") uniquely identifying the node through its UCMDB ID are also included. This is the very same ID that was created by the NNMi–UCMDB integration, after network elements discovered by NNMi were instantiated in the UCMDB. By providing direct access to the CI ID of the causing (or affected) node, HP OMi can easily resolve an event to its CI Type and the associated Event Type Indicators, and ultimately be able to perform topology-based event correlation. By referencing the auxiliary data provided in the events' CMAs, we can define tools to connect back into NNMi from OM's event context.

Incidents forwarded from NNMi into HPOM remain closely linked during their lifecycle. Incident state changes (incidents being closed in NNMi, or HPOM users acknowledging an event) will be synchronized between the products, assuring that no "stale data" is presented in either product's console. However, it is important to note that NNMi will simply close/acknowledge an incident once it determines that a fault condition has ended. NNMi does so by correlating the new object status with an existing incident, consequently leading to the incident's closure, in turn causing the HPOM event to be acknowledged automatically. It is of special note that in these situations, NNMi does not forward a separate incident to HPOM describing the end of a failure or impact condition (that is, a "Node Up" event). The fact that no unique reset event is generated is important to consider when defining Indicators in HP OMi. Indicators are "transported" by events, and the consequence of not having "reset events" as a transport mechanism for updated status and health indicators is significant: based on the NNMi integration behavior, we are forced to forego the use of Health Indicators in HP OMi. (To recap, Health Indicators are Event Type Indicators that can take on multiple values, often times mapping to good, impacted and non-operational states.) As only "cause events" (describing a problem) are at our disposal, we must limit ourselves to mapping these events to basic Event Type Indicators. Contrary to Health Indicators, basic ETIs can assume only a single value. Typically, these are used to signal that a certain situation or problem has occurred. In the NNMi integration context, such an occasion could be "Node Down", or "Interface Not Responding".

Based on this bi-directional integration between NNMi and OM, all events sent by NNMi will also automatically be forwarded to HP OMi. HP OMi continues to maintain the closely synchronized event lifecycle, so that, for example, the closure of an event in HP OMi will cause the same event to be acknowledged in HPOM, resulting in the original incident being closed in NNMi (and vice-versa).

For additional details on installing and configuring the NNMi–HPOM integration, refer to the "Integrations and Plug-Ins" chapter in the *HP NNMi Deployment and Migration Guide*, available at http://support.openview.hp.com/selfsolve/manuals.

Mapping Events to CIs

When a network event originating in NNMi is received by HP OMi, it is automatically resolved to a related UCMDB CI. This is facilitated by NNMi including the "OPR_CI_INFO" custom message attribute in all events sent to HPOM. As a result, HP OMi is able to associate the new event with the correct network element CI.

However, the current version of both NNMi's integration with the UCMDB and HPOM has one important limitation: NNMi will only track the UCMDB IDs for network node objects. While NNMi maintains (and forwards to HPOM) the UCMDB ID of managed objects that are network nodes (i.e. routers, switches, servers, load balancers, firewalls, etc.), it is not aware of the UCMDB IDs of other object types, even though it instantiates these non-node objects in UCMDB. Incidents forwarded to HPOM describing problems with, for example, network interfaces or IP addresses, cannot therefore contain an OPR_CI_INFO value identifying the correct UCMDB object. Instead, NNMi includes the UCMDB ID of the object's host node. Consequently, such events will resolve to a different CI type ("Host", or one of its subtypes instead of "Network Interface" or "IP" address). The different CI types may have different Event Type Indicators defined, so care must be taken to consider any possible "mis-mapping" of ETIs.

If CI resolution to the correct CI type is important for your use case, it is possible to remove the OPR_CI_INFO data from the event while it is being received and processed by HPOM. If HPOM for UNIX is being used, HP Correlation Composer can be employed to easily modify the event. HPOM for Windows currently does not provide access to Composer functionality, but does maintain APIs to allow access to the event's data programmatically.

Removing Wrong OPR_CI_INFO Values from Events (HPOM for UNIX Only)

The Incident Web Services (IWS) component in HPOM for UNIX injects events into the HPOM management server's Message Stream Interface (MSI). HPOM for UNIX also provides a mechanism to configure an MSI client's priority relative to other clients. Therefore, HPOM for UNIX can be configured to process IWS events injected into the MSI prior to all other MSI clients. It is then possible to use another MSI client, for example HP Correlation Composer, to manipulate the injected events while they are progressing through the MSI.

To define the relative priority of a process in the Message Stream Interface, edit the file /etc/opt/OV/share/conf/OpC/mgmtsv/msiconf on the HPOM for UNIX server. This configuration file consists of entries in the form "`process priority`". `Process` is the name of the process instance registration with the MSI; `priority` is a numeric value between -127 and 127, specifying the process priority relative to other processes. Lesser values have higher priority, for example, a process with priority -20 is served prior to a process with the value 0. To achieve our desired configuration of injecting events that can then be modified by HP Correlation Composer, modify the msiconf file to contain these (or similar) entries:

```
j38965328  -1
opcecm     10
```

Now, the IWS process instance (always named "j38965328" in OMU) will get served by the MSI prior to the Event Correlation Manager process ("opcecm") that runs the HP Correlation Composer logic.

To prepare HP Correlation Composer to recognize and let us reference the Custom Message Attributes included by NNMi, edit the file "/opt/OV/bin/CO.conf" and insert the lines "OPR_CI_INFO" and "EventTypeIndicator" at the location indicated:

```
…
TIME_ZONE_DIFF
TROUBLETICKET
TROUBLETICKET_ACK
UNMATCHED
OPR_CI_INFO
EventTypeIndicator
#CMIP
…
```

Also ensure that the HP Correlation Composer template has been assigned and distributed to your HPOM for UNIX management server, taking extra care it is indeed assigned to the server and not the agent running on the server. Verify that the Server MSI is enabled. Once everything is configured

correctly, it is necessary to restart the HPOM for UNIX server processes. From now on, HP Correlation Composer will be able to process events created in the IWS module, and let us manipulate those events by means of a HP Correlation Composer correlator.

In order to define a HP Correlation Composer correlator to remove the misleading OPR_CI_INFO value from an NNMi event, start HP Correlation Composer by running **`ovocomposer -ui`**. Open a new or existing correlator store, and then create a new correlation of type "Enhance". Name the correlator **`Remove_OPR_CI_INFO`**. Set the "Alarm Signature" section in the "Definition" tab to:

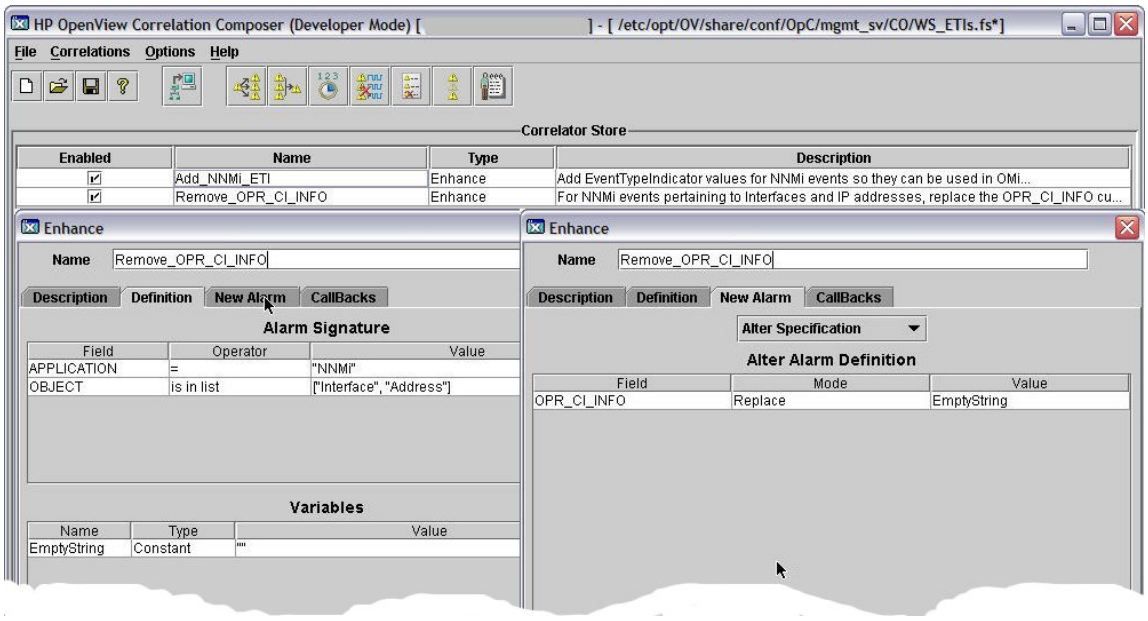
```
APPLICATION =                  "NNMi"
OBJECT          is in list     ["Interface","Address"]
```

and define a new variable in the "Variable" section:

```
  EmptyString      Constant     ""
```

In the "New Alarm" tab, select "Alter Specification" from the drop-down menu, then add the following Alter Alarm Definition:

```
  OPR_CI_INFO      =      EmptyString
```



(When clicking in the "Field" column, note that the OPR_CI_INFO variable is listed all the way at the end of the pull-down menu.)

Press **OK**, make sure the correlator is enabled, then save the correlator store somewhere on your HPOM for UNIX server. To activate this correlator, merge it into any correlator store that is already active, or, if used as the only correlator, install it to the runtime HP Correlation Composer engine by executing the command:

```
  ovocomposer -install -fs <composer_store_name> -ms
```

(where `<composer_store_name>` is the full pathname to where you saved the HP Correlation Composer store.) From now on, every event created by the NNMi integration that reports problems with network interfaces or IP addresses will be processed by this HP Correlation Composer correlator. Based on the Alarm Signature (the event's application field is "NNMi", and the object is either "Interface" or "Address"), HP Correlation Composer will discard the value of the OPR_CI_INFO custom message attribute. This in turn leads HP OMi to evaluate all other event attributes when it tries to resolve the event to a particular CI. When sufficient data is available, the event is mapped to the proper UCMDB CI of CI type "Network Interface" or "IP".

Removing Wrong OPR_CI_INFO Values from Events (HPOM for Windows Only)

Unfortunately, HPOM for Windows does not include the HP Correlation Composer product to facilitate event manipulation on the fly. However, HPOM for Windows does provide all the required API hooks to achieve a similar result if a customer or user is not afraid of creating a custom program to access the HPOM for Windows server's Message Stream Interface.

Conceptually, such a program would need to execute the following steps:

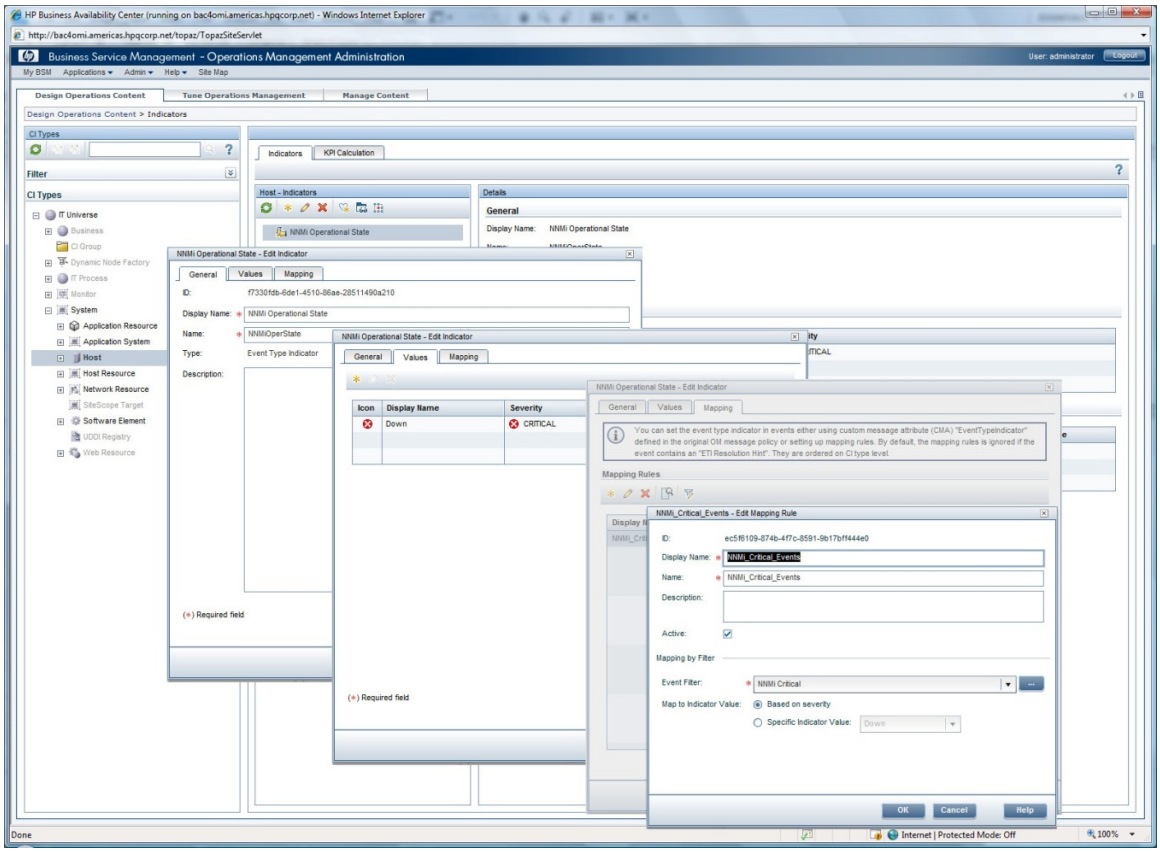
- Open the OMW server's MSI
- Create a filter specification ("APPLICATION=NNMi", and "OBJECT=Interface|Address") and register the filter with the MSI
- Wait for notification of a new event from the MSI
- Read the event, then blank out the OPR_CI_INFO Custom Message Attribute
- Write the modified event data structure back into the MSI
- Go back to waiting for a new event notification

Setting ETIs on NNMi Events

As we observed in the "HP NNMi–HPOM/HP OMi Integration" section on page 34, because NNMi does not send "reset" or "status update" events, Health Indicators cannot be meaningfully employed in HP OMi. Instead, we are limited to single-value Event Type Indicators (ETIs), typically indicating that a certain condition or situation has occurred.

By using indicator mapping rules (as illustrated in the section "Example: BPM integration" on page 25) we can easily signal an ETI value to HP OMi. Typically, we would want an NNMi-generated event to set an ETI for the operational state of a network element CI. To facilitate this process, the required steps are:

1. Define valid ETIs and their allowed value for the desired CI Types.
2. Create a mapping rule to set the ETI value upon receipt of an NNMi event, or, alternatively, use other means to insert a CMA into the event that will be interpreted as the name and value of an ETI.



To define the actual ETI and its value, in HP OMi navigate to **Admin** → **Operations Management** → **Design Operations Content** → **Indicators**. As NNMi monitors the operational state (besides many other aspects) of networked hosts, in this example we create a new ETI for the "Host" CI Type. Recall that ETIs are inherited by CI subtypes from their respective parent CI Type. Here, that implies that any Host ETI is automatically available on, for example, Switch or Windows CITs.

1. Select CI Type **Host**.
2. Select the "New item" icon (*) under "Host – Indicators".
3. Set Display Name ("NNMi Operational State") and Name ("NNMiOperState").
4. Ensure the type selector is set to "Event Type Indicator", then select **Next**.
5. In the Values screen, select the default value **Occurred**, and then select the delete button **X**.
6. Select "New item" (*), set both Display Name and Name to **Down**.
7. Select Event Severity **CRITICAL**, then select **OK**.
8. Press **Next**. On the Mapping screen, select "New item" (*) to create a new mapping rule.
9. Set Display Name and Name to **NNMi_Critical_Events**, and ensure the "Active" box is checked.
10. Select the Browse button (…) to create a new filter for the Event Filter list.
11. Press "New item" (*) and set the Display Name **NNMi Critical**.
12. In the General tab, deselect all severities except Critical
13. Go to the Advanced tab, place a checkmark into the Application selection box, select **equals** and fill in the application name (NNMi). Press **OK**. Select your newly created filter in the list, and select **OK**.

14. Back in the "Create New Mapping Rule" window, your new filter is now selected. Ensure "Map to Indicator Value" is set to "Based on severity". Select **OK**, then **Finish**.

The preceding steps have created an ETI mapping rule that will set the ETI "NNMiOperState" to value "Down" for every critical NNMi event received. Of course, the event must also resolve to a "Host" CI type (or any of its subtypes, including all "Computer" and "Net Device" subtypes), as otherwise the mapping rule will not be triggered.

If desired, repeat the above procedure for additional CI Types that may be affected by NNMi events, for example "Network", "Network Interface", or "IP" addresses.

Setting ETIs on NNMi Events (HPOM for UNIX Only)

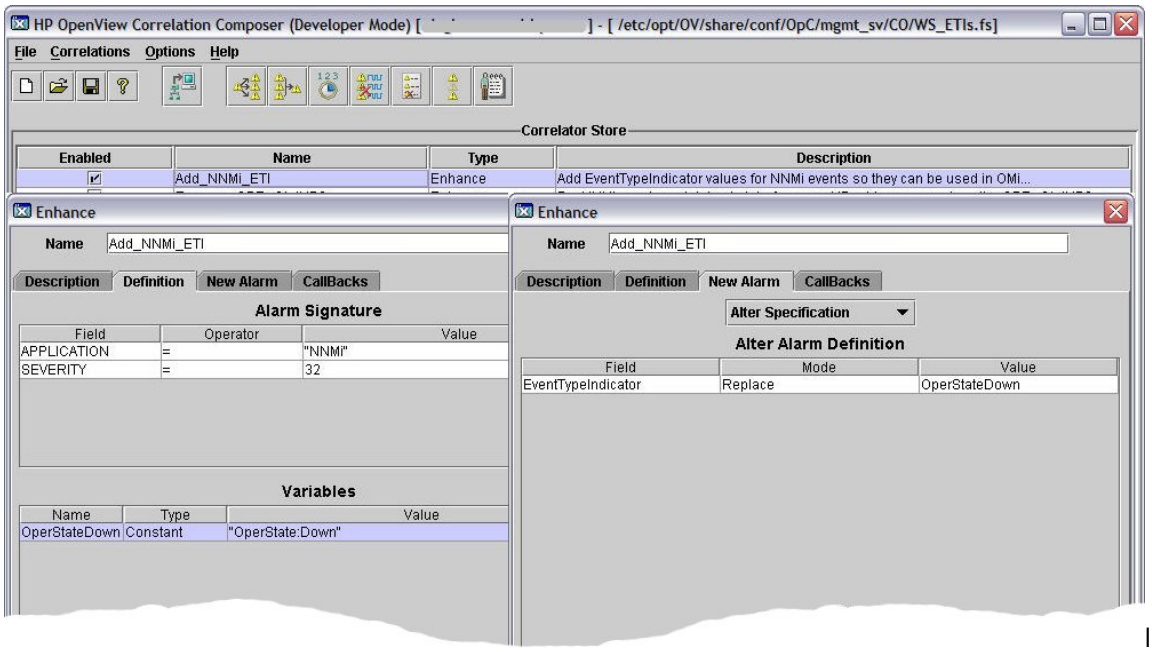
When integrating HP OMi with HP OMU, the capabilities of the MSI also allow for a different mechanism to associate an ETI with an event. Instead of performing the ETI mapping in HP OMi, HP Correlation Composer can be used to add a CMA of name "EventTypeIndicator" to the event while it is progressing through the MSI. In order for HP OMi to recognize the value of the CMA as an ETI, it needs to be named "EventTypeIndicator" and have values in the format "`ETI_name:ETI_value`".

*To use HP Correlation Composer to add a CMA to the event stream at runtime, make sure all prerequisites are fulfilled for using HP Correlation Composer on the OMU management server. Refer to section "*Removing Wrong OPR_CI_INFO Values from Events (HPOM for UNIX Only)*" on page 36 for specifics on how to configure processing priorities and enable additional CMAs in the* `CO.conf` *configuration file.*

Run **`ovocomposer -ui`**, open or create a Correlator Store, then add a new correlator of type "Enhance". Name the correlator **`Add_NNMi_ETI`**. Set the "Alarm Signature" in the "Definition" tab to

```
APPLICATION     =     "NNMi"
SEVERITY        =     32
```

(Here, the value 32 represents the critical severity of an event. For a complete list of possible severity values, check the **Correlations** → **Global Constants** menu function in the HP Correlation Composer UI.)



In the "Variables" section, define a new variable:

```
OprStateDown      Constant      "NNMiOperState:Down"
```

Switch to the "New Alarm" tab, select **Alter Specification** from the pull-down menu and set the "Alter Alarm Definition" section to:

```
EventTypeIndicator     Replace     OprStateDown
```

Save the definition and the correlator store, and then install the correlator store to the OMU management server's runtime engine, for example by running

```
ovocomposer -install -fs <composer_store_name> -ms
```

From now on, any new event received by OMU from NNMi with critical severity will be enhanced by adding a custom message attribute of name "EventTypeIndicator", with its value getting set to "NNMiOperState:Down". In HP OMi, this CMA in turn sets the ETI "NNMiOperState" to the value "Down", thus achieving the same effect as the ETI Mapping Rule created earlier.

Setting ETIs on NNMi Events (HPOM for Windows Only)

While OMW does not provide the HP Correlation Composer tool to define MSI event stream correlators, the product does include a complete set of application programming interfaces (APIs) to programmatically access the message stream. Conceptually, these steps are required to create a new Custom Message Attribute while an event progresses through the OMW server's MSI:

- Open the OMW server's MSI
- Create a filter specification ("APPLICATION=NNMi", and "SEVERITY=SEV_CRITICAL") and register the filter with the MSI
- Wait for notification of a new event from the MSI
- Read the event, then add a new CMA to the event's data structure
- Write the modified event back into the MSI
- Go back to waiting for new event notification

Correlation Rules

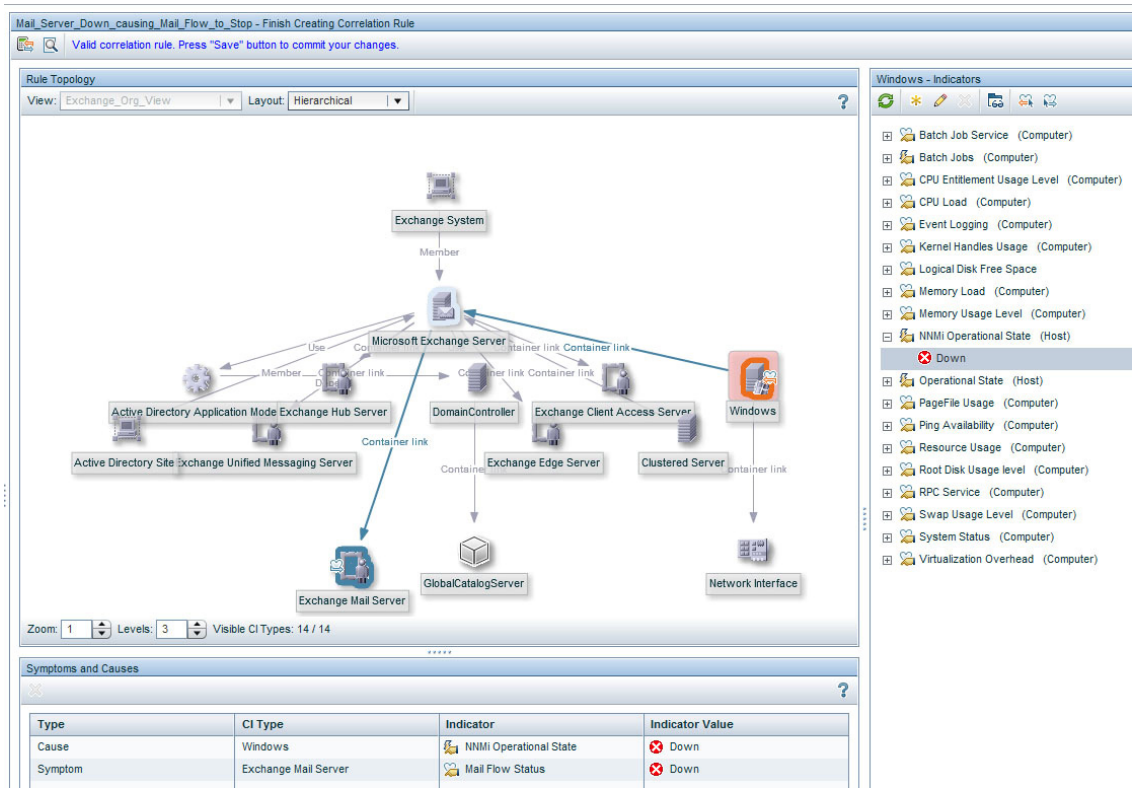
Having defined event type indicators for CI Types monitored by NNMi, and having provided HP OMi with mapping rules (or using HP Correlation Composer, or the MSI APIs) to determine an ETI's value, we can now use NNMi events for HP OMi topology-based event correlation (TBEC). All that is needed for TBEC are enhanced (or new) correlation rules in HP OMi, referencing the newly added event type indicators.

As an example, imagine a scenario where NNMi monitors your critical network elements, plus the business-critical server nodes hosting an MS Exchange installation. While HP OMi provides many out-of-the-box correlation rules for various Exchange scenarios, you also need to be able to determine that a stopped mail flow in the Exchange installation is caused by a crashed or stopped Exchange Mail Hub server.

To have HP OMi perform this correlation and identify a "Node Down" event for the Mail Hub to be the cause for the stopped mail flow, navigate to **Admin** → **Operations Management** → **Design Operations Content** → **Correlation Rules** and create a new correlation rule by selecting the "New item" (*) button.

Name the Correlation Rule (**`Mail_Server_Down_causing_Mail_Flow_to_stop`**), and base it on the UCMDB view "Exchange_Org_View". Mark the correlation rule as "Active" and continue by selecting the **OK** button. In the resulting Rule Topology view, select the "Windows" CI type to display a list of its event type indicators to the right. Drill into the "NNMi Operational State (Host)" ETI, right click the **Down** value and select **Add as a Cause**.

Back in the Rule Topology view, select on the **Exchange Mail Server** icon to display its list of ETIs and Health Indicators. Open the "Mail Flow Status" HI, right-click its **Down** value and select **Add as a Symptom**. To finalize, save the correlation rule.



Once saved, this new correlation rule will properly identify a "Node Down" event sent by NNMi for the Exchange Mail Hub server as the root cause event, when at the same time (or within the TBEC correlation time window) the MS Exchange Smart Plug-In for HPOM generates an event indicating that the server's "Mail Flow Status" is "Down".

Tools in the Network Management Context

In addition to identifying root cause events, a common requirement for the NNMi–HP OMi integration is the need to drill into network element-specific diagnostics, or simply to look up asset data for the network element, both maintained by NNMi. It is therefore desirable to launch into (context- and event-specific) screens of the NNMi web-based user interface directly from HP OMi's event browser, in either the Event or Health Perspective view.

Through the addition of Tools in the HP OMi UI it is possible to connect to CI Type-specific NNMi views directly from a selected HP OMi event. We can define the tool parameters from the **Admin** → **Operations Management** → **Design Operations Content** → **Tools** screen.

To define a new tool, select the desired scope of the tool (that is, upon which CI Types it can be invoked) from the CI Types navigation tree. Then press the "New item" (*) button to open the Create New Tool configuration dialog.

Name the new tool (**`NNMi Layer2 Neighbor View`**) and select **Next**. Select the tool type. Because NNMi provides an extensive selection of URL-based functions, select **URL** for this example configuration and then select **Next**. On the resulting screen, we need to construct the complete URL to the desired NNMi view.

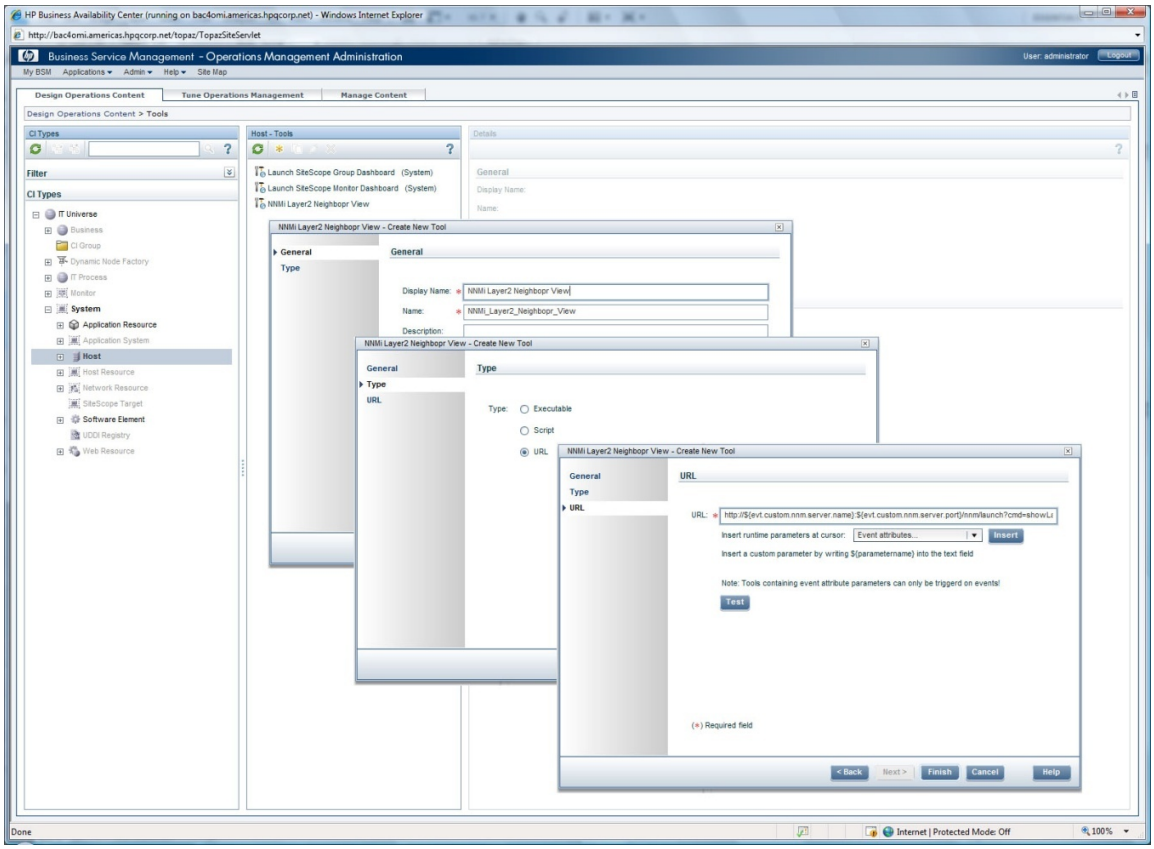
For a complete overview of which functions or views NNMi makes available under which URL, in NNMi, invoke **Help** → **NNM Documentation Library** → **URL Launch Reference**. The general format to launch NNMi functions from outside the program is:

`http://<host>:<port>/nnm/launch?cmd=<some_command_and_parms>`

From the URL Reference, we can see that to invoke a Layer 2 Neighbor view, we can connect to

http://<host>:<port>/nnm/launch?cmd=showLayer2Neighbors&
objType=Node&objuuid=<objectUUID>

At runtime, <host> and <port> need to be replaced with the name and port number of the NNMi server, and <objectUUID> with the UUID of the affected (or causing) network element. Luckily, the event that originated with NNMi does contain all the required data in a number of custom message attributes named **nnm.server.name, nnm.server.port and nnm.emittingNode.uuid**.

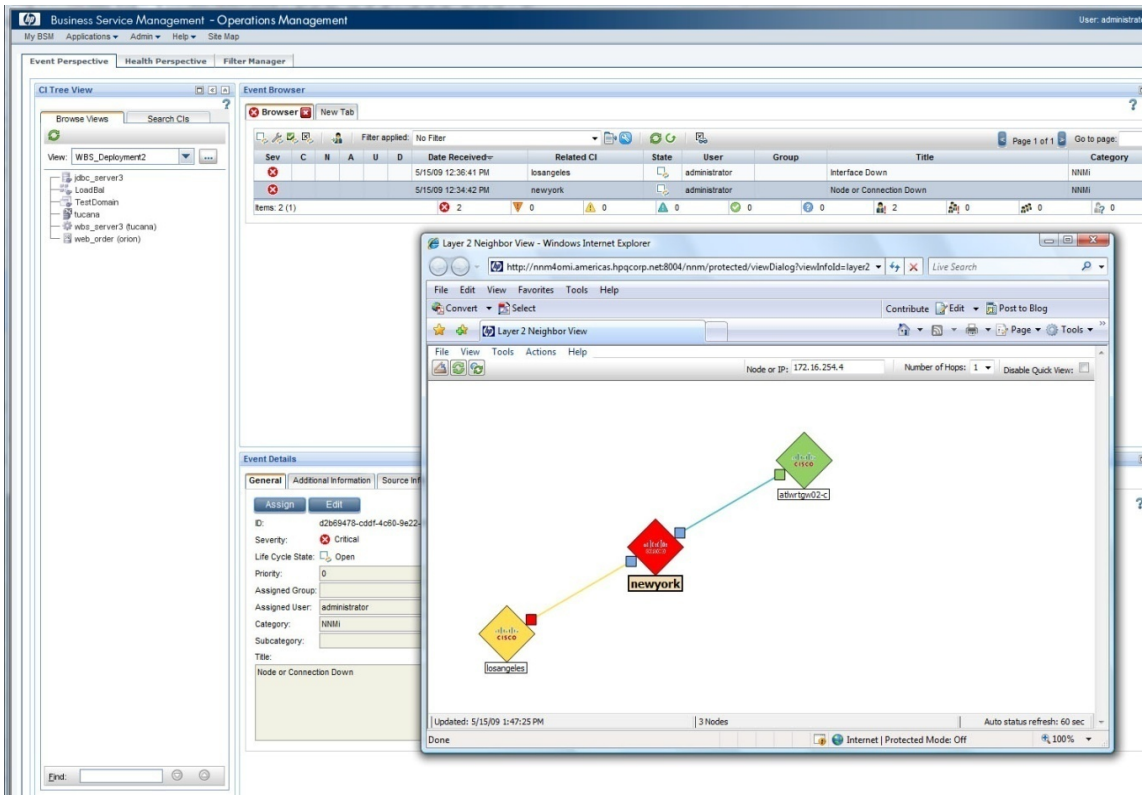


To construct the final URL for our Tool definition, select "Event attributes …" in the "Insert runtime parameters at cursor:" menu, then select **Insert**. While standard event attributes could be selected from the event attribute list, custom message attributes require us to specify their name in the "Custom event attribute" choice's text field. To insert the <host> portion of the URL, enter **nnm.server.name** in the "custom" field and select **Insert**.

Upon returning to the URL definition screen, it becomes obvious that the syntax used to refer to Custom Message Attributes in the URL field is "`${evt.custom.<CMA name>}`". Therefore, the final URL can be specified as

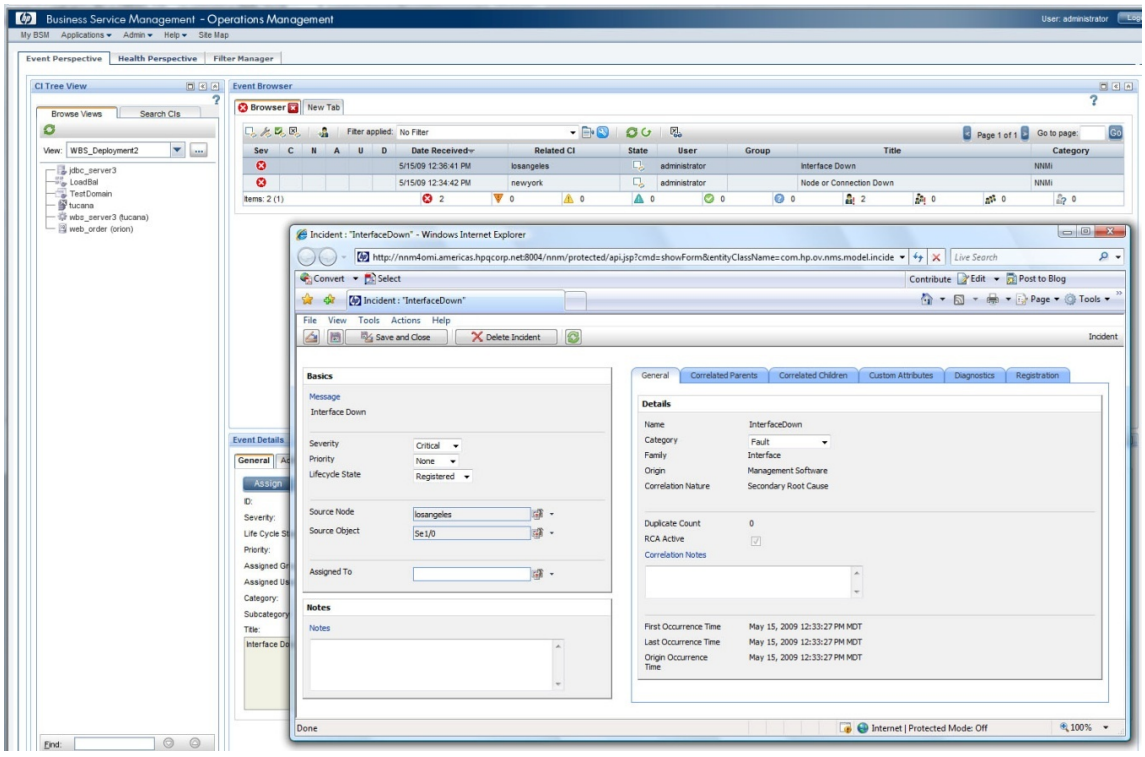
http://${evt.custom.nnm.server.name}:${evt.custom.nnm.server.port}/
nnm/launch?cmd=showLayer2Neighbors&objType=Node&
objuuid=${evt.custom.nnm.emittingNode.uuid}

After saving the URL and Tool definition, it is then available for invocation directly from the HP OMi event browser, provided the event resolves to a CI of the correct CI Type. By right-clicking the event and selecting the "Launch Tool …" menu choice, HP OMi presents a list of all configured tools for the particular event's CI Type. Select a tool by clicking its name in the presented list. While you can directly invoke it using the "Run Tool" button, it is considered another best practice to use the "Next" button to verify the URL invoked. Also, when on the "Specify Parameters" screen, you have the option to provide values for any missing parameters that could not be resolved from runtime data. Click through to the "Preview Tool Execution" screen, or invoke the tool directly by selecting **Run Tool**.



In addition to invoking a Layer 2 Neighbor view, many other functions and forms are available for direct invocation. For a complete list, again refer to the URL Launch Reference.

Example: To display an incident's details in NNMi, use these URL parameters:
`cmd=showFormandobjtype=Incidentandobjuuid=${evt.custom.nnm.incident.uuid}`



Example: To show all open incidents assigned to my NNMi user ID:
`cmd=showViewandview=myIncidentsTableView`

Miscellaneous

HP OMi and HP Service Manager

HP Service Manager (HP SM) and HP OMi are integrated through an underlying HP OM. As described is this paper earlier, HP OMi is an add-on to HP OM. This underlying HP OM server is the integration point for several management products. The integration of HP OMi and HP SM works via the well known SCAuto integration between HP OM and HP SM. SCAuto for OM allows you to automate the process of creating, updating, and closing trouble tickets. It has the capability to annotate, own, and acknowledge OM messages from modifications done on Service Manager Incident tickets.
This approach of integrating further management products with HP OMi via the underlying HP OM server is applied also for integrations with AlarmPoint and BMC Remedy. AlarmPoint provides a direct integration with HP OM whereas BMC Remedy is integrated with HP OM through the HP Smart Plug-in for Remedy AR System.
HP BAC provides a rich set of integration points with HP SM as well. Use case such as Incident submission, Problem Isolation, Dashboard and Service Level Management are covered by this integration. You might consider applying these use cases in your BSM solution deployment. For details see HP BAC 8 Solutions and Integrations guide.

HP OMi and HP Operations Orchestration

HP Operations Orchestration (HP OO) helps reduce operational costs by automating routine IT tasks, such as repetitive maintenance, change provisioning, and incident resolution. HP OO integrates with your IT environment to ensure minimal impact on current procedures and tools while fully utilizing your existing IT investments in management product such as HP OM, HP BAC and HP SM. The integration between HPOM and HP OO is very useful for a solution deployment including HP OMi. For example, HP OO actions for accessing and modifying OM messages update events in HP OMi as well, because events are fully synchronized between HP OM and HP OMi. The HP OM integration in HP OO makes it possible to start OO flows automatically or per user request. For details see the HP OO product manual “HP Operations Manager Integration Guide”. Alternatively, you can make use of the HP BAC / HP OO integration. This integration enables user to launch HP OO flows from the BSM Console in content of a selected CI.

HP OMi and HP Dependency Mapping Automation

HP OMi does not require HP Dependency Mapping Automation (DMA). However, there are two use cases where DMA provides additional value in an HP BSM solution with HP OMi as outlined above.

DMA can Automatically Setup New Nodes in HPOM

If your UCMDB is kept up to date when new systems are configured and used in your IT environment, then DMA can be used to automatically setup corresponding nodes in HPOM and to deploy policies to start the monitoring of these systems.
This assumes that the UCMDB is populated using HP DDM or UCMDB tools and not using HP OMi’s topology synchronization.

DMA can Provide UCMDB-based Views to HPOM Operators

To benefit from HP OMi, operators obviously have to use the HP OMi console. However, if some operators still remain on the HPOM operator console, then DMA can provide them with the UCMDB-based service views, similar to the views that HP OMi operators see. DMA allows you to gradually move from a SPI discovery-based service view to a UCMDB-based service view.
See the DMA documentation for more details.

DMA and HP OMi's Topology Synchronization

HP OMi's topology synchronization can be used to populate the CMDB based on the HPOM service model. DMA populates the HPOM service model based on UCMDB data. Does it make sense to use both mechanisms? Will using one mechanism break the other?

There are two options:

- *Do Not Use Topology Synchronization at All*

  The primary reason to use topology synchronization is to create the model (CIs and relationships) that is necessary for the key HP OMi features: to map events to CIs, to assign and set ETIs and to correlate events based on the underlying topology.
  If this model already exist in the CMDB (discovered by HP DDM), then there is no need to use topology synchronization.
- *Use Topology Synchronization Selectively*

  The default HP OMi topology synchronization creates hosts, CI groups and HP Operations agent CIs and corresponding relationships. Host CIs should have been discovered by HP DDM already and HP Operations Manager agent CIs can be created using DMA's node discovery (DMA 8.2 has been aligned so that its node discovery creates the same HP Operations Manager agent CIs as topo-sync), so the only CIs that will be added by the default topology synchronization are the CI group CIs (equivalent to the node groups in HPOM).

It can make sense to synchronize SPI service models and to create application CIs based on a discovered SPI service tree, but only if the HP DDM discovery does not already create a suitable model. You can only do that as long as the SPI discovery has not been disabled by DMA. If you link those SPI services with other CIs (for example: service CIs) in the UCMDB and sync that back to HPOM, then you have to make sure that you do not synchronize the SPI services again. Instead just sync the service CIs and use external links in DMA to link them to the existing services created by the SPI.

It definitely does not make sense to synchronize HPOM services that originate from the UCMDB back to the UCMDB as they already exist there. So there is no need to setup topology synchronization rules for these.

CI Resolution and DMA Smart Mapping

DMA hooks into the HPOM message stream interface and changes the HPOM message service_id field. Does that affect the CI resolution on the HP OMi side?
No, it should not. DMA adds the UCMDB id in the service_id field. HP OMi can directly handle UCMDB ids and does not have to go through the CI resolver mechanism again to find the correct CI.

For more information

http://www.hp.com/go/software
http://support.openview.hp.com/selfsolve/manuals

Technology for better business outcomes

December 2009