Moving to Service Centric Management with Operations Bridge

Recommended Implementation Steps



Table of Contents

| Introduction | 3 |
|--|---|
| Overview | 3 |
| Document Structure | |
| Planning Activities | |
| Step 1: Consolidate Events | |
| Step 2. Control Events | |
| Step 3: Add End-User Perspective | |
| Step 4: Business Service Centric Management | |
| Step 5: Self-Service and Monitoring Automation | |
| | |

Introduction

Micro Focus Operations Bridge monitors your IT environment and consolidates data from existing tools. It applies automated discovery, monitoring, analytics, and remediation to data across traditional, private, public, and multi-cloud and container-based infrastructure. It exploits automated AIOps machine learning and analytics for event reduction and accelerated root cause identification and then executes automated remediation. Executive stakeholders gain actionable insight on their mobile devices with tailored dashboards showing key status, business and IT KPIs.

Operations Bridge Manager (OBM) is a key component of Operations Bridge. It provides capabilities to collect and consolidate event and metric information from various sources, perform valuable event correlation allowing to take corrective actions from a central console, link the information against a run-time service model consisting of infrastructure and business services, display service health information and integrate universally to external components.

The target audiences for this document are architects and consultants planning the implementation of an Operations Bridge solution.

It is assumed that the reader has a basic knowledge of Operations Bridge, its concepts and related products. This document is based on Operations Bridge 2018.08.

Overview

OBM provides instant value after the initial deployment due to its out-of-the box capabilities.

Using OBM with Management Packs and out-of-the-box integrations simplifies the deployment of a solution delivering value immediately without the need of dealing with the underlying technology-related complexity.

Management Packs are available for example for the following domains:

- Infrastructure
- Databases
- Microsoft environments
- SAP environments
- Middleware
- Cloud
- Big Data

and more.

Numerous out-of-the-box integrations are supported with OBM, for example:

- Microsoft SCOM
- Nagios
- Oracle Enterprise Manager
- IBM Tivoli and NetCool
- BMC Remedy

NNMi

and more.

A complete list of available Management Packs and integrations is available on the <u>Micro Focus Marketplace</u>. OBM is highly flexible and extensible. A successful implementation of OBM cannot achieve the maximum of its capabilities in one step. Therefore, it is recommended to use a stepwise iterative approach described further in this document. Using Management Packs enables you to extend OBM with your specific integrations. <u>Note</u>: A key component of OBM is its run-time service model (RTSM), which builds the foundation for the OBM powerful event correlation and service management functions. Management Packs and out-of-the-box integrations supply a discovery-based basic topology model automatically into the RTSM and maintain the RTSM data in a dynamic manner. This eliminates the need to spend the implementation effort upfront and brings immediate benefits.

Document Structure

This document contains the following sections:

• Planning activities

This section describes the aspects to be considered when planning an OBM implementation.

Consolidate events

This section recommends to start with the event consolidation and provides some background information and typical tasks to be performed to gather the event information. It also outlines the benefits related to consolidating events.

- **Control events** This section describes the activities that can be performed to gain control over the events and to tune the event processing.
- Add end-user perspective This sections describes how to complement monitoring with the status information from the enduser perspective.
- Business service centric management
 This section provides information on how to leverage the outcome of previous phases to build a
 business service centric management solution and to provide business service centric views and
 dashboards.
- Self-service and Monitoring Automation This section describes how to increase administration efficiency of OBM in dynamic and virtualized environments.

Planning Activities

This section summarizes considerations for planning an OBM implementation.

Environmental Analysis

As an initial step in a successful OBM implementation, Micro Focus recommends to perform the environment analysis. This helps determine boundary conditions for a project and elaborate the scope of systems and applications to be covered (e.g. identify external systems, such as existing element managers that will feed the information into OBM, higher-level systems that will consume the information gathered and processed by OBM, as well as its users and user groups).

To gather information upfront, Micro Focus recommends to go through questionnaires and refine important topics through workshops with subject matter experts.

The following items can be collected as a starting point:

- The scope of systems (including system properties, such as an OS type and version, system responsibilities and ownerships, as well as the relation between systems and applications)
- Processes and organizations to be included or integrated with OBM
- Important organizations, user groups and users and the stakeholders for these groups
- Information about the relevant network topology

- An overview of the monitored aspects (if monitoring of systems and applications is already in place)
- Element managers to be integrated into the Operations Bridge that can act as a source of qualified event and status information
- External systems, such as mail servers, SMS notification and trouble ticket tools that can act as consumers for the information consolidated in OBM

Deployment Planning and Scenarios

Operations Bridge is the containerized service-oriented autonomous monitoring solution for Hybrid IT. It can be deployed in different ways with different set of capabilities, thus delivering an optimal fit for your requirements.

Drivers for planning the deployment are summarized below:

• A set of capabilities you can choose from depends on your business needs. You can deploy a single capability (for example, Operations Bridge Manager), or a combination of various capabilities (OBM with Operations Bridge Analytics to gain a complete view of your IT operations analysis, or OBM with Operations Bridge Reporter for cross-domain business service-driven resource reporting). The overview of all capabilities that are part of Operations Bridge can be found <u>here</u>.

Note: The focus of this white paper is the OBM deployment.

- Containerized or classic deployment.
- OS and database platform; virtualization. Both Linux and Windows can be used with OBM. OBM supports MS SQL, PostgreSQL, and Oracle as database platforms. Virtualization is in general supported by OBM.
 <u>Note:</u> The target system(s) should be dedicated to OBM and cannot host additional applications.
- Role of system. In small scaled environments, it is possible to develop the OBM configuration on a single OBM instance and use the same instance to run this configuration. However, this setup is not recommended in larger-scale environments, where a designated development system may be needed to separate development activities from the productive environment. Furthermore, it may be necessary to use a third instance of an OBM system to verify any developed configuration or a configuration change on a reference test/integration test environment before applying it in the production environment. Development systems typically have lower demand regarding scale and high availability, while production systems must meet the demands regarding scale, security, etc. required by the target environment. It is recommended to implement integration test/reference systems in the same way as implemented on the production systems.
- Connectivity. Depending on the specific target network, it may be necessary to place the OBM system in the appropriate network segment. The following considerations need to be taken into account:
 - From where do users connect to the system? The location of the systems that forward the events to OBM ("southbound")?
 - o Does the event information need to use certain networks/management interfaces?
 - What systems consume information from OBM ("northbound")?

While in simple small-sized networks this aspect is not important and easy to evaluate, connectivity may become rather complex in service provider contexts. As a result, it may be necessary to implement network setup changes, such as opening ports on firewalls, establishing certain routes on the network, etc.

- Scalability. OBM allows to implement different deployment scenarios: "typical" (single-node) deployment and distributed deployment for scalability by spreading the OBM components across several systems. Implementing a distributed deployment requires at least two systems: a Data Processor Server (DPS) and a Gateway Server (GW). A load balancer allows distributing communication load from users and event feeding systems evenly across a number of Gateway Servers. The number of Gateway Servers can be increased at a later time by installing additional Gateway Servers and connecting them to an existing OBM installation. This allows flexible adaptation to increase scalability, e.g. due to the growing number of users or growing event rate.
- Security. OBM UI communication can be secured with HTTPS/TLS. To meet specific security requirements, OBM can be configured to run in a non- root context. Network-related measures can be taken to restrict accessibility of the OBM application, e.g. by using firewalls. OBM features a user concept that allows a specific restriction of capabilities based on users/user group memberships.
- High availability. Monitoring capabilities of a load balancer can be used to increase the Gateway Server availability: if a monitored Gateway Server is down, a load balancer recognizes this and reroutes the incoming request to one of the available Gateway Servers. For increased availability, install a standby backup Data Processing Server. You can increase availability of an existing OBM system at a later time by installing a standby Data Processing Server and additional Gateway Server instances.
- Use of RTSM. See the section RTSM Considerations.

There are three basic types of how OBM can be deployed:

- Single-server ("typical") deployment.
 In this deployment, both Data Processor and Gateway Server components are installed on a single system. This is a good solution for the environments with medium to low scalability, which do not demand high availability (for example, small or development environments).
 The "typical" (single node) deployment is not recommended for use in production environments.
- Distributed deployment on two systems. This deployment is done on two servers, separating Data Processor Server and Gateway Server components. Compared to the single-server deployment, this setup has increased scalability.
- Distributed deployment on multiple systems. This deployment typically consists of one or two Data Processor Server(s) with multiple Gateway Server instances and utilizes a load balancer or a reverse proxy to distribute the load. This setup meets demands for increased scalability and high availability.

<u>Note</u>: As stated above, a container deployment of Operations Bridge is also available. However, the containerized Operations Bridge version 2018.08 does not yet offer similar scalability as the deployment options shown above.

<u>Note</u>: Micro Focus recommends to host the OBM database on a dedicated database server (in general, in all of the deployments described above the database can be local or remote).

OBM Building Blocks

OBM consolidates all events from the IT infrastructure monitoring in a central console and allows linking the event information against IT services. Combined with the complementary service status information from business service management components, OBM provides a complete overview of the overall IT infrastructure and provided services.

To achieve this, OBM provides the integration of various event sources through available out-of-the-box integrations.

On Event Data Acquisition, the integration covers (but is not limited to) the following event sources:

- Operations Manager Agents (can be connected via Monitoring Automation)
- Operations Manager for Unix/Windows
- Business Process Monitor
- Real User Monitor
- SiteScope
- Systems Insight Manager
- Alerts (e.g. CI status alerts, SLA alerts, event-based alerts)
- Integration of third-party element managers for specific domain environments using Operations Connector:
 - o Microsoft Systems Center Operations Manager (SCOM)
 - o Oracle Enterprise Manager
 - o Nagios
 - o IBM Tivoli
 - o ... and more

Monitoring solutions are available for a variety of platforms, middleware components, and applications:

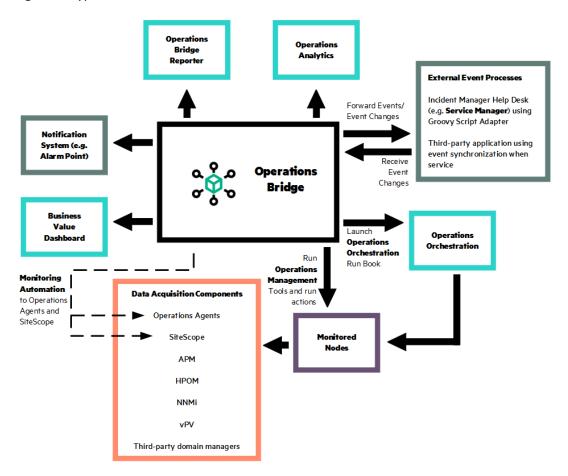
- Management Packs for OBM. Available for infrastructure (including virtualization and high availability), databases, MS applications, Cloud, Big Data, and more.
- <u>ITOM Marketplace</u> contributions. A growing number of Management Packs is contributed for re-use via the ITOM Marketplace.

Northbound Integrations:

- Alerting and Notification
- Incident Management
- Generic integration into external event processing through flexible groovy script-based interfaces

The following figure shows an example architecture draft of an OBM solution with potential event sources.

Figure 1: Typical OBM solution architecture



Note: For an introduction into OBM, refer to the Get Started documentation section.

RTSM Considerations

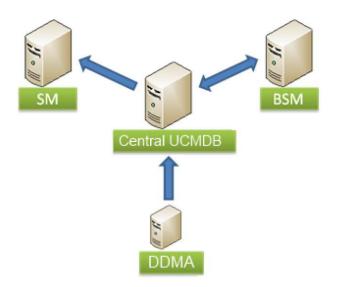
The RTSM is the configuration management database instance embedded into OBM. Its objective is to host an operational run-time service model consisting of all configuration items that are relevant for the infrastructure and service operations, including the Cl's relationship.

Within OBM, the run-time service model is the basis for a rich set of functions of the Operations Bridge, such as sophisticated event correlation, hosting of the service model and calculation of the KPI-driven service health. The RTSM focus is clearly on monitoring as it reflects the dynamics of the managed environment in near real time.

The RTSM is not a replacement for a configuration management solution. A configuration management solution has a different scope as it acts as a master for the CI information. It is a single point of integration supporting various configuration management processes and supplying the right set of data for different use cases. The operational service model is only one use case.

Depending on the scope, it may be feasible to consider the RTSM only (for example, if an operational service model is the only relevant model). However, if the requirements demand a comprehensive support of configuration management processes or if extended integrations are required, the implementation of a central configuration management solution is highly recommended. Such solution can be integrated with the operational RTSM of OBM.

Figure 2: BSM (RTSM) integrated with a central CMDB



Note that it is not required to implement a sophisticated and comprehensive data model within the RTSM upfront. Instead, the data model can grow over time and be refined and extended at a later stage. As an example, out-of-the-box integrations supply the CI and CI relation information into the RTSM with little effort, which can act as a basis for operations at early stages.

<u>Note</u>: Management Packs and out-of-the-box integrations supply a basic topology to the RTSM with no effort. Micro Focus recommends starting with this basic topology to consolidate and control events.

While extending and modifying the RTSM can be done in later project phases, Micro Focus recommends to understand the RTSM concepts upfront and perform planning of the data model to identify the system contributing the Cl information.

A good practice is to establish a basic topology consisting of nodes and the software running on these nodes during the event consolidation phase. Later, this can be complemented by the additional CI information, e.g. based on discovery.

<u>Note</u>: For more information on designing, implementing and tuning a fully compliant model, see the <u>RTSM</u> <u>Best Practices</u> document.

Project Implementation Approach

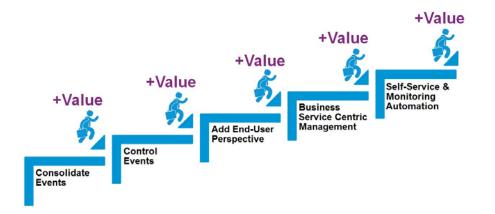
An OBM implementation project can be divided into several phases following the environmental analysis and deployment planning described in previous sections. This sets the basis the solution is built upon. Micro Focus recommends to perform the steps for implementing an OBM-based Operations Bridge in a defined sequence. This allows a clear and structured implementation approach. Less is more: avoid too many functions to be performed at once.

Although a comprehensive approach may work under a well-defined scope with good starting conditions, a stepwise incremental approach consisting of the following steps is recommended:

- Consolidate events
- Control events
- Add end-user perspective

- Business service centric management
- Self-service and Monitoring Automation

Figure 3: Steps and Sequence for a Successful Implementation



These steps are explained in more detail in the following sections.

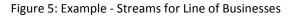
<u>Note:</u> You do not need to implement all steps at once to benefit from using OBM. Implementing initial steps, such as Consolidate Events and Control Events, already returns a lot of value to operators and allows comprehending the OBM concepts and functions.

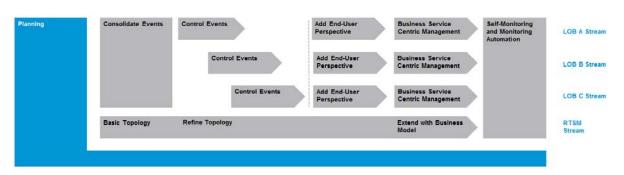
Carry out the steps sequentially. Benefits are gained with the implementation of every step. Activities related to the RTSM topology implementation are typically done in parallel.

Figure 4: Single Sequence

| Planning | Consolidate Events | Control Events | Add End-User Perspective | Business Service Centric Management | Self-Monitoring and Monitoring Automation | Add Value to Solution |
|----------|-----------------------|-----------------|-----------------------------|--|---|--------------------------|
| | Basic Topology | Refine Topology | | Extend with Business Model | | RTSM Stream |
| | | | | | | |

For larger-scale projects, it can be feasible to initially consolidate all events with a basic topology in the RTSM. Thereafter, the steps can be performed in smaller units, e.g. for each line of business.





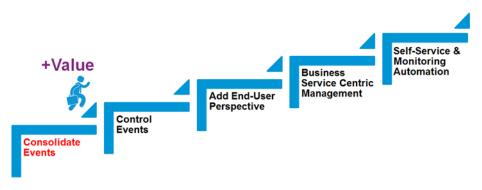
Note that this is just an example. A synchronization point after control events is not mandatory.

Step 1: Consolidate Events

This step combines all activities required to feed the event information into the OBM. Although this is the basis for all consecutive steps, its implementation already provides the value associated with relatively low effort.

It is not mandatory to execute consecutive steps immediately, which enables you to gain experience with the solution before evolving it.

Figure 6: First Step – Consolidate Events



Event Sources

The following considerations are relevant for the event consolidation:

- OM Agents can be integrated directly with OBM using the Monitoring Automation functionality. By using policy templates on systems with OM Agents, you can retrieve the event information from various sources, such as textual log files, windows event logs, monitoring services and processes, measuring numeric metrics for comparison against thresholds, intercepting SNMP traps and more.
- Using Management Packs together with the Monitoring Automation functionality allows you to implement a consistent and easy integration of monitoring of common OS and virtualization platforms, Oracle database platforms, Hadoop and Vertica, and so on. These Management Packs contain everything required to implement management of the related object in a quick, defined and efficient manner.
- Integration of already existing Operations Manager instances. With little effort, it is possible to gather and consolidate events of Operations Manager Servers. This approach incorporates synchronization of the event information derived from the already existing monitoring solution thus leveraging the investment done on this solution, e.g. Smart Plug-In based realization of monitoring. Furthermore, it is easy to include all existing CIs, offering a starting point to elaborate event-driven correlations in a next possible step.
- Connection of additional element managers through existing connectors. Element managers and management tools, such as Microsoft SCOM, Nagios, Oracle Enterprise Manager, IBM Tivoli and many more, can be easily integrated.
- Generic use of the Operations Connector. For specific element managers or integrating specific event sources, you can use the Operations Connector, as it provides a rich set of easy-to-use interfaces to accommodate these tasks.

At this point, the CI Information is already present in the RTSM. The integration of element managers, such as Microsoft SCOM, and the integration of OM Servers synchronize the set of CI instances. OM Agents connected to OBM directly via Monitoring Automation introduce the CI representation of the managed node

in the RTSM. Depending on used Smart Plug-Ins or Management Packs, additional more specific CI information will be gathered through discovery and be present in the OBM RTSM.

In general, this CI information already provides a CI model and is the basis for more value-adding functions that can be built in the next phase.

Manual configuration in the RTSM allows the creation of CI collections to supply operations with a structured view.

Mapping Events to Cls

OBM maps incoming events to the CIs contained in the RTSM. Depending on the CI information available in the RTSM, the mapping will be performed on a node CI-type level. If an event source supplies more CI-related information and the topology synchronization matches this specific CI model, a more specific mapping of the events to the CI information occurs.

For the event sources that supply the source node information with the event but do not have the related CI in the RTSM at this point, you can consider using the automatic node generation option. This option populates the RTSM with node CIs based on the incoming events.

It is recommended to establish a basic model of nodes and running software in the RTSM as early as possible. Other RTSM-related activities can be performed later.

CI resolution

As events do not normally carry a detailed RTSM identifier of the related CI, OBM implements a resolution algorithm that maps an event to the closest possible CI. To perform the CI resolution efficiently, a CI resolution cache is used for caching CIs relevant for the event resolution.

<u>Note</u>: See the <u>Micro Focus Documentation Portal</u> for additional information about the CI Resolution algorithm and the CI resolver cache.

The CI event resolution can be controlled by supplying additional information with the event that has a special meaning for the resolution: CI resolution hints. If the algorithm cannot identify the detailed CI for an event, the node CI is used as a fallback. It is important to ensure that the CIs for relevant events are resolvable. This is achieved by tuning of monitoring policies and templates to extend the current monitoring and supply CI resolution hints with the events.

Within OBM, CIs are associated with a set of indicators, which are stored in a repository and defined on a CItype level. Incoming events are analyzed to determine the indicator status. Indicators are used to support event processing functions, abstract the event information to determine the health state of a CI and set the CI's KPIs. Three kinds of indicators exist: Event Type Indicator (ETI), Health Indicator (HI) and Key Performance Indicator (KPI).

Micro Focus recommends performing the indicator tuning in the next phase.

Populate RTSM topology with CIs and CI relations

At this point, it is important to supply a basic model of the managed environment CIs. The basic model covers a node representation of the elements that are part of the environment. In addition, this basic model may cover relations to the running software, e.g. a database instance.

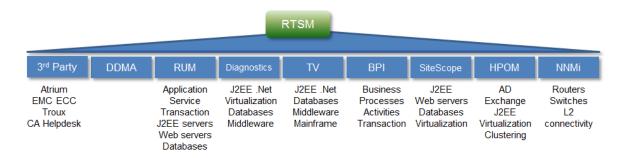
Cls can be created using available interfaces or through discovery. This is explained later in this section.

For more information on this topic, see the following documents:

- The RTSM Best Practices document
- The Effective Modeling for BSM 9 document
- Dynamic Environments and Creating Nodes from Events on the Micro Focus Documentation Portal

The CI and CI relation data can be maintained in the RTSM topology manually. However, as the RTSM has to accurately reflect the CIs and CI relations in the scope of the Operations Bridge, maintaining the changes manually is not efficient due to the amount of changes occurring in daily operations. Therefore, Micro Focus recommends to create and maintain the RTSM model automatically.

Figure 7: Sources for RTSM Topology



Information on managed objects existing in management solutions, such as Operations Manager and NNMi, can be integrated as well.

While some solutions, such as RUM, create CI instances together with relations across domains, other monitoring sources and solutions typically focus on single aspects and certain monitored elements. For example, the relationship between cross-domain CIs is often not established. The RTSM provides mechanisms that allow creating relationships between CIs automatically, e.g. through RTSM enrichment rules.

To feed the CI information into the RTSM, it is recommended to use the Operations Connector (OpsCx) in combination with a discovery policy.

OpsCx provides a set of out-of-the-box topology scripts for a basic set of CIs. If increased flexibility is required, custom topology scripts allow importing the CI information into the RTSM thus offering more control over discovery and mapping of the discovered information.

<u>Note:</u> Refer to the <u>Operations Connector documentation</u> on the Micro Focus Documentation Portal for more information on this topic.

As an alternative, a UD-based discovery solution can be integrated to perform a comprehensive discovery of the domain in a flexible and extensible manner. For this setup, consider implementing a central CMDB that will act as a consolidation point for the CI-related information. The central CMDB is more flexible in supporting change management processes.

Integration of an existing CMDB (or other RTSM instances) with the RTSM is possible at any time and is recommended.

Table 1: Integrating the CI information – Recommendations

| Integration of the external CI Information | Recommendation for the RTSM integration |
|---|---|
| Products NNMi, OM Linux, OM Windows, RUM, SiteScope, BPI | Use out-of-the-box topology integration. |
| Element managers via connector (e.g. Microsoft SCOM, Nagios, Oracle Enterprise Manager, Zenoss, IBM Tivoli, BMC Impact Manager) | Use the topology discovery and synchronization supplied with the Operations Connector. |
| Creating Node CIs from Events | Use the OBM capability to create node CIs based on incoming events for the instant use of OBM. |
| Custom element manager integration | Use the Operations Connector capabilities: out- of-the-box topology script or custom topology script. |
| Custom CI-related information from a set of defined sources | Use the Operations Connector for custom data sources. |
| Flexible and/or extensive discovery to be performed to gather the CI information. | Implement a separate CMDB instance and use UD to perform a discovery. Integrate the RTSM with CMDB. |

Derived Values and Benefits

Event consolidation into OBM provides a number of benefits enabling you to:

- Use the event perspective on the OBM application to gain an overall overview.
- Use event filters to analyze and focus on a certain subset of events. Predefined event filters can be supplied upfront; custom event filters can be saved for future reuse. Events can be searched and compared in an efficient manner.
- Use the event dashboard to design and implement predefined views on the events.
- Use event related KPIs (unassigned, unresolved) for the event filtering.
- Use automatic event assignment rules to increase operator efficiency.
- Use tools and actions to provide the right set of utilities to your operators. The out-of-the-box range of tools can easily be extended.

The example below shows an event dashboard. The upper part of the dashboard contains the elements allowing a quick and easy filtering of the events according to the category, event assignment, resolution state, and event age. Clicking an element performs filtering of the events; the result of the filtering is shown in the event browser on the bottom of the dashboard, from where all event data can be accessed. Note that event dashboards offer flexible layout capabilities and are easily customizable.

| xamp | le: Ope | ratior | ns Ma | nage | r Ove | ervie | w | | | | | | | | | b <i>F</i> | 0 « |
|------------|---------------|----------|--------|-------------|-------------|-------|-----|--|---|------|--|----------------|--|---------------------|----------------|--|----------|
| Jnass | signed I | High | Priori | ty | | | | | Unassigned Even | nts | Unre | solved > 1 | hour | Unresolved > | 3 hours | | |
| | | | | | Crit Maj | | | | | | 2 Critical 2 Major 6 Minor 1 Waming 2 Normal | | 2 Critical 2 Major 6 Minor 1 Waming 2 Normal | | | 2 Critical 2 Major 6 Minor 1 Waming 2 Normal | |
| |) MS | Activ | e Dir | ectory | | | | 1 | | | 0 Over | all | 2 | Cool Product | | 2 | |
| | MS | Exch | ange | | | | | , | O Oracle I | DB | 0 | | | | 5 | 2 | |
| | MS | SQL | Serv | er | | | | | D Infrastru | uctu | re 8 Even | nt History | | | | | |
| → ; | MS | Lync | | | | | | | O 🎧 OM inte | rnal | | 59.57 1C | 00.02 10.0007 10.00:12 11 | 0.00:17 10:00:22 10 | 9.00.27 10 | 00:32 10:00:37 | 10:00:42 |
| Even | nt Brow | | | | | | | | | | | | | | | | 0 « |
| | 5 1 6. | R | 1 | | • | 2 | 7 🛄 | <no filt<="" td=""><td>er></td><td></td><td> Select an event filter> </td><td></td><td>💌 🕙 🐑 🖇</td><td>Cearch Ev</td><td>vents (Ctrl+F,</td><td></td><td>9</td></no> | er> | | Select an event filter> | | 💌 🕙 🐑 🖇 | Cearch Ev | vents (Ctrl+F, | | 9 |
| Sev | Prio | С | N | I. | A | U | D | State | Time Received T | - | ïtle | | Related CI | User | 0 | Group | |
| | ~ | | | - | | | | - | 2/4/14 02:26:51 PM | | utomatic archiving of closed events has com | | | | | | |
| 7 | A | | | <i>(</i> 2) | | | | | 2/3/14 09:42:03 AM 1/27/14 01:34:04 PM | - | filesystem or logical disk on oml-db.mambo.n annot connect to Database. | iet is running | | | | | |
| 2 | ▲ ♦ | е. В. | | | | | | | 1/27/14 01:34:04 PM | | annot connect to Database. ode oml-db down | | orderdb (openview) bx002 (oml-db) | | | | |
| 3 | \$ | | | \$ | | | | | 1/23/14 11:47:58 AM | | BSPI-144: DBSPI40-1: Unable to fetch data fr | om table 'V' | | | | | |
| | ~ | | | <i>§</i> | | 8 | | | 1/23/14 11:22:03 AM | | mp daemon process is not running | and a start w | bx002 (oml-db) | | | | |
| 7 | 4 | | | 1 | | 8 | | | 1/23/14 11:21:08 AM | - | eSystem space utilization for Logical FileSys | tem /tmp/akl | | | | | |
| | ▼ | | | 1 | | 8 | | - | 1/23/14 11:21:08 AM | | eSystem space utilization for Logical FileSys | | | | | | |
| 1 | | | | 1 | | 8 | 2 | 5 | 1/23/14 11:21:03 AM | T | ne NFS server related process rpc.mountd is | not running | bx002 (oml-db) | | | | |
| <u>A</u> | ∇ | | | | | | | | | | | | | | | | |

Figure 8: Example: Event Dashboard with Quick Event Filtering Capabilities

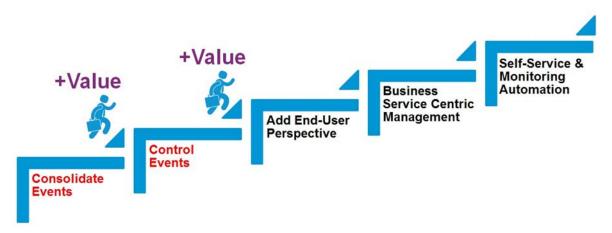
Step 2. Control Events

The Control Events step summarizes the activities that elaborate on the amount, meaning and significance of the events. OBM supports this with a rich set of high-performance functions acting on the stream of incoming events.

You do not have to apply the activities mentioned in this section all at once. Instead, consider implementing some functions early within a single step and applying other functions as tuning at a later time.

The following functions are applicable with node centric monitoring and do not require much elaboration on the RTSM and its CI information upfront:

- Event qualification
- Event correlation
- Event automation



Event Qualification

A significant amount of tuning is typically performed on the source side of the events by tuning the policies that intercept the event information and form it with its attributes. At this point, the filtering should be adapted in order to avoid unwanted events from appearing in OBM. Also, the events should be reviewed regularly to ensure that they appear with the correct severity and other attributes. OBM provides the following event-centric functions that help to control the events:

- Tuning of monitoring parameters. Monitoring Automation allows the adjustment of monitoring based on parameters.
- Event filtering. If unwanted events occur in OBM, filtering at the source side can be adjusted to suppress such events. If events have inappropriate attributes, event filters at the source side can be adjusted to ensure that events are generated with correct attributes.
- Event suppression. To ensure the availability of the OBM solution in case of erroneous or excessive event generation, the event storm detection facility can be used to identify such situations and take mitigation measures to protect OBM. The event storm detection enables the definition of exceptions.
- Event enrichment and processing through EPI. To provide high-quality meaningful events, it is possible to enrich events with the additional information through the event-processing interface (EPI). Event enrichment can be performed at different steps during the event processing. Based on a scripting interface, this functionality is highly flexible and efficient. The range of functionality is not limited to enrichment, but also enables performing modifications on events, such as reworking event attributes to normalize the event format.

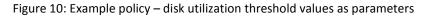
• Event typing.

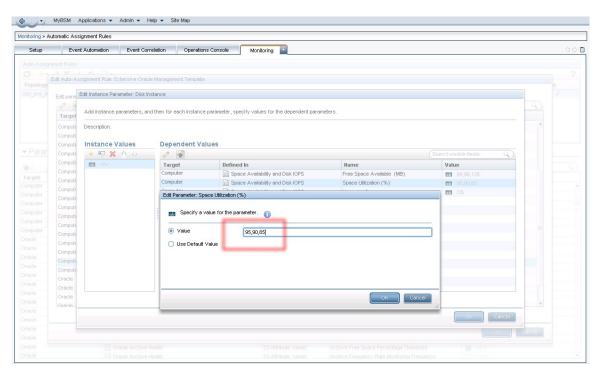
One of the OBM key concepts is event typification. Events are associated with event type indicators and health indicators. Such indicators are driving OBM correlation functions and provide the CI status and health information through key performance indicators.

Monitoring Parameter Tuning

If Monitoring Automation is used, the monitoring provides customizable parameters. Parameters can be adjusted for all OM Agent systems connected to OBM or for a defined subset group of OM Agents.

Tunable parameters are typically thresholds and event attributes, such as event category and event severities.





Event Filtering Through Policies

Event information sources are integrated with OBM as described below.

Monitoring Automation is used to connect systems monitored through an OM Agent to OBM. Monitoring Automation policies define the set of events accepted by the systems. They also determine the events' attributes, such as title and severity. Policy tuning is a regular task performed to ensure that unwanted events are removed from the system and the wanted events are displayed correctly.

When the events are integrated from an underlying OM system, policy tuning works in a similar way; however, it needs to be performed on the OM level.

Other event sources integrated through an Operations Connector are also based on policies, which can be adjusted according to your needs.

The following picture shows an example log file policy with the list of event conditions including the pattern of a particular event.

Figure 11: Example Policy – Oracle Alert Log Policy Conditions

| 🔶 🗸 🔸 MyBSM Appli | ications 🔻 Adr | lmin 🔻 Help 🔻 | Site Map | | |
|-------------------------------|---|---|-------------------------|---|---|
| Monitoring > Policy Templates | | | | | |
| Setup Event Au | tomation i | Event Correlation | n Operations Console Mo | nitoring | 0 ¢ 🕻 |
| | I Oracle A I 10 I Oracle C I Syr Alb I Syr I Sy I Sy I Sy I Sy I Sy | P ★ ★ P Mert Log Template Alert Log Template CRS Alert Log Template Oracle Alert Log Properties Source Defaults Rules Options | | sions exceeded sions licenses exceeded emory while obtaining enqueue while waiting for resource | Search Rules Rule Type Event on matched rule Event on commeter Frequency of Ora Image: State of the state of t |
| | | | | | OK Apply Cancel Help |

Event Suppression

You can define event suppression rules based on event filters. The events matching one of these rules are rejected from further processing.

The following screenshot shows an event suppression rule based on a simple event filter, which matches the events containing the word test in the event title.

Event suppression rules can be enabled or disabled at any time.

Figure 12: Event Suppression Based on Event Filter

| etup Event Automation Event Correlation Operations Console Mon | itoring | | | |
|---|--|---|---|--|
| | | | | |
| | 1 | | | |
| Create Event Suppression Rule | Create New Event Filter | | | |
| Event Suppression Rule Drops all events matching the assigned fiter. Dropped events cannot be recovered. | Display Name: match to Description: match to | est in title | | |
| Display Name suppress event containing test | General Dates A | dditional Event Properties | | |
| Descriptor: | Severity | Lifecycle State | Priority | |
| + Files | V Major Minor Minor Marning Normal | ✓ ^A/₂ in Progress ✓ ^B/₂ Resolved ✓ ^B/₂ Closed | ✓ ▲ High ✓ ♦ Medium ✓ ♥ Low ✓ ℧ Lowest | |
| (+) Required feet | 🗹 🕥 Unknown | All events 🔘 All top level e | Mone None | |
| Activate Rule after creation | - | | | |
| | | ntains 💌 🚺 | est | |
| | Description | iala 💌 | | |
| | Category eq | zals 💌 | | |
| | Subcategory eq | uals v | | |
| | □ Туре eq | zālā 💌 | | |
| | | | Convert to Advanced | |
| | (*) Required field | | | |
| | | GK | Cancel Help | |

Event typing - Event Type Indicators

An event type indicator (ETI) is an event attribute supporting event processing functions defined on a CI-type level and set on incoming events. ETIs are used to typify similar events across various event sources. It is a clear and concise indicator of the event's nature.

ETIs are the basis for performing event correlations, such as event de-duplication, pairwise correlation and topology-based event correlation.

ETI resolution is done in OBM on incoming events. The resolution process is performed based on hints contained in the events. In addition to a hint-driven resolution, indicator mapping rules can be configured to set the ETI on certain events or even to override the hints provided with the incoming events.

OBM comes with a large set of predefined ETIs. Those can be leveraged and extended if needed; however, they should not be deleted or modified without understanding the impact of the change to potentially affected health indicators.

Indicators are configured under Admin -> Operations Management in the Monitoring section. They mainly consist of a name and a set of possible states defined by the display name, status and a status icon. Event type indicators are shown together with health indicators and can be distinguished by an icon. An ETI can be converted into an HI.

The following figure shows a sample ETI available for the Unix CI type. This ETI is inherited from the Computer CI Type and has the name BatchJobService. Its status is Available or Unavailable, associated with Normal and Major criticality.

Figure 13: Event Type Indicator Example

| etup Event Automation Event Corre | lation Operations Console Monitoring | | | |
|---|---|---|--|------------------------------------|
| Types | CI Type - Unix | | | |
| Stebel Application Stebel Component Stebel Component Stebel Component Stebel Component Stebel Web Server Extension Of TheorResource Of Web Server Nithal Host Of Web Server Netsource DeplicationSystem Of CommunicationEndpoint Of Component Resource Of Otherwer Endsystem Note Note Note Note Note Note Note Note | Indicators | Type: Event | lobService Type Indicator Jiffy of Betch Job Service (UNIX/Linux Cror III | n, Windows Schedule Task Service ¥ |
| Cluster Resource Group Computer HP KonSop HP KonSop UDM Server HP KonSop UDM Server Mainframe Cycla Partis CopenVMS CopenVMS HMmainframe CSX Server HMmainf Server | Event Loging Service (Computer) Event Sorm (Node) File Server Service (Computer) FilesystemUsage Firewall Service (Computer) FilesystemUsage Firewall Service (Computer) Hot Disk Wildsath (Computer) Hot Disk Wildsath (Computer) Hot Disk Wildsath (Computer) Hetriace Error Rate (Computer) Hetriace Utilization (Computer) Hot Disk Wildsath (Computer) Legacy Disk (Computer) Legacy Disk (Computer) Legacy Disk (Computer) Legacy Vietwork (Computer) Legacy Vietwork (Computer) Legacy Vietwork (Computer) Memory Usage Level (Computer) Memory Usage Level (Computer) | Display Name Available Unavailable ETI Mapping Rules C Display Name 		 1 | Status Normal Major | Icon |

Event Typing - Health Indicators

A Health Indicator (HI) is similar to an ETI. It reflects the health of a specific operational characteristic of a CI and sets the status of a specific aspect of a monitored CI.

An HI impacts the propagation and calculation of Key Performance Indicators (KPIs) in the Service Health and Service Level Management components. Metric-based HI instances are also available, which are set via the metric information gathered by data collectors, such as SiteScope, BPI, etc.

One of HI examples is CPU Load. Its values can be Normal, Constrained, Overloaded, Bottlenecked, or Busy.

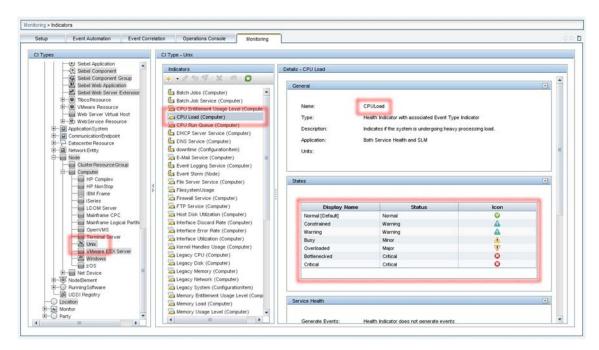
Note: Service Health will be covered later in this document. It is not necessary to focus on HIs at this stage, although they may already be present and functional, e.g. supplied by the out-of-the-box Management Packs.

Like ETIs, HIs can be used in OBM functions, such as event de-duplication and pairwise correlation. The value of an event's HI is resolved on the OBM Server. The resolution process is influenced by the hints contained in the event. An ETI can be converted into an HI.

A rich set of pre-defined HIs is supplied with OBM out of the box. Those can be leveraged and extended if needed; however, they should not be deleted or modified without understanding the impact of the change to potentially affected HIs.

The following figure shows a sample HI named CPULoad, available for the Unix CI type. This HI is inherited from the Computer CI Type. Its status can be set to Normal, Constrained, Warning, Busy, Overloaded, Bottlenecked, or Critical. These states correspond to severity levels from Normal to Critical as shown in the figure.

Figure 14: Health Indicator Details



Event Typing - Key Performance Indicators

KPIs are high-level indicators of the CI performance and availability. A KPI is related to a CI type and defined in the KPI repository. Its status is calculated by HIs using business rules. In addition, KPIs are influenced by the KPIs of child CIs. They represent a high-level aggregated view on the CI status.

A set of pre-defined KPIs, such as performance and availability KPIs, is supplied with OBM out of the box. Additional KPIs aggregate information about the events present for a specific CI (or its child CIs): Unresolved Events KPI and Unassigned Events KPI. The OBM user interface displays the KPI status and status trend information in the health view with colored icons tagged to the related CIs.

<u>Note</u>: KPIs will be covered later in this document. It is not necessary to focus on KPIs at this stage, although they may already be present and functional, e.g. supplied by the out-of-the-box Management Packs.

Event Enrichment and Custom Processing Through EPIs

Event processing customization enables you to implement custom script-based event processing directly on events. This is possible at four different processing stages: before the CI/ETI resolution, after the CI/ETI resolution, before storing the event in the database and after storing the event.

The range of events fed into the custom event processing can be controlled by specifying event filters. Different scripts can be enabled or disabled during runtime.

The script-based event processing logic has to be supplied as a groovy-based script. A number of sample scripts is available.

Figure 15: Sample EPI Script to Modify Event Attributes

```
import java.util.Date;
import java.util.List;
import com.hp.opr.api.scripting.Action;
import com.hp.opr.api.scripting.Event;
import com.hp.opr.api.scripting.EventActionFlag;
import com.hp.opr.api.scripting.LifecycleState;
import com.hp.opr.api.scripting.MatchInfo;
import com.hp.opr.api.scripting.NodeInfo;
import com.hp.opr.api.scripting.PolicyType;
import com.hp.opr.api.scripting.Priority;
import com.hp.opr.api.scripting.ResolutionHints;
import com.hp.opr.api.scripting.Severity;
\ast This example set all possible event attribute to some example values.
class SimpleExample
ł
 def init()
  {
  }
 def destroy()
  {
  }
 def process(List<Event> events)
  ł
      events.each {
      event -> modifyEvent(event);
  }
}
        def modifyEvent(Event event)
        {
    String application = event.getApplication();
    event.setApplication("Modified by EPI: " + application);
    long groupId = event.getAssignedGroupId();
    event.setAssignedGroupId(groupId);
    int assignedUserId = event.getAssignedUserId();
   event.setAssignedUserId(assignedUserId);
    Action autoAction = createSampleAction();
    event.setAutoAction(autoAction);
   ResolutionHints hints = createSampleResolutionHints();
    event.setNodeHints(hints);
    String ciInfo = event.getRelatedCiHint();
    event.setRelatedCiHint("Modified by EPI: " + ciInfo);
        }
 def ResolutionHints createSampleResolutionHints()
   ResolutionHints hints = new ResolutionHints(false);
    hints.setCoreId("CoreId");
    hints.setDnsName("mydqdn.com");
   hints.setHint("My Hint");
```

```
hints.setIpAddress("0.0.0.0");
return hints;
}

def Action createSampleAction()
{
    NodeInfo actionNodeInfo = new NodeInfo(false);
    Action action = new Action(false);
    actionNodeInfo.setCoreId("CoreId");
    actionNodeInfo.setIpAddress("0.0.0.0");
    action.setCall("Call");
    action.setCall("Call");
    action.setStatus(EventActionFlag.AVAILABLE);
    return action;
  }
}
```

The following figure shows the configuration dialog where EPI scripts are specified.

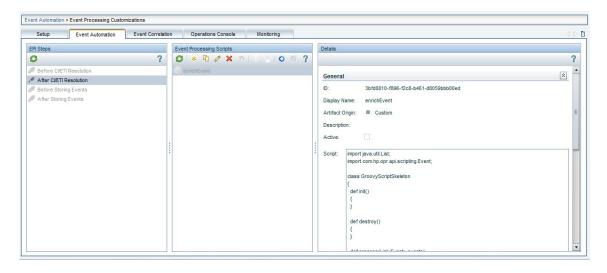


Figure 16: Sample EPI customization as displayed in OBM

Event Correlation Functions

OBM offers a rich set of event correlation functions running centrally on the OBM server. They help to reduce the amount and improve the quality of events.

Event correlation functions relevant to this phase include:

- Duplicate event suppression. It is possible to detect duplicate occurrences of events and suppress such duplicates based on the event attributes, such as the event key and others. Multiple occurrences are counted.
- Pairwise correlation. For the events representing status information, OBM allows pairwise correlation of such events to display only the latest status information. At this stage, pairwise correlation of events based on event keys is available.
- Stream-based event correlation. Stream-based event correlation capability of OBM addresses a number of common correlation scenarios:

o Suppression of repetitive events

Frequently re-occurring events can be suppressed and eliminated from further processing. This saves processing bandwidth as no duplicates have to be counted. Suppression is possible based on the rate and time interval.

• Check for missing events

If the absence of an event within a given timeframe represents an incident (example: "Backup completed successfully" is missing for more than one day), OBM can generate an alert for such a situation.

• Combination of events

OBM can combine a set of various events to raise a more meaningful alarm. This is useful if a certain event pattern stands for a specific cause. In this case, suppression of the originating events is optional.

- Topology-based event correlation. Using the RTSM topology information, OBM can automatically determine relationships between events and identify causal and symptom events.
- Event storm suppression

Duplicate Event Suppression

Duplicate events can be identified in OBM based on different event attributes. Like in classic OM, duplicate event suppression can be based on the event key; however, in OBM, it can also be based on a configurable selection of event attributes. In addition, a specific event type indicator can be used to detect duplicate events (it can be configured in the Settings Manager). OBM allows to control its behavior.

Figure 17: Duplicate Event Suppression – Configuration Settings

| p and Maintenance Report Scheduling Location | s Content Packs Users and Permissions Recipients Downtime Management Event Management | | | 9 |
|--|---|-------|---|---|
| astructure Settings Manager | | | | |
| ect Context: | | | | |
| Applications Operations Management | • | | | |
| oundations Alerting | | | | |
| 4 | | | | |
| Operations Management - Duplicate Events Suppression Set | ings | | | |
| Name | Description | Value | | |
| Detect Duplicate Events by ETI | Use ETIs to find original event. Duplicate events must have the same CI, ETI, and ETI value, and the ETI must contribute to health. | true | 0 | |
| Detect Duplicate Events by Identical Attributes | Use selected attributes to find the original event. All selected attributes must be identical. | false | 0 | |
| Detect Duplicate Events by Key | Use the key attribute to find the original event. Duplicate events must have identical keys. | true | 0 | |
| Enable Duplicate Events Suppression | If enabled, new events that are duplicates of an existing event are not retained and the original event is updated. | true | 0 | |
| Generate history lines for Duplicate Event Suppression | Adds, for each received duplicate event, a history line entry for the original event. | false | 0 | |
| Maximum Age of Duplicate Events | Maximum number of seconds difference between the received times of the original and new event (0 = infinite). | 0 | 0 | |
| Select Application | Duplicate events must have the same application. | true | 0 | |
| Select Category | Duplicate events must have the same category. | true | 0 | |
| Select Cl | Duplicate events must have the same CI. | true | 0 | ŝ |
| Select CI Hint | Duplicate events must have the same CI hint. | true | 0 | |
| Select ETI Hint | Duplicate events must have the same ETI hint. | true | 0 | |
| Select ETI Value | Duplicate events must have the same ETI and ETI value. | true | 0 | |
| Select HPOM Service ID | Duplicate events must have the same HPOM service ID. | true | 0 | |
| Select Node | Duplicate events must have the same node. | true | 0 | |
| Select Node Hint | Duplicate events must have the same node hint. | true | 0 | |
| Select Object | Duplicate events must have the same object. | true | 0 | |
| Select Policy Condition ID | Duplicate events must have the same policy condition ID. | true | 0 | |
| Select Severity | Duplicate events must have the same severity. | true | 0 | |
| Select Subcategory | Duplicate events must have the same subcategory. | true | 000000000000000000000000000000000000000 | |
| Select SubComponentid | Duplicate events must have the same SubComponent ID | true | 0 | |
| Select Title | Duplicate events must have the same title. | true | 0 | |
| Select Type | Duplicate events must have the same type. | true | 0 | |
| Update Severity of Original Event | Update severity of original event based on selected mode. | No | 0 | |
| Update Title of Original Event | Update title of original event with title of last duplicate event. | faise | 0 | |

Event keys and other regular event attributes are typically defined on the event source / policy level. Although they can be manipulated on the OBM server side (e.g. through an EPI script or with indicator mapping rules based on event filters), this approach should be limited to a low number of events.

Pairwise Correlation

OBM can automatically close a subsequent event based on the information contained in the event using pairwise correlation. The identification of the related events is based on the event key or the event type indicator attribute of the event. The behavior can be configured in the Infrastructure Settings.

| p and Maintenance 🔛 Rej | port Scheduling Locations Content Packs Users and Permissions Recipients Downtime Management E | Event Mana | geme 🔇 |
|--|--|--|------------------|
| astructure Settings Manager | | | |
| istructure Settings manager | | | |
| ect Context: | | | |
| pplications Operations Mana | gement 🗸 | | |
| oundations Alerting | Ť | | |
| All | | | |
| AII. | | | |
| Operations Management - Cl | nange State of Related Events Settings | | |
| | | | |
| Name | Description | Value | - |
| Name [≜] Change State | Description Change state of all related events to the selected value. | Value Closed | <i>[</i>] |
| Change State | Change state of all related events to the selected value. Consider old events that have the same key as the incoming event for related event correlation (e.g. close the old event when new event with a | Closed | Ø |
| | Change state of all related events to the selected value. Consider old events that have the same key as the incoming event for related event correlation (e.g. close the old event when new event with a matching close key pattern arrives). Setting this value to false allows the later deduplication step to suppress the new event as a duplicate of the old | | 0 |
| Change State Change state of events | Change state of all related events to the selected value. Consider old events that have the same key as the incoming event for related event correlation (e.g. close the old event when new event with a matching close key pattern arrives). Setting this value to false allows the later deduplication step to suppress the new event as a duplicate of the old one. | Closed true | 0 |
| Change State Change state of events having the same key Detected Related Events by ETI | Change state of all related events to the selected value. Consider old events that have the same key as the incoming event for related event correlation (e.g. close the old event when new event with a matching close key pattern arrives). Setting this value to false allows the later deduplication step to suppress the new event as a duplicate of the old | Closed | 0 |
| Change State Change state of events having the same key Detected Related Events by ETI Detected Related Events by | Change state of all related events to the selected value. Consider old events that have the same key as the incoming event for related event correlation (e.g. close the old event when new event with a matching close key pattern arrives). Setting this value to false allows the later deduplication step to suppress the new event as a duplicate of the old one. | Closed true | 0 |
| Change State Change state of events having the same key Detected Related Events by ETI Detected Related Events by Key Matching Pattern Enable Changing State of | Change state of all related events to the selected value. Consider old events that have the same key as the incoming event for related event correlation (e.g. close the old event when new event with a matching close key pattern arrives). Setting this value to false allows the later deduplication step to suppress the new event as a duplicate of the old one. Existing events must have same CI and ETI as new event, but a different ETI value. Only if ETI contributes to health. Key of existing events must match the closeKeyPattern value of the new event. When enabled, for each newly received event, the existing events are inspected to find events related to the new event. The state of any events that | Closed true true true | 0 |
| Change State Change state of events having the same key Detected Related Events by ETI Detected Related Events by Key Matching Pattern | Change state of all related events to the selected value. Consider old events that have the same key as the incoming event for related event correlation (e.g. close the old event when new event with a matching close key pattern arrives). Setting this value to false allows the later deduplication step to suppress the new event as a duplicate of the old one. Existing events must have same CI and ETI as new event, but a different ETI value. Only if ETI contributes to health. Key of existing events must match the closeKeyPattern value of the new event. When enabled, for each newly received event, the existing events are inspected to find events related to the new event. The state of any events that are related to the new event will be changed. | Closed true true true true | 0 0 0 |
| Change State Change state of events having the same key Detected Related Events by ETI Detected Related Events by Key Matching Pattern Enable Changing State of | Change state of all related events to the selected value. Consider old events that have the same key as the incoming event for related event correlation (e.g. close the old event when new event with a matching close key pattern arrives). Setting this value to false allows the later deduplication step to suppress the new event as a duplicate of the old one. Existing events must have same CI and ETI as new event, but a different ETI value. Only if ETI contributes to health. Key of existing events must match the closeKeyPattern value of the new event. When enabled, for each newly received event, the existing events are inspected to find events related to the new event. The state of any events that | Closed true true true true | 0 0 0 0 |
| Change State Change state of events having the same key Detected Related Events by ETI Detected Related Events by Key Matching Pattern Enable Changing State of Related Events | Change state of all related events to the selected value. Consider old events that have the same key as the incoming event for related event correlation (e.g. close the old event when new event with a matching close key pattern arrives). Setting this value to false allows the later deduplication step to suppress the new event as a duplicate of the old one. Existing events must have same Cl and ETI as new event, but a different ETI value. Only if ETI contributes to health. Key of existing events must have same Cl and ETI as new event, but a different ETI value. Only if ETI contributes to health. When enabled, for each newly received event, the existing events are inspected to find events related to the new event. The state of any events that are related to the new event will be changed. When set to true, the events close key patterns evaluate "OR" patterns (using the "]" symbol) in order to allow for alternatives in the pattern. When set | Closed true true true true | 0 0 0 0 |

Figure 18: Pairwise Correlation – Infrastructure Settings

Like in classic OM, an event key in combination with a close event with key pattern can be used to trigger the pairwise correlation.

Figure 19: Pairwise Correlation – Example Event Key and Close Events with Key Event Attributes

| Key: | SI-FileSystemUtilizationMonitor:omI-db.mambo.net:2049:/dev/sda1:/boot:FileSystemUsageLevel:NearCapacity |
|------------------------|---|
| Close Events with Key: | SI-FileSystemUtilizationMonitor:omI-db.mambo.net:2049:/dev/sda1:/boot:<*> |

The event type indicator, which is an event attribute linked to the CI instance, allows the implementation of the pairwise correlation: if an event matches the CI instance and the event type indicator of a previous event, it closes the previous event.

In the example below, the event type indicator File Server Service is set to Unavailable. A consecutive event setting the same indicator to a different value will close this event.

Figure 20: Pairwise Correlation – Example Event Type Indicator

| The NFS server related | process | s rpc.mountd is r | not running | Event Details | | | | | | | |
|------------------------|----------|-------------------|--------------|------------------|-------------------|-------------------|-------------|---------------------|--------------|------------|------|
| General Additiona | l Info | Source Info | Actions | Annotations | Custom Attributes | Related Events | History | Resolver Hints | Instructions | Forwarding | |
| | | | | | | | | | | | ? |
| ID: | fdf1d7 | 7da-8417-71e3- | 1c82-c0a8fe | 140000 | | Related CI: | <u>bx00</u> | 2 (oml-db) [Unix] | | | |
| Severity: | <u> </u> | linor | | | - | Node: | <u>bx00</u> | 2 (oml-db) [Unix] | | | |
| Lifecycle State: | ې 🖓 | Open | | | • | Source CI: | | | | | |
| Priority: | ⊽ ا | .ow | | | - | Time Created: | 1/23 | V14 11:20:33 AM | | | |
| Assigned Group: | | | | | • | Time Received: | 1/23 | /14 11:21:03 AM | | | |
| Assigned User: | | | | | • | Time State Chang | ed: | | | | |
| Category: | OS | | | | | Event Type Indica | tor: File | Server Service: Uni | available | | |
| Subcategory: | | | | | | Duplicate Count: | 2 | | | | |
| Control Transferred: | - | | | | | Туре: | | | | | |
| Title: | The N | IFS server relate | ed process r | pc.mountd is not | running. | | | | | | |
| | | | | | | | | (| ок | Cancel | Help |

Stream-Based Event Correlation: Repetition, Combination and Missing Recurrence

The stream-based event correlation of OBM allows the implementation of various correlation scenarios acting on a range of events potentially occurring within a defined time window. This functionality addresses a rich set of common correlation scenarios:

• Repetition rule

A repetition rule acts on events identified as potential event duplicates. It is defined on an event filter combined with a time window. The events matching the event filter are counted: if a defined threshold is reached, an event action is triggered. The event action can close all events, release the last event (closing the others) or release all events. As an option, a new event can be raised with defined attributes.

• Combination rule

A combination rule combines different events. In order to identify the events, two or more event filters can be configured combined with a time windows. As an option, a certain order of events can be demanded. The action performed once a combination rule matches can be configured as follows: the related (or all but the last) events can be closed or the events can be released with or without modifications. As an example, event modification, events can be marked as cause and symptom events. In addition, it is possible to raise a new event with specific attributes.

• Missing recurrence rule

Some events are expected to occur at a regular interval. As an example, an important positive event would be a "backup successful" event. If such an event does not occur within a given time window, an incident is indicated. The missing recurrence rule allows creating events if an expected event (identified by a filter) does not appear within the time window.

The missing recurrence rule has the option to release, close or discard the events matching the filter.

<u>Note</u>: For further information on the stream-based correlation capabilities and configuration of these scenarios, see <u>Stream-Based Event Correlation</u> on the Micro Focus Documentation Portal.

Topology-Based Event Correlation

OBM allows correlating events based on a topology relationship of the CI instances related to events. The correlation is not limited to relating events to each other; it also identifies and marks Cause and Symptom events allowing the operators to focus on Cause Events for a quick and effective problem resolution. Topology-based event correlation (TBEC) is based on the CI instance and CI relationship information contained in the RTSM. The RTSM consolidates CI instances across different domains potentially managed by different element managers. As an example, the CI instance and CI relationship information can be consolidated from system monitoring, network monitoring, as well as from a DB element manager, such as Oracle Enterprise Manager allowing to identify the events related to each other (although the event might originate from different sources).

The TBEC is built on top of event/health indicators, as well as on the topology information between the CI instances. The following figure shows an out-of-the-box TBEC correlation rule, which is an example of how TBEC is configured.

The TBEC correlation rule Generic::Node >> Node Status >> Database Server Status relates Node Status events, such as "Node Down", "Node Hang" and other node statuses that indicate an unavailable system with the events originating from the database monitoring. In particular, the "Database Server Status Down" event is related to the "Node Status Down" event. The events are specified with the ETI/HI for particular events.

To perform such a correlation, node CI instances and the database instance must be related to each other. The RTSM models this relationship with a Composition relationship. The rule topology is configured using a corresponding topology query.

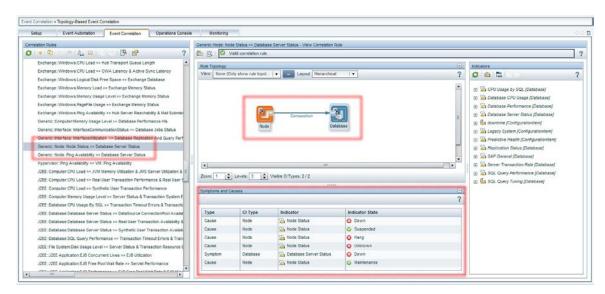


Figure 21: TBEC Example – Relates Node Down and Database Server Down

The events from related CI instances matching the topology query and having the right ETI/HI set are related to each other. The relation allows marking cause and symptom events. This information is displayed in the event and health perspective and can be used to filter the event browser, allowing the users to focus on the important events.

Figure 22: TBEC Example – Node Down Cause Event and Related Database Server Down Symptom Event

| | | .2 | | 9. | 2, | <selec< th=""><th>t a View></th><th></th><th>Select an ever</th><th>nt filter></th><th colspan="3">💌 📖 🕙 🚰 🛛 🧗 😂 🥵 (Search Events (Ctrl+F)</th></selec<> | t a View> | | Select an ever | nt filter> | 💌 📖 🕙 🚰 🛛 🧗 😂 🥵 (Search Events (Ctrl+F) | | | |
|-----|------|-----|-------|----|----|--|-----------|---------------------|----------------|---------------------------------------|---|------------------------------|--|---------------|
| Sev | Prio | C | N | 1 | A | U | State | Time Received | | Title | | | | Related C |
| 0 | | 0.0 | | | | | 5 | 1/27/14 01:34:04 PM | | Cannot connect to Database. | | | | orderdb (op |
| 4 | • | • | | | | | | 1/27/14 01:33:12 PM | | Node omi-db down. | | | | bx002 (oml- |
| 0 | • | - | Cause | 3 | | | 5 | 1/23/14 11:47:58 AM | 1 | DBSPI-144: DBSPI40-1: Unable to fetcl | ch data from table "V" [ORA-00942: ta | ble or view does not exist]. | (Event Message Text) [Policy: DBSPI-Mest | s orderdb (op |
| A | V | 1 | cause | n. | | 6 | 5 | 1/23/14 11:22:03 AM | | snmp daemon process is not running | | | | bx002 (oml-d |

Event Storm Suppression

The event storm suppression is an OBM feature that constantly monitors the rate of occurring events per node. If the amount of events received from a particular node exceeds a threshold for a given time window, the event storm suppression assumes an abnormal situation and discards the events from the node until the situation stabilizes.

The settings of the event storm suppression, such as thresholds and a time window, are configurable. In addition, it is possible to define the exception rules based on event filters, meaning that events matching the exception rules are not taken into consideration. The event storm suppression raises the events when an event storm starts or ends.

Figure 23: Event Storm Suppression

| | n Suppression | |
|--|--|-----|
| Setup Event A | utomation Event Correlation Operations Console Monitoring | 0.0 |
| ent Storm Suppression | | |
| 0 00 | | ? |
| ✓ General | | |
| Active: 🗸 | | |
| Artifact Origin: 🗆 Pred | Ined | |
| | | |
| Conditions | | |
| | ssion when more than 1000 events are received from the same node within 5 minute(s) ion when less than 100 events are received from the same node under storm conditions within 5 minute(s) | |
| | Change Event Storm Suppression | |
| ✓ Begin Event | | |
| itie: | Condition Exceptions Begin Event End Event | |
| Tì Hint | Specify the number of events that must be received within the analysis period to identify the start of an event storm, and the maximum number of events that may be received within the analysis period to identify the end of an event storm. | |
| leventy: Category: | Number of events to Begin Event Storm Suppression: 1000 | |
| Subcategory: | Number of events to End Event Storm Suppression: 100 | |
| lose Previous End Event | | |
| | * Analyze Time Period: S 💼 minute(s) 💌 | |
| End Event | | |
| itle: Event stor Ti Hint: EventStor | | |
| ieventy: O Norma | | |
| ategory: Internal | | |
| Subcategory: Event Stor | | |
| .og Only: - | | |
| | | |
| | | |
| | (*) Required field | |
| | ОК Салсе Нер | |
| | | |
| | | |

Event Automation

This section describes the OBM features related to event automation:

• Automatic and operator-initiated actions. Events can be related with defined corresponding actions. Such actions can be performed automatically and unattended or can be configured to require an operator initiation. Results of actions are typically stored in the event context as an annotation. A target can be any system running an OM Agent. In case of using Operations Orchestration integration, actions can trigger OO run books.

- Automatic user assignments
- Time-based event automation. A set of optional time-based event automation functions can be used to further tune the event processing. Examples of such functions include increasing an event priority after a certain period of time or automatic-time based restarting of failed actions.
- Automatic or manual forwarding of events to external systems. OBM allows the setup of flexible event forwarding rules to different targets, such as a trouble ticket system or a notification solution.

If an environment contains multiple instances of OBM or OM systems, it is possible to set up flexible and hierarchical event forwarding to these systems based on event filters. This is also known as the Manager-of-Managers concept.

Automatic and Operator-Initiated Actions

Automatic and operator-initiated actions related to events are typically defined on a policy level. If Monitoring Automation is used, the policies can be easily reworked within OBM in order to define automatic or operator-initiated actions for the events.

The policies for the events integrated via the OM or Operations Connector can be modified accordingly in the OM/Operations Connector. As an alternative, an event-processing interface (EPI) script can be used to set automatic or operator-initiated actions within OBM.

Automatic User Assignments

OBM allows performing automatic assignments of events to user groups. The events to be assigned are defined by an event filter. The event can be adjusted in a flexible manner to ensure the right set of events matches the filter. Automatic user assignment is initiated as soon as events arrive in OBM.

Time-Based Event Automation

Time-based event automation enables to configure the actions to be executed after a defined time. The set of events relevant for the event automation is defined using an event filter. Actions performed on events matching the event filter cover:

- Re-run of an automatic action associated with the event
- Modification of event attributes
- Forwarding events to external systems
- Assigning events to a user group
- Run of groovy scripts
- Launch of Operations Orchestration run books

The following example shows a time-based event automation scenario, which increases the severity of an open event after 1 hour.

| Event Automation > Time-Based Event Automation | | |
|--|--|----|
| Setup Event Automation Event Correlat | on Operations Console Monitoring | 00 |
| Time-Based Automation Rules | Increase Severity And Priority - Details | |
| 😏 * 🗅 🖉 🗙 🔊 🗘 🚿 🍞 💇 📍 | | ? |
| Close Old Events | General | * |
| B Forward To Trouble Ticket System | Display Name: Increase Severity And Priority | |
| D Increase Severity And Priority | Description: Increase the severity and priority of open events after 1 hour. | |
| B Restart Failed Automatic Action | Active: false | |
| | Artifact Origin: Custom | |
| | | \$ |
| | Conditions | |
| | Event Filter: Open | |
| | Execute, if filter matched for 1 hour(s). Unlimited execution. | |
| | Omminiced execution. | |
| | Actions | * |
| | 🖗 Modify event | * |
| | Modifies event attributes for the matching event | |
| | Attributes: Severity: Increase value | |
| | Priority: Increase value | |
| | | |
| | | |

Figure 24: Time-Based Event Automation Example – Increase Event Severity and Priority after 1 Hour

Event Forwarding and Other Functions

OBM allows to forward events matching configurable filters to target servers, including other OBM instances or a trouble ticket system. Target systems must be configured as connected servers. For the generic event forwarding, a custom external event processing can be triggered via script or web service.

<u>Note:</u> For further information, see <u>Event Forwarding</u> and <u>Connected Servers</u> on the Micro Focus Documentation Portal.

Other Activities

During this phase, Micro Focus recommends to regularly review the CI model contained in the RTSM. Typically, the event typing will identify gaps in the RTSM CI model and missing relations between CI instances. Plan and implement the generation of CI instances and establish relations as described in RTSM Considerations. This will result in more specific mapping of the events and, through a model of the CI relations, in assessing the impact of events cross-domain wise.

This model can also be used to implement more sophisticated event-related functions, such as topologybased event correlation (TBEC). From the operators' perspective, this step enables providing meaningful events and cross-domain views on CIs.

Derived Values and Benefits

At this stage, an operator works with the event perspective, which provides a rich set of functions based on the events. It can be used to identify situations to act upon, analyze the situation in more detail and remediate it.

- Use the tuning capabilities on policies to provide important high-quality events to your operators.
- Use correlation functions mentioned previously to reduce the amount of events and enrich the event information.
- Supply CI collections with grouped CIs to provide the operations with the views on managed objects, allowing them to filter the events and supply the additional information through the CI collection.
- Extend the use of automatic or operator-initiated actions on events.

- Offer performance graphs of a related system to allow operators an enhanced drilldown when analyzing a problem.
- If available, supply problem remediation capabilities using the Operations Orchestration integration.
- Supply instruction texts to the operators in order to solve problems quicker.
- Provide a problem escalation path by integrating OBM with a trouble ticket system allowing the operators to raise tickets upon an event and transfer the problem resolution to the process implemented within the trouble ticket system.
- Use TBEC to classify events as Cause and Symptom Events allowing the operators to focus on important Cause Events.
- Provide filters based on Cause Events to focus operators on these events.

The following figure shows an example event perspective. In the View explorer, an out-of-the-box view shows node CI instances derived from OM node groups, which can be used to filter the events.

The event browser shows the event details when selecting an event. Available tools are shown on the Actions pane on the right-hand side.

| Event Perspective × Health Perspective × | | | | | | | | | Select Page |) 😅 🖗 🔠 🏠 🖓 🗷 😌 関 |
|--|---------|---------|-------|-----------|--------|----------|-----------|---------------------|---|---------------------------------|
| View Explorer 🗈 🖉 🗠 ک | < 🕭 Eve | ent Bro | wser | for linu | х | | | | b⊮ ∂ « × | Actions 🗈 🖋 🗗 « × |
| Browse Views Search | Π., | 12 💐 | | 1 🦛 | | • | 🎝 📜 | OM CI Collecti 💌 | ielect an event 💌 📖 🚫 🛃 🎉 🥵 Sear | O ? |
| 6 B | Sev | Prio | С | NI | A | U | D Sta | Time Received | Title | Event: snmp daemon process is n |
| OM CI Collections | 4 | 4 | ۹., | | | | ۵. | 1/27/14 01:33:12 PM | Node oml-db down. | CI: Ix002 (oml-db) |
| CI-VCS SLES Nodes | | V | | 1 | | 2 | ۵. | 1/23/14 11:22:03 AM | snmp daemon process is not running | Node: bx002 (oml-db) |
| CI-VCS Windows Nodes | | | | 1 | | 8 2 | L. | 1/23/14 11:21:03 AM | The NFS server related process rpc.mountd is not running. | Search: Search Actions |
| CI-VCS-Solaris Nodes | | ▼ | | 1 | | 8 | Π., | 1/23/14 11:20:54 AM | Sendmail daemon process is not running. | |
| ESX and ESXi Host Outage | | ▼ | | 1 | | 8 | Π., | 1/23/14 11:20:44 AM | DNS daemon is not running. | Filter: 💿 All 🔿 Cl 🔾 Source |
| | | ▼ | | 1 | | 8 | Π., | 1/23/14 11:19:54 AM | dhcp daemon process is not running. | Sispi-ovperl ExecuteService.pl |
| 2 kx001 (bsm-gw2) | 0 | 윰 | | 1 | | | 5 | 1/23/14 11:15:21 AM | Open Agent Service terminated. (OpC40-471) | a Show Performance Graphs (|
| 2 k002 (oml-db) | | | | | | | | | | : 🏡 Show Performance Graphs (|
| net_devices | 4 | | | | | | | | | T Disable Oracle MP Monitoring |
| OMU Administration UI Servers | | 7 of 11 | (1) | | Q. | linux | (| 30 🐨 0 🗥 5 🛕 | 1 🕥 1 🞯 0 🔒 0 🌆 0 🔊 0 🎧 7 💂 | |
| OMU Administration UI WebApps | | | | | | | | ***** | | T Disable Oracle MP Trace |
| SI-Deployment | snmp | daemon | proce | ss is no | t runn | ning - E | vent Deta | ils | | T Display Oracle MP Error File |
| SI-HPUX | Ger | neral | Addit | ional Inf | | Source | Info | Actions Annotations | Custom Attributes Related Events History Ret | T Enable Oracle MP Monitoring |
| SI-IBM AIX | B | L | 71001 | | | 000100 | | | | T Enable Oracle MP Trace |
| SI-MS Windows | | 22 | | 122 | | | 2000 | | | T Ping node from NNMi server |
| SI-RHEL | ID: | | | | | | -71e3-1c | 83-c0a8fe140000 | Related CI: bx002 (oml-db) [U | |
| SI-SLES | Ser | verity: | | | 💧 Min | | | | ▼ Node: <u>bx002 (oml-db) [U</u> | T Ping node from NNMi server (|
| SI-Solaris | Life | cycle S | tate: | E | 👌 Op | en | | | Source Cl: | TRun Self-Healing Collector for |
| SiteScope 🔻 | | | | | | | | | • | |

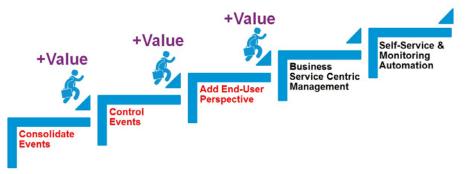
Figure 25: Example Event Perspective

Step 3: Add End-User Perspective

At this stage, it is recommended to complement the infrastructure monitoring with the status information gained through probing the end-user perspective on services and applications. The BPM and RUM solutions can be used to achieve such probing-based monitoring.

The health status gathered through probing provides the meaningful status information, while the event information gained through the infrastructure monitoring complements the status with the technical information on the cause of the incident.

Figure 26: Third Step – Adding End-User Perspective and Moving on to CI-Centric Monitoring



Complement Infrastructure Monitoring with End-User Perspective

Services and applications provide their functions through defined interfaces, such as web-based protocols (web services, http, soap, ftp and many more). Consumers of services and applications are the end users, as well as other systems.

BPM allows active probing of services and their transactions from probe systems. It measures the availability of the services and transactions and gathers additional metrics about the performance.

RUM performs similar monitoring; however, it is not based on active probing but on listening to the user traffic.

Through the APM-OBM integration, availability and performance KPIs calculated by BPM or RUM are supplied to OBM. This represents the important information from the end-user perspective and complements the events gathered from infrastructure-focused monitoring.

Both infrastructure monitoring and end-user perspective probing are key information for operations as they bring together the current state of services with potential causes and symptoms derived from the infrastructure monitoring. The OBM correlation functions allow correlating these events.

Evolve RTSM

Use RTSM enrichment rules to refine CIs and CI relations in the RTSM as described below.

Use RTSM enrichment rules to add additional CIs and to relate them, as well as to add or update the attribute values of specific CI instances.

Implement views that allow the operators to focus on their relevant area of responsibility. The out-of-thebox configuration provides a number of infrastructure and technology-focused views that can be used as a basis. Custom views can easily be created by copying and adjusting existing views.

Derived Values and Benefits

Adding the end-user perspective to monitoring brings the following benefits:

- It provides the operational staff with the events related to the service availability and performance, enabling them to get notified about service problems instantly.
- It supplies the views allowing the operational staff to use them on event filtering in the event and health perspective. Event filtering is thus extended by the view-based filtering, which allows the operator to quickly focus on the right set of events.

• It provides view mappings enabling the health perspective to display a pre-selected view on related CIs in the event browser.

| rw Explorer D 🚈 🗗 🕬 🗴 | O Ever | nt Brow | ser f | or OR | Deple | yme | nt | | | | | | | | | | | D1 44 × | Actions | * A d | р «с |
|--|----------|---|-------|-------|---------|-------|---------|-----------|--|--------|---|--|-----------|----------------------|---|------------------------|------------|-------------------------------|----------------------|--|------|
| rowse Views Search | G. 1 | 6 8 | 0 | -1 | | | | ORA_ | Deployment | | | lelect an event filter> | | | | Search (Search | h Events | 27 | 0 | | ? |
| 9.6 | Sev | Prio | C | N | | U | D Sta | te 1 | Time Received | Title | | | | | Related CI | | | User | Event N | iode omi-db down. | |
| RA_Deployment | W | * | | 4 | | | 9 | | 5/3/14 09:42:03 AM | A file | A filesystem or logical disk on omi-db mambo net is running ou bx002 (omi-db) | | | | | | | Ct N | x002 (omi-db) | | |
| GRA_Deployment | 0 | 4 | 10 | | | | 5 | | 1/27/14 01:34:04 PM | | | ect to Database. | | | orderdb (openvi | ew) | | | Node: | | |
| Gorderdb (operview) Mo02 (omi-db) Go5056A22821 | | ۲ | | | | | | | 1/27/14 01:33:12 PM | 1.20 | omi-db | | | | bx002 (omi-db) | | _ | | Search: (| Search Actions | - |
| | 0 | ٠ | | 4 | | | | C 1 1 1 1 | 1/23/14 11:47:58 AM | - | | BSPH0-1: Unable to fetch | th data t | from table "V" [O | and the second se | ew) | | | Fiter: @ | | |
| | | v | | 4 | | 8 | 9 | | 1/23/14 11:22:03 AM 1/23/14 11:21:03 AM | | | n process is not running | | | bx002 (oml-db) | | | | | | |
| | | ~ | | 4 | | 60 60 | 2 | | | | The NFS server related process rpc.mountd is not running. br002 (oml-db) | | | | | _ | | Rep Show | Performance Graphs (| (C | |
| | Â | ~ | | 4 | | | 9 | | 1/23/14 11:20:54 AM | | | mail daemon process is not running. bx002 (omi-db) | | | | | | Rep Show Performance Graphs (| (0 | | |
| | 4 | ~ | | 4 | | 8 | 9 | | 1/23/14 11:20:44 AM | | | is not running. | | | bx002 (oml-db) | | _ | | 17 Disable | e Oracle MP Monitoring | (0 |
| | 4 | ~ | | 4 | | 8 | 9 | | 1/23/14 11:19:54 AM | 1000 | | n process is not running. | | | bc002 (omi-db) | | YT Disable | e Oracle MP Trace | (0 | | |
| | 0 | 8 | - | 4 | 0 | | 9 | | /23/14 11:15:21 AM | Oper | Agent | Service terminated. (OpC | 40-471 | <u>}_</u> | bx002 (oml-db) | | _ | _ | VT Display | y Oracle MP Error File | (0 |
| | | | | | | | | | | | | | | | | | | | 11000000 | | (0 |
| | | | | | | | | | | | | | | | | | | | 100 | T Enable Oracle MP Trace T Ping node from NNMI server T Ping node from NNMI server (| |
| | | | _ | - | | | | - | | | | | | | | | | | TT Ping no | | |
| | | | | | | | | | | | | | | | | | | | T Ping no | | |
| | 4 | | | | 0 | | Deplo C | | | | 1.0 | | 00 | A: 0 | 20 0 | 1.4.4 | | | | elf-Healing Collector for | . (0 |
| | items: 1 | 0 01 13 | (1) | | a.A. | ORA | Depio C | 2. | V 1 🛕 | 5 | 4 | 1 01 | 00 | - 10 B | 20 0 | 1 ¹¹ 0 | lig 10 | 8 | | Layer 2 Neighbors to r | |
| | Node o | mi-de de | wit | Event | Details | | | | | | | | | | | | | | | | |
| | - | | | | | | | | | | | | | | Layer 2 Neighbors to r | | | | | | |
| | 1000 | General Additional Info Source Info Actions Annotations Custom Attributes Related Events History Resolver Hints Instructions Forwarding | | | | | | | | | | | -1 | | Layer 3 Neighbors to r | | | | | | |
| | 10 | n o ? | | | | | | | | | | | | | 17 Show | Layer 3 Neighbors to r | (0 | | | | |
| | ID; | D: 4d71c28c-esbb-4087-9053-e4e82ce4583b Related Ct (add2 (cmi-dp) 5Unix) | | | | | | | | | | | | T Show | NNMi console | (0 | | | | | |
| | Sev | Severity: A Warning Node: | | | | | | | | | | 1 | IT Show | NNMi console (https) | (0 | | | | | | |
| | Life | | | | | | | | | | | 5 | IT Show | NNMi server status | (0 | | | | | | |
| | Prior | Priorty: Assigned Group: | | | | | | | | | Time Created 1/27/14 01:33:12 PM | | | | | | | 1.000 | | (0 | |
| | Ass | | | | | | | | | | | Time Received: | | 14 01:33:12 PM | | | | 5 | 1000 | | |
| | 1000 | igned Us | | Ē | | | | | | - | - | | mail. | 14 91:33:12 PM | | | | | | node information in NNM | |
| | | | | F | | | | | | | | Time State Changed: | - | | | | | | T Show | node information in NN | (0 |
| | 10.0 | gory: category | | | | | | | | | _ | Event Type Indicator: | Ping | Availability: Una | vallable | | | | IT Show | related NNM incident | (0) |

Figure 27: CI-Centric Monitoring – Filter by View on Event Perspective

In addition to the event perspective, the health perspective is supplied, where the CI-centric monitoring offers comprehensive health information including relationships and CI status. The event information is complemented with the status information through HIs and KPIs (if already available, e.g. supplied by Management Packs).

Figure 28: CI-Centric Monitoring – Health Perspective with Event-Related Mapped Health Top View and HIs

| New Explorer D / D « » | O Eve | nt Brow | wser | for orde | rdb (op | enview | 1 | | | | | | | | 1 2 4 | | | |
|---|----------------------|--------------|-------|----------|---------|----------|---------------------|---|---|-----------------------------|----------------------------|---|--|--|--|--|--|--|
| Browse Views Search | ц. | 6 R | . 0, | | ۰. | 27 | 0 | RA_Deployment | 💌 📖 🗑 «No F | ter> | × - 😢 i | 3 4 1 | O 🕄 (See | | 9 4) 7 | | | |
| 0 % | Sev | Prio | C | N 1 | A U | D | Sta | Time Received | Title | | | | | | R | | | |
| DRA_Deployment | S. | | - | 0 | _ | | 5 | 2/3/14 09:42:03 AM | A filesystem or logical disk on om- | db mambo net is running out | of space. Usage is current | ly at 100.00 | at 100.00 % [Threshold value: 95] | | | | | |
| GRA_Deployment | 0 | Q A . | | | | 5 | 1/27/14 01:34:04 PM | Cannot connect to Database. | | | | | | or | | | | |
| Conderdb (openview) | 4 | | | | | | | 1/27/14 01:33:12 PM | Node oml-db down. | | | | | | bd | | | |
| | 0 | • | | Ø | | | 5 | 1/23/14 11:47:58 AM | DBSPI-144: DBSPI40-1: Unable to 1 | | -00942: table or view does | s not exist). | (Event Message | Text) (Policy: DB | | | | |
| | ▲ ▼ Ø & 2 G 1/2 | | | | | | 1/23/14 11:22:03 AM | snmp daemon process is not runn | 1. A. | | | | | br | | | | |
| | | | | | | | 1/23/14 11:21:03 AM | The NFS server related process ry | | | | | b | | | | | |
| | | <u>A</u> V Q | | | | | 5 | 1/23/14 11:20:54 AM | Sendmail daemon process is not n | anning. | | | | | be be | | | |
| | | V V | | 0 | 8 | | 000 | 1/23/14 11:20:44 AM | DNS daemon is not running. | | | | | | | | | |
| | 4 | 8 | | 1 | | | | 1/23/14 11:19:54 AM | dhcp daemon process is not running. Open Agent Service terminated. (OpC40-471) | | | | | | 5 | | | |
| | | | | 1 | | | -0 | 1000119111.12001.000 | open Agent Service terminates. (c | (post-171) | | | | | | | | |
| | | | | | | 1 | | | | | | | | | | | | |
| | the second second | 10 of 13 | 3 (1) | | by o | rderdb (| (og 😳 | | A 5 A 1 | Q 1 Q 0 | Ar 0 | da o | 250 | Eg 10 | | | | |
| orden to de continu | d beense | | | | | | | | | | | | | | | | | |
| ealth Top View | | | | | | Dit. | 102.44 | Health Indicators | | | DAGH | Acti | ons | | 2 2 4 | | | |
| All and the second s | | | - | | | D(A | 9.4 | Health Indicators orderdb (openview | | | D # 6 4 | Intel Address | ons | | 10 0 0 | | | |
| Selected View: ORA_Deployment | | 5 | j, | | | D.C | 91 et | orderdb (openview | /) (Oracle) + | | 5 × 6 « | Acti | | | | | | |
| raviture Concesso | | 5 | j, | | | D(2 | 97 ee | orderdb (openview | | | 5+18* | 0 | t Cannot conr | ect to Database. | | | | |
| raviture Concesso | | 5. | j | | | DIA | <u>ि</u> रु | orderdb (openview | /) (Oracle) + | | D A BR | 0 Even | it: Cannot conr orderdb (op | ect to Database. | | | | |
| Selected View: ORA_Deployment | Deplayn | | | | | B(c | 5 ⁷ et | orderdb (openview Health Indicate | /) (Oracle) + | | | Even Ct Node | it. Cannot conr orderdb (op t. | ect to Database. enview) | | | | |
| 1990009-000-000 | Deployn | | | | | D(2 | 27 et | orderdb (openview | /) (Oracle) + | Health Indicators | | Ct Node | t Cannot conr orderdb (op t: ch: (Search Ac | ect to Database. mvlew) | q | | | |
| Selected View: ORA_Deployment | Deployn | | | | | DI A | 9. ec | orderdb (openview Health Indicate |) (Oracle) + ors Contributing to KPIs | Health Indicators | | Ct Node Filter | t: Cannot conr orderdb (op t: cht: (Search Ac : • At () | ect to Database. enview) lons CI () Source C | G O Node | | | |
| Selected View: ORA_Deployment | Besloyn | | | | | D(A | <u>8</u> e | orderdb (openview Health Indicate |) (Oracle) + ors Contributing to KPIs | 2010/01/01/01/01/01 | | Ct Node Filter | t Cannot conr orderdb (op t: ch: (Search Ac | ect to Database. enview) lons CI () Source C | G O Node | | | |
| Selected View: ORA_Deployment | | nent 🗖 | J | | | D(A | <u>8</u> « | orderdb (openview Health Indicate 20 Mpi KPI |) (Oracle) + ors Contributing to KPIs | 2010/01/01/01/01/01 | | Ct Node Sear Fitter | t: Cannot conr orderdb (op t: cht: (Search Ac : • At () | ect to Database. enview) long Cl O Source C ce Graphs (| C Node T Cor | | | |
| Selected View: ORA_Deployment | | nent 🗖 | J | | | - Di A | ी की (| orderdb (openview Health Indicate 20 Mpi KPI |) (Oracle) + ors Contributing to KPIs | 2010/01/01/01/01/01 | | Ct Node Sear Filter | t Cannot conr orderdb (op r. ch: <u>Search Ac</u> r. (e) At () Show Performan | ect to Database. enview) CI O Source C ce Graphs ((C ce Graphs ((C | G O Node T Cor T Dat | | | |
| Selected View: ORA_Deployment | | nent 🗖 | J | | | D) / | 0 e | orderdb (openview Health Indicate 20 Mpi KPI |) (Oracle) + ors Contributing to KPIs | 2010/01/01/01/01/01 | | Ct Node Sear Fitter Do Do TT | t Cannot conv ordersb (op r: (0): (Search Ac r: (8) Al () Show Performan Show Performan Archive Device F | ect to Database. miview) Cons Col O Source C Ce Graphs ((C ce Graphs ((C ree Space (C | G Node T Cor T Dat T Dat | | | |
| Selected View: ORA_Deployment |) (openv | ient 🖬 | J | | | B(c | ð « | orderdb (openview Health Indicate 20 Mpi KPI |) (Oracle) + ors Contributing to KPIs | 2010/01/01/01/01/01 | | Ct Node Sear Fitter D D D T T | t Cannot conv orderdb (op r: ch: <u>Search Ac</u> : @ Al _ Show Performan Show Performan Archive Device F Archive Log Write | ect to Database. miview) loca Cl. O Source C ce Graphs ((C ce Graphs ((C ree Space (C Rate (C | G TT Cor TT Cor TT Dat TT Dat | | | |
| Selected View: ORA_Deployment |) (openv | ient 🖬 | J | | | B(c | 20 et | orderdb (openview Health Indicate 20 Mpi KPI |) (Oracle) + ors Contributing to KPIs | 2010/01/01/01/01/01 | | Ct Node Sear Fitter Do IT IT IT | t. Cannot conr orderdb (op r. ch: <u>Search Ac</u> r. @ All _ Show Performan Archive Device F Archive Log Wrb Beckground Dum | ect to Database. nview) CL Source C ce Graphs ((C ce Graphs ((C red Space (C rate (C c Device Us (C | G Node T Cor T Dat T Dat T Das T Das | | | |
| Selected View: ORA_Deployment | o (openv 2 (omi-a | ient = | J | | | B(c | 20 et | orderdb (openview Health Indicate 20 Mpi KPI |) (Oracle) + ors Contributing to KPIs | 2010/01/01/01/01/01 | | Ct Node Sear Fitter Do IT IT IT | t Cannot conv orderdb (op r: ch: <u>Search Ac</u> : @ Al _ Show Performan Show Performan Archive Device F Archive Log Write | ect to Database. miview) loca Cl. O Source C ce Graphs ((C ce Graphs ((C ree Space (C Rate (C | G Node TT Con TT Dat TT Dat TT Dat | | | |

The health perspective provides information beyond the CI-related to an event and broadens the view by allowing to discover the CI neighborhood and related CIs and their KPI status.

Performance graphs allow a detailed drilldown into a wide range of metrics provided by data collectors, such as Operations Agent, SiteScope, BPM, RUM and more.

Step 4: Business Service Centric Management

This phase of the product implementation deals with transforming OBM from the CI-centric monitoring to a service management level in order to provide meaningful service status and health information. To achieve this, the RTSM model is extended with a model of business services. This relates the event information to business services allowing the operational staff to identify and understand the impact of the events on business services and enable them to focus on the most important ones.

To support the service centric management, the RTSM must be extended by defining a model of business services and applications. Such a model must contain a relationship to all relevant CI instances contributing to the business service.

Unlike the CI-centric monitoring (where CI instances and relationships are maintained through discovery, integrations and automatic processes), the modeling of business services and their relations is a manual procedure. It requires a profound understanding of the service and its dependencies to the contributing service elements. Additionally, the structure of the service presentation is service-specific and needs to consider the views on the business service model as required by the user.

In this context, the details, such as service health status calculation, are specific to the business service and the consumers of the related views. Different users will require specific views on the business service model. As the service status and health information consolidated in OBM is often fed into service reporting and service management, the service model has to support such requirements and processes by supplying the right level of information for these solutions.

Beside the operational perspective on events and their impact on business services, a complementary view can be elaborated providing a view on the service health of particular services, e.g. for displaying this information on a read-only dashboard and for sharing it with management or end users.

Depending on the service model, monitoring and expectations of the users viewing the information, it may be possible to satisfy these requirements with a single service health model. In other cases, it may be required to implement different service models e.g. using a different set of KPIs and business rules to address different requirements.

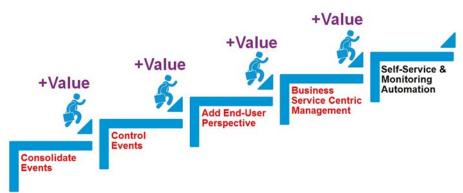


Figure 29: Fourth Step – Moving on to Service-Centric Management

To efficiently calculate the service health information, monitoring of services from the end-user perspective should be implemented.

Service Modeling

Figure 30: Logical Business Service Structure and Monitored Service Contributors

| Business Service – Retail Banking Business Service / LOB |
|--|
| Business Application – Online Bank Business Application |
| Business Transaction Flow – Transfer Business Transaction Flow |
| Business Transaction – To Checking Business Transaction |
| J2EE Application – JavaProg J2EE Application |
| WebLogic Server Server |
| Windows Server Host |
| CPU Network Disk |

Note: For more information on this topic, see:

- Effective Modeling for BSM Best Practices document
- <u>End-to-End Service Monitoring and Event Management Best Practices</u> (provides information on how to deploy and implement end-to-end service monitoring solutions to ensure adherence to the level agreed upon between the service provider and the service consumer)
- Modeling Studio sections on the Micro Focus Documentation Portal: <u>Building a Business View</u>, <u>Business CI Models</u>, <u>How to Build a Business CI Model – Scenario</u>

Example Business Service Model – Overview

This section describes the steps required to build an example business model in the RTSM based on a pattern view.

Business Service Models can be created in different ways, e.g. as an instance-based model. Instance-based models are easy to create and maintain by building a nested structure of business CIs and dragging the related CIs into this structure. However, an instance-based model is static and requires manual maintenance, e.g. if new CI instances relevant to the service model need to be included.

Pattern-based models are defined via a TQL query, which determines the structure and the CIs returned as a result. This leads to a more dynamic service model, meaning that new CI instances fitting into the service model are automatically included if they match the related TQL query. This provides more automation.

The example described in this section uses a pattern-based model. The following steps are shown:

- 1. Designing the structure of the service model
- 2. Creating the structure of the service model based on a pattern view
- 3. Creating CI instances required for the example service model
- 4. Creating relations between the example service model and underlying infrastructure services

As the first step, the structure of the business service model has to be drafted. In the example, a fictional service named "Cool Product" consists of two major entities with different channels, the customer self-service and the branch office application. The customer self-service is elaborated in more detail below.

Workshops held with subject matter experts revealed that the customer self-service has two major work components with different systems and responsibilities involved: the Ordering and the Fulfillment.

The Ordering depends on a number of contributing elements that are essential for the correct function: a frontend, a backend, a database and storage. These elements group the related monitored infrastructure Cls.

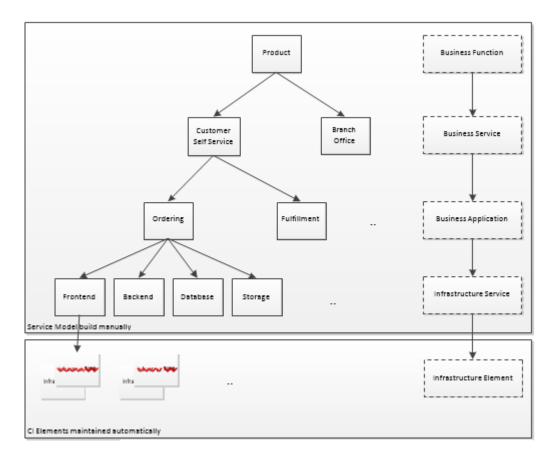


Figure 31: Example Simple Business Model – Structure and CI Types

The elements within the example business service model are built on CI types supplied with the RTSM out of the box. Their mapping is shown in the figure above.

The Name of the CI elements created manually is a textual string. It is recommended to use a naming schema and prefixes for easer CI element search. The text displayed in the UI can be changed easily by setting the display label attribute of the CI for enhanced readability once the service is constructed. In this example, generic CI names are used in the name attribute. CI instances created by discovery are also listed. Those CI instances are about to be linked into the example business service model.

Table 2: Example Business Service – Elements

| Business Service Element | Name | Display Label | СІ Туре | | |
|---------------------------------|---------------------------|--------------------------------|------------------------|--|--|
| Cool Product | aa_SimpleApp | Cool Product | business_function | | |
| Customer Self Service | aa_service1 | Customer Self Service | business_service | | |
| Ordering | aa_service1_app1 | Ordering | business_application | | |
| Frontend | aa_app1_frontend | Frontend | infrastructure_service | | |
| _ | (CI lx001, unix node, al | ready existing through discov | very) | | |
| uses Infrastructure Backend | aa_app1_backend | Backend | infrastructure_servic | | |
| _ | (CI lx002, unix node, al | ready existing through discov | very) | | |
| uses Infrastructure Database | aa_app1_database | Database | infrastructure_service | | |
| | (CI orderdb, oracle inst | ance, already existing throug | h discovery) | | |
| uses Infrastructure Storage | aa_app1_storage | Storage | infrastructure_service | | |
| uses Infrastructure | (Cl orderstorage, file sy | rstem, already existing throug | gh discovery) | | |
| Fulfillment | aa_service1_app2 | Fulfillment | business_application | | |
| | | | | | |
| ranch Office | aa_service2 | Branch Office | business_service | | |
| | | | | | |

Example Business Service Model – Create Pattern-Based View

Next, a pattern-based view is created in order to implement a TQL query matching the structure of the example business service. A new pattern view named "aa_MyService" is created in the Modeling Studio under Admin -> RTSM Administration, Modeling -> Modeling Studio.

Drag and drop the CI types that constitute the simple business view (BusinessFunction, BusinessService, BusinessApplication, InfrastructureService, InfrastructureElement CI types) into the query definition pane. Extend the TQL query by specifying the required relationship between the CI types. Choose the relationship type "Containment" between the Business Elements.

Choose the relationship type "Usage" between the InfrastructureService and the InfrastructureElement CI type.

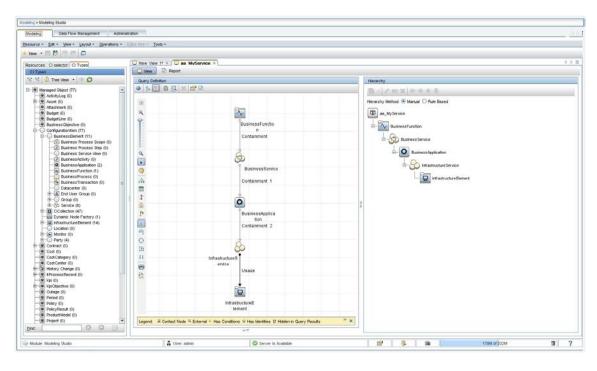
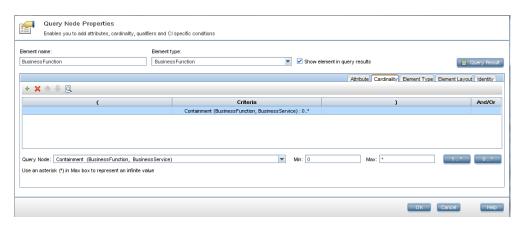


Figure 32: Example Simple Business Model – Create a Pattern-Based View

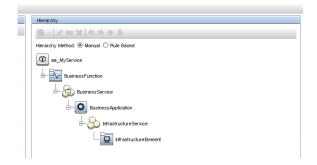
Refine the query node properties for the nodes in the TQL query. For this simple business model example, set the query cardinality for all "Containment" relationships of the query to the type "0..*". Refine the query node property of the "Usage" relationship between InfrastructureService and InfrastructureElement to "1..*".

Figure 33: Example Simple Business Model – Set Relationship Cardinality on the TQL Query



The result of the query representation can be influenced by defining the hierarchy method in the Hierarchy Pane. For the simple business model, a tree-like presentation of the structure is desired. This means that the hierarchy is refined as shown in the following screenshot.

Figure 34: Example Simple Business Model – Hierarchy Layout



After this, the pattern view can be saved under the name aa_MyServices.

Example Business Service Model - Create the CI Instances

Now that the pattern-based view has been created, the CI instances of the simple business view (see Table 2: Example Business Service - Elements) are added manually to the RTSM.

This is done in under Modeling -> IT Universe Manager. Selecting the aa_MyServices view restricts the CI types available for the CI creation, which is recommended.

Figure 35: Example Simple Business Model – Add CIs

| eling > IT Universe Manager Vodeling Data Flow Management Administrativ | | | | | |
|--|--|--|--|-------------------|------------------------------|
| R + View + Layout + Operations + Editor Item + Tool | | | | | |
| CI Selector Erowse Views Search Cis | | | | | View Results Related Cla |
| 9 · · · · · · · · · · · · · · · · · · · | Text | 1. 🗊 🖈 🖛 🔛 😸 🔊 🖸 | | | THE REPORT PRODUCT CON |
| n: aa_MyService 💌 🚥 | | the second se | | | |
| Cool Product - Cool Product - Cool Product D - Cool Product D - Cool Product D - Cool Product D - Cool Product | 9 | New CI | | 55 D | |
| | IT I | Select CI Type | | | |
| | a 💽 | Configuration them Configuration them Do Business theorem Do | ⑧ C1 Types from the currer ○ All C1 Types | et view | |
| | 2 | Define New CI Properties | 44444 | | |
| | * 8 4 | Lutino han C / rooms Pequined properties * Name Specific properties of class BusinessFun BusinessFunctionType Properties interified from class Business Properties interified from class Configurat | Dement | | |
| | s Legend: ±) Adde | | | | × |
| | Cool Product | Name | | 014 3:5228 PM CET | Properties History Discovery |
| | 12 2. i | | Save Cancel | Heb | |
| | Actual Deletion Per Allow CI Update BODY_ICON BusinessCriticality BusinessFunction | | True line_of_business | | 40 |
| d 0 0 | classification | | | | |

Once all CI instances are created, the relationship between the business service instances must be established. This can be performed via the menu item "Insert Relationship", which brings up a dialog that allows lookup/find of CI instances.

The CI instance's relationship in the example needs to be added as a containment relationship as this has been defined within the view.

Figure 36: Example Simple Business Model – Insert Relationship, CI Selection

| Select target CIs | Target CIs |
|---|------------|
| Searches: Simple Advanced CI Name: database CI Type: Node;BusinessElement; | |
| Name ▲ CI Tvoe Eind: | |

The CI Type can be restricted (or extended) in order to return the right type of CI instances in the lookup. Multiple target CI instances can be selected.

Figure 37: Example Simple Business Model – Insert Relationship, Relationship Type and Direction Selection

| Select Relation | nship |
|------------------|--------------------------------------|
| Source CI: | Fulfillment (BusinessApplication) |
| Target CI(s): | |
| rarger en(a). | Database |
| Relationship: | |
| Relationship. | |
| Direction: | Managed Relationship Orntainment |
| | Dependency |
| □ Define Relatio | |
| Deline Relatio | Backbone |
| | |
| | - Siber Channel Co |
| | - layer 2 |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | < CI Selection Save Cancel Help |
| | Cancel Hep |

Once all CI instances have been created and the CI instance's relationship is defined, this step is complete.

Example Business Service - Relate Monitored CI Instances

Now the monitored CI instances are linked into the business service model below the infrastructure service instances.

In the example, the following existing infrastructure elements are related to the example business service model:

- Frontend: Infrastructure element Ix001 (Unix node)
- Backend: Infrastructure element Ix002 (Unix node)
- Database: Infrastructure element orderdb (Oracle instance)
- Storage: Infrastructure element orderstorage)

The workflow establishing these relationships is similar to the previous section; however, the relationship is of type "Usage".

Example Business Service - Viewing the Result

The example business service model is available in service health and in the event, health and performance perspective. Its structure and related CI instances can be reviewed in these components.

The following screenshots show the example business service in the service health top view, the service health topology map (including KPIs) and the OBM health perspective where all related events are displayed.

Select Page ٠ 360° View x Top View X Custom Image x Geographic Map x Watch List x Topology Map x 💽 💋 🗁 🖪 📩 Top View 6 × 0 × × 🔻 🍳 Find ດ : 🕲 aa_MyService Display Options - Q 0 😣 aa_M e - 1 Ð 0 0 8 E 📀 Storage 🖽 😢 Ordering 🗖 C Fulfillment Prontend # 😣 Data se -H 🙁 or w) 🖪 V Last Update: 1/28/2014 07:07:12 AM

Figure 38: Example Simple Business Model – Service Health Top View

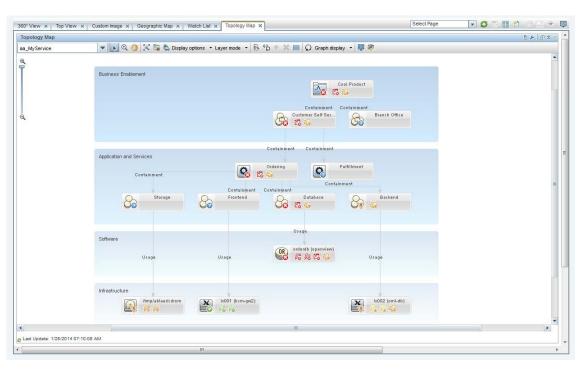


Figure 39: Example Simple Business Model – Service Health Topology Map

As an example within the health perspective, the example business service model can be used for event filtering purposes.

Figure 40: Example Business Model –View Explorer

| View Explorer 6 (A) d3.44 - | O Eve | nl Brow | raer for | Cool Pr | oduct | | | | | 21.00 |
|---------------------------------------|---------------------------|-------------|----------|----------|----------|--------|-------------|--|---|--|
| Browse Views Search | G | 8 2. | E. | 4 | · 2 | | sa_NyServia | pe . | 💌 👞 «Select an event filter- | 90 |
| 0 % | Sev | Prio | C N | 1 1 | A U | D St | te Time i | Received | * Title | Related CI |
| aa_MyService | 0 | | 10 | | | 5 | 1/27/14 | 01:34:04 PM | Cannot connect to Database. | orderdb (openview) |
| E-Q as_MyService | - | | 1 | | | 1 | | 01:33:12 PM | Node omi-db down. | bx002 (omi-db) |
| - Cost Product | 0 | 4 | | 13 | 1 | 5 | | 11:47:50 AM | DBSPI-144: DBSPI40-1: Unable to fetch data from table V [ORA-00942: table or view does not exist]. (Event Message Text) [Policy: DBSPI.Men | ss orderdb (openview) |
| Su Branch Office | 4 | v | | 2 | 8 | 5 | | 11:22:03 AM | snmp daemon process is not running | bx002 (omi-db) |
| 😑 🚱 Customer Self Service | V | • | | 2 | 8 | 5 | | 11:21:08 AM | FieSystem space utilization for Logical FieSystem Amplakaeniodrom - Major threshold exceeded. Space utilization in percentage = 100.00 (Th | |
| O Fulliment | | v | | 20 | A 2 | 8 E | | \$1:21:03 AM | The NFS server related process rpc mountd is not running. | 5x002 (omi-db) |
| Condening | 4 | V | | 1 | 8 | 5 | | 11.20.54 AM | Sendhall daemon process is not running. | bc002 (omi-db) |
| B & Backerd | | V | | 1 | 8 | 5 | | 11/20/44 AM | DNS daemon is not running. | bx002 (omi-db) |
| N be502 (omi-db) | 1 | V | | 1 | 8 | 5 | | 11:19:54 AM | dhop daemen process is not running. | bx002 (omi-db) |
| 🗐 🚷 Database | 0 | 8 | | 1 | | 5 | 1/23/14 | 11:15:21 AM | Open Agent Service terminated. (0pC45-471) | bx002 (omi-db) |
| Contendb (openview) | | | | | | | | | | |
| 🕞 🚷 Frentend | | | | | | _ | | | | |
| -B br001 (bsm-gw2) | | | | | | | | | | |
| ⊕ -8b Storage | | | | | | _ | | | | |
| Ampfaklasticdrom | 4 | | | | | | | | | 22 10 |
| | tens: | 10 of 11 | (1) | 121 | Y Cool P | reduct | Q 2 | 1 1 | <u>A</u> 5 <u>A</u> 1 Q 1 Q 0 <u>A</u> 0 <u>A</u> 30 <u>A</u> 30 | 2g 10 |
| ealth Top View | | | | | | ti / | S 46 1 | Health Indica | | 00 |
| A | | | | | | | | Ix002 (oml-db | b) (Unix) - | |
| Selected View: Systems_Infrastructure | | • | | | | | | | Event Node onl-db down. | |
| | | | | | | | | Health Inc | dicators Contribution to KPIs | |
| | | | | | | | | Treatment into | Ct by02 (onids) | |
| | | | | | | | | the second se | CE WAVE (OFFICE) | |
| | | | | | | | | | Ct Reve (on-de) Node: | |
| T System | Infrastu | ucture | | | | | | the second se | L. With (04-60) Note: Search: George Actions | |
| | infrastr comi-di | ucture | | 0.55 | 3) | | | 809 | CL Brance (Bried and Compared And Compa | |
| | _infráste 1 (omi-dt | ucture | | O | | | | ich KM V System | Note: Note: Nealth indextors Search: (Search dottors Analability If Proj Availability | Source Cl. 🔘 Node |
| • | Antrasta Comi-dit | | | 0 | | | | 809 | Node Node Medifit indicators Node Availability Image: Availability Image: Availability Image: Availability | Source Cl. () Node ha ((Cl)] T Ping node |
| | Lenin esta 2 (amil-alt | ucture | - | 0 50 | | 1 | | KPI KPI V System | Availability | Source Cl 🔘 Node ha ((Cl) 🏾 IT Ping nod ha ((Cl) 🔄 Run Serf |
| • | Frifriesto Comilect | ucture 9 | - | _ | | J | | Km Km V System Health Inc | Availability Prog Availability Prog Availability Prog Availability Control Contributing to KPIs | Source CI () Node Is ((CI) [] Ping rod Is ((CI) [] Run Self ring (CI) [] Show Le |
| | trifi éste comi-di | | 0 11 | 12, 168 | | J | | V System | Availability Pog Availability Pog Availability Control Control Public Control Control Control Public Control Cont | Source CI O Node ta ((CI) 17 Ping nod ta ((CI) 17 Run Self ring (CI) 17 Show La (CI) 17 Show La |
| • | enfreste Comi-di | | 0 11 | 12, 168 | | J | | Km Km V System Health Inc | Meetilit indicators Node Availability ♥ Prog Availability ♥ Availability # # ● dicators Not Contributing to KPis III Obasie Orace We Trace | Source CI O Node ta ((CI) 17 Ping nod ta ((CI) 17 Run Self ring (CI) 17 Show La (CI) 17 Show La |
| | | | 0 11 | 12, 168 | | J | | V System | Interaction indicators Note Availability ♥ Prg Availability ♥ # ♥ Prg Availability ♥ # ● ● ● ● # ● ● ● ● ● ● # ● <t< td=""><td>Source CI O Node ha ((C) 17 Ping nod ha ((C) 17 Run Sett nng (C) 17 Show Li (C) 17 Show Li le (C) 17 Show Li</td></t<> | Source CI O Node ha ((C) 17 Ping nod ha ((C) 17 Run Sett nng (C) 17 Show Li (C) 17 Show Li le (C) 17 Show Li |
| | tali ast Comi-di | | 0 11 | 12, 168 | | J | | View Constraints and the set of t | Analability Prog Availability | Source CI O Node ta ((CI) 1T Ping rock ta ((CI) 1T Run Ser- nng (CI) 1T Show La (CI) 1T Show La le (CI) 1T Show La ing (CI) 1T Show La |
| | tahast Comi-st | | 0 11 | 12, 168 | | J | | Constant indice Constant indice Constant indice Constant indice Constant indice | Mode (unitability Pog Availability Po | ha ((Cl) 1T Ping rode ha ((Cl) 1T Run Ser- ring (Cl) 1T Show La (Cl) 1T Show La ie (Cl) 1T Show La ing (Cl) 1T Show La (Cl) 1T Show La |
| | | | 0 11 | 12, 168 | | J | | Constant indice Constant indice Constant indice Constant indice Constant indice | Intertity indicators Note Intertity indicators Feelth indicators Intertity indicators Feelth indintors Intertity in | Source CI O Node tha ((CC) IT Ping rodu the ((CC) IT Run Self- ming (CC) IT Show La (CC) IT Show La (CC) IT Show La (CC) IT Show La |

Service Status and Service Health

The Service Health application combines service-related information derived from the event and metric data consolidated in OBM and transforms this information into a near real-time service model.

To achieve this, Service Health combines topology information from business services and their related monitored CI instances contained in the RTSM with business rules that calculate the services' status regarding health and performance.

Service Health comes with the out-of-the-box configuration that allows instant use of the Service Health application. Depending on the requirements, it is possible to tune and adjust Service Health to present the desired information and perform the appropriate service health calculation, depending on the group consuming the information.

This section provides some information on service health concepts and customization options.

Note: The documentation section Service Health gives additional information on this topic.

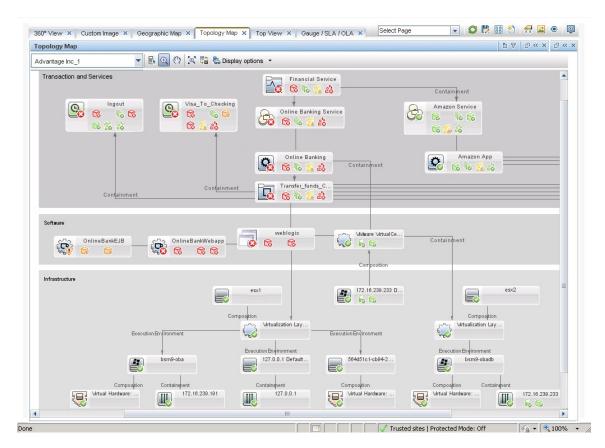


Figure 41: Example of Business Service Health View, with KPIs and CI Status

Key Performance Indicators

The KPI is defined in the KPI repository. Its status is calculated by HIs using business rules. In addition, KPIs of a CI are influenced by the KPIs of child CIs. They represent a high-level aggregated view on the CI status. A set of out-of-the-box KPIs is defined, such as performance and availability KPIs. Additional KPIs aggregate information about events that are present for a specific CI (or its child CIs): Unresolved Events KPI and Unassigned Events KPI. The OBM user interface displays the KPI status and status trend information in the health view with colored icons tagged to the related CIs.

The following figure shows KPIs, such as Application Availability and Application Performance, defined within service health out of the box.

| Figure 42: Key Performance | Indicators Overview | / Renository |
|----------------------------|---------------------|--------------|
| Figure 42. Key Ferrormance | inuicators overview | / Repusitory |

| rw Builder CI Indicators Custom Imag | e CI Status Alerta Assignmenta Repositories 🔤 | | 16 |
|--------------------------------------|---|------------|----------------------------------|
| Pia | | | |
| B / X 0 0 ? R B | | | |
| Name | Domain | Туре | Default Group Rule |
| | | | × |
| plication Availability | Application | Predefined | Worst Status Pule |
| splication Performance | Application | Predefined | Worst Status Pule |
| icklog | Business | Predefined | Worst Status Pule |
| siness Heath | Business | Predefined | BPI Business Health Status Rule |
| siness impact | Business | Predefined | Worst Status Pule |
| usiness Performance | Business | Predefined | Worst Status Rule |
| Aays | Application | Predefined | Worst Status Rule |
| inition | Buzinesz | Predefined | Worst Status Rule |
| leacy | Buzinezz | Predefined | Worst Status Rule |
| ceptions | Application | Predefined | Worst Status Pule |
| Nures | Application | Predefined | Worst Status Pule |
| Ineric | Unassigned | Predefined | Summary of values |
| gacy System | System | Predefined | Worst Status Pule |
| cations | Application | Predefined | Locations Grouped Parent Pule |
| stwork Availability | Network | Predefined | Worst Status Pule |
| stwork Performance | Network | Predefined | Worst Status Pule |
| unber Of open incidents | Business | Predefined | Sum of Open Incidents |
| T Impact | Business | Predefined | Sum of Values Rule |
| perational Status | Butters | Predefined | 8PI Operational Status Pule |
| VB | Butters | Predefined | Service Heath PNR Pule |
| rformance Analytics | Business | Predefined | Worst Status Pule |
| r Impact | Business | Predefined | Sum of Values Pule |
| P | Application | Predefined | Worst Status Pule |
| JF Alert | Application | Predefined | Worst Status Pule |
| curty | Application | Predefined | Worst Status Pule |
| ribel | Application | Predeficed | Worst Status Pule |
| ebel Errora | Application | Predefined | Number of Taoka in Error Pule |
| ebel Sessions | Application | Predefined | Number of Punning Sessions Pule |
| eScope Health | System | Predefined | Worst Status Pule |
| ftwore Availability | Software | Predefined | Worst Status Pule |
| fiware Performance | Software | Predefined | Worst Status Pule |
| stem Availability | System | Predefined | Worst Status Pule |
| stem Performance | System | Predefined | Worst Status Pule |
| roughput | Business | Predefined | Worst Status Pule |
| ansactions | Application | Predefined | Transactions Grouped Parent Rule |
| nassigned Events | System | Predefined | Worst Status Rule |
| resolved Events | System | Predefined | Worst Status Rule |
| alue | Business | Predefined | Worst Status Rule |
| olume | Business | Predefined | Worst Status Rule |

The following sample configuration shows how Software Availability KPI of an Oracle database instance is determined from HI states. The assignment between CI types, KPIs and HIs is under Service Health Administration -> Assignments -> KPI Assignments.

Figure 43: KPI Assignment Example – Oracle CI and Oracle KPI Assignments

| Турез | Assignments for CI Type: Oracle | | | Heath Indi | cator Assignments KPT/ | Assignments Propagation Ru |
|---|------------------------------------|---------------------|--------------|---|------------------------|----------------------------|
| Filter Show all assignments | | ිදි 📄 🔲 🕄 Synchrone | ze CI Type 💋 | | | |
| | Assignment Name | Monitored By | Status | KPIs | Type | Description |
| B-G Agent | | | | | | |
| P-Q ApplicationServer | Disgnostics Context Menu Assignm | Diagnostics | Punning | Software Performance | Predefined | Assigns drill to Diagnost |
| B-Q CusterSoftware | OM KPI Assignments (Configuration) | | Punning | Unassigned Events, Unresolved Events | Predefined | KPI enrichment assignm |
| ⊕-@ Communication Server ⊕-@ Database | Oracle KPIs Assignment | | Running | Software Availability, Software Performance | Predefined | KPI Assignments for |
| - C 082 | Purning Software KPI Assignments(| RUM | Running | | Predefined | Purning Software KPI A |
| - A HanaDatabase | SHA assignments (Infrastructure Be | | Punning | Performance Analytics | Predefined | |
| - Max DB | SteScope Related KPIs Assignment | SteScope | Punning | SAP | Predefined | KPIs Assignment for SA |
| - MySQL | SiteScope Related KPIs Assignment | SiteScope | Punning | Legacy System | Predefined | KPIs Assignment for Cli |
| Contact Solution S | | | | | | |

Navigating to the details of the Oracle KPI Assignment shows the details, such as related HIs and the business rule, used to aggregate the KPI status from related HIs.

| ID: * Name: Description: | Oracle KPIs Assignm | I-9ea9-1c8e40779414 ent Oracle Health Indicators | |
|--------------------------------|---------------------|--|-------------------|
| | - | | |
| Description: | KPI Assignments for | Oracle Health Indicators | |
| | | | |
| ondition PI Configurations | | | E [|
| * 🖉 🗶 🗞 | | | |
| KPI | Calculated Base | Related Heath Indicators | Business Rule |
| Software Availability | HIs and child KPIs | Database Object Status, Database Segment Status, Database Segment Usage Level, D | Worst Status Rule |
| Software Performance | HIs and child KPIs | | Worst Status Rule |

Figure 44: KPI Assignment Example – Software Availability KPI, Related Health Indicators and Business Rule

KPI and List of Related HIs

A list of related HIs is maintained within this configuration.

In the following figure, the HI "Database Service Status" is a member of an HI list related to the Software Availability KPI of an Oracle database.

Figure 45: KPI Assignment Example – List of HIs assigned to KPI

| KPI | | CI Type Propert | es |
|--|---|---|-----------|
| KPI: Business Rule: Calculated Based On: Related Health Indicators: | Software Availability Worst Status Pule His and child KPIs Database Object Status Database Segment Usage Leve Database Server Status | General Pri Binary Application Pass Calculated ID Boolean Allow Cl Update Change Is New Centre Annual Control Database Object Sta Database Object Sta Database Server Sta Database Server Sta Database Server Sta | evel |
| ticenses ⊕-s Licenses | - | Apply | ncel Help |

KPI and CI Status Calculation

A KPI status is calculated from the KPI-related HIs based on a business rule. By default, the worst status rule is applied. The calculation is based on HIs and the child KPIs.

The CI status is determined by the status of its KPIs according the "worst status rule". The worst status of CIrelated KPIs determines the overall status of the CI.

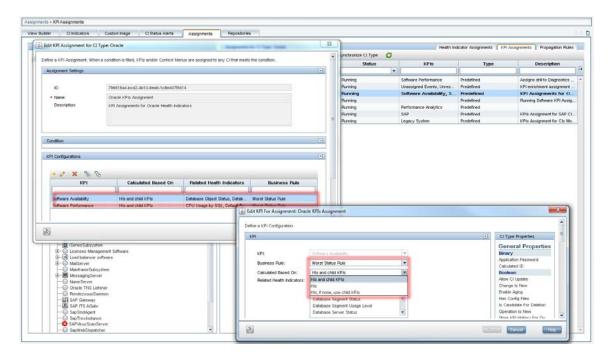


Figure 46: KPI Status – Business Rule, Calculation Basis

Example: KPI and HI in Health Perspective

The following figure shows a sample "database down" event, which sets the Database Service Status HI impacting the Software Availability KPI.

Figure 47: KPI Example – KPI and HI Related to Database Down Event

| View Explorer 👘 🌮 🗇 « 🗴 | O Ev | ent Brow | wser fo | r aa_My | Service | | | | | | | | ē & d |] << |
|-------------------------------|--------|----------|---------|----------|---|-------------------|---------------|---------------------------------------|--|--------------------------|--|---|--|-----------|
| Browse Views Search | | 10 20 | | -1 | ۰. | 27 | 🚊 aa | a_MyService | <select an="" event="" filter=""></select> | 💌 🚳 🍝 |) 🖗 i | 😅 🖳 (Se | arch Events |) * |
| 6 B | Sev | Prio | CI | 1 1 | A U | D | Sta | Time Received v Title | | | | | | |
| aa_MyService 💌 | 0 | - 🔺 . | - | | | | ₽. | 1/27/14 01:34:04 PM Canno | t connect to Database. | | | | | |
| ∃ I aa_MyService | | • | 0 | - | | | г, | 1/27/14 01:33:12 PM Node | oml-db down. | | | | | |
| Cool Product | • | • | | 1 | | | - | 1/23/14 11:47:58 AM DBSP | -144: DBSPI40-1: Unable to fetch data | a from table "V" [ORA-00 | 942: table | or view does not | t exist]. (Event | Mes |
| Branch Office | | ▼ | | 1 | 2 | | ۵, | 1/23/14 11:22:03 AM snmp | daemon process is not running | | | | | |
| Customer Self Service | · | • | | 1 | 2 | | 5 | 1/23/14 11:21:08 AM FileSy | stem space utilization for Logical File | System /tmp/aklaer/cdro | m - Major t | hreshold exceed | ed. Space util | izat |
| | | | | 1 | 2 | | Γ., | 1/23/14 11:21:03 AM The N | S server related process rpc.mount | d is not running. | | | | |
| | | | | 1 | 2 | | Γ. | | nail daemon process is not running. | | | | | |
| | | | | <i>(</i> | 2 | | 5 | | aemon is not running. | | | | | |
| | | | | a | 8 | | - | 1/23/14 11:19:54 AM dhcp | laemon process is not running. | | | | | |
| | | 10 of 11 | 1 (1). | 1 | aa | MySer | v 🙆 | 2 1 1 \Lambda 5 | A 1 Q 1 Q | 0 | 3 81 0 | asi 0 | lang 10 | Ţ |
| ealth Top View | - | | | - | | | _ | ***** | | | | | | _ |
| | | | ۍ ن | < X | | ndicato | | | | | | | ē / 6 | 1 << |
| Sum top view | | | | | | | | | | ē≁ ∂≪× | Action | · | | |
| Selected View: ORA_Deployment | 1 | _ | | - 0 | | | | (Oracle) 👻 | | × >> 0 4 8 | | | | 1 |
| | t | | | • • | rderdb | (openv | view) | | | × » • • • • • • | | Cannot conne | ct to Database. | - |
| | t | | | , ° | rderdb Health | (openv | view) | (Oracle) – rs Contributing to KPIs | | | S | | | - |
| Selected View: ORA_Deployment | | | | , ° | rderdb | (openv | view) | | | | C Event: | Cannot conner | | - |
| | | | , | | rderdb Health | (openv | view) | | Health Indicators | | C: | Cannot conne orderdb (open | view) | - |
| Selected View: ORA_Deployment | | | | | rderdb Health | (openv | view) | | | | Event: CI: Node: Search: | Cannot conner orderdb (open | view) ns | |
| Selected View: ORA_Deployment | hent 🖃 | | | | rderdb -lealth 82 () KPI | (openv | riew) cato | rs Contributing to KPIs | Health Indicators | | Event: Cl: Node: Search: Filter: | Cannot conner orderdb (open | view) ns I () Source C | 1 (|
| selected View: ORA_Deployment | hent 🖃 |) | | | rderdb Health E III KPI Sof | (openv n India | riew) cato | rs Contributing to KPIs | | | Event: CI: Node: Search: Filter: | Cannot conne orderdb (open Cearch Actio All O Ci ow Performance | view) ns O Source C Graphs ((C | 1 (|
| Selected View: ORA_Deployment | hent 🖃 |) | | | rderdb Health E III KPI Sof | (openv n India | riew) cato | rs Contributing to KPIs | | | Event: CI: Node: Search: Filter: | Cannot conner orderdb (open | view) ns O Source C Graphs ((C | 1 (|
| Selected View: ORA_Deployment | iew) 🛢 |) | | | rderdb Health E III KPI Sof | (openv n India | riew) cato | rs Contributing to KPIs | | | Event: CI: Node: Search: Filter: | Cannot conne orderdb (open Cearch Actio All O Ci ow Performance | ns Source C Graphs ((C Graphs ((C | I ()) |
| Selected View: ORA_Deployment | iew) 🛢 |) | | | rderdb Health E III KPI Sof | (openv n India | riew) cato | rs Contributing to KPIs | | | Event: CI: Node: Search: Filter: | Cannot conner orderdb (open : <u>Search Actio</u> All Cl ow Performance ow Performance | ns Source Graphs () Graphs () | 0 |

CIs and KPI Propagation

When a KPI is assigned to a CI, the propagation mechanism propagates the KPI to the parent CIs based on propagation rules. By default, a KPI of a CI is automatically propagated to parent CIs (exceptions include "Unresolved Events" and "Unassigned Events").

Propagation rules are defined on the CI-type level and can be changed. They are maintained under the Service Health Administration section under Assignments -> Propagation Rules.

Propagation rules defined on higher-level CI types are inherited by descendant CI types. See <u>Propagation</u> <u>Rules</u> for more information.

| ypes | Assignments for CI Type: Configurationitem | | | Heath | Indicator Assignments KP | Assignments Propagation Rules | | | | | | |
|---|---|--|---|----------------------------------|----------------------------|--|--|--|--|--|--|--|
| Filter: Show all assignments | Propagation Rules define exceptions to the default KPI propagation. By default, when a KPI is assigned to a Ci the XPI is propagated to the CI's parents, with the default group rule (as defined in the KPI Repository). | | | | | | | | | | | |
| Configuration tem | * * 0 × 0 % % 0 | | | | | | | | | | | |
| CiCollection | Pule Nome | Parent CI Ty | pe Assigned KPI | Propagated KPIs | Туре | Description | | | | | | |
| Image: Second Seco | | | | | | | | | | | | |
| -G Location | OM Unaccigned Event KPI Propagation OM Unresolved Event KPI Propagation | Configurationitem Configurationitem | Unassigned Events Unresolved Events | None None | Predefined | Propagation of Unassigned . Propagation of Unresolved | | | | | | |
| 0-12 Monitor | OW Offestives Every KPT Propagation | Corriguiationitem | One loved Events | INCIDE | Precenced | Propagation or Unresolved . | | | | | | |
| - Charly | | Edit Propagation Rule for CI | Type: ConfigurationItem | | | | | | | | | |
| | | Define a VPI Procession Date W | hen a condition is filled on a CI, KPIs are | assigned to the CFs parent based | on the Task definitions | | | | | | | |
| | | 1 | | | writers rear with more to. | | | | | | | |
| | | Pute Settings | | | | | | | | | | |
| | | ID: | c1ecc4f5-a857-4071-b073-44eddd4e | 1766 | | | | | | | | |
| | | * Name: | | | | | | | | | | |
| | | Nome: Description: | OM Unassigned Event kPI Propagat | | | | | | | | | |
| | | | Propagation of Unassigned Event KP | to to corregulation send | | | | | | | | |
| | 1 | | | | | | | | | | | |
| | | | | | | | | | | | | |
| | | Condition | | | | A | | | | | | |
| | | | | | | | | | | | | |
| | | Source CI Type: | Configurationitiess | | | | | | | | | |
| | | Parent CI Type: | Configurationitem | * | | | | | | | | |
| | | Assigned KPI Type: | Unassigned Events | * | | | | | | | | |
| | | | | | | | | | | | | |
| | | Task | | | | | | | | | | |
| | | | | | | | | | | | | |
| | | Do not propagate the KPI | 6 | | | | | | | | | |
| | | | using a differt rule: Select a rule | T | | | | | | | | |
| | | Propagate custom KPIs | | | | | | | | | | |
| | | | | | | | | | | | | |

Figure 48: Sample Propagation Rule – Do Not Propagate Unassigned Events KPI

Customize KPIs Displayed in Views

Service Health allows selecting a range of KPIs that are displayed for a view. For all KPIs potentially available within a view, a subset can be selected. In addition, you can define which KPIs are considered for the overall CI status calculation.

This customization is performed in View Builder of the Service Health Administration.

This is useful for providing custom views for different audiences. As an example, the operational staff should typically have all KPIs visible. A dashboard view provided to an end user can be limited to contain the basic availability and performance KPI information.

| w Selector | D View Builde | r | | | | | | D.C. |
|--|--|---------------------------------------|-----------------|-------------------------|-------------|--|-----------------|--|
| □ - □ - ▷ - @ × ▷ ¥ ▷ ¥ ○ ? | 000 | e 0 | | | | | | |
| Poot | CI selector | CI Types | | Query Definition | 20 | | | Herarchy |
| | Ci select | × | | 1. 11 0 | Q 🗶 📅 🖓 | | | B-J=X 0000 |
| - In MuSarvine without cale | | Browse Views | Search Cis | | | 100.00 | 1 1 1 1 4 | 2 Internet internet in the second sec |
| as_PatternView1 | CO | | | | | ~ | | Hierarchy Method: Manual |
| - Ca as Perspective_Based_View | Statistics of the local division of the loca | | | | | 0 | | |
| aa_Perspective_Based_View_2 | View; ao | MyService | × 100 | 9 | | BusinessFunctio | | aa_MyService_without_calc |
| 8- 🚰 Application | as MyService withou | t calc Properties | | | | Concession in which the local division in which the local division is not the local division in the local dint | * | Tem |
| ē-🧱 Business | | | _ | | | | | BusinessFunction |
| Business Services | Include in View | Include in CI Sta | | KPI | Domain | Business User | Operations User | To |
| 2-Coud | Include in view | anclude in CI Sta | | - KPT | Domoin | Business Oser | Operations User | E- D Business Service |
| 2 Configuration Manager | | · · · · · · · · · · · · · · · · · · · | 1 | | | | | |
| - Database | | | OT impact | | Business | ~ | v • | - O Business Applicatio |
| Datacenter and Power | | | Operational Sta | tus | Business | ~ | 1 | |
| - CotNet | | | PNR | | Business | 1 | 1 | B- Co Infrastructur |
| GettingStarted Guide | | | Performance Av | whiles | Business | 4 | 1 | 1.000 |
| E IBM WebSphere MQ | i iii | | RT Impact | CONTRACTOR OF THE OWNER | Business | 1 | 1 | - 🖵 ktras |
| 0- 🤤 Integrations | i i | | SAP | | Application | 3 | 2 | |
| 3- 🚰 (Series | i i | | SAP Alert | | Application | | | |
| 9-100 J200 | i ii | | Security | | Application | | 1 | |
| - Eil Local impact views | | | Siebel | | Application | 1 | | 4 |
| Locations Mainframe | i i | | Siebel Errors | | Application | | 2 | 2 |
| | | | Siebel Sessions | | Application | 2 | | |
| E Network | | | | | | * | , | |
| - Operations Management | | | Site Scope Heal | | System | ~ | | |
| Perspectives | | | Software Availa | | Software | v, | | |
| 5-10 SAP | | | Software Perfor | | Software | 2 | 1 | |
| 0- 📴 Service Deck | 2 | | System Availab | | System | | | |
| 0-🔚 SteScope | | G | System Perform | whice | System | 1 | 1 | |
| - 📴 SOA | | | Throughput | | Business | 4 | ~ | 2 |
| 9 🚰 Storage | | | Transactions | | Application | ~ | 1 | |
| 9- 🚰 Virtualization | | | Unassigned Ev | | System | ~ | 1 | |
| - an_aki | | | Unresolved Eve | enta | System | 4 | - | |
| CI Alerts View Giognostics Probe Group and Infrastructure | | | Value | | Business | ~ | ~ | |
| Model Herarchy | | | Volume | | Business | 1 | V | |
| - | | | | | | Constant of the | | |
| | | | | | | Store | Cancel Hep | |
| | - | | | | | | | |
| | | | _ | - | | | | |

Figure 49: View Properties – KPIs to Be Included in View, CI Status Calculation

The following figure shows two sample views taken at the same time, with the identical TQL. The view on the left-hand side uses default view properties leading to all propagated KPIs shown in the view. The view on the right-hand side is limited to show the System Availability KPI only.

| Busmess Enablement | | | Cod Product | Business Enablement | | | Coal Product |
|--|---|--|--|-------------------------|-----------------|---|--|
| | | Containment Set Ser | Containment Branch Office | | | Curtainer Eaf Ser | Containment Branch Office |
| pplication and Services Containment | | Containing | Fulliment | Application and Environ | 2 | Containment Containment | Fullment |
| Contract Research | | Status: Business Rule: Last Status Change: | Critical Critical Worst Status Rule 1/27/2014 1:18 PM | Co Broge | Contribution | Details - Syst Status: Business Rule Last Status Change: | em Availability Major Worst Status Rule 1/27/2014 1:33 PM |
| finare Urage | Unge onderde lopensiewj Ro Ro Ro Go | Urage | Usige | Safivers | Urage | Unige anderth (spendere) | Unge |
| hashurhare Angslødaerikdrem | | idet (borrgaz) | 1402 (mri 44) | Inflactsucture | Idot (torr-yea) | | Intit (smide) |

Figure 50: Two Views on the Same Service – Full KPIs (Left) and System Availability KPI Only (Right)

Event Priority

Once events and their related CIs impact business services, OBM instantly calculates an additional event attribute allowing to classify events depending on their impact on a business service. The event attribute, Event Priority, can be used by the operational staff to identify the events to work on.

The calculation of the event priority is based on the event severity and the business service impact, as shown in the following table.

Table 3: Event Priority Mapping

| Business Service | Event Severity | | | | | | | | |
|---------------------|----------------|--------|---------|--------|---------|----------|--|--|--|
| | Unknown | Normal | Warning | Minor | Major | Critical | | | |
| No Impact | Lowest | Lowest | Low | Low | Medium | Medium | | | |
| Low | Lowest | Lowest | Low | Low | Medium | Medium | | | |
| MediumLow | Low | Low | Low | Medium | Medium | High | | | |
| Medium | Medium | Low | Medium | Medium | High | High | | | |
| MediumHigh | High | Medium | Medium | High | High | Highest | | | |
| High | Highest | Medium | High | High | Highest | Highest | | | |

Figure 51: Example Business Impact of a CI – Medium Low

| Hierarchy | | | | | | | 5 F 19 X |
|--|-----------|--------|-----------------------|--------------------------|----------------------|----------------------|--------------------|
| aa_NyService 💌 📴 🦉 😵 [global Show errors 🔹 🍇 🛐 🚍 | 4 | | | | | | |
| Name | Business | Status | Last Status Change | Software | 🕒 System | | |
| | Impact | | | Software Availability | Unassigned Events | Unresolved Events | System Availabilit |
| B 33_M/Service | | | | | | | |
| Cool Product | | 0 | 1/27/2014 1:18 PM | 0 | | | V |
| - B B Customer Self Service | | 0 | 1/27/2014 1:18 PM | 0 | 200 | | V |
| Ordering | | 0 | 1/27/2014 1.18 PM | 0 | | - | V |
| - 🕫 👶 Database | | 0 | 1/27/2014 1:18 PM | 0 | (inc) | | V |
| (openview) | (000000) | 0 | 1/27/2014 1:18 PM | 0 | 2 | 2 | V |
| | Medium Lo | w | | | | | |

<u>Note</u>: To display the Business Impact information in Service Health's 360° View, the business impact bar needs to be enabled in Service Health's infrastructure settings. See How to Configure the Business Impact Component for further information.

Figure 52: Example: Resulting Event Priority in the Event Browser

| Ever | nt Brow | /ser | for o | order | rdb (| ope | nviev | v) | | |
|-----------------|---------|------|-------|-------|-------|-----|-------|-----|-------------------------|---|
| 🕞 🔏 🖳 🔒 🐁 🔓 🖓 🛱 | | | | | | | | 0 | ▼ ▼ <no filter=""></no> | |
| Sev | Prio | С | N | I | Α | U | D | Sta | Time Received • | Title |
| ∇ | | | | ø | | | | Γ., | 2/3/14 09:42:03 AM | A filesystem or logical disk on oml-db.mambo.net is running out of space. I |
| 0 | | 2. | | | | | | Γ., | 1/27/14 01:34:04 PM | Cannot connect to Database. |
| | | n | | | | | | Π., | 1/27/14 01:33:12 PM | Node oml-db down. |
| 0 | | | | ø | | | | Γ., | 1/23/14 11:47:58 AM | DBSPI-144: DBSPI40-1: Unable to fetch data from table "V" [ORA-00942: tal |

Derived Values and Benefits

Implementation of Service Centric Management brings the following benefits:

- Using the event priority allows classification of the events according to their business impact. This enables the operators to focus on key events.
- Providing business service-related context with the trouble ticket integration.
- Extending the dashboard by views. This provides a business service perspective by combining the status information collected across different domains.

• Supplying dashboards that provide tailored views with the right level of the service-related information to customers, end users and management personnel.

Step 5: Self-Service and Monitoring Automation

This phase targets administrative aspects of the solution. The objective of this phase is to increase administration efficiency.

+Value +Value +Value +Value elf-Service Ionitoring +Value utomation usines ervice Centric Management dd End-Use erspective Control Events Consolidate Events

Figure 53: Fifth Step – Moving On to Self-Service and Automation

Today's monitored environments are highly dynamic: the number of applications is growing; the underlying infrastructure elements are frequently changed or replaced. With virtualization and cloud concepts, changes are performed instantly and often automatically.

Implementing this phase of the solution will allow achieving the following goals:

- Reduced administrative effort needed to configure the underlying monitoring solution
- Self-provisioning of monitoring by customers
- Standardized monitoring, which at the same time allows defined adaptation to target environments via parameters modifiable by the subject matter experts
- Providing monitoring as a service

Automation of Monitoring

Current IT environments are highly dynamic. The number of IT services, applications and infrastructure components that need to be monitored is growing. Infrastructure components are constantly added or removed by means of virtualized environments. OBM addresses the resulting administrative demand with its Monitoring Automation (MA) component that enables automatic configuration of monitoring in changing environments.

This section contains a brief description of the OBM MA components. It also shows how Monitoring Automation helps in monitoring virtualized or dynamic environments and provides the efficient monitoring configuration by using parameterized monitoring.

Monitoring Automation

Monitoring Application provides the following capabilities:

- It allows configuring monitoring products in a single place. It covers the OM Agent-based monitoring, as well as the agentless SiteScope-based monitoring. This enables the maintenance and distribution of policies from OBM.
- Monitoring Automation's aspects concept allows definition of monitoring in a CI-specific manner.
- It provides a powerful mechanism to implement adjustable monitoring standards that can be tuned to the target environment via a set of defined parameters. This hides the complexity and details of monitoring and its underlying technology allowing defining relevant parameters for tuning purposes.
- The reporting capabilities can be used to quickly gain an overview about the monitoring configured in the environment.
- Management Templates concept allows definition of monitoring in a service-specific context.
- Monitoring Automation allows designing flexible models of distributed, composite applications corresponding to the CI topology contained in the RTSM. It is able to configure the corresponding distributed monitoring with minimal configuration activities.
- The deployment of monitoring of distributed composite applications can be automated based on discovery results. The monitoring configuration can react on the changes occurred due to dynamicity.

Coping with Monitoring of Infrastructure That Is Changing

Monitoring Automation enables OBM to perform monitoring of provisioned systems in an efficient manner. At the startup of an installed system, it is possible to bundle system images with the OM Agent software. During the initial startup, the process for registering the OM Agent with the OBM server can be handled automatically, e.g. through implementing an automatic certificate granting through OBM server-based scripts.

In addition, Monitoring Automation allows defining triggers for the automatic deployment of discovery policies extending the RTSM with the CI information about the provisioned system based on discovery results. This can be used as a trigger to deploy an additional defined set of monitoring configuration matching the results of the discovery. Subsequent changes on the system are detected by regular rediscovery routines, propagating the related changes to the RTSM, which leads to adaptation of monitoring if necessary.

The Infrastructure Management Pack provides a set of common monitoring configuration targeting the system and OS resources and events and is ready to leverage the MA functionality. Other Management Packs for common middleware components, such as databases, are available as well (the number of Management Packs is constantly growing).

Monitoring Automation also allows customers to design and develop their own tailored set of monitoring configuration as a management template, leveraging these concepts and fitting and extending the existing out-of-the-box capabilities.

Monitoring Automation Management Templates allow the definition of monitoring based on the RTSM CI topology information in a service context. It allows taking different CI types, relations and patterns into account to determine the applicable monitoring configuration.

Monitoring configuration developed with OM Windows or OM Linux can be imported into Monitoring Automation to build and evolve a monitoring solution on investment done in the past.

<u>Note:</u> The <u>OBM Management Pack Development Guide</u> gives some background information and provides details on how to design and develop custom Management Packs.

Virtualization Monitoring

Operations Bridge Cloud Optimizer and the <u>Management Pack for Infrastructure</u> provides a rich set of functionality that targets the monitoring of virtualized environments from the guest and the host perspective. In addition to provisioning monitoring of a virtualized system upon startup as described in the previous section, the Management Pack for Infrastructure can also gather the event, status change and performance metric information available by the virtualization platform itself covering common solutions, such as KVM and XEN.

Cloud Optimizer can be integrated into OBM and is covering X86-based virtualization platforms such as VMware.

Standardize Monitoring with Parameters

Monitoring Automation allows adaptation of the monitoring configuration to a specific target based on parameters. Typical parameters include a category, a set of thresholds, environmental parameters, such as installation paths or filenames, and environment variables or the credential information needed by monitoring to access the monitored application.

The MA parameter concept allows defining a certain baseline of standard default monitoring that can be adjusted to the needs of the target environment.

Parameterization of the monitoring configuration allows an easy adaptation of monitoring to different target systems and application instances. The parameters can have a default value (if no specific parameters are required for a target system) or can be tuned for specific needs.

Delegate Monitoring administration to SMEs

Tuning a parameter does not require an in-depth knowledge on how the monitoring is configured. The parameters can be used to hide the complexity of the underlying monitoring configuration and technology.

Designing and implementing the monitoring configuration should be separated from deploying the monitoring (the deployment task can be delegated to domain subject matter experts, reducing the amount of work for the OBM administrators).

Automation of OBM Configuration

OBM provides web service interfaces for automating configuration tasks and steps. These web services can be used to script the configuration changes and to integrate OBM configuration tasks into external workflow systems.

Monitoring Service Provisioning

Today's IT is dominated by concepts, such as cloud, infrastructure as a service, platform as a service, software as a service and more. Various solutions are available that support the implementation of such concepts.

Cloud Service Automation CSA combined with Operations Orchestration provides a solid and proven solution that is scalable and capable of handling the complexity of heterogeneous customer environments. It provides various functions, such as automated service lifecycle support, as well as an open and extensible architecture.

Provisioning of monitoring and service management is an important aspect, since service level objectives and the service levels related to provisioned services must be fulfilled instantly at the startup of the service with little or no latency.

By providing interfaces through exposing a rich set of web services, OBM can perform the necessary steps required to configure, deploy and maintain all monitoring and service management aspects in an automatic manner.

OO workflows can use web services to instrument all tasks required for the OBM configuration. Monitoring and service monitoring provisioning go along with the provisioning of services instantly and reliably. As an example, CSA can be used allowing to choose a desired service. CSA and the underlying OO do not simply provision the infrastructure, platform and software; they also extend OBM and its RTSM with related CI instances and relations, as well as the views and business service representation in the service model. The provisioned systems register with the OBM system at the startup, which automatically triggers the deployment and activation of related monitoring.

Figure 54: Integrated Service Provisioning Covering OBM

Delivering an Integrated Service - CSA & BSM

