

HP Business Service Management

for the Windows/Linux operating system

Software Version: 9.x

Multi-Tenancy Using BSM 9.x – Best Practices

Document Release Date: January 2012
Software Release Date: August 2011



Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notices

© Copyright 2012 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe® and Acrobat® are trademarks of Adobe Systems Incorporated.

Intel®, Pentium®, and Intel® Xeon® are trademarks of Intel Corporation in the U.S. and other countries.

Java is a registered trademark of Oracle and/or its affiliates.

Microsoft®, Windows®, Windows NT®, and Windows® XP are U.S registered trademarks of Microsoft Corporation.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

Acknowledgements

This product includes software developed by the Apache Software Foundation (<http://www.apache.org>).

This product includes software developed by the JDOM Project (<http://www.jdom.org/>).

Documentation Updates

The title page of this document contains the following identifying information:

Software Version number, which indicates the software version.

Document Release Date, which changes each time the document is updated.

Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: <http://h20230.www2.hp.com/selfsolve/manuals>

This site requires that you register for an HP Passport and sign-in. To register for an HP Passport ID, go to: <http://h20229.www2.hp.com/passport-registration.html>

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

HP Software Support

Visit the HP Software support web site at: <http://www.hp.com/go/hpsoftwaresupport>

The web site provides contact information and details about the products, services, and support that HP Software & Solutions offers. It provides customer self-solve capabilities, and is a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage a support contract
- Look up HP support contracts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

Table of Contents

- Table of Contents 4
- 1 Introduction 5
 - 1.1 Scope and motivation 5
 - 1.2 What is a tenant? 5
 - 1.3 What is multi-tenancy? 5
 - 1.4 Solution overview 6
 - 1.5 Use Cases 8
 - 1.5.1 Dashboard 8
 - 1.5.2 uCMDB 8
 - 1.5.3 Monitoring 8
- 2 Step-by-Step Configurations 9
 - 2.1 Class model configuration in the central BSM 9
 - 2.2 Configuration changes in the tenant’s CMS 11
 - 2.3 Configuration for End User Monitoring 16
 - 2.4 Configuration for OMi 17
 - 2.4.1 Enriching events which are forwarded from OM 17
 - 2.4.2 Enriching events which are generated from within BSM 19
 - 2.4.3 Automatically assigning events to user groups 20
 - 2.4.4 Viewing the Tenant Owner attribute in the Event Console 22
 - 2.5 Create Views in BSM 23
 - 2.6 Create MyBSM Pages 26
 - 2.7 Configure Users and Permissions 26
 - 2.8 Multi-Tenancy and Other Data Collectors 28

1 Introduction

1.1 Scope and motivation

Multi-tenancy means different things to different people. Different service providers have different multi-tenancy needs (for example SaaS – Software as a Service, MSP – Managed Services Provider, or IaaS – Infrastructure as a Service), and there may be shared or split ownerships between service providers and clients.

The implementation of global system integrators like MSP, and the common use of the BSM platform in a shared services environment, make it necessary to provide multi-tenant capabilities in a shared services platform. The following use cases were identified to demonstrate the multi-tenant capabilities of BSM in a shared service platform.

This document provides the following:

- Definition of multi-tenancy, solution overview, and high level architecture of multi-tenancy.
- Use cases supporting multi-tenancy in BSM.
- Detailed explanations on how to configure BSM to provide multi-tenancy capabilities.

1.2 What is a tenant?

A **tenant** is a set of people (customers, service providers, or suppliers), whose access to data in support of customer contracts is managed as a collective whole. It is commonly referred to as an organization.

In the world of enterprise systems, a tenant is referred to as a customer receiving organization (CRO). Tenants are managed as part of the customer on-boarding process of the service provider.

For larger customers, a tenant can be defined based on business unit, contract segment, or geography, depending on contract entitlement. For example:

- Acme Manufacturing vs. Acme Sales
- Acme Web Hosting vs. Acme Workplace Services
- Acme North America vs. Acme Europe

Note that a user can be associated with more than one tenant; this is common for service provider personnel, but less common for customer personnel. A tenant is *not* a user group or resource group.

1.3 What is multi-tenancy?

Multi-tenancy refers to a principle in software architecture whereby a single instance of software runs on a logical system and serves multiple client organizations (tenants), thereby increasing resource utilization and reducing operational complexity and cost.

Multi-tenancy is contrasted with a multi-instance architecture, whereby separate software instances (or hardware systems) are set up for different client organizations. Multi-instance architecture increases the costs to clients and support organizations. Supporting multi-tenancy within a single system reduces these costs by leveraging common infrastructure and support models.

The advantages of multi-tenancy are not universally supported within the software industry, and this may be a source of competitive differentiation.

1.4 Solution overview

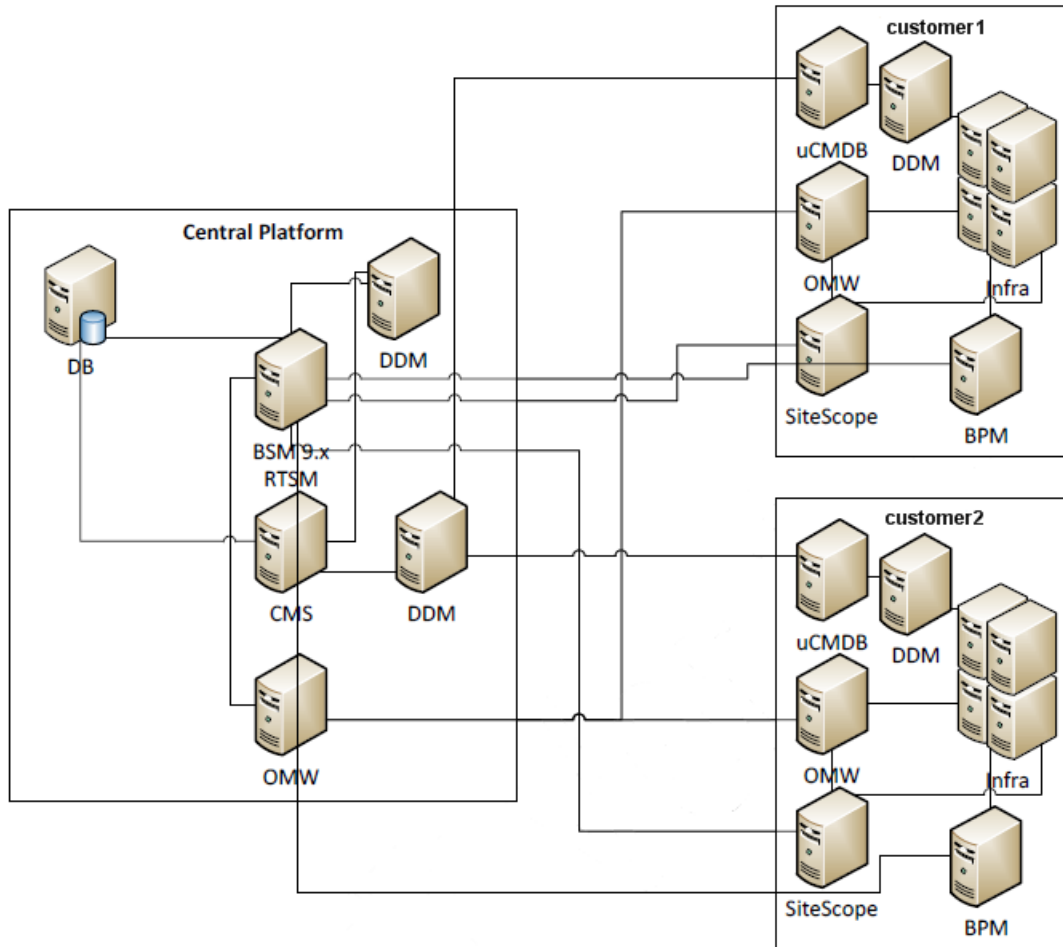
BSM 9.x does not support multi-tenancy out-of-the-box. This document describes the configuration steps required to enable BSM 9.x to support multi-tenancy use-cases.

The solution is based on the following:

- 1) **Tenant Management.** Each tenant's data is identified within BSM by a dedicated attribute; this is the foundation of the entire solution. In some cases this attribute is automatically enriched, but in some cases the MSP administrator must update this attribute manually. This attribute identifies the tenant who owns the element, which means that the element is managed under the lifecycle of that tenant.
- 2) **Data-to-Tenant Relationship.** All the elements in BSM - CIs, events, views, and so on - are related to one (and only one) tenant. Based on this information, data can be filtered in the different BSM applications to show data for each tenant; for example you can filter events or RTSM views based on the tenant attribute.
- 3) **Security Management.** Using the BSM security model you can set permissions for different resources; BSM allows you to assign different permissions to different user groups. This can be used to define different permissions for different tenants. This security model is managed by the MSP administrator, and will probably be the most significant area requiring administration in the multi-tenancy solution on top of BSM 9.x.

The solution architecture is designed to address the use cases described in the following section. The main concept of the solution architecture is based on a local IT Operations environment for each customer, which is monitored by different monitoring tools (such as SiteScope, OM, BPM), and discovered by DDM. All this information is then sent to a central BSM/CMS platform, and accessed by different tenants.

The diagram below is one example of a deployment scenario; there are many variations to multi-tenancy but the basic concept will be similar. For example, some of the monitoring products such as BPM or SiteScope can also be leveraged across tenants (using remote monitoring); in this case they are configured and managed by the service provider using the central BSM platform, and the monitored data is accessed through the various BSM applications (Event Console, Service Health and so on).



1.5 Use Cases

The multi-tenancy solution should provide the following capabilities:

1.5.1 Dashboard

- **Different Dashboards for Different Tenants.** Show all IT components managed by the various monitoring tools (OM, SiteScope, BPM, and so on) in separate views for each tenant. This includes Service Health, Event Management, Reports, SLM, etc.
- **Integrated Dashboard for the Operations Team.** Show integrated views with data from all tenants, to be used by the MSP Operations team.
- **Event Correlation.** Provide event correlation for each tenant independently.

1.5.2 uCMDB

- **Discovery and Integration.** Provide integration of multiple CMDBs with one CMS system.
- **Independent Discovery.** Provide discovery and mapping capabilities for each tenant independently.

1.5.3 Monitoring

- **Distributed Monitoring.** Integrate all remote monitoring platforms using a central BSM platform.
- **Event Views.** Provide different event views for different tenants.
- **Reports.** Provide scheduled reports for different tenants.

2 Step-by-Step Configurations

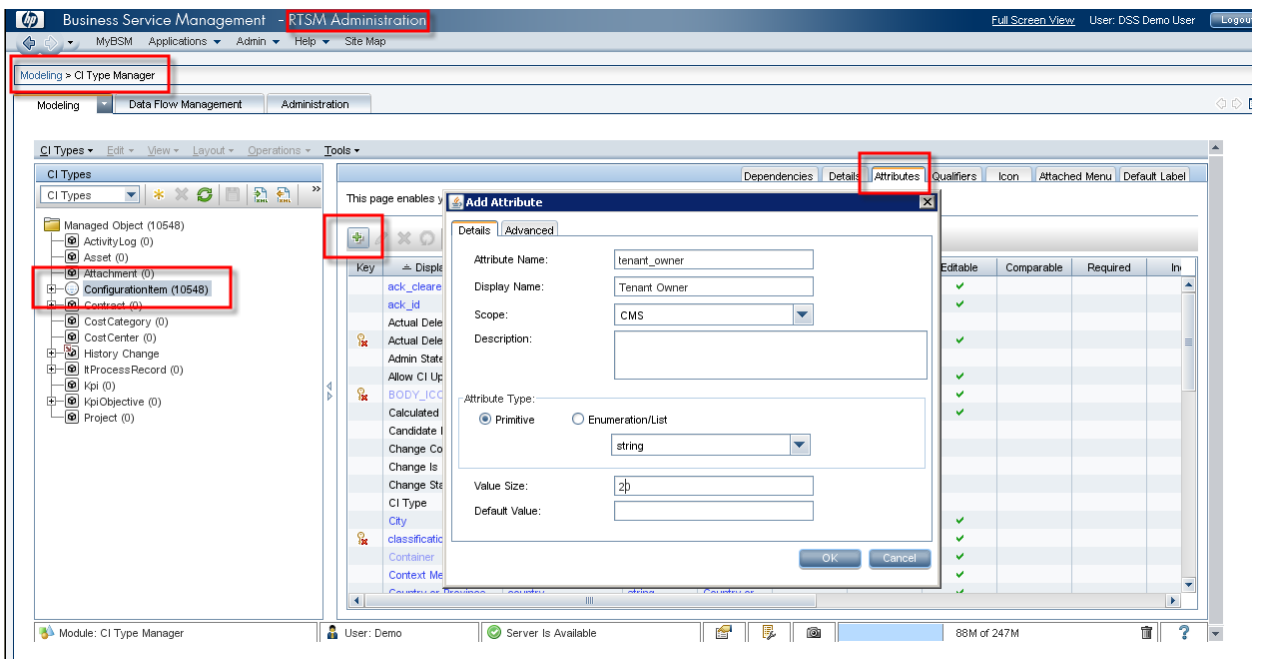
2.1 Class model configuration in the central BSM

One of the major requirements of this solution is that each CI be associated with a specific tenant. To achieve this, we need to add an attribute to the RTSM class model which will identify the tenant owner.

Note: In UCMDB 9.02, the class model has already been defined to support multi-tenancy. You might want to consider creating the Tenant Owner attribute in BSM the same as in uCMDB 9.02, since it may be supported in a future upgrade, although this is not certain. You should also note that BSM 9.10 is compatible with uCMDB 9.03, which means these attributes will be available out-of-the-box.

If you are using BSM 9.0x, perform the following steps to extend the class model:

- 1) Within BSM, access **Admin > RTSM Administration > Modeling > CI Type Manager**.
- 2) Select the **Configuration Item** CI type.
- 3) Within the **Attributes** tab, click **Add** to add a new attribute.
- 4) Define the new attribute as follows:



- 5) Click **OK** to close the **Add Attribute** dialog box, then click **Save** in the left pane.

Note: This is how the Tenant Owner attribute is defined in uCMDB 9.02:

The screenshot shows the 'Edit Attribute' dialog box with the following fields and values:

- Attribute Name:** Tenant Owner
- Display Name:** Tenant Owner
- Scope:** CMS
- Description:** The Tenant Owner ID
- Attribute Type:** Primitive (selected), Enumeration/List
- Value Size:** 32
- Default Value:** (empty)

Buttons: OK, Cancel

Edit Attribute

Details | Advanced

Attribute Name:

Display Name:

Scope:

Description:

Attribute Type:

Primitive Enumeration/List

Value Size:

Default Value

OK Cancel

2.2 Configuration changes in the tenant's CMS

If your tenant will have discovery capabilities, the DDM probe in the tenant environment should be connected to the tenant's (local) CMS. You can then synchronize the relevant topology from this CMS into the central BSM/CMS.

To do this, you must perform the following configuration changes on each tenant's CMS:

- 1) Repeat the procedure described in section 2.1 on the tenant's CMS. This updates the tenant's class model with the same attribute that you added in the central BSM.
Although you can assign a default value for the Tenant Owner attribute in the CMS, this does not affect CIs which already exist. To update these CIs you need to perform the next step.
- 2) Create an enrichment rule which will fill the Tenant Owner attribute for each CI in this local CMS. This is necessary so that when you synchronize CIs from the local CMS to the central CMS, the CIs will be associated with the correct tenant.
 - a. In each tenant's CMS, access **Modeling > Enrichment Manager**.

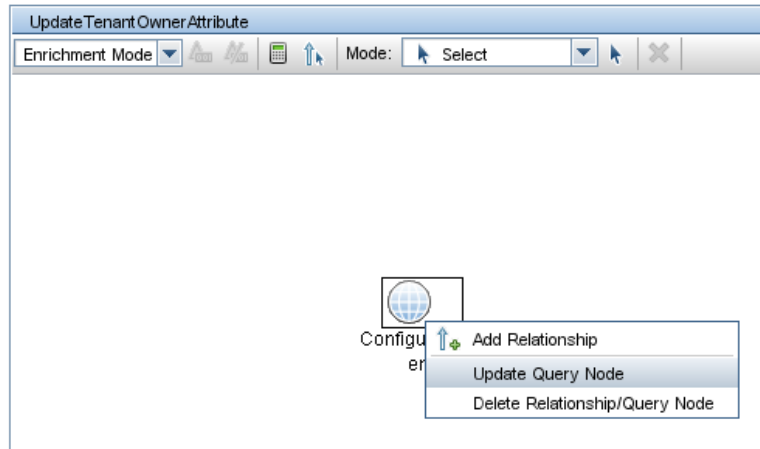
- b. Create a new enrichment rule with a meaningful name. Activate the rule, and select **Base the Enrichment on a new Query**, as seen in the following images:

The screenshot shows the 'New Enrichment Rule' wizard in the 'Rule General Attributes' step. The title bar reads 'New Enrichment Rule'. Below the title, there is a yellow asterisk icon and the text 'Rule General Attributes' and 'Fill in the general attributes of the new enrichment rule.' On the left, a 'Steps' pane shows 'Rule General Attributes' as the current step, with 'Rule Base Query' and 'Wizard Completion' below it. The main area contains a 'Rule Name' field with the text 'Update Tenant Owner Attribute', a 'Rule Description' text area, and a checked checkbox labeled 'Rule is Active'. At the bottom, there are navigation buttons: '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'.

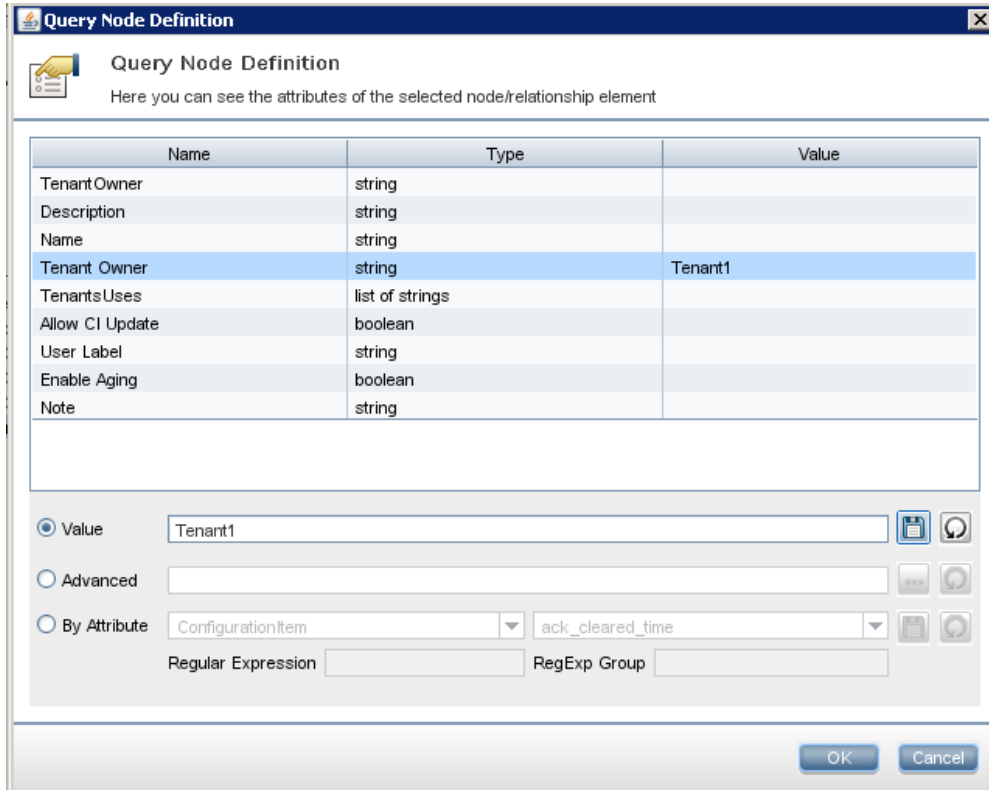
The screenshot shows the 'New Enrichment Rule' wizard in the 'Rule Base Query' step. The title bar reads 'New Enrichment Rule'. Below the title, there is a yellow asterisk icon and the text 'Rule Base Query' and 'Fill in the information about the base query of the new enrichment rule.' On the left, a 'Steps' pane shows 'Rule Base Query' as the current step, with 'Rule General Attributes' and 'Wizard Completion' below it. The main area contains a 'Base Query Type' section with two radio buttons: 'Base the Enrichment on a new query' (selected) and 'Base the Enrichment on an existing query'. Below this is a 'Base Query Attributes' section with three fields: 'Base Query Name' (text field with 'UpdateTenantOwnerAttribute'), 'Base Query Description' (text area), and 'Base Query Priority' (dropdown menu with 'Medium' selected). At the bottom, there are navigation buttons: '< Back', 'Next >', 'Finish', 'Cancel', and 'Help'.

- c. Click **Finish**.
d. In **Query Mode**, select the **Configuration Item CI** type.

- e. Within **Enrichment Mode**, right-click **Configuration Item** and select **Update Query Node**:



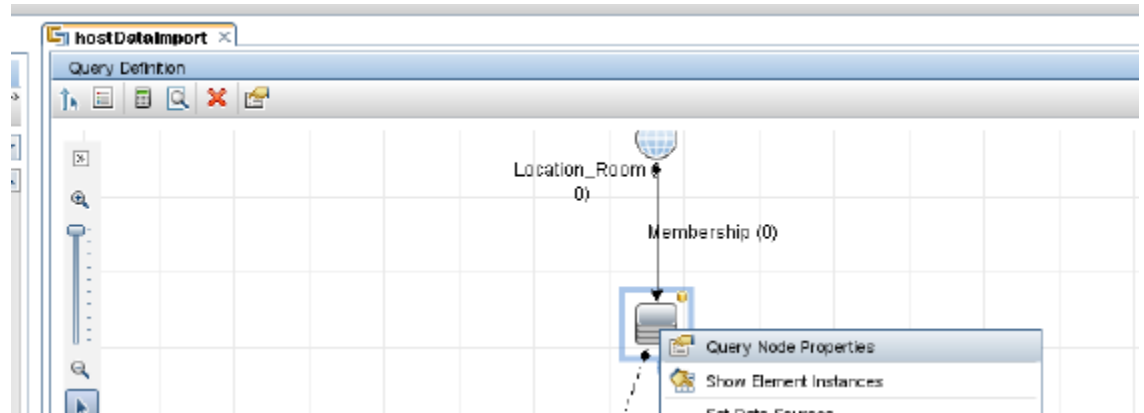
- f. Select the attribute you want to update; enter its value, and click **Save**:



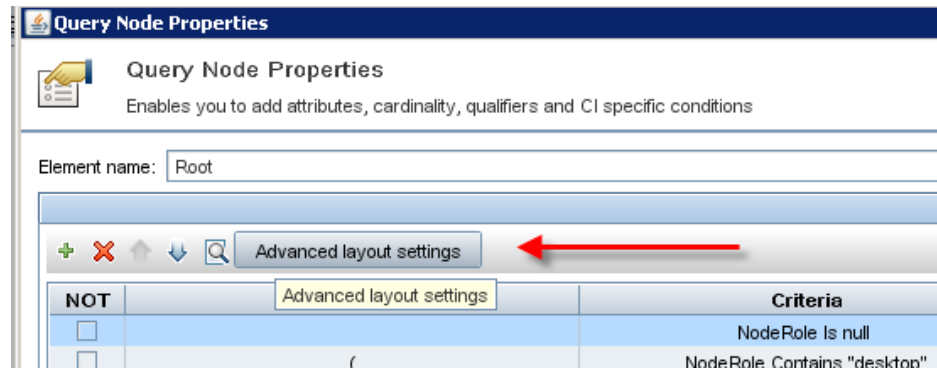
- g. Click **Save** to save the enrichment rule and activate it.
 h. You can validate the enrichment results in IT Universe Manager.

- 3) Modify the Integration TQLs that you will use to synchronize topology from the local CMS to the central CMS, to also synchronize the Tenant Owner attribute:
- On each tenant's CMS, within **Modeling Studio > Resource Type** access **Queries**.

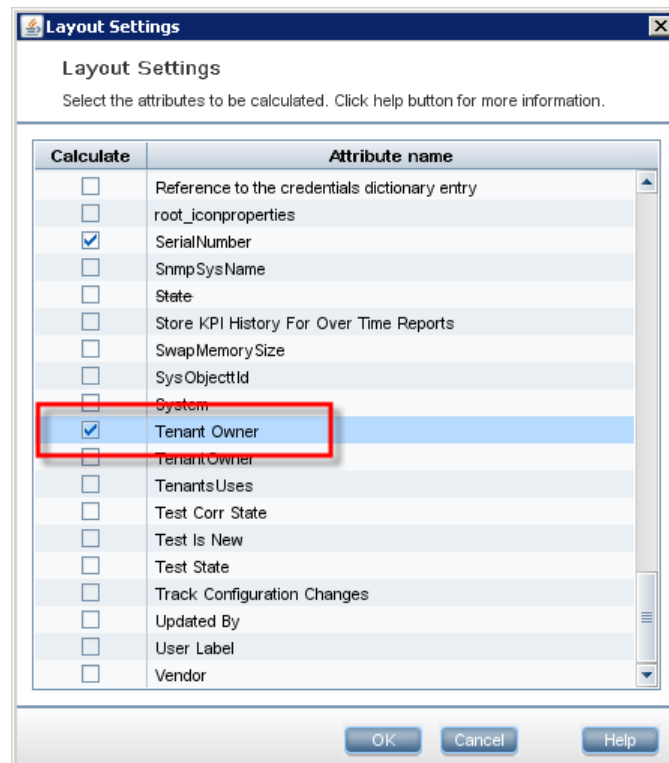
- b. In each Integration TQL, right-click each of the query nodes and select **Query Node Properties**:



- c. Click **Advanced layout settings**:



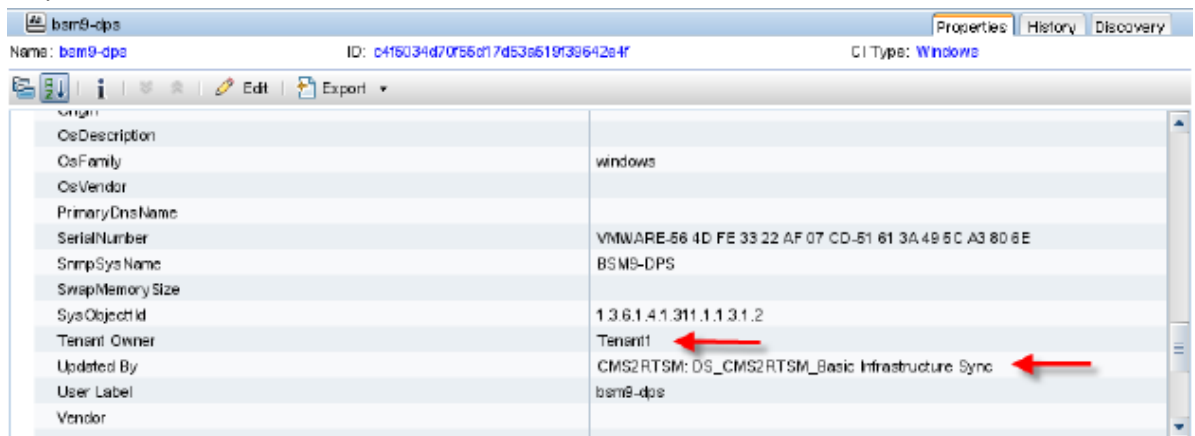
- d. Add the attribute that you want to synchronize:



- e. Click OK to close the **Layout Settings** dialog, then click OK to close the **Node Properties** dialog.
 - f. Repeat the above step b - step e for each element in the query, and then click **Save** in the left pane.
- 4) Within the central BSM, define an integration point to the tenant's CMS in order to synchronize CIs into BSM:
- a. Within the central BSM, access **Admin > RTSM Administration > Data Flow Management > Integration Studio**.
(Note that you must connect the Data Flow Probe to BSM *before* performing these steps, and configure credentials to CMS in Probe administration. This is done using the Data Flow Probe set up; for details refer to the RTSM Data Flow Management pdf.)
 - b. Create a new integration point:

- c. Add new jobs; for each job select the Integration TQL for which you configured Layout Settings in step 4 above.
- d. Save the integration point, and perform full synchronization.

- e. Within IT Universe Manager, check that you received CIs from the local CMS, and that they contain the Tenant Owner attribute:



Note: In your deployment scenario, if each tenant's DDM will be connected directly to the central BSM, you need to employ a different approach to enrich the Tenant Owner attribute in BSM, such as using an Enrichment Rule.

2.3 Configuration for End User Monitoring

You can configure EUM-level permissions for users to restrict access to tenant-specific applications (see section 2.6). However, to separate the applications and application views in RTSM, you must add the **TenantOwner** attribute *manually* for all the Business Applications. This needs to be done once for each Business Application CI; it can be done before or after creating the Business Application configuration in EUM Administration.

- If you prefer to do this before EUM configuration, create the Business Application in **Admin > RTSM Administration > Modeling Studio**, and fill the **TenantOwner** attribute with the relevant value. In EUM Administration you will edit the CI and add BPM/RUM configuration.
- If you prefer to do this after EUM configuration, edit the Business Application CI's properties in **Admin > RTSM Administration > IT Universe Manager**, and fill the **TenantOwner** attribute with the relevant value. If you will use naming conventions for the Business Applications (such as using tenant name as suffix), you can easily assigning the TenantOwner attribute with an enrichment rule.

Once the application is configured for monitoring, you will use the Custom event template to add the TenantOwner attribute for automatic assignment of events to the respective event consoles (see section 2.4.2).

2.4 Configuration for OMi

To enable each tenant to only see their relevant events, you will assign a custom attribute to events. This custom attribute will be used for filters, and for auto-assignment of event to groups. There are several ways to assign this custom attribute, depending on the event source:

2.4.1 Enriching events which are forwarded from OM

If you cannot set the custom attribute within the OM monitoring policy directly (or the integration policy when using an integration adapter), you can enrich incoming events with tenant information on the OMi server as follows:

1. Access **Admin > Operations Management > Tune Operations Management > Event Processing Customization**.
2. Select **Before CI/ETI Resolution**, and add a new script.
3. Fill the Groovy script form as specified in the image below. Note that you must add each OM server and each tenant separately. For performance considerations, we recommended doing this in one script. A similar approach can be used for events received via the Integration Adapter, where the **source ci hint** can be used in the **if** statement.

AddTenantOwner - Create New Script

General

ID: 3b452c2d-f744-f021-d9fd-1ae3d7fcef1

Name: * AddTenantOwner

Description: Add TenantOwner attribute for events which originated from OM servers

Script: *


```

NodeInfo originatedServer = event.getOriginatingServer
String serverDnsName = originatedServer.getDnsName
if ("omServer1.hp.com".equals(serverDnsName))
    tenantOwner = "TenantGroup1";
else if ("omServer2.hp.com".equals(serverDnsName))
    tenantOwner = "TenantGroup2";
// else if ... for each OM server/Tenant info

event.addCustomAttribute("TenantOwner", tenantOwner);
    
```

Active:

< Back Next > Finish Cancel Help

Here is the basic script:

```

import java.util.List;
import com.hp.opr.api.scripting.Event;
class GroovyScriptSkeleton
{
    def init()
    {

    }

    def destroy()
    {

    }

    def process(List events)
    {
        events.each {Event event ->
            String tenantOwner = null;
        }
    }
}
    
```

```

NodeInfo originatedServer = event.getOriginatingServer();
String serverDnsName = originatedServer.getDnsName();
if ("omServer1.hp.com".equals(serverDnsName))
    tenantOwner = "TenantGroup1";
else if ("omServer2.hp.com".equals(serverDnsName))
    tenantOwner = "TenantGroup2";
// else if ... for each OM server/Tenant info

If (tenantOwner != null)
    event.addCustomAttribute("TenantOwner", tenantOwner)
}
}
}

```

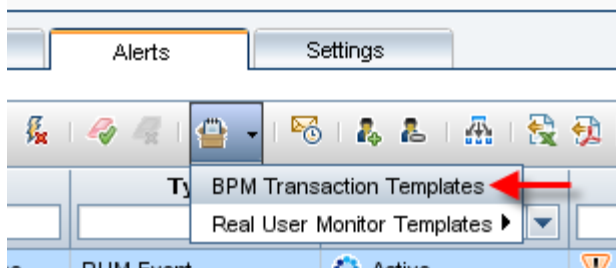
4. Click **Next**.
5. You can specify additional elements for your script in the **Advanced Tab** (for example, you can add an external jar which will be used by the script or events filter), or click **Finish**.

2.4.2 Enriching events which are generated from within BSM

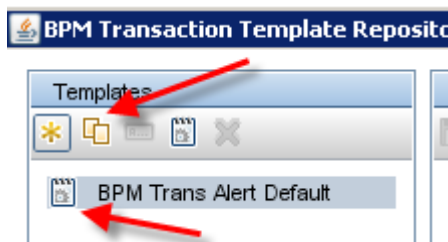
Some events are generated from within BSM, such as alert actions, health indicator (HI) status changes, or events from SiteScope. For these events you must add the custom attribute in the event template, as described in the following section.

(This example is for BPM Alerts, but the procedure is similar for other events generated from within BSM.)

1. Within **Admin > End User Management > Alerts**, click **BPM Transaction Templates**:



2. Select the **BPM Trans Alert Default** template (marked with the Default icon), and click **Duplicate selected template**:



3. Edit the duplicated template, and rename the template name to be unique for a tenant.

- In the **Custom Attributes** tab, add a new key. The key name should be **TenantOwner** (as in the Groovy script), and the value should be the tenant name:

BPM Trans Alert Default(1)

Properties

* Template name

Set the selected template to be the default template

General Custom Attributes

Name	Value
Transaction Name	<<Transaction Name>>
Location Name	<<Location Name>>
TenantOwner	Tenant1

- Save your changes.
- When you create a BPM transaction alert on a tenant's business application, make sure you select the correct event template:

Definition Details

when trigger conditions occur [even once](#).

Actions

Generate Event with [BPM Trans Alert template for Tenant1](#) template and [End User event:Critical](#) Event Type Indicator

Notification Frequency

Send alert for every triggered occurrence

2.4.3 Automatically assigning events to user groups

As part of the solution, role-based access will be used to configure and provide access permissions to RTSM, Service Health, OMi and EUM for the respective tenants; for details see section 2.6. Events which are assigned to these groups/users contain associated permissions; events which are not assigned to them are not displayed.

Perform the following steps to automatically assign events to user groups based on the TenantOwner attribute:

- Within **Admin > Operations Management > Tune Operations Management > User Group Assignments**, create a new assignment rule.
- Create a new event filter by clicking **Browse... > New Event Filter**.
- Provide a meaningful name for the filter.

4. In the **Additional Event Properties** tab, mark the **Custom Attribute** checkbox. Add **TenantOwner** as the attribute name, and enter the tenant name as its value (as you did in the Groovy script and event templates):

The screenshot shows a 'Filter Configuration' dialog box with the following fields and settings:

- Filter Display Name: * Tenant1 events
- Filter Description: (empty)
- General tab: (selected)
- Dates tab: (selected)
- Additional Event Properties tab: (selected)
- Application: equals []
- Object: equals []
- Key: equals []
- Original Data: equals []
- CI Type: derived from [] ...
- Solution: equals []
- Custom Attribute: TenantOwner equals [Tenant1]

Buttons at the bottom: OK, Advanced Mode, Cancel, Help

5. Click **OK**, and select your event filter.
6. In the assignment rule, select a user group or user that represents the tenant (creating user groups is explained in section 2.6):

Tenant1 - Create New Event Assignment Rule

Filter: * Tenant1 events

User Group: * Tenant1

Description: Automatically assign Tenant1 events to the relevant user group

(* Required field)

OK Cancel Help

7. Click **OK**, and save your rule.

2.4.4 Viewing the Tenant Owner attribute in the Event Console

To enable a super-admin who is working across tenants to see the **Tenant Owner** event attribute as a column in the Event Console, configure the following infrastructure setting:

1. Within **Admin > Platform Administration > Infrastructure Settings**, select **Applications: Operations Management**.
2. In the section **Operations Management - Custom Attribute Settings**, edit the setting **Available Custom Attributes**.
3. Add a semicolon (;), and then type **TenantOwner**. Save your changes.

You can then add a column that will show the tenant owner attribute in the Events Console.

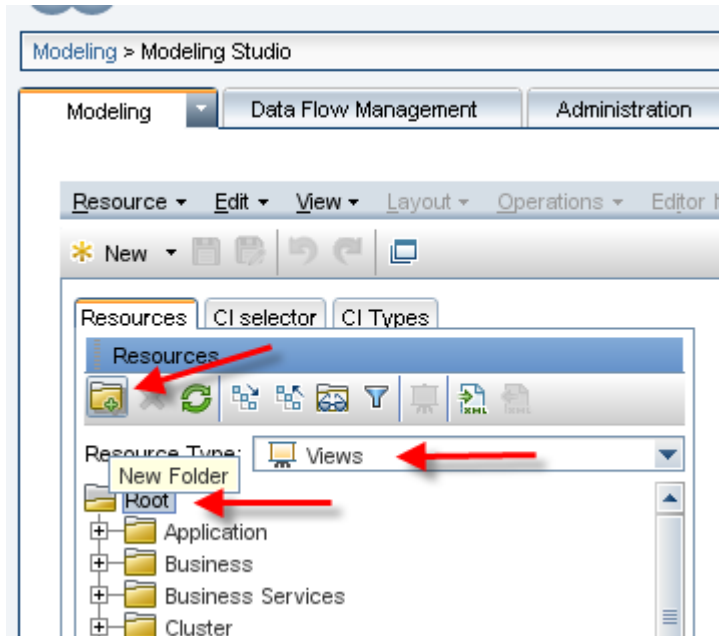
You can also create Event Browser filters to only view events from a specific tenant. You can reuse the same filters you created in section 2.4.3.

A useful filter for the super-admin can be to show all events whose tenant owner attribute is empty.

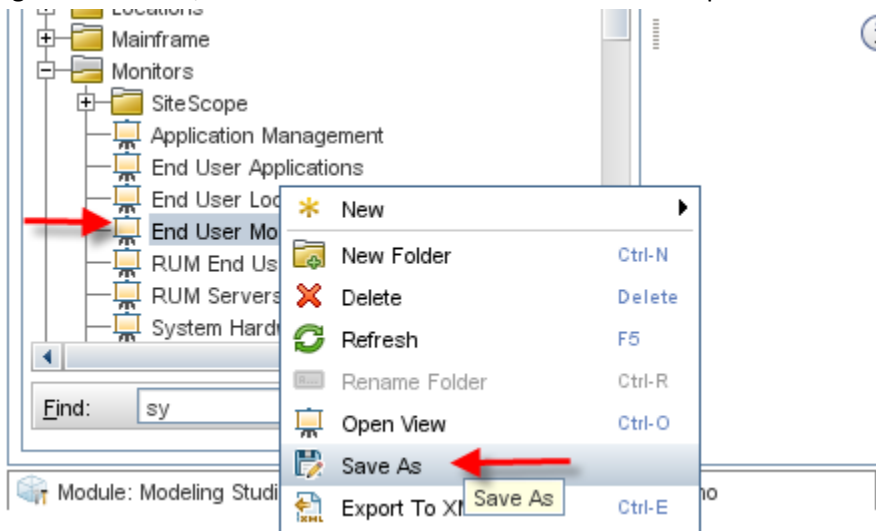
2.5 Create Views in BSM

Use the following procedure to create different pattern-based views for different tenants, and to add conditions to the CITs in each view to collect data relevant to the tenants. In this procedure, the TenantOwner attribute is used to identify and limit the data relevant to the tenants, and user level permissions are configured for the respective tenants.

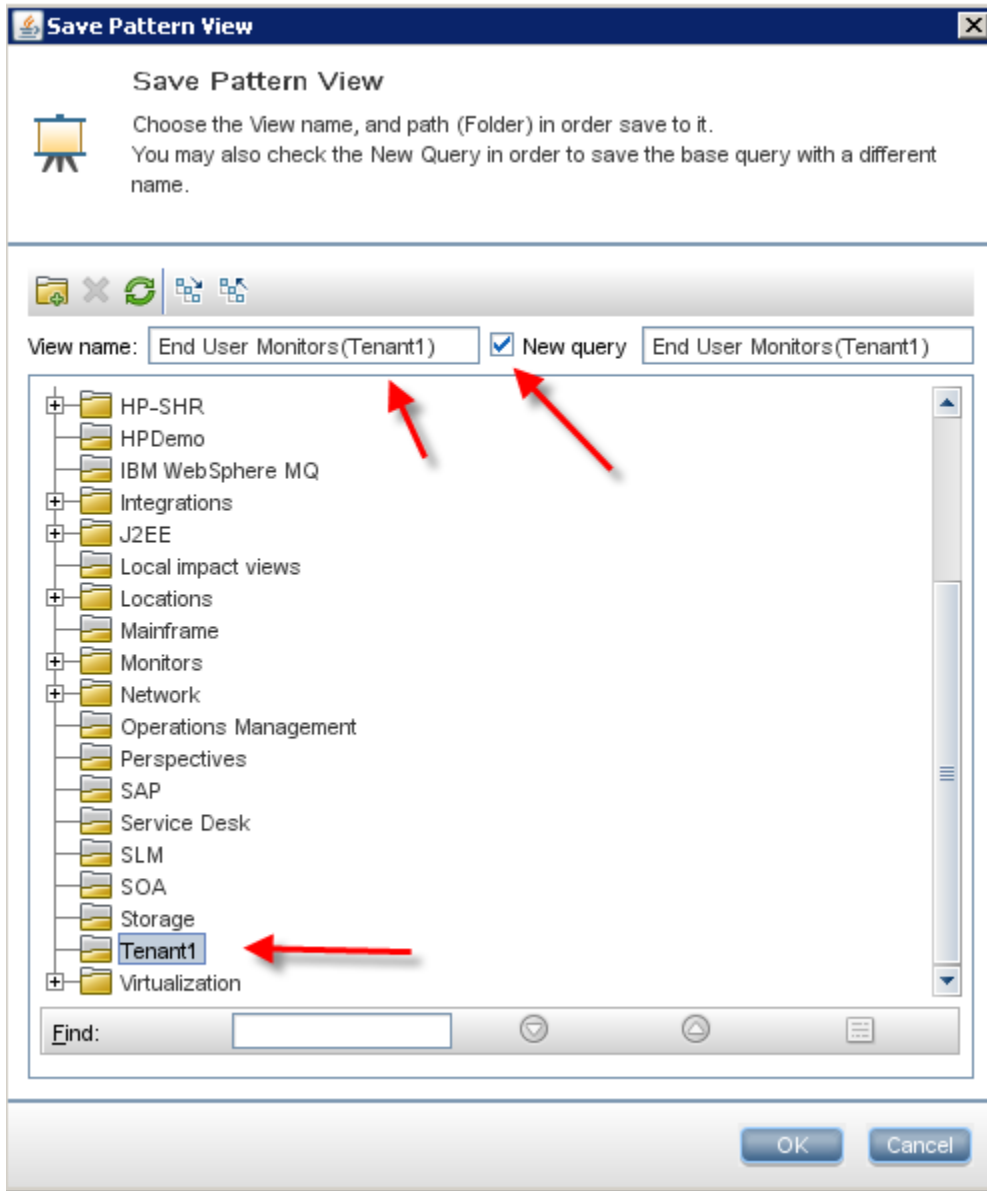
1. Within **Admin > RTSM Administration > Modeling Studio**, create a folder for each tenant. This will hold all the views belonging to that tenant and make it easy to manage the views.



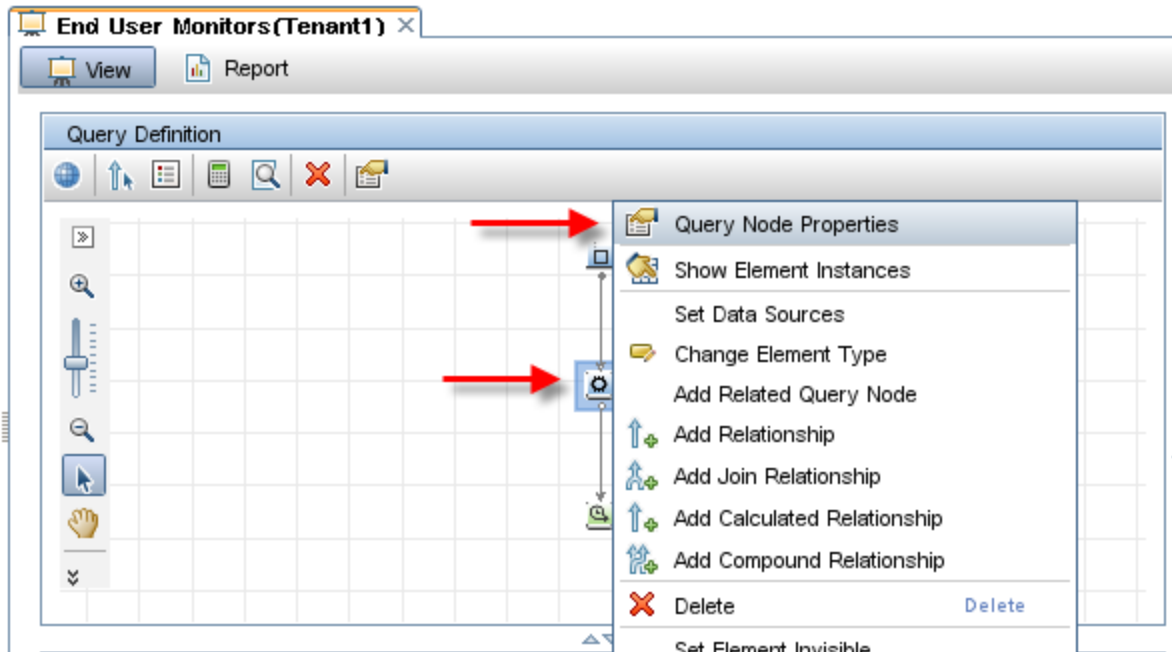
2. Right-click a view, and select **Save As** to create a view for a specific tenant.



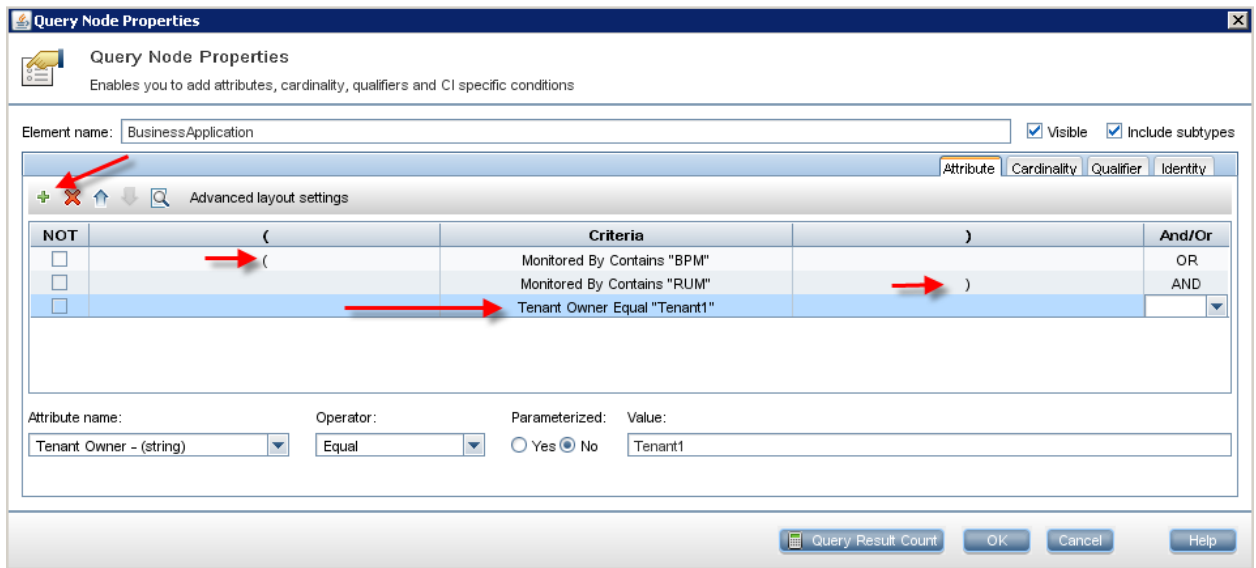
3. Give the new view a relevant name and save it in the correct folder. Make sure you select the **New query** checkbox.



4. Select the cloned view, and edit the node which will have tenant owner information for filtering. In this example, we will edit the Business Application CI (see section 2.3):



5. Add a new property condition; select the TenantOwner attribute, and enter the relevant value. Do not harm the logic of the existing condition (if it exists); if necessary add parentheses to maintain the existing condition:



6. Click **OK**.
7. When you finish filtering the query nodes, click **Preview** to validate the view results.
8. When you finish validating, click **Save**.

Note: You might want to create a view which shows all the CIs that do not have the TenantOwner attribute defined as expected, or that are connected to such CIs. Such a view is very useful for the super-admin.

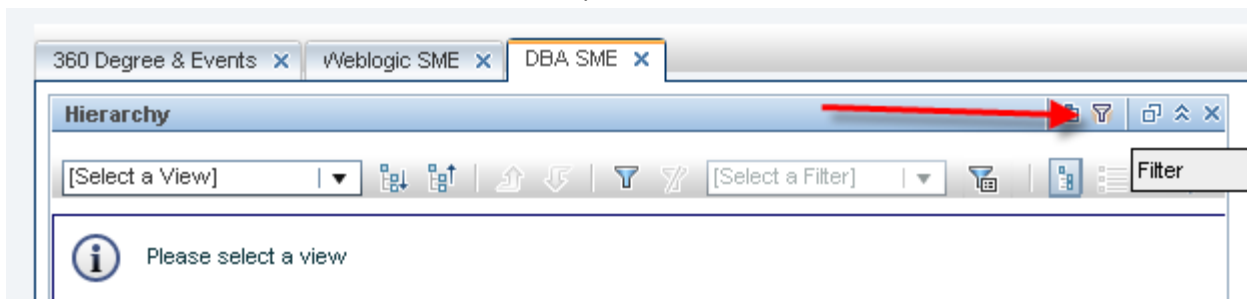
2.6 Create MyBSM Pages

You can create different MyBSM pages for different tenants. These pages can be filtered to specific views in order to differentiate tenants' workflows and provide each tenant with specific Dashboards. You can also configure permissions for each page, so users will not be able to modify the pages.

Components which contain a view selector are only populated with the views that a user has permissions to access.

You can also create new URL-based components in MyBSM, for example using a tenant's knowledge base portal. You can then assign permissions for these components (see section 2.7).

1. Within MyBSM, create a new page or open an existing page and save it with a new name.
2. Build your page by adding the relevant components.
3. Click **Filter** on a Service Health, SLM or EUM component.



4. Select the relevant filter criteria (view, SLAs or Business Applications) and click **OK**.
5. Click **Save**.

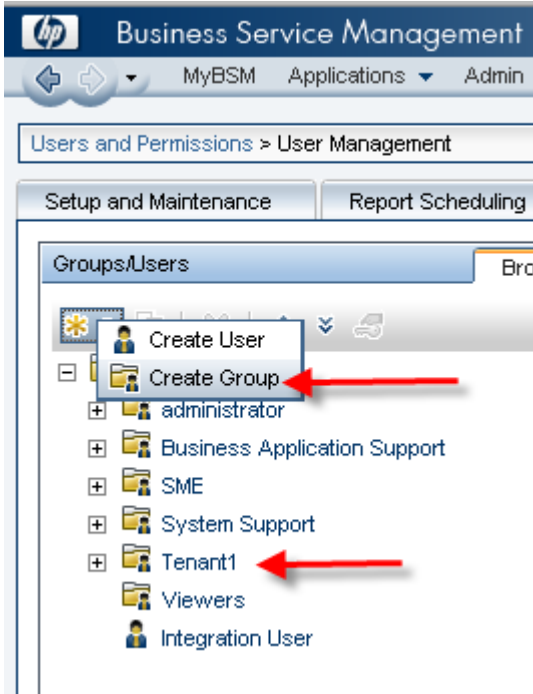
For more information on working with MyBSM, refer to the *Using MyBSM* pdf.

2.7 Configure Users and Permissions

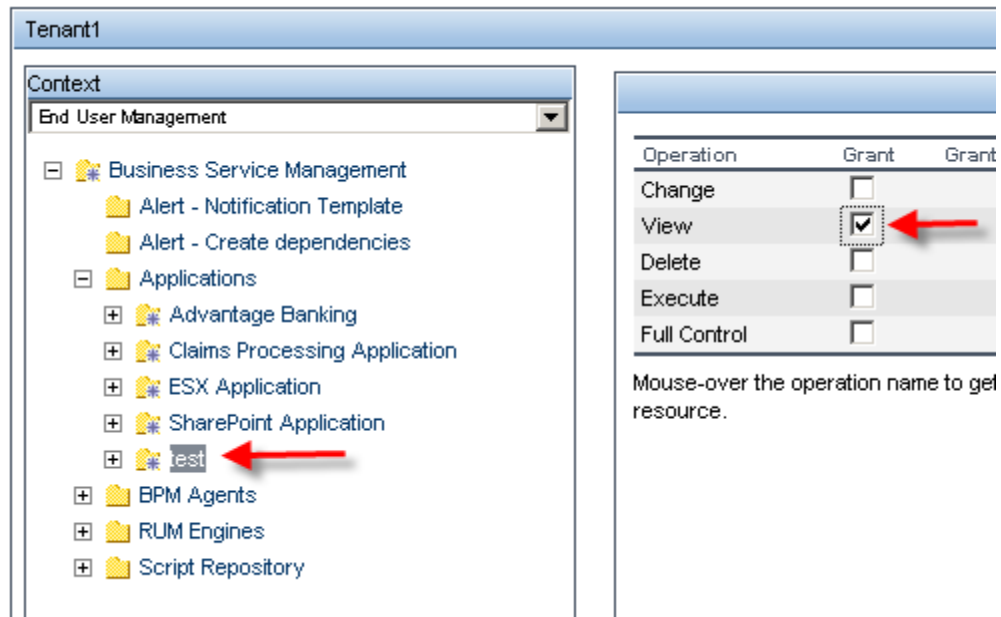
Use the following procedure to define group-level role-based permissions. You will use user groups to identify tenants and distinguish between users of each tenant. By defining permissions on the user group level, you avoid managing permissions on the user level. Moreover, when a new user is added to a group, it inherits permissions from the group. If a user belongs to two groups, the user gains access to events and views of two tenants.

1. Access **Admin > Platform Administration > Users and Permissions > User Management**.

2. For each tenant, create a group:



3. Edit permissions for each group you create, as follows:
 - a. Select the group.
 - b. Select the **Permissions** tab; click **Apply Permissions** after each modification.
 - c. Select the **End User Management** context, and assign operations for each element (for example, you can grant View permissions for the tenant's applications only:



- d. Select the **Operations Management** context. Make sure that **Events not assigned to user** is not selected, and **Events assigned to user** is selected. You can also set permissions on tools, actions and other administration tasks.

- e. Select the **RTSM** context and assign View permissions to the specific views you created for this tenant. Make sure other tenants' views are not selected. You will probably want to de-select the out-of-the-box views for this tenant, they are Pattern views which are not filtered for specific tenant owner.
- f. Select the **Service Health** context and assign permissions to MyBSM pages for this tenant.
- g. If you created SLAs for this tenant, select the **Service Level Management** context and assign permissions for these SLAs.
- h. If you create custom, trend or service reports for this tenant, select the **User Defined Reports** context and assign permissions for each category for this tenant.

For more details on users and permissions, refer to the *BSM Platform Administration* pdf.

2.8 Multi-Tenancy and Other Data Collectors

Some data collectors support multi-tenancy more easily than others:

- 1) **BPM**. BSM can handle multiple BPM agents, which means that you can have different BPM agents for different tenants. This allows fully segregation between tenants.
- 2) **RUM**. BSM can handle multiple RUM engines and each engine can handle multiple RUM probes. This allows fully segregation between tenants.
- 3) **SiteScope**. BSM can handle multiple SiteScope instances, so you can have different SiteScope instances for different tenants. Since SiteScope creates topology in RTSM, you can also use the RoutingDomain attribute to segregate the topology that each SiteScope creates. You can populate the TenantOwner attribute using enrichment, or you can use the SiteScope Profile CI in your enrichment.
- 4) **Diagnostics**. Only one Diagnostics server/commander can be connected to BSM. You can connect multiple Diagnostics probes to the server/commander, but the data and topology that are sent into BSM will not have any tenant information; you must assign topology and data to specific tenants manually. You can assign permissions on the Application within User Management, in the context **Transaction Management**.
- 5) **NNMi**. Only one NNMi server can be connected to BSM. Future versions of NNMi will enable multi-tenancy, but currently you must manually assign the topology to tenants (or using enrichment, for example based on IP address ranges). This is also true for events with more complicated Groovy script.
- 6) **BSM Connector/IA**. BSM can work with multiple connectors, which means you can have a different connector for each tenant. You can also customize topology, and include the TenantOwner attribute in the topology sent by the connector. You can also customize events and include the TenantOwner attribute in the event itself.