



Business Intelligence Platform Multitenancy Guide

■ Business Intelligence Platform Multitenancy Guide 4.1 SP1

2013-08-03

Copyright

© 2013 SAP AG or an SAP affiliate company. All rights reserved. No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice. Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors. National product specifications may vary. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty. SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries. Please see <http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark> for additional trademark information and notices.

2013-08-03

Contents

Chapter 1	Document history.....	5
Chapter 2	Introduction.....	7
Chapter 3	Quick start.....	9
3.1	Installation requirements.....	9
3.2	Create the tenant template.....	10
3.2.1	To create a new user group for a tenant.....	10
3.2.2	To create a new folder for a tenant.....	11
3.2.3	To set access rights for a tenant.....	12
3.3	Define the tenant configuration file.....	13
3.4	Run the Multitenancy Management Tool	14
3.5	Confirm that the tenant was created.....	15
Chapter 4	Setting up a tenant template.....	17
4.1	Creating a token phrase.....	17
4.2	Adding public folders to the tenant template.....	18
4.3	Adding user groups to the tenant template.....	18
4.4	Including event folders in the tenant template.....	19
4.5	Including categories in the tenant template	19
4.6	Including profiles in the tenant template	20
4.7	Including security settings in the tenant template	20
4.8	SAP Crystal Reports 2011.....	21
4.9	SAP Crystal Reports for Enterprise	24
4.10	Managing universes and connections.....	26
4.10.1	Unshared universes and connections.....	26
4.10.2	Shared universes and connections.....	30
4.10.3	Restricting data access for shared universes (classic UNV universes only).....	31
Chapter 5	Running the Multitenancy Management Tool	33
5.1	Command line options.....	33

5.2	Tenant definition configuration file reference.....	34
5.2.1	Mandatory Configuration.....	38
5.2.2	Tenant on-boarding/provisioning configuration options.....	47
5.2.3	(Database) Type specific configuration options.....	83
5.2.4	tenant_template_def.properties.....	89
Chapter 6	Managing tenants in the CMC.....	93
6.1	To set tenant properties	93
6.1.1	To change the name of the tenant	94
6.1.2	To change the description of the tenant	94
6.1.3	To change keywords for the tenant	94
6.1.4	To change the number of concurrent users for a tenant	95
6.2	To assign access rights to a tenant user group	95
6.2.1	To remove access rights from a tenant.....	96
6.3	Managing user groups for a tenant.....	96
6.3.1	To view user and group associations for a tenant.....	96
6.3.2	To add a user group to a tenant.....	97
6.3.3	To remove a user group from a tenant.....	97
6.4	To delete a tenant.....	98
Chapter 7	Troubleshooting.....	99
7.1	Multitenancy Management Tool errors	100
7.1.1	Multitenancy Error Messages	106
7.2	To configure tracing for the Multitenancy Management Tool	116
Appendix A	More Information.....	119
Index		121

Document history

The following table provides an overview of the enhancements made to this document.

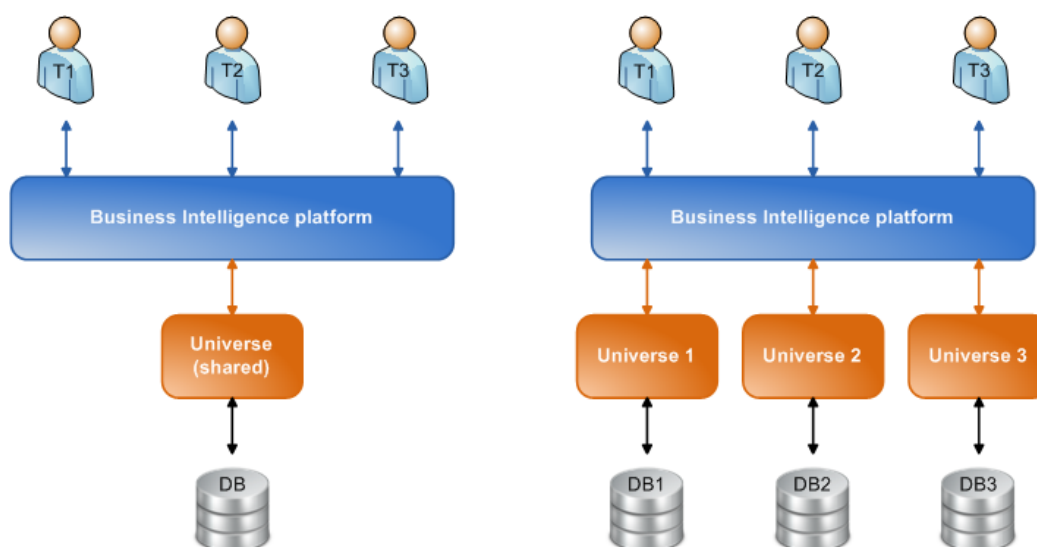
Version	Date	Description
SAP BusinessObjects Business Intelligence Suite 4.1	November, 2012	First release of this document.
SAP BusinessObjects Business Intelligence Suite 4.1 SP1	August, 2013	

Introduction

The SAP BusinessObjects Business Intelligence platform Multitenancy Management Tool is a Java-based program that SAP OEM partners can use to standardize and automate the creation of objects and settings for new customers in a multitenant BI platform deployment.

In a multitenant deployment, multiple tenants are hosted on a single instance of BI platform. Tenants can view their own data, but other tenants cannot see it. For example, a tenant deployment may include:

- Tenants that share the same universe and database.
- Tenants that use different universes and databases.



In most cases, you manually configure BI platform for each tenant. For example, you can do the following:

1. Create separate user groups for each tenant.
2. Create separate folders for tenant documents.
3. Restrict universe data views for each tenant.
4. Apply appropriate security settings to isolate each tenant's experience.

The Multitenancy Management Tool automates these and other steps to make it easier to create new tenants.

Who should use this information?

This section is for content and system administrators who are responsible for configuring, managing, and maintaining a multitenant BI platform installation. You'll need to be familiar with the basic concepts

and tools used to manage a BI platform installation. An understanding of how to design universes for reports and analytics may also be required, depending on your tenant deployment requirements. For all levels of administration, this section provides sufficient background and conceptual information to clarify all administrative tasks and features.

For information about setting up the security and server infrastructure of BI platform, see the *SAP BusinessObjects Business Intelligence Platform Administrator Guide*.

For information about managing, scheduling, and distributing BI content in the repository, see the *SAP BusinessObjects Business Intelligence Platform User Guide*.

For information about designing classic universes (`.unv` files), see the *SAP BusinessObjects Universe Design Tool User Guide*.

Terminology in this guide

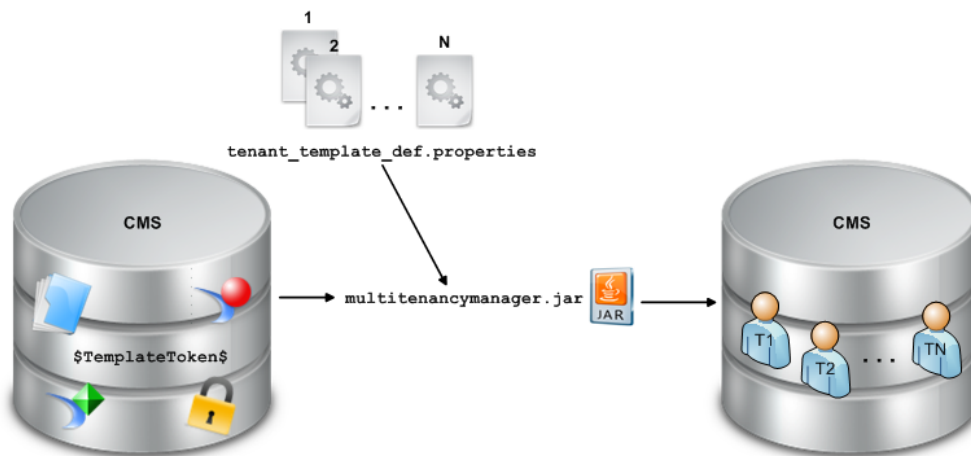
Term	Definition
Tenant	A customer of an SAP OEM partner who shares a single instance of the BI platform infrastructure and services with other customers, but data and the user experience are kept separate.
Tenant template	A collection of repository objects, rights, and settings that are the structural basis for creating new tenants in a multitenant BI platform installation.
Template token	A string that identifies a tenant template object or setting. The Multitenancy Management Tool replicates the template token when creating a new tenant.
Tenant definition configuration file	A Java properties file (<code>tenant_template_def.properties</code>) that you use to define options for tenants, before running the Multitenancy Management Tool.

Quick start

This quick start provides a series of tasks to get you up and running with the Multitenancy Management Tool in BI platform. The tasks help familiarize you with the essential steps required to create new tenants based on a template.

Before you can use the Multitenancy Management Tool, these conditions have to be met. Links to more detailed instructions and information are provided throughout this section where necessary.

1. The tool must be installed by default or selected during a custom installation.
2. You must create a collection of objects and settings in BI platform that serve as a tenant template.
3. For each new tenant, you must create a tenant configuration file (copied from `tenant_template_def.properties`).
4. You must run the Multitenancy Management Tool to create each tenant. The tool uses the tenant template described in step 2, and the settings defined in the tenant configuration file.



This quick start shows how to create a new tenant template (includes one BI platform user group, two public folders, and granted rights on a folder), how to set up a tenant configuration file, then you learn how to run the Multitenancy Management Tool to create a new tenant.

3.1 Installation requirements

To use the Multitenancy Management Tool, you must have the following software installed:

- JRE 1.6.

- SAP BusinessObjects Business Intelligence platform 4.1 or later.

The Multitenancy Management Tool is installed by default with BI platform and is located in the `\java\apps\` folder:

- **Windows:** `<InstallDir>\SAP BusinessObjects Enterprise XI 4.0\java\apps\multitenancyManager\jars\`
- **Unix:** `<InstallDir>/sap_bobj/enterprise_xi40/java/apps/multitenancyManager/jars/`

If you did a custom installation, you add the tool by modifying your installation and selecting the following feature:

- **Windows:** **Administrator Tools > Multi-tenancy Manager**
- **Unix:** `Administrator Tools > MultitenancyManager`

For detailed instructions on modifying the installed features in your BI platform deployment, see the *SAP BusinessObjects Business Intelligence Platform Installation Guide*.

3.2 Create the tenant template

For this quick start example, create the following template objects and settings in the Central Management Console (CMC):

- A user group named "\$TemplateToken\$"
- A public folder named \$TemplateToken\$ at the root level
- A public folder structure `_tenants/$TemplateToken$_temp` at the root level
- Assigned rights to the \$TemplateToken\$ folder

To begin, start the CMC and log on with appropriate credentials to create objects. By default, you can start the CMC with the `http://<WebServerName>:8080/BOE/CMC` URL.

Related Topics

- [Setting up a tenant template](#)

3.2.1 To create a new user group for a tenant

1. In the **Users and Groups** management area of the Central Management Console (CMC), select **Manage > New > New Group**.
The "Create New User Group" dialog box appears.
2. Type the group name \$TemplateToken\$ and a description.
3. Click **OK**.

A template user group is created for the tenant.

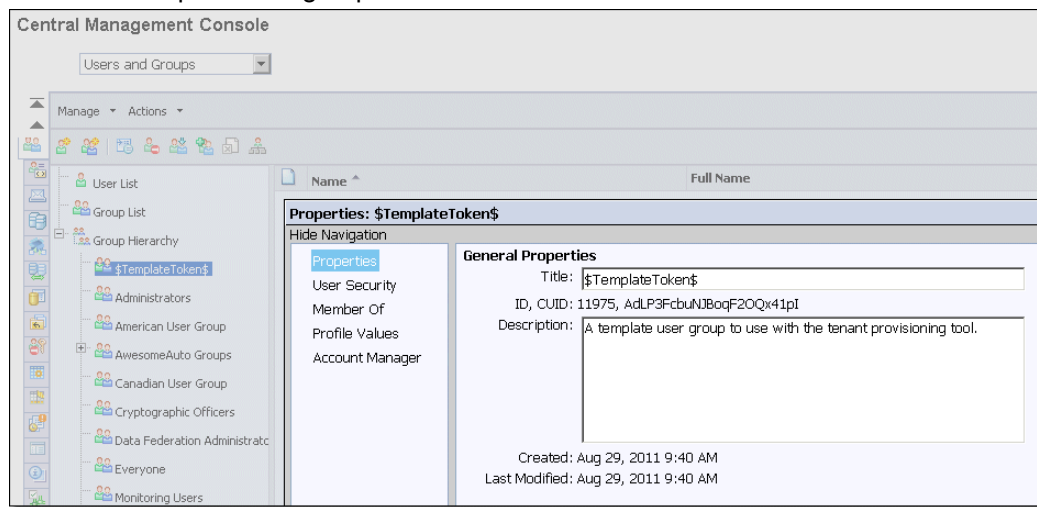


Figure 3-1: Properties: \$TemplateToken\$ dialog box for creating a tenant user group (English example)

3.2.2 To create a new folder for a tenant

1. In the **Folders** management area of the Central Management Console (CMC), navigate to the root public folder.
2. Select **Manage > New > Folder**.
3. Type \$TemplateToken\$ as the name of the new folder.
4. Click **OK**.
5. Repeating steps 2 to 4, create a folder called `_tenants` and a subfolder called `$TemplateToken$_temp` inside the `_tenants` folder.

The new folders appear in the list of folders and objects for the tenant.

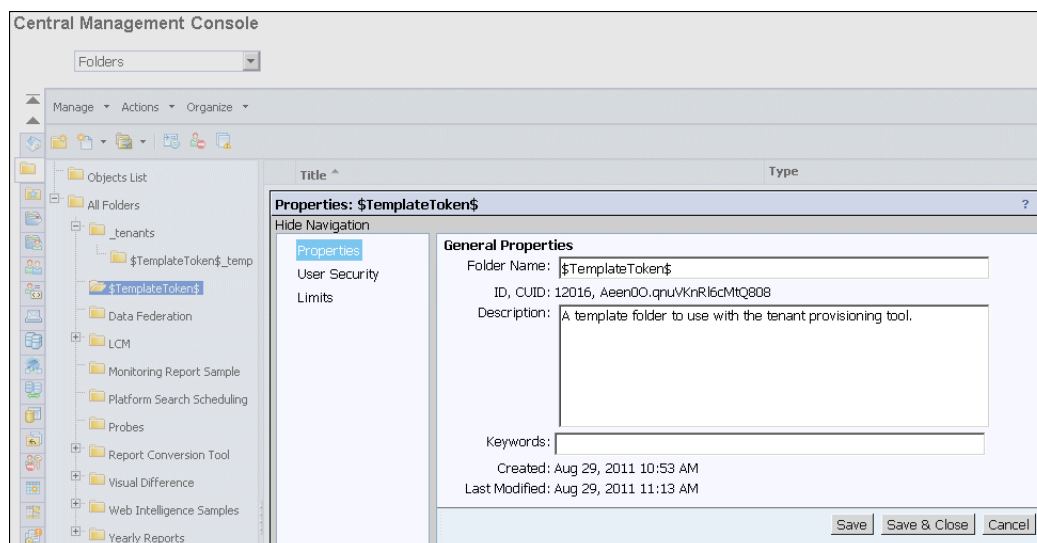


Figure 3-2: Properties: \$TemplateToken\$ dialog box for creating a tenant folder (English example)

3.2.3 To set access rights for a tenant

1. In the **Folders** management area of the Central Management Console (CMC), select the \$TemplateToken\$ folder.
2. Select **Manage > User Security**.
The "User Security" dialog box appears. A list of names of individuals and groups that have been granted access is displayed.
3. Click **Add Principals**.
4. In the "Add Principals" dialog box, move the template user group "\$TemplateToken\$" from the **Available Users or Groups** list to the **Selected Users or Groups** list.
5. Click **Add and Assign Security**.
6. In the "Assign Security" dialog box, select access levels to grant to the \$TemplateToken\$ user group.
For example, select **View** to give the group "View" access rights.
7. To enable folder inheritance, select the **Inherit From Parent Folder** check box.
To disable folder inheritance, clear the check box.
8. To enable group inheritance, select the **Inherit From Parent Group** check box.
To disable group inheritance, clear the check box.
9. Click **OK**, and click **Close**.
10. Repeating steps 1 to 9, select the \$TemplateToken\$_temp folder and assign rights to it for the template user group.

The template user group has rights assigned to the template folders:

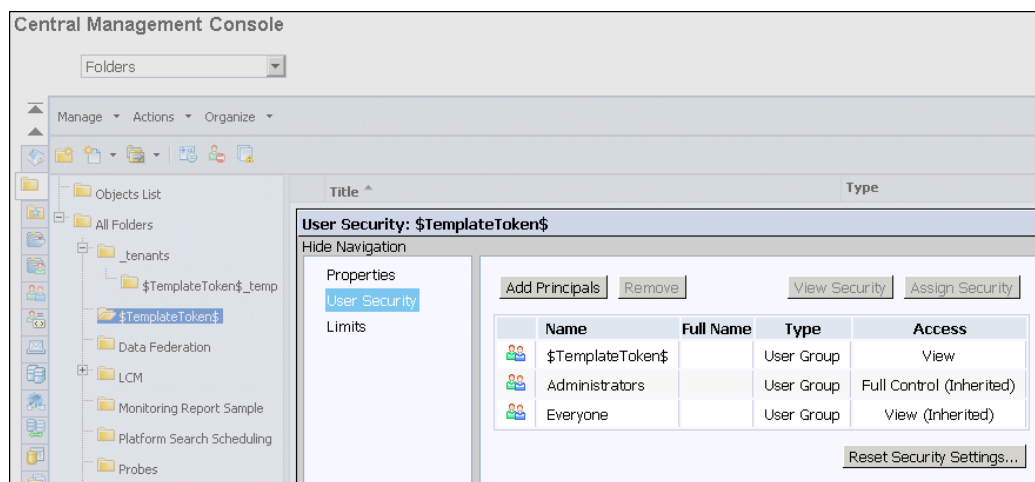


Figure 3-3: User Security: \$TemplateToken\$ dialog box for setting tenant access rights (English example)

3.3 Define the tenant configuration file

Create a copy of the original tenant configuration file (`tenant_template_def.properties`) in the `multitenancyManager` folder. Do not change the original file—use the copy you made to configure tenants.

You must set options in the tenant configuration file that tell the Multitenancy Management Tool to log on to your Central Management Server (CMS), identify template objects by the template token string, and create new copies of the objects with a specific tenant name.

The `tenant_template_def.properties` file is located in the `multitenancyManager` folder at:

- **Windows:** `<InstallDir>\SAP BusinessObjects Enterprise XI 4.x\java\apps\multitenancyManager\jars\`
- **Unix:** `<InstallDir>/sap_bobj/enterprise_xi4x/java/apps/multitenancyManager/jars/`

Note:

If you decide to use the original tenant configuration file to define new tenant configurations (instead of using a copy of the file), back up the original file before making changes to it. During a BI platform upgrade (for example, from version 4.0 FP3 to version 4.1), the original tenant configuration file will be replaced by a new version with added features.

1. In a text editor, open the copied `tenant_template_def.properties` file.
2. Under (Mandatory) Logon information, change values for the `cms`, `auth`, `user`, and `password` options to reflect your authentication credentials for logging on to the CMS.

The `auth` option values are `secEnterprise`, `secLDAP`, `secWinAD`, or `secSAPR3`.

For example:

```
cms=MyCMSDomainName:port
auth=secEnterprise
user=Administrator
pwd=MyPassword
```

3. Under (Mandatory) Name of the tenant being added, change the value for the tenant Name option to the name of the new tenant you are defining.

For example:

```
tenantName=abc_customer
```

4. Under (Mandatory) Template token identifier used for tenant name replacing, change the value for the templateToken option to the string used to identify your template folders and user group.

Use the string "\$TemplateToken\$" as you did in the "Create your tenant template" task. For example:

```
templateToken=$TemplateToken$
```

5. Under (Optional) Document folder template, change the value for the templateContentFolder option to specify the top-level template folder paths.

Separate multiple values with a semicolon, and include the full path of the public root folder.

In this quick start, you've created two template folders with the template token in the name. For example:

```
templateContentFolder=$TemplateToken$;_tenants/$TemplateToken$_temp
```

6. Save and close the tenant_template_def.properties file.

3.4 Run the Multitenancy Management Tool

The Multitenancy Management Tool is installed here by default with BI platform:

- Windows: <InstallDir>\SAP BusinessObjects Enterprise XI 4.0\java\apps\
- Unix: <InstallDir>/sap_bobj/enterprise_xi40/java/apps/
- Go to the apps folder where the Multitenancy Management Tool is located, open a command prompt, and run multitenancymanager.jar with the tenant definition file passed to the -configFile option, using the following syntax:

```
java -jar multitenancymanager.jar -configFile tenant_template_def.properties
```

After the tool finishes running successfully, you can log on to the Central Management Console (CMC) and view the newly-created public folders, user group, and security settings for the new tenant ("abc_customer" in the example).

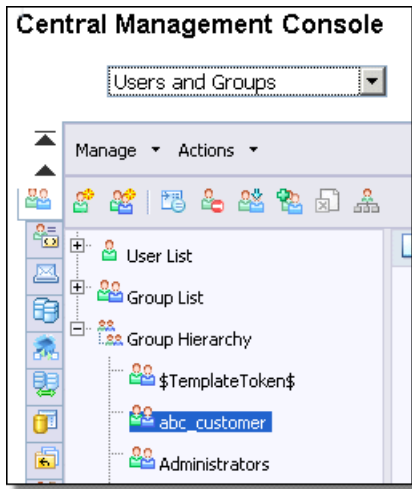


Figure 3-4: New tenant listed in Group Hierarchy in CMC (English example)

The tenant is created in the CMC and a log file named `multitenancymanager<yyyy><mm><dd><time>.csv` is created in the following folder:

- Windows: `<InstallDir>\SAP BusinessObjects Enterprise XI 4.0\logging\`
- Unix: `<InstallDir>/sap_bobj/logging/`

3.5 Confirm that the tenant was created

Once a tenant is configured and added to the BI repository, you can see and manage the tenant in the **Multitenancy** management area of the CMC.

1. To make sure you added a tenant in the quick start, verify that the tenant appears in the main "Multitenancy" window.

The quick start showed how to set up a basic tenant template and create a new tenant with a few simple objects and security settings. However, you can use the Multitenancy Management Tool for more complex scenarios and to replicate many other objects, including universes and report database connection information. This guide explains how to set up all types of template objects and optional configurations in the tenant definition file.

Setting up a tenant template

A tenant template is a collection of repository objects, rights, and settings that serve as the structural basis for creating new tenants in a multitenant BI platform installation. For example, tenants in your deployment may have common traits, such as structurally identical user groups and public folders.

Rather than re-creating the similar objects and settings each time you need to create a new tenant, you can create the objects and settings in a template and then run the Multitenancy Management Tool to create new tenant instances based on the template.

Parts of the tenant template are identified by a template token—a string that identifies an object or setting to be replicated by the Multitenancy Management Tool.

For instructions on creating new objects (such as user groups and folders) in the Central Management Console (CMC), see the *Business Intelligence Platform Administrator Guide*.

Example: Template token

1. Choose the string "\$TemplateToken\$" as the template token.
2. Create a user group in the CMC with "\$TemplateToken\$" in its name, such as "\$TemplateToken\$" or "\$TemplateToken\$_usergroup".
3. Run the Multitenancy Management Tool.

A new user group with the same structure and settings as the template user group is created for the tenant. The token string ("TemplateToken") is replaced with the actual name defined in the tenant configuration file.

4.1 Creating a token phrase

Decide on a token phrase to be used on the objects you wish to have replicate to your tenant template, for example, a user group. Dollar signs "\$" are used to designate the beginning and end of the phrase.

To create a template token phrase, apply it to a user group then provision a new tenant, follow these steps:

1. Choose the string such as "\$TemplateToken\$" as the template token.
2. Create a user group in the CMC with "\$TemplateToken\$" in its name, such as "\$TemplateToken\$" or "\$TemplateToken\$_usergroup".
3. Run the Multitenancy Management Tool as described in the section "Run the Multitenancy Management Tool".

A new user group with the same structure and settings as the template user group is created for the tenant. The token string ("{\$TemplateToken\$}") is replaced with the actual name defined in the tenant configuration file.

Related Topics

- [To create a new folder for a tenant](#)
- [To create a new user group for a tenant](#)

4.2 Adding public folders to the tenant template

You use the **Folders** management area of the Central Management Console (CMC) to add public folders to the tenant template. To add public folders to the tenant template, do the following:

1. Create a folder and include the template token in its name.
2. In a text editor, open the tenant configuration file, and under the `templateContentFolder` option, specify the top-level folders, separating folder names with a semicolon.
3. On a command line, run the Multitenancy Management Tool.

All subfolders and objects contained in top-level folders are copied automatically.

Example: Copying Documents folder and Crystal report to {\$TemplateToken\$} folder

In the following example, after specifying the `{$TemplateToken$}` and `{$TemplateToken$}_temp` folders in the tenant configuration file, the `Documents` folder and the Crystal report will be copied automatically as children of the top-level `{$TemplateToken$}` folder.

```
Public Folders
{$TemplateToken$
  Documents
    Tenant_report.rpt
  Reports Samples
    {$TemplateToken$}_temp
```

Related Topics

- [templateContentFolder](#)

4.3 Adding user groups to the tenant template

User groups are prepared for addition to the tenant template by using the **Users and Groups** management area of the Central Management Console (CMC), and including a template token string

when they are named. To add user groups that include the template token string in their name so they can be added to the tenant template, do the following:

1. Create a new user group with the template token in its name—for example, "\$TemplateToken\$" or "\$TemplateToken\$_usergroup".
2. Create empty user groups with titles and descriptions to use as templates.

Note:

To avoid permission conflicts, do not include user accounts in template user groups. Doing so makes the template group a member of all new tenant user groups. Tenants must not share users.

Note:

The Multitenancy Management Tool automatically searches for user groups; you cannot exclude them from the tenant template.

Related Topics

- [To view user and group associations for a tenant](#)
- [To add a user group to a tenant](#)
- [To remove a user group from a tenant](#)

4.4 Including event folders in the tenant template

You use the **Folders** management area of the Central Management Console (CMC) to include event folders in the tenant template.

1. Create an event folder with the template token in its name.
2. In a text editor, open the tenant configuration file in a text editor, and set `optionIncludeEvents` to `true`.
3. Under `templateEventFolder`, specify the event folders, separating folder names with a semicolon.
4. On a command line, run the Multitenancy Management Tool.

You must specify only top-level event folders in the tenant template when running the Multitenancy Management Tool. All subfolders and events contained in the folders will also be copied.

4.5 Including categories in the tenant template

You use the **Categories** management area of the Central Management Console (CMC) to include categories in the tenant template.

1. Create a category with the template token in its name.
2. In a text editor, open the tenant configuration file, and set `optionIncludeCategories` to `true`.

3. Under `templateCategoryFolder`, specify the categories, separating category names with a semicolon.
4. On a command line, run the Multitenancy Management Tool.

4.6 Including profiles in the tenant template

You use the **Profiles** management area of the Central Management Console (CMC) to include profiles in the tenant template.

1. Create a profile with the template token in its name.
2. In a text editor, open the tenant configuration file, and set `optionIncludeProfiles` to `true`.
3. On a command line, run the Multitenancy Management Tool.

Users, user groups, profile values, and global profile targets are copied to the new tenant profile.

4.7 Including security settings in the tenant template

You use the **Folders** management area of the Central Management Console (CMC) to include access levels (groups of rights) in the tenant template.

1. Create an access level with the template token in its name.
2. In a text editor, open the tenant configuration, and set `optionIncludeLevels` to `true`.
3. On a command line, run the Multitenancy Management Tool.

All included rights are copied to the new tenant access level.

Importing access levels and rights

Any rights or access levels granted on template objects to principals are copied to the new tenant object. If the target tenant object already exists before running the Multitenancy Management Tool (for example, when running the tool multiple times for the same tenant), use the `optionImportSecMode` option to specify how to handle existing principal rights on the tenant object:

- `optionImportSecMode=0` (merge mode): Merge template principals and rights with the existing tenant object. Preserves the original rights granted to principals on the existing tenant object.

Note:

In the event of a conflict, the template value takes precedence during the merge. For example, the template object specifically grants a right to a principal, but the existing tenant object denies that same right. In that case, the tenant is granted the right.

- `optionImportSecMode=1` (principal-level overwrite): Replace the rights of identical principals with template rights. Add unique principals and rights from the template, and preserve unique principals and rights on the tenant object.

- `optionImportSecMode=2` (object-level overwrite): Remove all principals and rights on the existing tenant object, and replace with principals and rights granted on the template object.

Example:

Assume you have a template folder (`$TemplateToken$`) and an existing tenant folder (ABC) with the access levels granted to principals as follows:

Table 4-1: Access levels before

Template folder = \$TemplateToken\$		Tenant folder = ABC	
Principal	Access level	Principal	Access level
User 1	View	User 1	Full Control
User 2	View	-	
-		User 3	View

After running the Multitenancy Management Tool, the resulting permissions granted on tenant folder ABC, based on the `optionImportSecMode` setting, are as follows:

Table 4-2: Access levels after

Tenant folder = ABC optionImportSecMode=0		Tenant folder = ABC optionImportSecMode=1		Tenant folder = ABC optionImportSecMode=2	
Principal	Access level	Principal	Access level	Principal	Access level
User 1	Full Control; View	User 1	View	User 1	View
User 2	View	User 2	View	User 2	View
User 3	View	User 3	View	-	-

For example, you can see that the previously individually set rights, such as the Full Control access for User 1 are replaced with View rights, and these would have to later be reinstated if required.

4.8 SAP Crystal Reports 2011

You can use the Multitenancy Management Tool to map direct-to-data connection information for SAP Crystal Reports 2011 template reports to new tenant connection settings.

To map tenant data source connection information for Crystal reports:

1. Create and upload a set of reports to a template folder in BI platform.

Template folder names are defined under `templateContentFolder` in the tenant configuration file.

2. Under `crystalreport.templatedb1` in the tenant configuration file, specify the template's DSN information.

The format of the value is `<database server>;<database name>;<data source type>;<username>;<password>`.

For example:

```
crystalreport.templatedb1=MyTemplateDSN;MyTemplateDatabase;odbc;administrator;password
```

Tip:

To find the correct values for `<database server>;<database name>`, right-click a Crystal report in the Central Management Console (CMC) and select **Database Configuration**, and then copy the values in the **Server** and **Database** fields to the tenant configuration file.

3. Under `crystalreport.tenantdb1` in the tenant configuration file, specify the tenant's new DSN information.

The format of the value is `<database server>;<database name>;<data source type>;<username>;<password>`.

For example:

```
crystalreport.tenantdb1=MyTenantDSN;MyTenantDatabase;odbc;tenantname;tenantpwd
```

4. On a command line, run the Multitenancy Management Tool.

After running the Multitenancy Management Tool, the copied Crystal reports in the tenant folders are mapped to their data source connection. You can specify additional template and tenant pairs of settings by adding `crystalreport.templatedb<n>` and `crystalreport.tenantdb<n>` option pairings to the tenant configuration file. For example, `crystalreport.templatedb2` and `crystalreport.tenantdb2`.

Consider the following before-and-after snapshots of the Central Management Server (CMS), assuming a tenant named "ABC" is created:

Before	After
<div>Public Folders</div> <div> \$TemplateToken\$_folder \$TemplateToken\$_report1.rpt \$TemplateToken\$_1report.rpt \$TemplateToken\$_report2.rpt </div> <ul style="list-style-type: none"> • \$TemplateToken\$_report1 and \$TemplateToken\$_1report1 use crystalreport.templateddb1 DSN settings. • \$TemplateToken\$_report2 uses crystalreport.templateddb2 DSN settings. 	<div>Public Folders</div> <div> \$TemplateToken\$_folder \$TemplateToken\$_report1.rpt \$TemplateToken\$_1report.rpt \$TemplateToken\$_report2.rpt </div> <div>ABC_folder</div> <div> ABC_report1.rpt ABC_1report.rpt ABC_report2.rpt </div> <ul style="list-style-type: none"> • ABC_report1.rpt and ABC_1report.rpt use crystalreport.tenantddb1 DSN settings. • ABC_report2.rpt uses crystalreport.tenantddb2 DSN settings.

Mapping table prefixes

The template database may use different table prefixes than the new tenant database that you are pointing reports to. In this case, use `crystalreport.templateprefixes<n>` and `crystalreport.tenanttableprefixes<n>` in the tenant configuration file to specify how to map tables. Use semicolons to separate multiple prefixes, and end the list with a semicolon so that empty strings ("") can be identified.

For example, the following table prefixes:

```
templatetableprefixes1=templateprefixa;templateprefixb;;
tenanttableprefixes1=;tenantprefixb;tenantprefixc;
```

Result in the following mapping:

Template prefix	Maps to tenant prefix
templateprefixa	(empty string)
templateprefixb	tenantprefixb
(empty string)	tenantprefixc

Tip:

To find the name of a table prefix in a template report, right-click the report in the CMC and select **Database Configuration**. The **Table Prefix** field displays the name.

Supported data types

The Multitenancy Management Tool supports changing the following direct-to-data connection types. Use these values in the `crystalreport.templateddb<n>` and `crystalreport.tenantddb<n>` options:

Table 4-4: SAP Crystal Reports 2011 data source type

Data source type
odbc
oracle
db2
sybase
informix
crdb_xml

Related Topics

- [Adding public folders to the tenant template](#)

4.9 SAP Crystal Reports for Enterprise

You can use the Multitenancy Management Tool to map SAP Crystal Reports for Enterprise reports that configure a direct-to-data connection through a `CCIS.DataConnection` connection object created with the SAP BusinessObjects Information Design Tool. Connection objects are hosted on BI platform and can be viewed in the `Connections` folder in the Central Management Console (CMC). Template reports are mapped to new tenant connection settings.

To map tenant data source connection information for reports that use a connection object, configure the following option in the tenant configuration file:

- `ccis.dataconnection.dbcredentials<n>=<template_CCIS.CONN_CUID>`

The CUID of a template relational connection object that the template report uses is `template_CCIS.CONN_CUID`.

Tip:

To find the connection object associated with a template report, right-click the report in the CMC and select **Tools > Check Relationships**. A "Relational Connection" object is listed in the results.

- If you have more than one template connection object, add additional `ccis.dataconnection.dbcredentials` options—for example, `ccis.dataconnection.dbcredentials2`, `ccis.dataconnection.dbcredentials3`, ... `ccis.dataconnection.dbcredentialsn`.

After running the Multitenancy Management Tool, the copied Crystal reports in the tenant folders are mapped to their data source connection.

Consider the following before-and-after snapshots of the CMS, assuming a tenant named "ABC" is created and the tenant configuration file is configured as `ccis.dataconnection.dbcredentials1=ZZZZZZZZZZ`.

Object type	Before	After
Folders and reports	<pre>Public Folders \$TemplateToken\$_folder \$TemplateToken\$_cr4ereport.rpt \$TemplateToken\$_cr4ereport.rpt: • CUID=XXXXXXXXXX • Connection=\$TemplateToken\$_ODBC Connection</pre>	<pre>Public Folders \$TemplateToken\$_folder \$TemplateToken\$_cr4ereport.rpt ABC_folder ABC_cr4ereport.rpt \$TemplateToken\$_cr4ereport.rpt: • CUID=XXXXXXXXXX • Connection=\$TemplateToken\$_ODBC Connection ABC_cr4ereport.rpt: • CUID=AAAAAAAAAA • Connection=ABC_ODBCConnection</pre>
Connections	<pre>Connections \$TemplateToken\$_unshared \$TemplateToken\$_ODBCConnection \$TemplateToken\$_ODBCConnection: • CUID=ZZZZZZZZZZ • type=CCIS.DataConnection</pre>	<pre>Connections \$TemplateToken\$_unshared \$TemplateToken\$_ODBCConnection ABC_unshared ABC_ODBCConnection \$TemplateToken\$_ODBCConnection: • CUID=ZZZZZZZZZZ • type=CCIS.DataConnection ABC_ODBCConnection: • CUID=CCCCCCCCCC • type=CCIS.DataConnection</pre>

Mapping table prefixes

The template database may use different table prefixes than the new tenant database that you are pointing reports to. In this case, configure the following options:

- `crystalreport.ccis.dataconnection.template_db<n>`: CUID of the template connection object
- `crystalreport.ccis.dataconnection.template_tableprefixes<n>`: Table prefixes of the template data source
- `crystalreport.ccis.dataconnection.tenant_tableprefixes<n>`: Table prefixes of the tenant data source to map to

Separate multiple prefixes with semicolons, and end the list with a semicolon so that empty strings ("") can be identified.

For example, the following table prefixes:

```
crystalreport.ccis.dataconnection.templateid=ZZZZZZZZZZ
crystalreport.ccis.dataconnection.templateprefixes1=templateprefixa;templateprefixb;;
crystalreport.ccis.dataconnection.tenanttableprefixes1=;tenantprefixb;tenantprefixc;
```

Result in the following mapping:

Template prefix	Maps to tenant prefix
templateprefixa	(empty string)
templateprefixb	tenantprefixb
(empty string)	tenantprefixc

Tip:

To find the name of a table prefix in a template report, view the details of the direct-to-data connection object using the SAP BusinessObjects Information Design Tool.

Related Topics

- [Managing universes and connections](#)
- [Adding public folders to the tenant template](#)

4.10 Managing universes and connections

The Multitenancy Management Tool is used to manage tenant universes and connection information. This section outlines the following scenarios to consider:

- Unshared universes and connections
- Managing universes and connections

Note:

This release supports classic universes (.unv files) with limited support for DSL universes (.unx files, with exceptions that include Access Restrictions (Security Profile) and multi-connection DSL universes).

4.10.1 Unshared universes and connections

Unshared universes and connections

In this scenario, reports and analytics connect to separate universes (with separate underlying connection objects). This lets you automate the distribution of template reports while isolating the data that is accessible by each tenant.

Note:

You must set the options described in this task in addition to other required options (such as folder templates) for running the Multitenancy Management Tool.

To provide separate, unshared universes and connection objects for each tenant:

1. In the **Multitenancy** management area of the Central Management Console (CMC), create a connection and include the template token in its name, and then save it to a template connection folder that has the template token string in its name.
2. Create a universe with the template token in its name and that uses the template connection you created in step 1, and then save it to a universe folder that has the template token string in its name.
3. Create and upload a set of reports or analytics that use the universe you created in step 2 to a template folder in BI platform.
4. Set the following required options in the tenant configuration file:

For this required option	Do this
<code>optionIncludeUniverses</code>	Set to <code>true</code> so that template universes are copied to each tenant
<code>optionIncludeConnections</code>	Set to <code>true</code> so that template connections are copied to each tenant
<code>templateUniverseFolder</code>	<p>Set to the folder path where your universe templates are located—for example, <code>\$TemplateToken\$_unshared</code></p> <p>The path is relative to the root <code>Universes</code> folder.</p>
<code>templateConnectionFolder</code>	<p>Set to the folder path where your universe templates are located—for example, <code>\$TemplateToken\$_unshared</code></p> <p>The path is relative to the root <code>Connections</code> folder.</p>

For this required option	Do this
<code>ccis.dataconnection.dbcredentials1</code>	<p>Set to the details of the template connection object that you want to replicate for each tenant—for example, <code><CUID>;<data source name>;<database name>;<username>;<password></code></p> <p>CUID is the CUID of a template connection object. Specify the DSN information (<code><data sourcename></code>, <code><database name></code>) for the new tenant connection that will be created.</p> <p>If you have more than one template connection object, add additional <code>ccis.dataconnection.dbcredentials</code> options—for example, <code>ccis.dataconnection.dbcredentials2</code>, <code>ccis.dataconnection.dbcredentials3</code>, ... <code>ccis.dataconnection.dbcredentialsn</code>.</p>

5. On a command line, run the Multitenancy Management Tool.

The new tenant has its own instances of reports, analytics, universes, and connection objects.

Consider the following before-and-after snapshots of the CMS, assuming the tool was run twice to create tenants named "ABC" and "DEF." In this example, the CUID in the `ccis.dataconnection.dbcredentials1` option is set to `ZZZZZZZZZZ`:

Object type	Before	After
Folders and reports	<p>Public Folders</p> <ul style="list-style-type: none"> \$TemplateToken\$_unshared \$TemplateToken\$_sales.wid <p>\$TemplateToken\$_sales.wid:</p> <ul style="list-style-type: none"> CUID=XXXXXXXXXX Universe=\$TemplateToken\$_ODBCUniverse.unv 	<p>Public Folders</p> <ul style="list-style-type: none"> \$TemplateToken\$_unshared \$TemplateToken\$_sales.wid ABC_unshared ABC_sales.wid DEF_unshared DEF_sales.wid <p>\$TemplateToken\$_sales.wid:</p> <ul style="list-style-type: none"> CUID=XXXXXXXXXX Universe=\$TemplateToken\$_ODBCUniverse.unv <p>ABC_sales.wid:</p> <ul style="list-style-type: none"> CUID=AAAAAAAAAA Universe=ABC_ODBCUniverse.unv <p>DEF_sales.wid:</p> <ul style="list-style-type: none"> CUID=DDDDDDDDDD Universe=DEF_ODBCUniverse.unv
Universes	<p>Universes</p> <ul style="list-style-type: none"> \$TemplateToken\$_unshared \$TemplateToken\$_ODBCUniverse.unv <p>\$TemplateToken\$_ODBCUniverse.unv:</p> <ul style="list-style-type: none"> CUID=YYYYYYYYYY Connection=\$TemplateToken\$_ODBCConnection 	<p>Universes</p> <ul style="list-style-type: none"> \$TemplateToken\$_unshared \$TemplateToken\$_ODBCUniverse.unv ABC_unshared ABC_ODBCUniverse.unv DEF_unshared DEF_ODBCUniverse.unv <p>\$TemplateToken\$_ODBCUniverse.unv:</p> <ul style="list-style-type: none"> CUID=YYYYYYYYYY Connection=\$TemplateToken\$_ODBCConnection <p>ABC_ODBCUniverse.unv:</p> <ul style="list-style-type: none"> CUID=BBBBBBBBBB Connection=ABC_ODBCConnection <p>DEF_ODBCUniverse.unv:</p> <ul style="list-style-type: none"> CUID=EEEEEEEEEE Connection=DEF_ODBCConnection
Connections		

Object type	Before	After
	<pre>Connections \$TemplateToken\$_unshared \$TemplateToken\$_ODBCConnection</pre> <p>\$TemplateToken\$_ODBCConnection:</p> <ul style="list-style-type: none"> • CUID=ZZZZZZZZZZ • type=CCIS.DataConnection 	<pre>Connections \$TemplateToken\$_unshared \$TemplateToken\$_ODBCConnection ABC_unshared ABC_ODBCConnection DEF_unshared DEF_ODBCConnection</pre> <p>\$TemplateToken\$_ODBCConnection:</p> <ul style="list-style-type: none"> • CUID=ZZZZZZZZZZ • type=CCIS.DataConnection <p>ABC_ODBCConnection:</p> <ul style="list-style-type: none"> • CUID=CCCCCCCCCC • type=CCIS.DataConnection <p>DEF_ODBCConnection:</p> <ul style="list-style-type: none"> • CUID=FFFFFFFFFF • type=CCIS.DataConnection

4.10.2 Shared universes and connections

In this scenario, reports and analytics connect to the same universes and connection objects.

Note:

You must set the options described in this task in addition to other required options (such as folder templates) for running the Multitenancy Management Tool.

To specify a shared environment:

1. Create a connection without the template token in its name, and then save it to a connection folder that is shared by all tenants.
Reminder: the connection must not have the template token in its name.
2. Create a universe that uses the template connection you created in step 1, and then save it to a universe folder that is shared by all tenants.
3. Create and upload a set of reports or analytics that use the universe that you created in step 2 to a template folder in BI platform.
4. Set the following options in the tenant configuration file:

For this required option	Do this
<code>optionUseSharedUniverses</code>	Set to <code>true</code> .
<code>optionUseSharedConnections</code>	Set to <code>true</code> .
<code>sharedUniverseFolder</code>	Set to the folder path where your shared universe is located—for example, <code>SharedUniverses</code> . The path is relative to the root <code>Universes</code> folder.
<code>templateConnectionFolder</code>	Set to the folder path where your universe templates are located—for example, <code>SharedConnections</code> . The path is relative to the root <code>Connections</code> folder.

5. On a command line, run the Multitenancy Management Tool.

After running the tool, the new tenant has its own instances of report, analytics that use a common universe and connection object.

Consider the following before-and-after snapshots of the CMS, assuming the tool was run twice to create tenants named "ABC" and "DEF":

4.10.3 Restricting data access for shared universes (classic UNV universes only)

When sharing universes between tenants, you may want to restrict the data in the universe that each tenant can view. To restrict access to data, you can assign rights on objects, rows, queries, and other aspects of a universe to certain user groups. This is done during universe design (and not in the tenant template), in the "Manage Access Restrictions" dialog box in the SAP BusinessObjects Universe Design Tool. When the universe is accessed by a member of the user group, the query generated at run time will return only data that the user group has access to.

Note:

This topic does not apply to `UNX` DSL universes.

The Multitenancy Management Tool helps you automate this task. Instead of restricting access for a specific user group, you can use the tool to set access for the template user group (that is, the group whose name contains the template token string). The name of the restriction you create must contain the template token string. After running the tool, the access restriction will be replaced by the tenant name, specified by the `tenantName` option in the tenant configuration file.

Note:

Ensure that the user group is also a template group (with "\$TemplateToken\$" in the name) so that conflicting restrictions are not applied to the same tenant user group. You cannot apply more than one restriction to the same user group. For instructions on setting access restrictions in classic universes (.unv files), see the *SAP BusinessObjects Universe Design Tool User Guide*.

Running the Multitenancy Management Tool

To run the Multitenancy Management Tool, go to the folder where the tool is located, open a command prompt, and run `multitenancymanager.jar` with the tenant definition file passed to the `-configFile` option:

```
java -jar multitenancymanager.jar -configFile tenant_template_def.properties
```

Note:

An update to the `Path` environment variable may be necessary.

By default, the Multitenancy Management Tool is installed with BI platform in the following location:

- **Windows:** `<INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\java\apps\multitenancyManager\jars\`
- **Unix:** `<INSTALLDIR>/sap_bobj/enterprise_xi40/java/apps/multitenancyManager/jars/`

Caution:

The Multitenancy Management Tool creates and commits new objects (such as user groups, folders, documents, and universes) to your CMS repository. Keep the `abortANDRollback` option set to `true` (the default), so that when an error is encountered, the provisioning tool will automatically roll back changes to the CMS. If further cleanup is required, use the **Delete** option in the **Multitenancy** area of the CMC to remove objects associated with the tenant in question.

5.1 Command line options

You can use the command line to pass options directly to the Multitenancy Management Tool.

When creating new tenants, parameters and their values are delivered through the command line. Parameter values entered at the command line override the same parameters in the tenant configuration file (`tenant_template_def.properties`). When you override options in the configuration file at the command line, the configuration file is not modified in the process.

The following example demonstrates how to override the `tenantName` option in the tenant configuration file:

```
java -jar multitenancymanager.jar -configFile tenant_template_def.properties -tenantName=xyz_customer
```

Increasing Java memory allocation

For some complex provisioning operations, you may encounter an out-of-memory exception when running the Multitenancy Management Tool. The exception message is similar to the following example:

```
There was an error running multitenancy management tool.LoggingoffException in thread
"AWT-Shutdown" java.lang.OutOfMemoryError: PermGen spaceException in thread
"sap.bobj.mainMultiTenancyManagerThread" java.lang.OutOfMemoryError: PermGen spaceException in thread
"Business Objects - Sessions Clean up" java.lang.OutOfMemoryError: PermGen space
```

To increase the permanent generation space allocation from the default (64 MB) to 128 MB, add the parameter `-XX:MaxPermSize=128m` as follows:

```
InstallDir\SAP BusinessObjects Enterprise XI4.0\win64_x64\sapjvm\bin\
java.exe" -XX:MaxPermSize=128m -jar multitenancymanager.jar -configFile tenant_template_def.properties
```

Note:

Do not confuse this memory adjustment with the JVM max heap size `-Xmx`.

Onboarding and provisioning DSL universes

Setting the Java system property `-Dbusinessobjects.connectivity.directory` to the BOE connection server folder is necessary when using MTM to onboard or provision DSL universes.

Before running the Multitenancy tool, set the path to the SAP BusinessObjects binaries on a Windows machine as follows:

```
set path=InstallDir\SAP BusinessObjects Enterprise XI 4.0\<machine architecture>;%path%
```

An example of setting a path on a Windows machine would appear as follows:

```
set path=c:\bip_autochain\boe\SAP BusinessObjects Enterprise XI 4.0\win64_x64;%path%
```

An example of setting a path on a Linux machine would appear as follows:

```
export PATH="/opt/boe/sap_bobj/enterprise_xi40/linux_x64:$PATH"
```

Examples of `<machine architecture>` would be `win64_x64`, `linux_x64`, `aix_rs6000_64`, `solaris_sparcv9` and so on.

Run the multitenancy tool to set `-Dbusinessobjects.connectivity.directory` as follows:

```
InstallDir\SAP BusinessObjects Enterprise XI 4.0\win64_x64\sapjvm\bin\java.exe"
-Dbusinessobjects.connectivity.directory="InstallDir\SAP BusinessObjects Enterprise XI 4.0\
dataAccess\connectionServer" -jar multitenancymanager.jar -configFile tenant_template_def.properties
```

Related Topics

- [tenantName](#)

5.2 Tenant definition configuration file reference

This is the reference section for configuration options contained in the tenant definition configuration (tenant_template_def.properties) file.

Category	Property name	Function	Example value
Mandatory configuration	---	---	---
	tenantName	Replaces templateToken	Demo Tenant 4
	templateToken	Placeholder string that will be replaced with the tenantName string.	\$Tenant_Template\$
Authentication	cms	Server address and port number (6400 is the default port)	localhost:8080
	auth	Authentication type	secEnterprise
	user	Logon name	tenantuser1
	pwd	Password	userpass1
	---	---	—
	reviewBeforeProceed	Pause provisioning process on error. Press Y to continue.	true or false
Optional on-boarding/provisioning configuration	---	---	---
	tenantConcurrentUserLimit	Maximum number of users for this tenant	100
	statusLog	Location of log file	c:\program files (x86)\SAP BusinessObjects\SAP BusinessObjects Enterprise XI 4.0\logging
	abortANDRollback	If true, upon encountering a fatal error, undo provisioning changes.	true or false
Option...	---	---	Where applicable, use with matching template... property settings
	optionImportSecMode	Security overrides	1 (overwrite), 0 (merge) , 2 (overwrite), 3 (ignore)

Category	Property name	Function	Example value
	optionIncludeUniverses		true or false
	optionIncludeConnections		true or false
	optionIncludeCategories		true or false
	optionIncludeProfiles		true or false
	optionIncludeEvents		true or false
	optionIncludeAccessLevels		true or false
	optionUseSharedUniverses		true or false
	optionUseSharedConnections		true or false
Template...	---	---	Use with matching <code>optionInclude...</code> property settings
	templateContentFolder	Starting folder	tenants/\$tenant\$_reports;tenants/\$tenant\$_programs/public
	templateUniverseFolder		tenants/\$tenant\$
	templateConnectionFolder		tenants/\$tenant\$
	templateCategoryFolder		\$tenant\$
	templateEventFolder		/Custom Events/\$tenant\$;/System Events/\$tenant\$_system
Shared...	sharedUniverseFolder		tenants/Test Shared Universe
	sharedConnectionFolder		tenants/Test Shared Connections
(Database) Type specific configurations (optional)	---	---	---

Category	Property name	Function	Example value
Crystal Reports 2011 database connections	crystalreport.templateedb	Database identification and credentials	crystalreport.templateedb1=template-server;templatedb-name;odbc;administrator;password
	crystalreport.tenantdb		crystalreport.tenantdb1=tenant-server;tenantdbname;odbc;administrator;mypassword
	crystalreport.templateprefixes	Use when table prefix update is needed.	crystalreport.templateprefixes1=templateprefixa;templateprefixb;;
	crystalreport.tenanttableprefixes		crystalreport.tenanttableprefixes1=;tenantprefixb;tenantprefixc;
Crystal Reports for Enterprise managed database connections	crystalreport.ccis.dataconnection.templateedb	Database identification and credentials	crystalreport.ccis.dataconnection.templateedb1=ARiy-ILQC6h5Ogwf8VolZzqQ
	crystalreport.ccis.dataconnection.templateprefixes	Use when table prefix update is needed.	crystalreport.ccis.dataconnection.templateprefixes1=templateprefixa;templateprefixb;;
	crystalreport.ccis.dataconnection.tenanttableprefixes		crystalreport.ccis.dataconnection.tenanttableprefixes1=;tenantprefixb;tenantprefixc;
CCIS.DataConnection database connections	ccis.dataconnection.dbcredentials	Update CCIS database connection	ccis.dataconnection.dbcredentials1=ARiyILQC6h5Ogwf8VolZzqQ;tenantDBN;tenantDB;user-ABC;Password123

Related Topics

- [Quick start](#)
- [tenant_template_def.properties](#)

5.2.1 Mandatory Configuration

5.2.1.1 tenantName

Name of the tenant to be added to the BI Platform by the tenant provisioning tool. This string replaces the [templateToken](#) string wherever it has been declared in the CMC in the tenant template that was created earlier.

For example, if `tenantName=abc`, `templateToken=$TemplateToken$` and you've created a user group template called "`$TemplateToken$_usergroup`", then the tool will create a new user group called "`abc_usergroup`".

Note:

Using the [Command line options](#), the name specified will be used to create a new tenant. This name overrides the value stored in the tenant definition properties file.

Category

Mandatory Configuration - Tenant setup

Required setting?

Yes

Default setting

None

Example

Type	Example	Note
alphanumeric	Demo Tenant 4	spaces and numbers are allowed in the name, but not \$ dollar signs.
alphanumeric	Tenant_xyz	

CMC - Multitenancy user Interface

Tenant names may be changed in **CMC > Multitenancy > Properties > Tenant Name**. For more information, see "To change the name of the tenant."

The tenant name must be entered or a warning alert appears: "Tenant Name cannot be empty".

The tenant name must be unique, or an error appears: [Duplicate object name in the same folder. \(FWB 00026\)](#).

Properties File comment

```
# (Mandatory) Name of the tenant being added.  
# - Tenant specific info, overridden by command-line option  
tenantName=ABC
```

Command line

New tenants may be created using a name specified in the command line that will override the name set in the tenant configuration file.

```
java -jar multitenancymanager.jar -configFile tenant_template_def.properties -tenantName=xyz_customer
```

5.2.1.2 templateToken

A unique string name surrounded by \$ dollar signs that is used in any tenant template object such as a folder name, connection name, event names and so on. When a new tenant is provisioned using the multitenancy tool, the `templateToken` string serves a placeholder that will be replaced by the string in `tenantName`.

Note:

Choose a unique string that is not commonly found in existing object names within your BI platform installation.

Category

Mandatory Configuration - Tenant setup

Required setting?

Yes

Note:

The name defined in [tenantName](#) replaces the search and replace string phrase defined in `templateToken`. For example the `tenantName` value `Tenant_xyz` will replace all occurrences of the `templateToken` value `$Tenant_Template$`.

Default setting

None

Example

Type	Example	Note
alphanumeric	\$Tenant_Template\$	All occurrences of the phrase within the dollar signs will be replaced with the tenantName.

Properties File comment

```
# (Mandatory) Template token identifier used for tenant name replacing
# - Can be any format but need to clearly identify the templates so that any regular
#   objects not intended as templates should not have the token embedded in their names.
# - E.g. User group template /$tenant$_users/ for tenant ABC will become /ABC_users/
templateToken=$tenant$
```

5.2.1.3 Authentication

5.2.1.3.1 cms

The domain name or IP number and port of the Central Management Server (CMS).

Category

Mandatory Configuration - Tenant setup - Authentication

Required setting?

Yes

Note:

Also requires setting configuration options `auth`, `user` and `pwd`.

Default setting

Type	Example	Note
URL	localhost	There is no default value. The port number may be omitted if it is 6400.
Port	6400	6400 is the default port if none is included with the URL.
URL	www.tenantabcfinance.com:8080	This is the domain name followed by the port assignment.

Example

Type	Example	Note
URL	localhost	local computer
URL	10.168.4.16	IP address
URL	localhost:6400	local computer running on the default port 6400
URL	www.exampleserver.com:2456	domain name using port 2456

Log file success messages

Logging on to cms localhost

Logon succeeds

Logging off

Properties file comment

```
# (Mandatory) Logon information
cms=
```

Possible errors and warnings

Error log location: <InstallDir>\SAP BusinessObjects Enterprise XI 4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

```
INSTALLDIR\SAP BusinessObjects Enterprise XI 4.0\logging\multitenancymanager20120626163937724.csv
```

See the *Error Messages Explained* guide for more information.

Error	Cause	Action
Enterprise authentication could not log you on. Please make sure your logon information is correct.	You could not log on because your user name could not be found, or because your password is invalid.	Ensure that your user name and password are correct.

Related Topics

- [auth](#)
- [user](#)
- [pwd](#)

5.2.1.3.2 auth

The authentication type to use for logging on. This includes enterprise authentication, LDAP, WinAD and SAP authentication.

Category

Mandatory Configuration - Tenant setup - Authentication

Required setting?

Yes

Note:

Also requires setting configuration options `cms`, `user` and `pwd`.

Default setting

None

Properties file comment

```
# (Mandatory) Logon information
auth=secEnterprise
```

Example

Type	Example	Note
Fixed value	secEnterprise	Enterprise authentication
Fixed value	secLDAP	Lightweight Directory Access Protocol authentication
Fixed value	secWinAD	Windows Active Directory (AD) authentication
Fixed value	secSAPR3	SAP authentication

Log file success messages

Logging on to cms localhost

Logon succeeds.

Possible errors and warnings

Error log location: <InstallDir>\SAP BusinessObjects Enterprise XI 4.0\Logging\multitenancymanager<yyyy><mm><dd><ttttttttt>.csv>

INSTALLDIR\SAP BusinessObjects Enterprise XI 4.0\logging\multitenancymanager20120626163937724.csv

See the *Error Messages Explained* guide for more information.

Error	Cause	Action
Enterprise authentication could not log you on. Please make sure your logon information is correct.	You could not log on because your user name could not be found or because your password is invalid.	Ensure that your user name and password are correct.

Related Topics

- [user](#)
- [pwd](#)
- [cms](#)

5.2.1.3.3 user

The username of the account used to log onto the CMS and run the multitenancy tool. A tenant username does not qualify.

Category

Mandatory Configuration - Tenant setup - Authentication

Required setting?

Yes

Note:

Also requires setting configuration options `cms`, `auth`, `pwd`

Default setting

None

Properties File comment

```
# (Mandatory) Logon information  
user=administrator
```

Example

Type	Example	Note
alphanumeric	P@s3w.or9	Password string
alphanumeric	Accounting_1xU3v	Password string

Log file success messages

Logging on to cms localhost

Logon succeeds.

Logging off

Possible errors and warnings

Error log location: `<INSTALLEDIR>\SAP BusinessObjects Enterprise XI 4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv`

```
INSTALLEDIR\SAP BusinessObjects Enterprise XI 4.0\logging\multitenancymanager20120626163937724.csv
```

See the *Error Messages Explained* guide for more information.

Error	Cause	Action
Enterprise authentication could not log you on. Please make sure your logon information is correct.	You could not log on because your username could not be found, or because your password is invalid.	Ensure that your username and password are correct.

Related Topics

- [auth](#)
- [pwd](#)
- [cms](#)

5.2.1.3.4 pwd

The password of the user account used to log onto the CMS and run the multitenancy tool.

Category

Mandatory Configuration - Tenant setup - Authentication

Required setting?

Yes

Note:

Also requires setting configuration options `cms`, `auth`, `user`

Default setting

None

Properties File comment

```
# (Mandatory) Logon information
pwd=
```

Example

Type	Example	Note
alphanumeric	P@s3w.or9	Password string
alphanumeric	Accounting_1xU3v	Password string

Log file success messages

Logging on to cms localhost

Logon succeeds.

Logging off

Possible errors and warnings

Error log location: <INSTALLEDIR>\SAP BusinessObjects Enterprise XI 4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

`INSTALLEDIR\SAP BusinessObjects Enterprise XI 4.0\logging\multitenancymanager20120626163937724.csv`

See the *Error Messages Explained* guide for more information.)

Error	Cause	Action
Enterprise authentication could not log you on. Please make sure your login information is correct.	You could not log on because your username could not be found, or because your password is invalid.	Ensure that your username and password are correct.

Related Topics

- [auth](#)
- [cms](#)
- [user](#)

5.2.1.4 reviewBeforeProceed

Indicates whether or not to allow the review of errors or warnings before final changes are made the MTM configuration. The script will stop and wait for confirmation to continue.

Category

Mandatory Configuration - Tenant setup

Required setting?

No

Default setting

True

Example

Type	Example	Note
boolean	reviewBeforeProceed=true	(default): Enables review prompts. At each error or warning that the configuration administrator must acknowledge before the configuration will resume. Messages include suggestions for fixing the errors.
boolean	reviewBeforeProceed=false	Disables review prompts. Errors and warnings are displayed and written to the log file, but the configuration will not stop and wait for permission to continue.

Properties File comment

```
# (Optional) Review configuration errors/warnings, if any, before proceeding to the actual MTM operation.
# - true: (default) review the configuration errors and warnings the tool found before giving instruction
on how to proceed
# - false: the tool will not prompt the user for review, instead it will display the errors/warnings and
ignore them
reviewBeforeProceed=true
```

Possible errors and warnings

Error log location: <INSTALLDIR>\SAP BusinessObjects Enterprise XI
4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttt>.csv>

5.2.2 Tenant on-boarding/provisioning configuration options**5.2.2.1 tenantConcurrentUserLimit**

The maximum number of users for a tenant that are allowed to logon simultaneously.

When the limit is reached, no more users will be able to logon until another tenant's user has logged out. This will ensure users of other tenants can logon. If the maximum number of users for all tenants is reached, the system will display a warning message.

Category

Tenant on-boarding/provisioning configuration

Required setting?

No

Default setting

100

Example

Type	Example	Note
number	tenantConcurrentUserLimit=100	100 is the default.
number	tenantConcurrentUserLimit=-1	Unlimited users allowed.
number	tenantConcurrentUserLimit=	No value entered means the system will use the default of -1

CMC Interface

Weight values for each tenant may be changed in **CMC > Multitenancy > Properties > Concurrent Users**. For more information, see "To change the number of concurrent users for a tenant."

Entering a **Value** of zero (0) effectively disables the tenant. A red alert will appear in CMC that reads: "A value of 0 means the Tenant cannot do anything". Choosing the **Unlimited** radio button removes all resource limitations.

Log file success messages

none

Properties File comment

```
# (Optional) Number of concurrent users allowed for the tenant being added.
# Defaults to -1 meaning unlimited in which case the number of concurrent users
# for this tenant will only be limited by what the BOE license key allows
# for the entire system. When not specified, this option will be set to default
# during the tenant's first on-board, or unchanged during tenant's provisioning.
# - Tenant specific info
tenantConcurrentUserLimit=100
```

Possible errors and warnings

Error log location: <INSTALLDIR>\SAP BusinessObjects Enterprise XI
4.0\Logging\multitenancymanager<yyyy><mm><dd><ttttttt>.csv>

Error	Cause	Action
Error: Invalid configuration of parameter "tenantConcurrentUserLimit".	The value provided is not acceptable.	Remove the value, or enter a positive number.

5.2.2.2 statusLog

The directory where the log file is created.

For troubleshooting purposes, the log file contains a trace of the program and any errors that occurred. For example:

```
statusLog="C:\TenantLogs\"
```

The log file is automatically named based on the tenantName value and the date and time stamp when the program is run. For example: multitenancymanager<yyyy><mm><dd><time>.csv

Category

Tenant on-boarding/provisioning configuration

Required setting?

No

Default setting

- **Windows:** <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\logging\
- **Unix:** <INSTALLDIR>/sap_bobj/logging/

Example

Type	Example	Note
filepath	statusLog=c:\biplogs	Custom drive path.
filepath	statusLog=c:\program files (x86)\SAP BusinessObjects\SAP BusinessObjects Enterprise XI 4.0\logging	Example
filepath	<INSTALLDIR>\SAP BusinessObjectsEnterprise XI 4.0\logging\	Windows default path when statusLog is left blank.
filepath	<INSTALLDIR>/sap_bobj/logging/	Unix default path when statusLog is left blank.

Properties File comment

```
# (Optional) Status log location - directory path for the status log file
# - The tenant name and timestamp (?) will be used to construct the log file name
# - Default status log goes to the logging directory in the BOE install, typically
#   c:\program files (x86)\SAP BusinessObjects\SAP BusinessObjects Enterprise XI 4.0\logging
statusLog=
```

5.2.2.3 abortANDRollback

In the event a tenant is not created due to a first encounter with a fatal error, the system will roll back to the original state before the provisioning process began. This option is on by default.

Setting `abortANDRollback=false` on a provisioning attempt you know may fail allows the process to complete to the end, then you can review the log file to correct the errors. In this case of an incomplete tenant creation, objects that are left behind can be deleted using the CMC > Multitenancy > Delete function, or by locating objects associated with that tenant and deleting them.

Note:

If some objects fail to commit to the CMS, and `abortANDRollback=false` your tenant is only partially created with the objects that were successfully committed. Use the log file to troubleshoot and correct any errors, then rerun the multitenancy tool for that tenant.

Category

Tenant on-boarding/provisioning configuration

Required setting?

No

Default setting

true

Example

Type	Example	Note
boolean	abortANDRollback=false	Allows an onboarding/provisioning process to complete even if errors are encountered.
boolean	abortANDRollback=true	true is the default value. Changes made will be undone if a fatal error is encountered. Check CMC for any residual objects left behind as a result of a rollback.

User interface available in CMC

No

Use CMC to check areas where objects may have been created, such as Universes, Connections, Categories, Profiles, Events, Folders

Log file success messages

none

Properties file comment

```
# (Optional) Rollback option
# - true: rollback if one of the infoobjects fail to commit
# - false: keep going, and log the failures
# - default is false
abortANDRollback=false
```

Possible errors and warnings

Error log location: <InstallDir>\SAP BusinessObjects Enterprise XI 4.0\Logging\multitenancymanager<yyyy><mm><dd><ttttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>" application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the running Multitenancy Management Tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as templateContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running Multitenancy Management Tool.	A general error alert	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter is incorrect.	Enter the correct user name.

5.2.2.4 Template support options

5.2.2.4.1 optionImportSecMode

Indicates how to import security settings (access rights) on objects created from template objects. Three settings are available:

- 0 (merge mode): Merges template principals and rights with the existing tenant object. This value preserves the original rights granted to principals on the existing tenant object.

Note:

If a conflict occurs, the template setting takes precedence during a merge. For example, if a template grants a right to a principal, but the existing tenant denies that right, the template takes precedence and the new tenant is granted the right.

- 1 (principal-level overwrite): This is the default value if nothing is specified. Replaces the rights of identical principals with template rights. This value adds unique principals and rights from the template and preserves unique principals and rights on the tenant object.

- 2 (object-level overwrite): Removes existing principals and rights on an existing tenant object and replaces them with principals and rights granted on the template object

Category

Tenant on-boarding/provisioning configuration > Template support options

Required setting?

No

Default setting

1

Example

Type	Example	Note
Integer	<code>optionImportSecMode=0</code>	Merge mode
Integer	<code>optionImportSecMode=1</code>	Principal-level overwrite. 1 is the default value.
Integer	<code>optionImportSecMode=2</code>	Object-level overwrite

User interface available in CMC

No

Log file success messages

none

Command line option

Use the command line to create a new tenant with parameters and values that override those in the properties file. This example sets `optionImportSecMode` property to a value of 2, and creates a new tenant using all other properties stored in the file `tenant_template_def.properties`.

```
java -jar multitenancymanager.jar -configFile tenant_template_def.properties -optionImportSecMode=2
```

Properties file comment

```
# (Optional) security import options
# 0: merge mode, or access control element level overwrite.
#   Template security info will be copied onto tenant's existing security info
# 1: (default) per principal overwrite mode, or principal level overwrite.
#   Template security info will overwrite tenant's security info on those assigned principals.
# 2: overwrite mode, or object level overwrite. Template security info will
#   overwrite tenant's existing security info
# 3: ignore mode. Template security info will not be copied over to the tenant
optionImportSecMode=1
```

Possible errors and warnings

Error log location: <InstallDir>\SAP BusinessObjects Enterprise XI 4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>"application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the Multitenancy Management Tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as template-ContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running Multitenancy Management Tool.	A general error alert	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter is incorrect.	Enter the correct user name.

5.2.2.4.2 optionIncludeUniverses

Indicates whether the program will search for universe objects using the [templateToken](#) string and create a new tenant version.

Category

Tenant on-boarding/provisioning configuration > Template support options

Required setting?

No

Default setting

true

Example

Type	Example	Note
boolean	optionIncludeUniverses=true	true, or unspecified is the default value
boolean	optionIncludeUniverses=false	false

User Interface available in CMC

No

Log file success messages

none

Properties File comment

```
# (Optional) enable template support options
# - All options listed below are optional. Document(Content) and User Group templates are mandatory.
# - If any of the listed options is not specified, the default will be false.
# - Each supported type has a configuration entry for template folders
#   (see Template Folders) except Access Level, Profile, and User Group.
#   The latter types are of flat folder structure, meaning all objects of that
#   type reside on the level directly below its root folder. If the type is
#   included, any template objects of that type identified using the template
#   token will be copied.
optionIncludeUniverses=true
```

Possible errors and warnings

Error log location: <INSTALLDIR>\SAP BusinessObjects Enterprise XI
4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

Error	Cause	Action
There was an error running multi-tenancy management tool.	A general error alert.	Read subsequent errors or turn on tracing.
optionIncludeUniverses is true, but there are no template objects of type Universe matching this option.	Either optionIncludeUniverses is correctly set to true and there is no matching value, the value is incorrect for template UniverseFolder, or, option IncludeUnivrsers setting is incorrectly set to true.	Set optionIncludeUniverses to false. Add the appropriate templateUniverseFolder value.

5.2.2.4.3 optionIncludeConnections

Indicates whether the program will search for connection objects using the `templateToken` string and create a new tenant version.

Category

Tenant on-boarding/provisioning configuration > Template support options

Required setting?

No

Default setting

true

Example

Type	Example	Note
boolean	optionIncludeConnections=true	true, or unspecified is the default value.
boolean	optionIncludeConnections=false	false

User interface available in CMC

No

Log file success messages

none

Properties file comment

```
# (Optional) enable template support options
# - All options listed below are optional. Document(Content) and User Group templates are mandatory.
# - If any of the listed options is not specified, the default will be false.
# - Each supported type has a configuration entry for template folders
#   (see Template Folders) except Access Level, Profile, and User Group.
#   The latter types are of flat folder structure, meaning all objects of that
#   type reside on the level directly below its root folder. If the type is
#   included, any template objects of that type identified using the template
#   token will be copied.
optionIncludeConnections=true
```

Possible errors and warnings

Error log location: <InstallDir>\SAP BusinessObjects Enterprise XI 4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>" application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the Multitenancy Management Tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as template-ContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running Multitenancy Management Tool.	A general error alert	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter is incorrect.	Enter the correct user name.

Related Topics

- [templateToken](#)

5.2.2.4.4 optionIncludeCategories

Indicates whether the program will search for BI platform categories using the `templateToken` string and create a new tenant version.

Category

Tenant on-boarding/provisioning configuration > Template support options

Required setting?

No

Default setting

true

Example

Type	Example	Note
boolean	optionIncludeCategories=true	true, or unspecified is the default value
boolean	optionIncludeCategories=false	false

User interface available in CMC

No

Log file success messages

none

Properties file comment

```
# (Optional) enable template support options
# - All options listed below are optional. Document(Content) and User Group templates are mandatory.
# - If any of the listed options is not specified, the default will be false.
# - Each supported type has a configuration entry for template folders
#   (see Template Folders) except Access Level, Profile, and User Group.
#   The latter types are of flat folder structure, meaning all objects of that
#   type reside on the level directly below its root folder. If the type is
#   included, any template objects of that type identified using the template
#   token will be copied.
optionIncludeCategories=true
```

Possible errors and warnings

Error log location: <InstallDir>\SAP BusinessObjects Enterprise XI 4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>" application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the Multitenancy Management Tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as template-ContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running Multitenancy Management Tool.	A general error alert	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter is incorrect.	Enter the correct user name.

Related Topics

- [templateToken](#)

5.2.2.4.5 optionIncludeProfiles

Indicates whether the program will search for BI platform profiles using the `templateToken` string and create a new tenant version.

Category

Tenant on-boarding/provisioning configuration > Template support options

Required setting?

No

Default setting

true

Example

Type	Example	Note
Boolean	optionIncludePro files=true	true or unspecified is the default value.
Boolean	optionIncludePro files=false	false

User interface available in CMC

No

Log file success messages

none

Properties file comment

```
# (Optional) enable template support options
# - All options listed below are optional. Document(Content) and User Group templates are mandatory.
# - If any of the listed options is not specified, the default will be false.
# - Each supported type has a configuration entry for template folders
#   (see Template Folders) except Access Level, Profile, and User Group.
#   The latter types are of flat folder structure, meaning all objects of that
#   type reside on the level directly below its root folder. If the type is
#   included, any template objects of that type identified using the template
#   token will be copied.
optionIncludeProfiles=true
```

Possible errors and warnings

Error log location: <InstallDir>\SAP BusinessObjects Enterprise XI 4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>" application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the Multitenancy Management Tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as template-ContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running the Multitenancy Management Tool.	A general error alert	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter is incorrect.	Enter the correct user name.

Related Topics

- [templateToken](#)

5.2.2.4.6 optionIncludeEvents

Indicates whether the program will search for Business Intelligence (BI) platform event folders using the `templateToken` string and create a new tenant version.

Category

Tenant on-boarding/provisioning configuration > Template support options

Required setting?

No

Default setting

true

Example

Type	Example	Note
Boolean	optionIn cludeEvents=true	true or unspecified is the default value.
Boolean	optionIn cludeEvents=false	false

User interface available in CMC

No

Log file success messages

none

Properties file comment

```
# (Optional) enable template support options
# - All options listed below are optional. Document(Content) and User Group templates are mandatory.
# - If any of the listed options is not specified, the default will be false.
# - Each supported type has a configuration entry for template folders
#   (see Template Folders) except Access Level, Profile, and User Group.
#   The latter types are of flat folder structure, meaning all objects of that
#   type reside on the level directly below its root folder. If the type is
#   included, any template objects of that type identified using the template
#   token will be copied.
optionIncludeEvents=true
```

Possible errors and warnings

Error log location: <InstallDir>\SAP BusinessObjects Enterprise XI 4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>" application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the Multitenancy Management Tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as template-ContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running the Multitenancy Management Tool.	A general error alert	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter is incorrect.	Enter the correct user name.

Related Topics

- [templateToken](#)

5.2.2.4.7 optionIncludeAccessLevels

Indicates whether the program will search for BI platform access levels (groups of rights) using the `templateToken` string and create a new tenant version..

Category

Tenant on-boarding/provisioning configuration > Template support options

Required setting?

No

Default setting

true

Example

Type	Example	Note
boolean	optionIncludeAccessLevels=true	true, or unspecified is the default value
boolean	optionIncludeAccessLevels=false	false

User interface available in CMC

No

Log file success messages

none

Properties file comment

```
# (Optional) enable template support options
# - All options listed below are optional. Document(Content) and User Group templates are mandatory.
# - If any of the listed options is not specified, the default will be false.
# - Each supported type has a configuration entry for template folders
#   (see Template Folders) except Access Level, Profile, and User Group.
#   The latter types are of flat folder structure, meaning all objects of that
#   type reside on the level directly below its root folder. If the type is
#   included, any template objects of that type identified using the template
#   token will be copied.
optionIncludeAccessLevels=true
```

Possible errors and warnings

Error log location: <InstallDir>\SAP BusinessObjects Enterprise XI 4.0\Logging\multitenancymanager<yyyy><mm><dd><ttttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>" application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the Multitenancy Management Tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as templateContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running Multitenancy Management Tool.	A general error alert	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter is incorrect.	Enter the correct user name.

5.2.2.4.8 optionUseSharedUniverses

Indicates whether template reports and analytics use shared universes. Use in conjunction with [sharedUniverseFolder](#).

Category

Tenant on-boarding/provisioning configuration > Template support options

Required setting?

No

Default setting

true

Example

Type	Example	Note
boolean	optionUseSharedUniverses=true	true, or unspecified is the default value
boolean	optionUseSharedUniverses=false	false

User Interface available in CMC

No

Log file success messages

none

Properties File comment

```
# (Optional) enable template support options
# - All options listed below are optional. Document(Content) and User Group templates are mandatory.
# - If any of the listed options is not specified, the default will be false.
# - Each supported type has a configuration entry for template folders
#   (see Template Folders) except Access Level, Profile, and User Group.
#   The latter types are of flat folder structure, meaning all objects of that
#   type reside on the level directly below its root folder. If the type is
#   included, any template objects of that type identified using the template
#   token will be copied.
optionUseSharedUniverses=true
```

Possible errors and warnings

Error log location: <INSTALLDIR>\SAP BusinessObjects Enterprise XI
4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>"application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the multitenancy management tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as templateContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running multitenancy management tool.	A general error alert.	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter incorrect.	Enter the correct user name.

5.2.2.4.9 optionUseSharedConnections

Indicates whether template reports and analytics use shared connections. Use in conjunction with [sharedConnectionFolder](#).

Note:

Category

Tenant on-boarding/provisioning configuration > Template support options

Required setting?

No

Default setting

true

Example

Type	Example	Note
boolean	optionUseSharedConnections=true	true, or unspecified is the default value
boolean	optionUseSharedConnections=false	false

CMC User Interface

No

Log file success messages

none

Properties File comment

```
# (Optional) enable template support options
# - All options listed below are optional. Document(Content) and User Group templates are mandatory.
# - If any of the listed options is not specified, the default will be false.
# - Each supported type has a configuration entry for template folders
#   (see Template Folders) except Access Level, Profile, and User Group.
#   The latter types are of flat folder structure, meaning all objects of that
#   type reside on the level directly below its root folder. If the type is
#   included, any template objects of that type identified using the template
#   token will be copied.
optionUseSharedConnections=true
```

Possible errors and warnings

Error log location: <INSTALLDIR>\SAP BusinessObjects Enterprise XI
4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>"application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the multitenancy management tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as templateContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running multitenancy management tool.	A general error alert.	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter incorrect.	Enter the correct user name.

5.2.2.5 Template folder options

5.2.2.5.1 templateContentFolder

A semicolon-separated list of folder paths that defines all top-level template folders in the repository. Template folders contain the template token string in their name.

Category

Tenant on-boarding/provisioning configuration > Template folder options

Required setting?

No

Default setting

- <blank>

Example

Type	Example	Note
alphanumeric name and file path	templateContentFolder=tenants/\$tenant\$	
alphanumeric name and file path	templateContentFolder=tenants/\$tenant\$_reports;tenants/\$tenant\$_programs/public	a semicolon ";" is used to separate values

User Interface available in CMC

No

Log file success messages

none

Properties File comment

```
# Template Folders - folder structures defining the location of each type of template objects
#
# - The folder structure should be specified with respect to the root folder of each
#   corresponding template type
#
# - The starting location of the tenant folder doesn't need to be at the top most level under
#   the root folder
#
# - Sub folders in the template folder will be mapped in tenant folder tree structure along with
#   documents contained in the folder subtree
#
# - The string substitution can be at any level and be substring of the folder name
#
# - Multiple template folders can be defined for each type of object
#
# - e.g /$tenant$/ and /public/$tenant$/reports will be mapped to /ABC/ and /public/ABC/reports for
#   tenant ABC along with all the content in the folder
#
# - If folder name happen to have the following characters ('\ ', '/', ';') in it, they will have to
#   be escaped using '\ '
#
# - and because this is a java source file, we have to escape '\ ' again
#
# - e.g if your folder is named "my;specialtest/folder\", you will have to specify it as
#   "my\\;specialtest\\/folder\\"
## (Optional) Document folder template
templateContentFolder=tenants/$tenant$_reports;tenants/$tenant$_programs/public
```

Possible errors and warnings

Error log location: <INSTALLDIR>\SAP BusinessObjects Enterprise XI
4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv

Error	Cause	Action
Invalid configuration of parameter "<config_option>"application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the multitenancy management tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as templateContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running multitenancy management tool.	A general error alert.	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter incorrect.	Enter the correct user name.

Related Topics

- [Adding public folders to the tenant template](#)

5.2.2.5.2 templateUniverseFolder

Set to the folder path where your universe templates are located. For example: \$TemplateToken\$_unshared

Category

Tenant on-boarding/provisioning configuration > Template folder options

Required setting?

No

Default setting

- <blank>

Example

Type	Example	Note
alphanumeric name and file path	templateUniverseFolder=tenants/\$tenant\$	
alphanumeric name and file path	templateUniverseFolder=tenants/\$tenant\$_univers;tenants/\$tenant\$_universe/public	a semicolon ";" is used to separate values

User Interface available in CMC

No

Log file success messages

none

Properties File comment

```
# Template Folders - folder structures defining the location of each type of template objects
#
# - The folder structure should be specified with respect to the root folder of each
#   corresponding template type
#
# - The starting location of the tenant folder doesn't need to be at the top most level under
#   the root folder
#
# - Sub folders in the template folder will be mapped in tenant folder tree structure along with
#   documents contained in the folder subtree
#
# - The string substitution can be at any level and be substring of the folder name
#
# - Multiple template folders can be defined for each type of object
#
# - e.g /$tenant$/ and /public/$tenant$/reports will be mapped to /ABC/ and /public/ABC/reports for
#   tenant ABC along with all the content in the folder
#
# - If folder name happen to have the following characters ('\\', '/', ';') in it, they will have to
#   be escaped using '\\'
#
# - and because this is a java source file, we have to escape '\\' again
#
# - e.g if your folder is named "my;specialtest/folder\\", you will have to specify it as
#   "my\\;specialtest\\/folder\\\\"
## (Optional) Universe folder template
templateUniverseFolder=tenants/$tenant$
```

Possible errors and warnings

Error log location: <INSTALLDIR>\SAP BusinessObjects Enterprise XI
4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>"application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the multitenancy management tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as templateContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running multitenancy management tool.	A general error alert.	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter incorrect.	Enter the correct user name.

5.2.2.5.3 templateConnectionFactory

Set to the folder path where your universe templates are located. For example: \$TemplateToken\$_unshared.

Category

Tenant on-boarding/provisioning configuration > Template folder options

Required setting?

No

Default setting

- <blank>

Example

Type	Example	Note
alphanumeric name and file path	templateConnectionFolder=tenants/\$tenant\$	
alphanumeric name and file path	templateConnectionFolder=tenants/\$tenant\$_connections;tenants/\$tenant\$_connections/public	a semicolon ";" is used to separate values

User Interface available in CMC

No

Log file success messages

none

Properties File comment

```
# Template Folders - folder structures defining the location of each type of template objects
#
# - The folder structure should be specified with respect to the root folder of each
#   corresponding template type
#
# - The starting location of the tenant folder doesn't need to be at the top most level under
#   the root folder
#
# - Sub folders in the template folder will be mapped in tenant folder tree structure along with
#   documents contained in the folder subtree
#
# - The string substitution can be at any level and be substring of the folder name
#
# - Multiple template folders can be defined for each type of object
#
# - e.g /$tenant$/ and /public/$tenant$/reports will be mapped to /ABC/ and /public/ABC/reports for
#   tenant ABC along with all the content in the folder
#
# - If folder name happen to have the following characters ('\\', '/', ';') in it, they will have to
#   be escaped using '\\'
#
# - and because this is a java source file, we have to escape '\\' again
#
# - e.g if your folder is named "my;specialtest/folder\\", you will have to specify it as
#   "my\\;specialtest\\/folder\\\\"
## (Optional) Connection folder template
templateConnectionFolder=tenants/$tenant$
```

Possible errors and warnings

Error log location: <INSTALLDIR>\SAP BusinessObjects Enterprise XI
4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>"application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the multitenancy management tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as templateContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running multitenancy management tool.	A general error alert.	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter incorrect.	Enter the correct user name.

5.2.2.5.4 templateCategoryFolder

Set to the folder path where your category templates are located. For example: \$TemplateToken\$_unshared.

Category

Tenant on-boarding/provisioning configuration > Template folder options

Required setting?

No

Default setting

- <blank>

Example

Type	Example	Note
alphanumeric name and file path	templateCategoryFolder=tenants/\$tenant\$	
alphanumeric name and file path	templateCategoryFolder=tenants/\$tenant\$_category;tenants/\$tenant\$_category/public	a semicolon ";" is used to separate values

User Interface available in CMC

No

Log file success messages

none

Properties File comment

```
# Template Folders - folder structures defining the location of each type of template objects
#
# - The folder structure should be specified with respect to the root folder of each
#   corresponding template type
#
# - The starting location of the tenant folder doesn't need to be at the top most level under
#   the root folder
#
# - Sub folders in the template folder will be mapped in tenant folder tree structure along with
#   documents contained in the folder subtree
#
# - The string substitution can be at any level and be substring of the folder name
#
# - Multiple template folders can be defined for each type of object
#
# - e.g /$tenant$/ and /public/$tenant$/reports will be mapped to /ABC/ and /public/ABC/reports for
#   tenant ABC along with all the content in the folder
#
# - If folder name happen to have the following characters ('\\', '/', ';') in it, they will have to
#   be escaped using '\\'
#
# - and because this is a java source file, we have to escape '\\' again
#
# - e.g if your folder is named "my;specialtest/folder\\", you will have to specify it as
#   "my\\;specialtest\\/folder\\\\"
## (Optional) Category folder template
templateCategoryFolder=$tenant$
```

Possible errors and warnings

Error log location: <INSTALLDIR>\SAP BusinessObjects Enterprise XI
4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>"application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the multitenancy management tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as templateContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running multitenancy management tool.	A general error alert.	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter incorrect.	Enter the correct user name.

5.2.2.5.5 templateEventFolder

Set to the folder path where your event templates are located. For example: \$TemplateToken\$_unshared.

Note:

Category

Tenant on-boarding/provisioning configuration > Template folder options

Required setting?

No

Default setting

- <blank>

Example

Type	Example	Note
alphanumeric name and file path	templateEventFolder=tenants/\$tenant\$	
alphanumeric name and file path	templateEventFolder=/Custom Events/\$tenant\$;/System Events/\$tenant\$_system	a semicolon ";" is used to separate values

User Interface available in CMC

No

Log file success messages

none

Properties File comment

```
# Template Folders - folder structures defining the location of each type of template objects
#
# - The folder structure should be specified with respect to the root folder of each
#   corresponding template type
#
# - The starting location of the tenant folder doesn't need to be at the top most level under
#   the root folder
#
# - Sub folders in the template folder will be mapped in tenant folder tree structure along with
#   documents contained in the folder subtree
#
# - The string substitution can be at any level and be substring of the folder name
#
# - Multiple template folders can be defined for each type of object
#
# - e.g /$tenant$/ and /public/$tenant$/reports will be mapped to /ABC/ and /public/ABC/reports for
#   tenant ABC along with all the content in the folder
#
# - If folder name happen to have the following characters ('\ ', '/', ';') in it, they will have to
#   be escaped using '\ '
#
# - and because this is a java source file, we have to escape '\ ' again
#
# - e.g if your folder is named "my;specialtest/folder\", you will have to specify it as
#   "my\\;specialtest\\/folder\\"
## (Optional) Event folder template
templateEventFolder=/Custom Events/$tenant$;/System Events/$tenant$_system
```

Possible errors and warnings

Error log location: <INSTALLEDIR>\SAP BusinessObjects Enterprise XI
4.0\Logging\multitenancymanager<yyyy><mm><dd><ttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>"application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the multitenancy management tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as templateContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running multitenancy management tool.	A general error alert.	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter incorrect.	Enter the correct user name.

5.2.2.6 Shared object options

5.2.2.6.1 sharedUniverseFolder

Universes in this folder are shared by multiple tenants.

If a shared Universe contains restriction (overload) templates, these templates need to be duplicated for each tenant that uses the shared Universe. Used in conjunction with [optionUseSharedUniverses=true](#).

Category

Tenant on-boarding/provisioning configuration > Shared object options

Required setting?

No

Default setting

- <blank>

Example

Type	Example	Note
alphanumeric name and file path	sharedUniverseFolder=tenants/Test Shared Universe	
alphanumeric name and file path	sharedUniverseFolder=tenants/Test Shared Universe1;tenants/Test Shared Universe2	a semicolon ";" is used to separate values

User Interface available in CMC

No

Log file success messages

none

Properties File comment

```
# (Optional) Shared Universe folder - Universes in this folder are shared by multiple tenants
# - If a shared Universe contains restriction (overload) templates, these templates
# need to be duplicated for each tenant that uses the shared Universe
sharedUniverseFolder=tenants/Test Shared Universe
```

Possible errors and warnings

Error log location: <INSTALLDIR>\SAP BusinessObjects Enterprise XI
4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>"application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the multitenancy management tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as templateContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running multitenancy management tool.	A general error alert.	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter incorrect.	Enter the correct user name.

5.2.2.6.2 sharedConnectionFolder

Connections that are shared by multiple tenants. A semicolon-separated list of connection folder paths where shared connections are located. Used in conjunction with [optionUseSharedConnections=true](#).

Category

Tenant on-boarding/provisioning configuration > Shared object options

Required setting?

No

Limitations

Windows Active Directory single sign-on accounts are not supported.

Default setting

- <blank> or use the pound sign # to comment out the string.

Example

Type	Example	Note
alphanumeric name and file path	sharedConnectionFolder=tenants/Test Shared Universe	
alphanumeric name and file path	sharedConnectionFolder=tenants/Test Shared Universe1;tenants/Test Shared Universe2	a semicolon ";" is used to separate values

User Interface available in CMC

No

Log file success messages

none

Properties File comment

```
# (Optional) Shared Connection folder - Connections in this folder are shared by multiple tenants
sharedConnectionFolder=tenants/Test Shared Connections
```

Possible errors and warnings

Error log location: <INSTALLDIR>\SAP BusinessObjects Enterprise XI
4.0\Logging\multitenancymanager<yyyy><mm><dd><tttttttt>.csv>

Error	Cause	Action
Invalid configuration of parameter "<config_option>"application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the multitenancy management tool. Often this is the result of mandatory options that are left blank.	Open your configuration file (tenant_template_def.properties) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as templateContentFolder. Note: A .csv log file will not be created when this error occurs.
There was an error running multitenancy management tool.	A general error alert.	Read subsequent errors or turn on tracing.
The <secEnterprise_junk plug-in does not exist in the CMS (FWM 02017). (FWB 00008)	user parameter incorrect.	Enter the correct user name.

5.2.3 (Database) Type specific configuration options

5.2.3.1 Crystal Reports 2011 database connections

For SAP Crystal Reports 2011, use `crystalreport.template<n>` and `crystalreport.tenantdb<n>` to enter the Data Source Name (DSN) information for the template and the new tenant.

If you have more than one template connection object, add additional `crystalreport.template<n>` and `crystalreport.tenantdb<n>` options. For example: `ccis.dataconnection.dbcredentials2, ccis.dataconnection.dbcredentials3, ... ccis.dataconnection.dbcredentials<n>`

The DSN values are semicolon-separated and are formatted as follows:

```
<database server>;
<database name>;
```

```
<data source type>;  
<network layer>;  
<username>;  
<password>
```

Note:

To find the correct values for <database server>;<database name>, right-click on the report in the CMC and select Database Configuration. The "Server" and "Database" fields can be copied to the tenant configuration file.

Template and tenant database table prefixes

Support for mapping database table prefixes is provided by using `crystalreport.templatetableprefixes<n>` and `crystalreport.tenantttableprefixes<n>`, by listing table prefixes separated by a semicolon for each table name.

You can specify additional templates and their new tenant settings by adding `crystalreport.templatetableprefixes<n>` and `crystalreport.tenantttableprefixes<n>` option pairings to the tenant configuration file. For example, `crystalreport.templateprefixes2` and `crystalreport.tenantttableprefixes2`.

Category

Database type specific configuration options

Required setting?

No

Syntax for DSN

```
crystalreport.templatedb<n>=<template_CCIS.CONN_CUID>;<DATA  
SOURCE>;<DATABASE>;<USER>;<PASSWORD>
```

```
crystalreport.tenantdb<n>=<template_CCIS.CONN_CUID>;<DATA  
SOURCE>;<DATABASE>;<USER>;<PASSWORD>
```

Syntax for table prefixes

```
crystalreport.templateprefixes<n>=<template_table_prefix_a>;<template_ta  
ble_prefix_b>
```

```
crystalreport.tenantttableprefixes<n>=<tenant_table_prefix_a> ;<tenant_ta  
ble_prefix_b>
```

Default setting

- None

Examples

Type	Example	Note
Alphanumeric	crystalreport.template<n>=template<n>server;template<n>db<n>name;odbc;administrator;password	A semicolon (;) is used to separate values
Alphanumeric	crystalreport.template<n>db1=ARiyILQC6h5Ogwf8VolZzqQ;tenantDBN;tenantDB;userABC;Password123	<p><###> 1</p> <p><template_CCIS.CONN_CUID> ARiyILQC6h5Ogwf8VolZzqQ</p> <p><DATASOURCE>tenantDBN</p> <p><USER>userABC</p> <p><PASSWORD>Password123</p>
Alphanumeric	crystalreport.template<n>tableprefixes<n>;<template_table_prefix_a>;<template_table_prefix_b>;	A semicolon (;) is used to separate values. Add a semicolon for empty table matches.
	crystalreport.template<n>tableprefixes1=templateprefixa;templateprefixb;	templateprefixa maps to an empty string in crystalreport.tenantttableprefixes1.
	crystalreport.tenantttableprefixes1=;templateprefixb;templateprefixc;	The first table entry is empty, which is matched from crystalreport.template<n>tableprefixes1 by templateprefixa. The last entry, templateprefixc is mapped to by an empty string from crystalreport.template<n>tableprefixes1.

Properties file comment

```
#####
# Type specific configuration
#####

# (Optional) CrystalReport Database connection info for the tenant
#
# - Replaces template database connection info with tenant database connection info.
#   Can be overridden by command-line option
# - Replaces template database table prefix with tenant database table prefix.
#   Can be overridden by command-line option
#
# [CR2011 Direct to Database]
# Syntax:
#     crystalreport.template<n>db<n>= <template_server_name>;
#                                     <template_database_name>;
#                                     <template_server_type>;
#                                     <template_user_name>;
#                                     <template_password>
```

```

#
#      crystalreport.tenantdb<n>=      <tenant_server_name>;
#                                     <tenant_database_name>;
#                                     <tenant_server_type>;
#                                     <tenant_user_name>;
#                                     <tenant_password>
#
#      (Optional when no prefix change)
#      crystalreport.templateprefixes<n>= <template_table_prefix_a>;
#                                     <template_table_prefix_b>
#
#      (Optional when no prefix change)
#      crystalreport.tenanttableprefixes<n>= <tenant_table_prefix_a>;
#                                     <tenant_table_prefix_b>
#
# - <n> is the nth database connection that needs to be updated.
# - When multiple database connections need to be updated, use templatedb1, templatedb2, templatedb3...
#   They must be paired with tenantdb1, tenantdb2, tenantdb3...
# - Corresponding nth templateprefixes and tenanttableprefixes represent the table prefix mapping
#   that needs update for this connection.
# - templateprefixes<n> and tenanttableprefixes<n> are semicolon separated and must always end
#   with a semicolon, in case the last one is actually an empty string
#
# The valid database server types include:
#      odbc
#      oracle
#      db2
#      sybase
#      informix
#      crdb_xml
#
# Example:
#      crystalreport.templatedb1=      templateserver;
#                                     templatedbname;
#                                     odbc;
#                                     administrator;
#                                     password
#
#      crystalreport.tenantdb1=      tenantserver;
#                                     tenantdbname;
#                                     odbc;
#                                     administrator;
#                                     mypassword
#
#      crystalreport.templateprefixes1= templateprefixa;
#                                     templateprefixb;
#                                     ;
#
#      crystalreport.tenanttableprefixes1= ;
#                                     tenantprefixb;
#                                     tenantprefixc;
#
# The above example shows that for report logon database number 1,
# templateprefixa maps to the empty string, templateprefixb maps to the tenantprefixb,
# and the empty string maps to tenantprefixc
#
#crystalreport.templatedb1=
#crystalreport.tenantdb1=
#crystalreport.templateprefixes1=
#crystalreport.tenanttableprefixes1=

```

5.2.3.2 Crystal Reports for Enterprise managed database connections

For SAP Crystal Reports for Enterprise, the Data Source Name (DSN) details of the template connection object are replicated for each tenant. For example: <CUID>;<data source name>;<database name>;<username>;<password>

CUID is the CUID of a template connection object, whereas the DSN information is of the intended new connection object for the tenant that will be created based on the template.

If you have more than one template connection object, add additional `ccis.dataconnection.dbcredentials` options. For example: `ccis.dataconnection.dbcredentials2`, `ccis.dataconnection.dbcredentials3`, ... `ccis.dataconnection.dbcredentials<n>`

Category

Database type specific configuration options

Required setting?

No

Default setting

- None

Example

Type	Example	Note
Alphanumeric	<code>crystalreport.ccis.dataconnection.template edb1=ARiyILQC6h5Ogwf8VolZzqQ</code>	
Alphanumeric	<code>crystalreport.ccis.dataconnection.tem platetableprefixes1=templateprefixa;templatepre fixb;;</code>	
Alphanumeric	<code>crystalreport.ccis.dataconnection.tenanttablepre fixes1=;tenantprefixb;tenantprefixc;</code>	

Properties file comment

```
# [(Optional) CR for Enterprise Managed Direct to Database]
# These options are only necessary when table prefix update is needed.
# Syntax:
#     crystalreport.ccis.dataconnection.template<n>=<template_connection_cuid_a>
#     crystalreport.ccis.dataconnection.template<n>=<template_table_prefix_a>;
#                                     <template_table_prefix_b>
#
#     crystalreport.ccis.dataconnection.tenanttableprefixes<n>=<tenant_table_prefix_a>;
#                                     <tenant_table_prefix_b>
#
# - Each ccis.dataconnection.template specifies one CUID for a template CCIS dataconnection
#   used in reports that needs to be updated to its tenant counterpart.
# - template<n> and tenanttableprefixes<n> are semicolon separated and must always
#   end with a semicolon, in case the last one is actually an empty string
#
# Note that the actual database information such as server name,
# database name, etc. should be updated through the
# corresponding CCIS.DataConnection database connection parameter.
#
# Example:
#     crystalreport.ccis.dataconnection.template1=ARiyILQC6h5Ogwf8VolZzqQ
#     crystalreport.ccis.dataconnection.template1=templateprefixa;templateprefixb;;
#     crystalreport.ccis.dataconnection.tenanttableprefixes1=;
```

```
# tenantprefixb;  
# tenantprefixc;  
#  
#crystalreport.ccis.dataconnection.templatedb1=  
#crystalreport.ccis.dataconnection.templatetableprefixes1=  
#crystalreport.ccis.dataconnection.tenanttableprefixes1=
```

5.2.3.3 CCIS.DataConnection database connections

The Data Source Name (DSN) details of the template connection object you want to replicate for each tenant. For example: <CUID>;<data source name>;<database name>;<username>;<password>

CUID is the CUID of a template connection object, whereas the DSN information is of the intended new connection object for the tenant that will be created based on the template.

For JDBC and OLEDB connections, the <data source name>;<database name> settings refer to the following:

- JDBC Oracle: <machinename:port>;<TNS service name>
- All other JDBC sources: <machinename:port>;<database name>
- OLEDB: <machinename>;<database name>

If you have more than one template connection object, add additional `ccis.dataconnection.dbcredentials` options. For example: `ccis.dataconnection.dbcredentials2`, `ccis.dataconnection.dbcredentials3`, ... `ccis.dataconnection.dbcredentials<n>`

Category

Database type specific configurations

Required setting?

No

Default setting

- None

Example

Type	Example	Note
Alphanumeric	#ccis.dataconnection.dbcredentials1=	Commented out
Alphanumeric	ccis.dataconnection.dbcredentials1=ARiyILQC6h5Ogwf8VolZzqQ;tenantDBN;tenantDB;userABC;Password123	Unique database connection string,

Properties file comment

```
# (Optional) CCIS.DataConnection database connection parameter
# - Updates the specified template CCIS connection with the provided database information
# - Please refer to the Tenant Provisioning Tool User Guide for more details
#
# Syntax:
# ccis.dataconnection.dbcredentials<###>=<template_CCIS.CONN_CUID>;<DATASOURCE>;<DATABASE>;<USER>;<PASSWORD>
#
# <###> = (String) (required) uniquely identify the connection db setting
# <template_CCIS.CONN_CUID> = (String) (required) CUID of the template CCIS Connection object
# <DATASOURCE> = (String) (required) name of the data source
# <DATABASE> = (String) (required) depending on datasource) name of the database
# <USER> = (String) (required) user id for the database
# <PASSWORD> = (String) (required) password for the supplied database user id.
# Leave empty if the password is blank.
#
# Example:
# ccis.dataconnection.dbcredentials1=ARiyILQC6h5Ogwf8VolZzqQ;tenantDBN;tenantDB;userABC;Password123
#
# (uncomment the following and configure)
#ccis.dataconnection.dbcredentials1=
```

5.2.4 tenant_template_def.properties

This is a copy of the tenant properties file `tenant_template_def.properties` as it appears in a new BI platform installation.

```
#
# Tenant Template Definition Properties File
#
# Note:
# - All the default template tokens used in this properties file are just
#   examples and can be replaced with other strings
# - All configurations can be overwritten by command line options of the tool
#
#####
# Mandatory configuration
#####

# (Mandatory) Name of the tenant being added.
# - Tenant specific info, overridden by command-line option
tenantName=ABC

# (Mandatory) Template token identifier used for tenant name replacing
# - Can be any format but need to clearly identify the templates so that any regular objects not intended
#   as templates
# should not have the token embedded in their names.
# - E.g. User group template /$tenant$_users/ for tenant ABC will become /ABC_users/
```

```

templateToken=$tenant$

# (Mandatory) Logon information
cms=
auth=secEnterprise
user=administrator
pwd=

# (Optional) Review configuration errors/warnings, if any, before proceeding to the actual MTM operation.
# - true: (default) review the configuration errors and warnings the tool found before giving instruction
# on how to proceed
# - false: the tool will not prompt the user for review, instead it will display the errors/warnings and
# ignore them
reviewBeforeProceed=true

#####
# Optional on-boarding/provisioning configuration
#####

# (Optional) Number of concurrent users allowed for the tenant being added. Defaults to -1 meaning unlimited
# in which
# case the number of concurrent users for this tenant will only be limited by what the BOE license key allows
# for the entire system. When not specified, this option will be set to default during the tenant's first
# on-board,
# or unchanged during tenant's provisioning.
# - Tenant specific info
tenantConcurrentUserLimit=100

# (Optional) Status log location - directory path for the status log file
# - The tenant name and timestamp (?) will be used to construct the log file name
# - Default status log goes to the logging directory in the BOE install, typically
# c:\program files (x86)\SAP BusinessObjects\SAP BusinessObjects Enterprise XI 4.0\logging
statusLog=

# (Optional) Rollback option
# - true: rollback if one of the infoobjects fail to commit
# - false: keep going, and log the failures
# - default is true
abortANDRollback=true

# (Optional) security import options
# 0: merge mode, or access control element level overwrite. Template security info will be copied onto
# tenant's existing security info
# 1: (default) per principal overwrite mode, or principal level overwrite. Template security info will
# overwrite tenant's security info
# on those assigned principals.
# 2: overwrite mode, or object level overwrite. Template security info will overwrite tenant's existing
# security info
# 3: ignore mode. Template security info will not be copied over to the tenant
optionImportSecMode=1

# (Optional) enable template support options
# - All options listed below are optional. Document(Content) and User Group templates are mandatory.
# - If any of the listed options is not specified, the default will be false.
# - Each supported type has a configuration entry for template folders (see Template Folders) except Access
# Level, Profile, and User Group.
# The latter types are of flat folder structure, meaning all objects of that type reside on the level di
# rectly below its root folder.
# If the type is included, any template objects of that type identified using the template token will be
# copied.
optionIncludeUniverses=true
optionIncludeConnections=true
optionIncludeCategories=true
optionIncludeProfiles=true
optionIncludeEvents=true
optionIncludeAccessLevels=true
optionUseSharedUniverses=true
optionUseSharedConnections=true

# Template Folders - folder structures defining the location of each type of template objects
# - The folder structure should be specified with respect to the root folder of each corresponding template
# type
# - The starting location of the tenant folder doesn't need to be at the top most level under the root
# folder
# - Sub folders in the template folder will be mapped in tenant folder tree structure along with documents
# contained in the folder subtree
# - The string substitution can be at any level and be substring of the folder name
# - Multiple template folders can be defined for each type of object
# - e.g /$tenant$/ and /public/$tenant$/reports will be mapped to /ABC/ and /public/ABC/reports for tenant
# ABC along with all the content

```

```

# in the folder
# - If folder name happen to have the following characters ('\\', '/', ';') in it, they will have to be escaped
  using '\\'
# - and because this is a java source file, we have to escape '\\' again
# - e.g if your folder is named "my;specialtest/folder\", you will have to specify it as "my\\;special
  test\\folder\\"

## (Optional) Document folder template
templateContentFolder=tenants/$tenant$_reports;tenants/$tenant$_programs/public

## (Optional) Universe folder template
templateUniverseFolder=tenants/$tenant$

## (Optional) Connection folder template
templateConnectionFolder=tenants/$tenant$

## (Optional) Category folder template
templateCategoryFolder=$tenant$

## (Optional) Event folder template
templateEventFolder=/Custom Events/$tenant$;/System Events/$tenant$_system

# (Optional) Shared Universe folder - Universes in this folder are shared by multiple tenants
# - If a shared Universe contains restriction (overload) templates, these templates
# need to be duplicated for each tenant that uses the shared Universe
sharedUniverseFolder=tenants/Test Shared Universe

# (Optional) Shared Connection folder - Connections in this folder are shared by multiple tenants
sharedConnectionFolder=tenants/Test Shared Connections

#####
# Type specific configuration
#####

# (Optional) CrystalReport Database connection info for the tenant
#
# - Replaces template database connection info with tenant database connection info. Can be overridden by
  command-line option
# - Replaces template database table prefix with tenant database table prefix. Can be overridden by command-
  line option
#
# [CR2011 Direct to Database]
# Syntax:
#     crystalreport.templateddb<n>=<template_server_name>;<template_database_name>;
#                                     <template_server_type>;<template_user_name>;<template_password>
#     crystalreport.tenantddb<n>=<tenant_server_name>;<tenant_database_name>;<tenant_server_type>;<tenant_us
  er_name>;<tenant_password>
#     (Optional when no prefix change) crystalreport.templateprefixes<n>=<template_table_prefix_a>;<tem
  plate_table_prefix_b>
#     (Optional when no prefix change) crystalreport.tenantprefixes<n>=<tenant_table_prefix_a>;<ten
  ant_table_prefix_b>
#
# - <n> is the nth database connection that needs to be updated.
# - When multiple database connections need to be updated, use templatedb1, templatedb2, templatedb3...
# They must be paired with tenantdb1, tenantdb2, tenantdb3...
# - Corresponding nth templateprefixes and tenantprefixes represent the table prefix mapping
  that
#     needs update for this connection.
# - templateprefixes<n> and tenantprefixes<n> are semicolon separated and must always end with
  a semicolon,
#     in case the last one is actually an empty string
#
# The valid database server types include:
#     odbc
#     oracle
#     db2
#     sybase
#     informix
#     crdb_xml
#
# Example:
#     crystalreport.templateddb1=templateserver;templatedbname;odbc;administrator;password
#     crystalreport.tenantddb1=tenantserver;tenantdbname;odbc;administrator;mypassword
#     crystalreport.templateprefixes1=templateprefixa;templateprefixb;;
#     crystalreport.tenantprefixes1=;tenantprefixb;tenantprefixc;
#
# The above example shows that for report logon database number 1, templateprefixa maps to the empty string,
  templateprefixb
# maps to the tenantprefixb, and the empty string maps to tenantprefixc
#

```

```

#crystalreport.templateddb1=
#crystalreport.tenantddb1=
#crystalreport.templateprefixes1=
#crystalreport.tenanttableprefixes1=

# [(Optional) CR for Enterprise Managed Direct to Database]
# These options are only necessary when table prefix update is needed.
# Syntax:
#     crystalreport.ccis.dataconnection.templateddb<n>=<template_connection_cuid_a>
#     crystalreport.ccis.dataconnection.templateprefixes<n>=<template_table_prefix_a>;<template_table_prefix_b>
#     crystalreport.ccis.dataconnection.tenanttableprefixes<n>=<tenant_table_prefix_a>;<tenant_table_prefix_b>
#
# - Each ccis.dataconnection.templateddb specifies one CUID for a template CCIS dataconnection used in reports that needs
#   to be updated to its tenant counterpart.
# - templateprefixes<n> and tenanttableprefixes<n> are semicolon separated and must always end with a semicolon,
#   in case the last one is actually an empty string
#
# Note that the actual database information such as server name, database name, etc. should be updated through the
# corresponding CCIS.DataConnection database connection parameter.
#
# Example:
#     crystalreport.ccis.dataconnection.templateddb1=ARiyILQC6h5Ogwf8VolZzqQ
#     crystalreport.ccis.dataconnection.templateprefixes1=templateprefixa;templateprefixb;;
#     crystalreport.ccis.dataconnection.tenanttableprefixes1=tenantprefixb;tenantprefixc;
#
#crystalreport.ccis.dataconnection.templateddb1=
#crystalreport.ccis.dataconnection.templateprefixes1=
#crystalreport.ccis.dataconnection.tenanttableprefixes1=

# (Optional) CCIS.DataConnection database connection parameter
# - Updates the specified template CCIS connection with the provided database information
# - Please refer to the Tenant Provisioning Tool User Guide for more details
#
# Syntax:
#     ccis.dataconnection.dbcredentials<###>=<template_CCIS.CONN_CUID>;<DATASOURCE>;<DATABASE>;<USER>;<PASSWORD>
#
# <###> = (String) (required) uniquely identify the connection db setting
# <template_CCIS.CONN_CUID> = (String) (required) CUID of the template CCIS Connection object
# <DATASOURCE> = (String) (required) name of the data source
# <DATABASE> = (String) (required depending on datasource) name of the database
# <USER> = (String) (required) user id for the database
# <PASSWORD> = (String) (required) password for the supplied database user id. Leave empty if the password is blank.
#
# Example:
#     ccis.dataconnection.dbcredentials1=ARiyILQC6h5Ogwf8VolZzqQ;tenantDBN;tenantDB;userABC;Password123
#
# (uncomment the following and configure)
#ccis.dataconnection.dbcredentials1=

```

Managing tenants in the CMC

After configuring tenants in the `tenant_template_def.properties` file and running the Multitenancy Management Tool to create tenants, you can manage the tenants in the Central Management Console (CMC).

To manage tenants, go to the **Multitenancy** management area of the CMC. You can perform the following tasks:

- Set some tenant properties, such as maximum concurrent users, and so on
- View user and group associations for a tenant
- Add a user group to or remove one from a tenant
- Delete a tenant

For more information about the `tenant_template_def.properties` file, see “Tenant definition configuration file reference”.

Related Topics

- [Tenant definition configuration file reference](#)

6.1 To set tenant properties

You can set the following properties, without modifying the tenant properties file, in the Central Management Console (CMC):

- Tenant name
- Description
- Keywords
- Concurrent users

You cannot edit these read-only tenant properties in the CMC:

- ID
- CUID
- Date created
- Date last modified

For detailed information about each property in the file, see “Tenant definition configuration file reference”.

Tip:

You can select a tenant and click  on the toolbar to go directly to the "Properties" dialog box.

6.1.1 To change the name of the tenant

1. In the Central Management Console (CMC), select the **Multitenancy** area.
2. Double-click the tenant.
The tenant "Properties" dialog box appears.
3. In the **Tenant Name** box, type a new name for the tenant.
4. Click **Save & Close**.
The name you entered appears for the tenant.

Related Topics

- [tenantName](#)

6.1.2 To change the description of the tenant

1. In the Central Management Console (CMC), select the **Multitenancy** area.
2. Double-click the tenant.
The tenant "Properties" dialog box appears.
3. In the **Description** box, type a description of the tenant.
4. Click **Save & Close**.
The description you entered appears for the tenant.

Related Topics

- [tenantName](#)

6.1.3 To change keywords for the tenant

1. In the Central Management Console (CMC), select the **Multitenancy** area.

2. Double-click the tenant.
The tenant "Properties" dialog box appears.
3. In the **Keywords** box, type keywords for the tenant
4. Click **Save & Close**.
The keywords you entered appear for the tenant.

Related Topics

- [tenantName](#)

6.1.4 To change the number of concurrent users for a tenant

1. In the Central Management Console (CMC), select the **Multitenancy** area.
2. Double-click the tenant.
The tenant "Properties" dialog box appears.
3. Under **Concurrent Users**, select the maximum number of concurrent users who can log on to the CMC for this tenant:
 - To enter a maximum number of users for this tenant who can log on to the CMC, select **Value**, and enter the number.

If the maximum number is exceeded, a message appears and the user cannot log on.
 - If you do not want to limit the number of concurrent users for this tenant, select **Unlimited**.
4. Click **Save & Close**.
The values you defined appear under the **Concurrent Users** column on the "Multitenancy" home page.

Related Topics

- [tenantName](#)

6.2 To assign access rights to a tenant user group

You can set access rights for tenant user groups in the Central Management Console (CMC), without modifying the properties file.

Tip:

You can select a tenant and click  on the toolbar to go directly to the "User Security" dialog box.

1. In the CMC, select the **Multitenancy** area.
2. Right-click the tenant and select **User Security**.
3. In the "User Security" dialog box, click **Add Principals**.
4. In the "Add Principals" dialog box, move the tenant user group for which you want to set access rights from the **Available Users or Groups** list to the **Selected Users or Groups** list.
5. Click **Add and Assign Security**.
6. In the "Assign Security" dialog box, select access-right levels to grant to the tenant user group.
7. To enable folder inheritance, select the **Inherit From Parent Folder** check box.
To disable group inheritance, clear the check box.
8. To enable group inheritance, select the **Inherit From Parent Group** check box.
To disable group inheritance, clear the check box.
9. Click **OK**, and click **Close**.

The user group is assigned the access rights you selected.

6.2.1 To remove access rights from a tenant

You can remove access rights from a tenant user group in the Central Management Console (CMC), without modifying the properties file.

1. In the CMC, select the **Multitenancy** area.
2. Right-click the tenant and select **User Security**.
3. In the "User Security" dialog box, click **Add Principals**.
4. In the "Add Principals" dialog box, move the tenant user group for which to remove rights from the **Available Users or Groups** list to the **Selected Users or Groups** list.
5. Click **Add and Assign Security**.
6. In the "Assign Security" dialog box, click **Remove Access**.
7. Click **OK**, and click **Close**.

All access rights are removed from the tenant user group.

6.3 Managing user groups for a tenant

6.3.1 To view user and group associations for a tenant

You can view user and user group associations for a tenant in the Central Management Console (CMC), without modifying the properties file.

Tip:


You can select a tenant and click  on the toolbar to go directly to the "User Groups" dialog box.

1. In the CMC, select the **Multitenancy** area.
2. Double-click the tenant for which you want to view user and group associations.
3. In the "Properties" dialog box, click **User Groups** in the navigation list.
The "User Groups" dialog box appears, listing the groups associated with this tenant.

6.3.2 To add a user group to a tenant

You can add a user group to a tenant in the Central Management Console (CMC), without modifying the properties file.

Tip:

You can select a tenant and click  on the toolbar to go directly to the "Add Groups to Tenant" dialog box.

1. In the CMC, select the **Multitenancy** area.
2. Right-click the tenant to add a user group to and select **Add Groups to Tenant**.
3. In the "Add Groups to Tenant" dialog box, move the user group to add from the **Available Groups** list to the **Selected Groups** list.
4. Click **OK**.

The user group is added to the tenant.

6.3.3 To remove a user group from a tenant

You can remove a user group from a tenant in the Central Management Console (CMC), without modifying the properties file.

1. In the CMC, select the **Multitenancy** area.
2. Double-click the tenant from which to remove a user group.
3. In the tenant "Properties" dialog box, click **User Groups** in the navigation list.
4. In the "User Groups" dialog box, select the user group to remove, and click **Remove**.

The user group is removed from the tenant.


6.4 To delete a tenant

You can delete tenants and all of their objects in the Central Management Console (CMC) from the BI repository.

Note:

Objects that are shared, or where modification rights are not granted are not deleted.

Tip:

You can select a tenant and click  on the toolbar to go directly to the "Delete" dialog box.

1. In the CMC, select the **Multitenancy** area.
2. Right-click a tenant, select **Delete**.
3. In the "Delete" dialog box, move the tenant to delete from the **Available** list to the **Excluded** list, and click **OK**.
4. In the confirmation dialog box that appears, click **OK** again.

The tenant is deleted from the Central Management Server (CMS) repository.

Troubleshooting

Consider the following best practices when troubleshooting errors.

Note the return code

The Multitenancy Management Tool supplies a return code (Java exit status) that can be captured by the batch files or shell scripts that run the tool. The return code enables you to determine success (return code 0) or failure (return code 1) after the tool runs. For example, you can print the return code to the command prompt as follows:

- Unix:

```
java -jar multitenancymanager.jar -configFile tenant_template_def.properties
echo "return code is $?"
```

- Windows:

```
java -jar multitenancymanager.jar -configFile tenant_template_def.properties
echo "return code is %ERRORLEVEL%"
```

If the tool fails, use the log files to troubleshoot the specific issue.

Review the tool log (.csv) file

Program messages displayed on the command line as the Multitenancy Management Tool runs are saved to a log file that you can review after the tool runs. By default, a tool log file named `multitenancymanager<yyyy><mm><dd><time>.csv` is created in the following folder:

- Windows: `<INSTALLEDIR>\SAP BusinessObjects Enterprise XI 4.0\logging\`
- Unix: `<INSTALLEDIR>/sap_bobj/logging/`

You can use the `statusLog` option in the tenant configuration file to change the location of the log file.

Enable and review the trace log (.glf) file

System-level messages generated by BI platform servers and applications can be traced and written to log files. Trace log files contain more detailed information for diagnosing problems than command-line output or the tool log (.csv) file.

To help diagnose problems, enable tracing. To configure tracing for the Multitenancy Management Tool, use the `BO_trace.ini` configuration file to specify the type and verbosity of information traced and sent to the log file.

Trace messages are collected in log files that are saved with the generic log file (.glf) extension. The .glf file name is formatted as a combination of shorthand identifier and number reference (for example, `multitenancymanager_trace.000001.glf`). A new trace log file is created when the log file approaches a preconfigured maximum size. Traces for the Multitenancy Management Tool are created in the following folder:

- Windows: <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\logging\
- Unix: <INSTALLDIR>/sap_bobj/enterprise_xi40/logging/

Determine where the error originated

Multitenancy Management Tool errors are often caused by incorrect values in the tenant configuration file (`tenant_template_def.properties`) or by problems on the machine where `multitenancy manager.jar` is run.

Errors that originate from the BI platform servers or security framework are accompanied by an error code—a three-letter string followed by a five-digit string. Common error codes are FWB and FWM. For example:

```
Enterprise authentication could not log you on. Please make sure your logon information is correct. (FWB00008)
```

For a list of possible error codes and solutions, see the *SAP BusinessObjects Business Intelligence Suite Error Message Guide*.

Related Topics

- [To configure tracing for the Multitenancy Management Tool](#)
- [Multitenancy Management Tool errors](#)

7.1 Multitenancy Management Tool errors

The following table provides a list of common errors displayed by the Multitenancy Management Tool and recorded in the `.csv` log file.

Error	Cause	Action
Invalid application configuration file, error parsing <config_option>.	The value provided for <config_option> cannot be processed by the Multitenancy Management Tool. Often this is the result of mandatory options that are left blank.	<p>Open your configuration file (<code>tenant_template_def.properties</code>) and ensure that the value for <config_option> is set correctly. Refer to the documentation and comments in the configuration file for examples. <config_option> is the name of the option, such as <code>templateContentFolder</code>.</p> <p>Note: A <code>.csv</code> log file will not be created when this error occurs.</p>

Error	Cause	Action
An unexpected error has occurred.	An unspecified exception occurred when running the Multitenancy Management Tool.	Check the trace log file (<code>multitenancymanager_trace.00000<n>.glf</code>) to perform further analysis.
No matching objects are found using query <code><query></code> .	No template objects can be retrieved based on the template paths specified.	<p>Open your configuration file (<code>tenant_template_def.properties</code>) and ensure that any template paths specified are correct. Check the following options:</p> <ul style="list-style-type: none"> • <code>templateContentFolder</code> • <code>templateUniverseFolder</code> • <code>templateConnectionFolder</code> • <code>templateCategoryFolder</code> • <code>templateEventFolder</code> • <code>sharedUniverseFolder</code> • <code>sharedConnectionFolder</code> <p>Note: You may not have templates for some object types. It is best practice to set the appropriate <code>optionInclude</code> option to <code>false</code> in these cases. For example, <code>optionIncludeCategories=false</code>.</p>
Failed to fetch object(s).	An error occurred when fetching objects from the Central Management Server (CMS).	<p>Check that you can connect to the CMS. If the CMS can be connected to, check the <code>.csv</code> log file for the object CUIDs and ensure that they exist in the CMS.</p> <p>If the CMS can be connected to and the objects exist, check the trace log file (<code>multitenancymanager_trace.00000<n>.glf</code>) to perform further analysis.</p>
Failed to map template object(s) to tenant object(s).	An error occurred when mapping a template object to a tenant object or when the Multitenancy Management Tool CMS application object is retrieved.	<p>Check that you have enough disk space on the home directory of the operating system user running <code>multitenancymanager.jar</code>. Ensure that the home directory is writable.</p> <p>If there is adequate disk space, check the trace log file (<code>multitenancymanager_trace.00000<n>.glf</code>) to perform further analysis.</p>

Error	Cause	Action
Failed to perform backup.	<p>An error occurred while creating a backup of existing tenant objects.</p> <p>This error may occur when running <code>multitenancymanager.jar</code> a second time (or more) on the same tenant. Since the tenant already exists, existing tenant objects are backed up locally by the tool to a BIAR file and this error may be thrown during this process.</p>	<p>Check that you have enough disk space on the home directory of the operating system user running <code>multitenancymanager.jar</code>. Ensure that the home directory is writable.</p> <p>If there is adequate disk space, check the trace log file (<code>multitenancymanager_trace.000000<n>.glf</code>) to perform further analysis.</p>
Failed to restore backup.	<p>An error occurred while restoring the backup of tenant objects to their original state.</p> <p>This error may occur when running <code>multitenancymanager.jar</code> a second time (or more) on the same tenant and if another error triggers a rollback (<code>abortANDRollback=true</code>). When rolling back changes, the tool attempts to restore the original objects and settings of a tenant from the local backup and this error may be thrown during this process.</p>	<p>Check that you have enough disk space on the home directory of the operating system user running <code>multitenancymanager.jar</code>. Ensure that the home directory is writable.</p> <p>If there is adequate disk space, check the trace log file (<code>multitenancymanager_trace.000000<n>.glf</code>) to perform further analysis.</p>

Error	Cause	Action
Failed to refresh tenant object(s) or update shared object(s).	An error occurred while attempting to update tenant documents with a tenant-specific data sources.	<p>Check for a more specific error thrown earlier in the <code>.csv</code> log file. Often the result of incorrect or mismatched configuration file options such as:</p> <ul style="list-style-type: none"> <code>crystalreport.template<n></code> <code>crystalreport.tenantdb<n></code> <code>crystalreport.templateprefixes<n></code> <code>crystalreport.tenanttableprefixes<n></code> <code>crystalreport.ccis.dataconnection.template<n></code> <code>crystalreport.ccis.dataconnection.templateprefixes<n></code> <code>crystalreport.ccis.dataconnection.tenanttableprefixes<n></code> <code>ccis.dataconnection.dbcredentials<n></code> <p>The CUID and name of the report or document is recorded.</p> <p>If your configuration file settings are correct, check the trace log file (<code>multitenancymanager_trace.00000<n>.glf</code>) to perform further analysis.</p>
Failed to load document refresh plugin for kind <code><SI_KIND></code> .	An error occurred while loading a JAR file from the docRefresh Plugins folder. <code><SI_KIND></code> refers to the object type, for example <code>Webi</code> (Web Intelligence documents).	<p>By default, the <code>docRefreshPlugins</code> folder is located at:</p> <ul style="list-style-type: none"> <code><INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\java\apps\multitenancyManager\jars\docRefreshPlugins\</code> <code><INSTALLDIR>/sap_bobj/enterprise_xi40/java/apps/multitenancyManager/jars/docRefreshPlugins /</code> <p>Ensure that this folder and all required JARs exist for the object types you are importing or updating, such as <code>ccisdataconnection.Refresh.jar</code>, <code>CrystalReportRefresh.jar</code>, <code>UniverseRefresh.jar</code>, and <code>WebiRefresh.jar</code>.</p>

Error	Cause	Action
Failed to refresh document for <code><document_details></code> .	An error occurred while attempting to update a tenant document with a tenant-specific data source.	Check the trace log file (<code>multitenancymanager_trace.00000<n>.glf</code>) to perform further analysis.
Failed to prepare tenant for importing.	An error occurred while retrieving a tenant's objects that are to be imported to the CMS.	Check the trace log file (<code>multitenancymanager_trace.00000<n>.glf</code>) to perform further analysis.
Missing template database configuration <code>crystalreport.template<n></code> or tenant database configuration <code>crystalreport.tenantdb<n></code> .	A template to tenant mapping for SAP Crystal Reports 2011 direct-to-data connections is missing or not configured correctly.	Open your configuration file (<code>tenant_template_def.properties</code>) and ensure that every <code>crystalreport.template<n></code> option has a corresponding <code>crystalreport.tenantdb<n></code> option. Check the trace log file (<code>multitenancymanager_trace.00000<n>.glf</code>) to perform further analysis.
Invalid template database configuration, error parsing <code><config_option></code> .	The template database configuration for an SAP Crystal Reports 2011 report does not have the right format.	Open your configuration file (<code>tenant_template_def.properties</code>) and ensure that all <code>crystalreport.template<n></code> options are set to valid values. For details on setting tenant options for SAP Crystal Reports 2011 direct-to-data connections, see SAP Crystal Reports 2011 .
Invalid tenant database configuration, error parsing <code>crystalreport.tenantdb<n></code> .	The tenant database configuration for an SAP Crystal Reports 2011 report does not have the right format.	Open your configuration file (<code>tenant_template_def.properties</code>) and ensure that all <code>crystalreport.tenantdb<n></code> options are set to valid values. For details on setting tenant options for SAP Crystal Reports 2011 direct-to-data connections, see SAP Crystal Reports 2011 .

Error	Cause	Action
Invalid database server type <code><dbserver_type></code> .	The specified data source type (<code><dbserver_type></code>) for a SAP Crystal Reports 2011 direct-to-data connection is not supported.	<p>Open your configuration file (<code>tenant_template_def.properties</code>) and ensure you are using supported data source types for both template and tenant reports. Check the following options:</p> <ul style="list-style-type: none"> <code>crystalreport.template_db<n></code> <code>crystalreport.tenant_db<n></code> <p>For a list of supported data source types, see SAP Crystal Reports 2011.</p>
Table prefixes mismatch for <code><template_prefix></code> and <code><tenant_prefix></code> , either one of them does not exist, or number of prefixes do not match.	The number of template table prefixes does not equal the number of tenant table prefixes.	<p>Open your configuration file (<code>tenant_template_def.properties</code>) and ensure that every template table prefix option has a corresponding mapping to a tenant table prefix option with an equal number of prefixes. Check the following Crystal Reports options:</p> <ul style="list-style-type: none"> <code>crystalreport.template_table_prefixes<n></code> <code>crystalreport.tenant_table_prefixes<n></code> <code>crystalreport.ccis.dataconnection.template_table_prefixes</code> <code>crystalreport.ccis.dataconnection.tenant_table_prefixes</code> <p>Tip: Remember to end the prefix lists with a semicolon. Multiple prefixes are separated by semicolons, and the list must end in a semicolon to be able to identify what is an empty string ("").</p>
Database configuration is invalid, please see previous item(s) for details.	The database configuration for a tenant's Crystal report is invalid. Error can be encountered for both SAP Crystal Reports 2011 and SAP Crystal Reports for Enterprise.	<p>Check for a data source update error thrown earlier in the <code>.csv</code> log file.</p> <p>Check the trace log file (<code>multitenancymanager_trace.00000<n>.glf</code>) to perform further analysis.</p>

Error	Cause	Action
Unexpected internal error has occurred while updating a crystal report document.	An unexpected error occurred while updating a SAP Crystal Reports for Enterprise report with a tenant-specific data source.	Check that the Crystal Reports Processing Server is enabled and running. If the server is running, check the trace log file (<code>multitenancymanager_trace.00000<n>.glf</code>) to perform further analysis.
The template connection <code><name of template connection></code> specified by option <code><parameter name></code> does not correspond to an imported tenant connection.	A specified template connection does not have a corresponding tenant connection imported. This could be the result of an incorrectly specified template connection, a missing connection object import, or incorrect configuration of the connection import option.	Check the template connection configuration of the specified option in the error message to make sure the template connection CUID is correct. Also check that the <code>optionIncludeConnections</code> import option is set to true and the specified template connection object is properly included by the specification of <code>templateConnectionFolder</code> option.

7.1.1 Multitenancy Error Messages

This section provides a list of common errors displayed by the Multitenancy Management Tool and recorded in the .csv log file.

Error code range	Category
MTM 00001 - MTM 00024	Properties configuration file errors
MTM 00101 - MTM 10104	Error reading data
MTM 01001 - MTM 01009	Database configuration errors
MTM 02000 - MTM 02002	User Group and Permissions errors
Multitenancy Management Tool	Warning messages

7.1.1.1 MTM 00001 - MTM 00013

(MTM 00001) Invalid application configuration file. Error: {0}

Cause

There were errors with the specification of some configuration options.

Action

Check the output or log file to see which option is incorrect and fix it according to the corresponding comment in the installed sample tenant template definition file.

(MTM 00002) Reserved

Cause (reserved for MTM 00002)

Cause placeholder text.

Action

Action placeholder text.

(MTM 10003) An unexpected error has occurred

Cause

An unexpected situation has occurred.

Action

Check the log file for more detailed error message.

(MTM 00004) Reserved

Cause

Cause placeholder text.

Action

Action placeholder text.

(MTM 00005) No matching objects under root folder "{1}" in "{2}", specified by configuration option "{3}", are found using query "{0}"

Cause

No template object can be retrieved based on the template paths specified.

Action

Verify that the template paths specified are correct using the configuration option. It is possible that no template is available under the specified root folder.

(MTM 00006) Failed to fetch object(s) with query {0}

Cause

An error occurred when fetching objects from the CMS or the Object Manager.

Action

Check log file for more detailed root cause. Check the query syntax, the CMS availability, and the object accessibility.

(MTM 00007) Failed to fetch template object(s) from "{1}", specified by configuration option "{2}", under root folder "{0}"

Cause

An error occurred when fetching template objects from the CMS or the Object Manager..

Action

Check log file for more detailed root cause. Check the query syntax, the CMS availability, and the object accessibility.

(MTM 00008) Failed to map template object(s) to tenant object(s)

Cause

An error occurred when mapping a template object to a tenant object. The error could be caused by IO operations during Object Manager related operation or BIAR file operations.

Action

Check the log for more details on the actual cause. Check if the template objects are properly defined in the tenant template definition file used. Also check if the user has sufficient access privilege to access IO resources.

(MTM 00009) Failed to perform backup of the previously imported tenant objects

Cause

An exception occurred while backing up existing tenant objects. The error could be caused by IO operations during Object Manager related operation or BIAR file operations.

Action

Check that you have enough disk space in user's home directory and the CMS can be accessed. Check log file for more details on the actual cause.

(MTM 00010) Failed to restore the backup tenant objects

Cause

An exception occurred while restoring the tenant objects to their original state.

Action

Check that the BIAR file for the backup tenant objects can be located in user's home directory and the CMS can be accessed. Check log file for more details.

(MTM 00011) Failed to refresh tenant object(s) or update shared object(s)

Cause

An exception occurred while refreshing some tenant objects with tenant specific data such as database connection info.

Action

Check log file for more detailed root cause relevant to different object types.

(MTM 00012) Failed to load document refresh plugin for kind {0}

Cause

An exception occurred while loading a document refresh plugin jar.

Action

Check that the document refresh plugin jar exists and is accessible. The refresh plugins are located under \$installdir\$\SAP BusinessObjects Enterprise XI 4.0\java\apps\multitenancyManager\jars\docRefreshPlugins

(MTM 00013) Reserved

Cause

Reserved.

Action

Reserved.

7.1.1.2 MTM 01001 - MTM 01009 Crystal Reports related errors

(MTM 01001) Missing template database configuration {0} or tenant database configuration {1}

Cause

A template to tenant database mapping is not configured correctly.

Action

Please check your configuration to see if any template to tenant database information is incorrect.

(MTM 01002) Invalid template database configuration, error parsing {0}

Cause

The template object referenced by the report object does not have a matching tenant object.

Action

Please check if the matching tenant object such as connection or universe object is missing from the importing.

(MTM 01003) Invalid tenant database configuration, error parsing {0}

Cause

The tenant database configuration does not have the right format.

Action

Please check the tenant database format described by the tenant template definition file.

(MTM 01004) Invalid database server type {0}

Cause

The specified database server type is not supported.

Action

Ensure that the database server types is supported. For a list of supported types please see our documentation.

(MTM 01005) Reserved

Cause

Reserved.

Action

Reserved.

(MTM 01006) Table prefixes mismatch for {0} and {1}, either one of them does not exist, or number of prefixes do not match

Cause

The number of template table prefixes doesn't match that of tenant table prefixes.

Action

Please check the table prefix configuration to fix the problem.

(MTM 01007) Database configuration is invalid, please see previous item(s) for details

Cause

A database configuration error was reported earlier.

Action

Please check the output or the log file for more detail on the earlier configuration error.

(MTM 01008) An unexpected exception occurred while refreshing a CR document

Cause

An unexpected exception occurred while refreshing a CR document.

Action

Please check log file for more detailed root cause.

(MTM 01009) The template connection {0} specified by option {1} does not correspond to an imported tenant connection

Cause

A specified template connection does not have a corresponding tenant connection imported. This could be the result of an incorrectly specified template connection, a missing connection object import, or incorrect configuration of the connection import option.

Action

Please check the template connection configuration of the specified option in the error message to make sure the template connection CUID is correct. Also check that the `optionIncludeConnections` import option is set to `true` and the specified template connection object is properly included by the specification of `templateConnectionFolder` option.

7.1.1.3 MTM 02000 - MTM 02002

(MTM 02000) The user '<username>' does not have sufficient permissions for tenant '<tenantname>'

Cause

A template to tenant mapping for SAP Crystal Reports 2011 direct-to-data connections is missing or not configured correctly.

Action

Open your configuration file (`tenant_template_def.properties`) and ensure that every `crystalreport.templateidb<n>` option has a corresponding `crystalreport.tenantidb<n>` option.

Check the trace log file (`multitenancymanager_trace.00000<n>.glf`) to perform further analysis.

(MTM 02001) Database configuration is invalid, refer to previous error logs for more details

Cause

There was error detected with the specification of the database configuration options related to CCIS Date Connection.

Action

Open your configuration file (`tenant_template_def.properties`) and ensure that every `crystalreport.templatedb<n>` option has a corresponding `crystalreport.tenantdb<n>` option.

Check the trace log file (`multitenancymanager_trace.00000<n>.glf`) to perform further analysis.

(MTM 02002) Principals are only allowed to join user groups belonging to the same tenant. Related principals are not in the same tenant:
`{type1} '{name1}' (id={id1}, tenant={tenant 1 name(tenant 1 id)})` and
`{type2} '{name2}' (id={id2}, tenant={tenant 2 name(tenant 2 id)})`

Cause

An attempt is being made to add a principal to a user group that belongs to another tenant.

Action

In CMC, review the users and groups

Example

Principals are only allowed to join user groups belonging to the same tenant.
 Related principals are not in the same tenant: usergroup 'Administrators' (id=33, tenant=shared(0)) and user 'George' (id=1234, tenant=TenantA(4567)).

7.1.1.4 FWB category MTM Errors

Duplicate object name in the same folder. (FWB 00026)

Cause

You have tried to create an object with a name that already exists.

Action

Give your object a unique name.

(FWB 00079) The category specified for the request is <tenantcategory1> but the user object indicates the category should be <tenantcategory2>. The request will not be processed.

Cause

<placeholder text>

Action

<placeholder text>

(FWB 00092) The user '<username>' does not have sufficient permissions for tenant '<tenantname>'.

Cause

The current user does not have the right to administer the tenant.

Action

In CMC, review Applications > Multi-tenancy Manager > User Security > [user name] > Assign Security, and add View rights.

(FWB 00093) Tenant with id '<tenantid>' could not be found.

Cause

A tenant with the given id does not exist in the system.

Action

Check to see that you are using the correct Tenant ID. Review tenant properties in CMC > Tenancy > Properties. Review the `tenantName` property in the `tenant_template_def.properties` located at `<BOE BIP installed path>\java\apps\multitenancyManager\jars\`. Re-run the tenant provisioning script with the correct value for the `tenantName` property.

(FWB 00094) Principals are only allowed to join user groups belonging to the same tenant. Related principals are not in the same tenant: %1 '%2'(id=%3, tenant=%4) and %5 '%6'(id=%7, tenant=%8)

Cause

Principals in different tenants are being associated.

Action

Make sure principals are in the same tenant beforehand.

7.1.1.5 MTM Tool warning messages

(MTM Tool 1) Retrieving objects of type Category, template path templateCategoryFolder is not specified

Cause

The MTM properties configuration file `optionIncludeCategories` is set to `true` and there is no valid path supplied for `templateCategoryFolder`.

Action

Set `optionIncludeCategories` to `false` or specify a valid path for `templateCategoryFolder`.

(MTM Tool 2) No matching shared objects for kind CCIS.DataConnection or CommonConnection

Cause

The MTM properties configuration file `optionUseSharedConnections` is specified, but the `sharedConnectionFolder` is not specified or the specification doesn't point to a valid connection object.

Action

set `optionUseSharedConnections` to `false` or specify a valid path for `sharedConnectionFolder`

(MTM Tool 3) There are errors/warnings encountered during validation of the plugin specific configuration for plugins CrystalReport.

Cause

One or more CR specific configuration (those starts with “crystalreport.” in the tenant definition file) is not valid.

Action

There should be an error logged or displayed for the invalid option. Correct it in accordance with the comments associated with these options in the tenant definition file.

7.2 To configure tracing for the Multitenancy Management Tool

1. Open the `BO_trace.ini` file.
 - The default location on Windows is `<INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\conf\`.
 - The default location on Unix is `<INSTALLDIR>/sap_bobj/enterprise_xi40/conf/`.
2. In the file, under `Trace Syntax and Setting`, uncomment the required lines.
3. Add an IF statement to specify tracing settings for the Multitenancy Management Tool.

For example:

```
if (process == "multitenancymanager")
{
    active = true;
    importance = xs;
    alert = true;
    severity = 'S';
    keep = false;
    size = 100 * 1000;
}
```

Tip:

The process must be specified as `multitenancymanager` in order to apply the tracing setting to the Multitenancy Management Tool.

The following table lists the parameters available for configuring tracing:

Parameter	Possible values	Description
active	false, true	Enables tracing for the current process or server if set to true. Default value is false.
importance	'<<', '<=', '==', '>=', '>>', xs, s, m, l, xl Note: importance = xs is the most verbose option available while importance = xl is the least.	Specifies the threshold for tracing messages. All messages beyond the threshold will be traced. Default value is m (medium).
alert	false, true	Specifies to automatically enable trace for severe system events. Default value is true.
severity	'S', 'W', 'E', 'A', 'F', success, warning, error, assert, fatal	Specifies the threshold severity over which messages can be traced. 'S' consumes the most disk space. Default value is 'E'.
size	Possible values are integers >= 1000	Specifies the number of messages in a trace log file before a new one is created. Default value is 100000.
keep	false, true	Specifies whether or not to keep the old log file after a new file is created. Default value is false.
administrator	Strings or integers	Specifies an annotation to use in the output log file. For example, if <pre>administrator = "hello"</pre> this string will be inserted into the log file.
log_dir		Specifies the output log file directory. By default log files are stored in the Logging folder.
always_close	on, off	Specifies if the log file should be closed after a trace is written to the log file. Default value is off.

4. Save and close the BO_trace.ini file.

The next time you run the Multitenancy Management Tool, a trace log file (multitenancymanager_trace.00000<n>.glf) will be created.

Instead of modifying the default `BO_trace.ini` file, you can create a copy of the file specifically for the Multitenancy Management Tool, and output the trace log file to a different location. For example, to use `C:\my_BO_trace.ini` for trace log settings and output the trace log file to `C:\myLogging`, edit the following logging options in the `multitenancymanagerSystem.properties` file:

```
<!-- logging -->
<entry key="mtm.systemVar.trace.logDir">C:\myLogging\</entry>
<entry key="mtm.systemVar.trace.iniDir">C:\</entry>
<entry key="mtm.systemVar.trace.iniFile">my_BO_trace.ini</entry>
```

Note:

This also changes the default output location of the `.csv` log file (`multitenancymanager<yyyy><mm><dd><time>.csv`).

For information about configuring additional tracing for BI platform servers and other applications, see “Managing and Configuring Logs” in the *Business Intelligence Platform Administrator Guide*.

More Information

Information Resource	Location
SAP BusinessObjects product information	http://www.sap.com
SAP Help Portal	<p>Navigate to http://help.sap.com/businessobjects and on the "SAP BusinessObjects Overview" side panel click All Products.</p> <p>You can access the most up-to-date documentation covering all SAP BusinessObjects products and their deployment at the SAP Help Portal. You can download PDF versions or installable HTML libraries.</p> <p>Certain guides are stored on the SAP Service Marketplace and are not available from the SAP Help Portal. These guides are listed on the Help Portal accompanied by a link to the SAP Service Marketplace. Customers with a maintenance agreement have an authorized user ID to access this site. To obtain an ID, contact your customer support representative.</p>
SAP Service Marketplace	<p>http://service.sap.com/bosap-support > Documentation</p> <ul style="list-style-type: none"> • Installation guides: https://service.sap.com/bosap-instguides • Release notes: http://service.sap.com/releasenotes <p>The SAP Service Marketplace stores certain installation guides, upgrade and migration guides, deployment guides, release notes and Supported Platforms documents. Customers with a maintenance agreement have an authorized user ID to access this site. Contact your customer support representative to obtain an ID. If you are redirected to the SAP Service Marketplace from the SAP Help Portal, use the menu in the navigation pane on the left to locate the category containing the documentation you want to access.</p>
Docupedia	<p>https://cw.sdn.sap.com/cw/community/docupedia</p> <p>Docupedia provides additional documentation resources, a collaborative authoring environment, and an interactive feedback channel.</p>
Developer resources	<p>https://boc.sdn.sap.com/</p> <p>https://www.sdn.sap.com/irj/sdn/businessobjects-sdklibrary</p>

Information Resource	Location
SAP BusinessObjects articles on the SAP Community Network	https://www.sdn.sap.com/irj/boc/businessobjects-articles These articles were formerly known as technical papers.
Notes	https://service.sap.com/notes These notes were formerly known as Knowledge Base articles.
Forums on the SAP Community Network	https://www.sdn.sap.com/irj/scn/forums
Training	http://www.sap.com/services/education From traditional classroom learning to targeted e-learning seminars, we can offer a training package to suit your learning needs and preferred learning style.
Online customer support	http://service.sap.com/bosap-support The SAP Support Portal contains information about Customer Support programs and services. It also has links to a wide range of technical information and downloads. Customers with a maintenance agreement have an authorized user ID to access this site. To obtain an ID, contact your customer support representative.
Consulting	http://www.sap.com/services/bysubject/businessobjectsconsulting Consultants can accompany you from the initial analysis stage to the delivery of your deployment project. Expertise is available in topics such as relational and multidimensional databases, connectivity, database design tools, and customized embedding technology.

Index

A

abortANDRollback 50, 54, 55
auth 42

C

CCIS.DataConnection database
 connection parameter 88
cms 40
Command line options 33
CrystalReport Database connection
 info for the tenant 83
CrystalReport for Enterprise Managed
 Direct to Database 86

E

error messages
 multitenancy 100

F

FWB 00026 113
FWB 00079 114
FWB 00092 114
FWB 00093 114
FWB 00094 115

M

MTM 00001 107
MTM 00002 107
MTM 00003 107
MTM 00004 107
MTM 00005 108
MTM 00006 108
MTM 00007 108
MTM 00008 108
MTM 00009 109
MTM 00010 109
MTM 00011 109
MTM 00012 109
MTM 00013 110
MTM 01001 110
MTM 01002 110
MTM 01003 110
MTM 01004 111
MTM 01005 111
MTM 01006 111

MTM 01007 111
MTM 01008 112
MTM 01009 112
MTM 02000 112
MTM 02001 113
MTM 02002 113
MTM Tool 00001 115
MTM Tool 00002 115
MTM Tool 00003 116
multitenancy
 (Mandatory) Logon information 13
 (Mandatory) Name of the tenant
 being added 13
 (Mandatory) Template token
 identifier used for tenant name
 replacing 13
 (Optional) Document folder
 template 13
 access rights for tenant, setting 12
 categories 19
 CMC tasks 93, 94, 95
 configuring properties 9
 configuring tenants 93
 Crystal Reports 2011 22
 Crystal Reports and 24
 deleting
 tenants 98
 error messages 100
 event folders 19
 folders 18
 groups 18
 installation prerequisites 9
 mapping tenant data source
 connections 22
 new folder for tenant, creating 11
 number of concurrent users 93
 passing options to a command line
 33
 profiles 20
 properties 94
 public folders 18
 quick start 9
 resource weight 93
 running the tool 9, 33
 security 20
 shared universes and connections
 30
 template token 17
 tenant
 user and group associations 97
 tenant definition 7

multitenancy (*continued*)
 tenant definition configuration file
 7
 tenant description
 command line option 94
 tenant keywords
 command line option 94
 tenant management 93
 tenant properties 93
 description 94
 keywords 94
 name 94
 number of concurrent users 95
 tenant template definition 7
 tenant template, creating 10
 tenant template, setting up 17
 tenant token definition 7
 tenant_template_def.properties 7,
 93
 tenant-definition configuration file
 93
 tenant's concurrent users
 command line option 95
 tenants
 adding user groups to 97
 description 93
 keywords 93
 name 93
 removing user groups from 97
 tenants, deleting 98
 tenants, viewing 15
 tracing for the tool 116
 troubleshooting 99
 universes and connections 26, 31
 unshared universes and
 connections 26
 user and group associations
 for tenants 97
 user group for tenant, creating 10
 user group rights, assigning 95
 user group rights, removing 96
 user group template 18
 user groups 18
 adding to tenants 97
 removing from tenants 97
 viewing tenants 15
Multitenancy Management Tool 9
 how to run 14
 location of 14
 location of files 33
 running 33

O

optionImportSecMode 52
 optionIncludeAccessLevels 63
 optionIncludeCategories 57
 optionIncludeProfiles 59, 61
 optionUseSharedUniverses 65, 67

P

pwd 45

R

reviewBeforeProceed 46

S

sharedConnectionFolder 81
 sharedUniverseFolder 79
 statusLog 49

T

template token
 definition of 17
 templateCategoryFolder 75
 templateConnectionFolder 73
 templateContentFolder 69, 71
 templateEventFolder 77
 tenant access rights
 assigning 95
 removing 96
 tenant configuration file
 (Mandatory) Logon information 13
 (Mandatory) Name of the tenant
 being added 13
 (Mandatory) Template token
 identifier used for tenant name
 replacing 13
 (Optional) Document folder
 template 13
 tenant properties
 description 93, 94
 keywords 93
 name 93, 94
 number of concurrent users 93
 resource weight 93

tenant template 17
 tenant_template_def.properties 7, 89, 93
 (Mandatory) Logon information 13
 (Mandatory) Name of the tenant
 being added 13
 (Mandatory) Template token
 identifier used for tenant name
 replacing 13
 (Optional) Document folder
 template 13
 tenant-definition configuration file 93
 tenantConcurrentUserLimit 47
 tenantName 38, 39
 tenants
 adding user groups to 97
 configuring in CMC 93
 deleting 98
 removing user groups from 97
 user and group associations 97
 viewing in CMC 15, 93

U

user 43