How to…

Build a Collaboration Room Template

ENTERPRISE PORTAL 6.0

PUBLIC

ASAP "How to…" Paper

Applicable Releases: EP 6.0

September 2003

1 Scenario

One of the core ideas behind Collaboration Rooms is to provide an easy creation process for new Rooms. To enable this, most mandatory technical information must be predefined. This is done in the Collaboration Room templates. These templates describe the default behavior and settings of the Rooms based on them. You can therefore define specific templates for Collaboration Rooms for all relevant business case scenarios in your company.

2 Result

This guide describes the creation of a new Collaboration Room template. These templates can be used to create new Collaboration Rooms.

3 The Step By Step Solution

3.1 Background Information

Every Collaboration Room is based on a Room template. A Room template is a blueprint for a Collaboration Room with room-specific views of the room content. The template combines iViews and pages in a task-specific blueprint, from which the user can generate Collaboration Rooms for corresponding tasks.

Room templates predefine a lot of technical information to make the creation process of the Room as easy as possible. In most cases, each Room template reflects a specific business scenario (for example, Meeting template, Project template). In general, a Room template consists of three layers:

- The first layer is mainly based on a common portal workset. This workset defines the visual layout structure and iView content that is to be part of the template. This information is stored in the Portal Content Directory (PCD). It can be accessed from the Portal Content Catalog (PCC).
- All information about the attributes of the template (for example, name, description, mappings, and room roles) is stored inside of the KM layer. It can be accessed from the *Room Template Administration* iView.
- Additionally, all content with which a Room instance is to be populated by default must be stored in the KM Repository, and is linked to the template. It can be accessed from the KM Content Navigator.

3.2 Summary

The following steps are involved in creating a new Collaboration Room template:

Prerequisites:

1. Check/obtain necessary portal roles in your portal

Steps in the Portal Content Catalog:

2. Create a new workset in the Portal Content Catalog
3. Create iViews and pages, or reuse existing iViews and pages

Steps in the Template Manager

4. Template List and Template creation
5. Define general attributes
6. Define Room roles and access permissions to the Room's pages
7. Define and create content areas to be used
8. Define mapping parameters to the iViews used
9. Test the Room template

3.3 Prerequisites

You need the following portal roles to access the tools necessary to fulfill all steps of the template creation process:

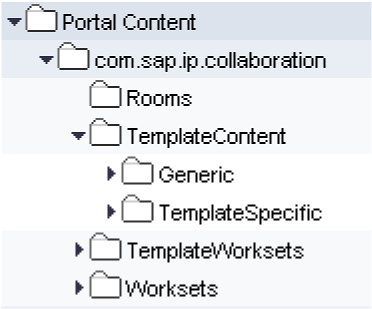
- ***com.sap.portal.content_admin_role (Content Administration)***
  This is a mandatory role that allows access to the Portal Content Catalog (PCC). You can create the necessary portal objects (worksets, pages, and iViews) there.
- ***com.sap.netweaver.coll.CollaborationAdministration (Collaboration Administration)***
  This is a mandatory role that allows access to the Room Template Manager. This tool allows you to create and maintain Room Templates.
- ***com.sap.netweaver.coll.CollaborationRoomCreation (Room Creation)***
  This role is strongly recommended. It allows you to create Rooms based on templates. This is necessary to test the Room templates that you create.
- ***com.sap.netweaver.coll.Collaboration (Collaboration)***
  This role is strongly recommended. It allows you to access the Room directory in order to enter a Room that has just been created.
- ***com.sap.km.ContentManager (Content Administration KM Content)***
  This role is recommended. If you want to prepopulate a Room with KM content (for example, documents), you need this portal role.

3.4 Steps in the Portal Content Catalog

3.4.1 Structure in the Portal Content Calatog

Each Room Template contains a workset including relevant pages and iViews. These elements are stored in the Portal Content Directory (PCD). They can be accessed from the Portal Content Catalog (PCC). To access the PCC, the user must be assigned to the relevant portal role (see 3.3). The PCC can be accessed under *Content Administration → Portal Content*.

The elements relevant for the template are located in the following structure:



You have to store all the worksets that are to be used to build a template in the `Worksets` folder. The `TemplateWorksets` folder contains worksets that are related to a template. Therefore you first place a new workset in the `Worksets` folder, and then create a template based on it. During this template creation process, the Template Manager copies the relevant workset into the `TemplateWorksets` folder.

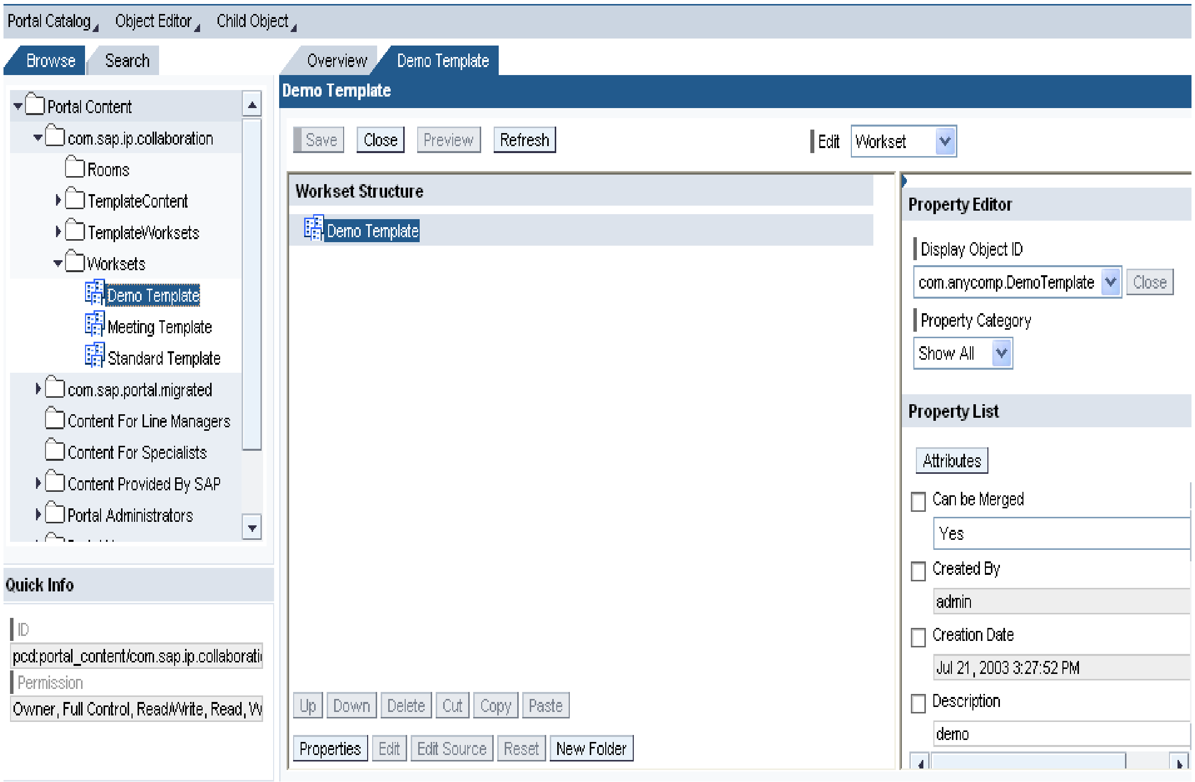
Portal content must be used to build the worksets. SAP recommends storing this template content in the `TemplateContent` folder. The `Generic` folder contains pages and iViews that can be reused in several templates. SAP provides a basic set of these pages and iViews. The `TemplateSpecific` folder contains pages and iViews that are specific to a particular template.

The `Rooms` folder contains the worksets that represent Room instances. This folder is not accessible by an administrator. Only the Room infrastructure writes data into this folder. Changes can only be made through the Room Maintenance which should be included in every room template.

3.4.2 Create a New Workset for the Template

The first step is to create a new workset that represents the Room template in the PCC. Store it in the folder *Portal Content* → *com.sap.ip.collaboration* → *Worksets*. SAP recommends using a namespace for each new workset. SAP itself always uses the namespace `com.sap.netweaver.coll`.

In the following example, we have created a new workset with the name “Demo Template” and the ID “com.anycomp.DemoTemplate”:



ID: pcd:portal_content/com.sap.ip.collaborati

Permission: Owner, Full Control, Read/Write, Read, W

Display Object ID: com.anycomp.DemoTemplate

Property Category: Show All

☐ Can be Merged: Yes

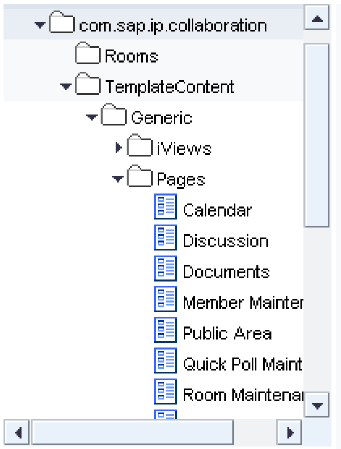
☐ Created By: admin

☐ Creation Date: Jul 21, 2003 3:27:52 PM

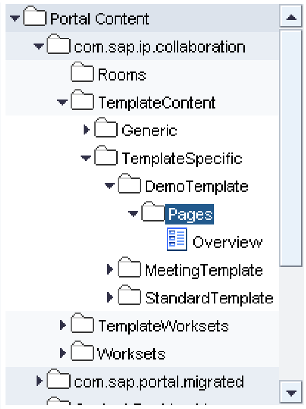
☐ Description: demo

3.4.3 Create or Reuse Existing Pages and iViews

The next step is to define the content (pages and iViews) for the new workset. As mentioned above, some content is already provided by SAP in the folder *Portal Content* → *com.sap.ip.collaboration* → *TemplateContent* → *Generic.* This is depicted in the graphic below.



In addition to these elements, most templates require additional pages and iViews. These elements should be stored in the folder *Portal Content* → *com.sap.ip.collaboration* → *TemplateContent* → *TemplateSpecific* → *MYTEMPLATENAME*. In our example, we have created one additional overview page:

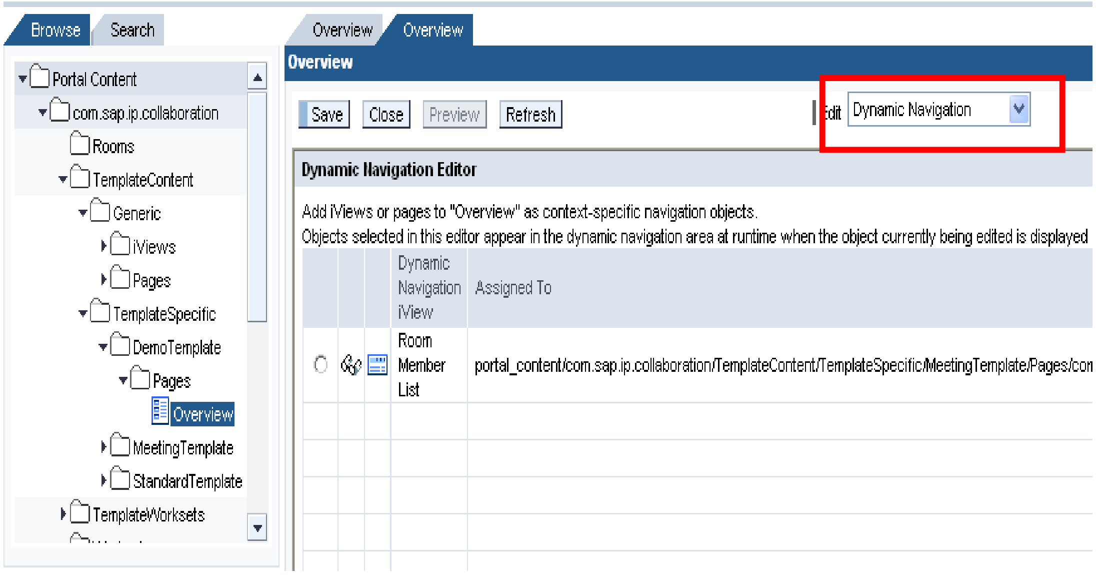


It is also possible to integrate content from anywhere else in the PCC into this workset. Only the workset has to be stored in the correct folder, not the integrated content. Nevertheless, the above order allows easier maintenance of the templates, so SAP recommends using it.

We provide two special iViews (RoomInformation and RoomInformationSpacer) for the purpose of displaying a header above each page with the Room's name. For details, see the topic *Room Information* in the documenation. Check their usage in SAP's example templates.

We recommend using an appropriate namespace for the ID of the portal object for all created worksets, pages and iViews. Regarding the isolation method and height type for iViews, refer to the corresponding documentation of the iViews.

You can place special iViews in the dynamic navigation area of a page. As a result, these iViews appear directly below the detailed level navigation (DTN) area of the portal. SAP provides a template with the *Room Member List* iView integrated using dynamic navigation. The graphic below shows what a page with an iView in dynamic navigation looks like:



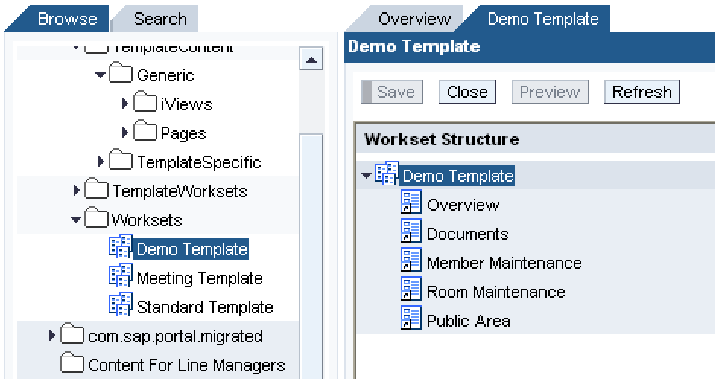
Dynamic Navigation Editor

Add iViews or pages to "Overview" as context-specific navigation objects.
Objects selected in this editor appear in the dynamic navigation area at runtime when the object currently being edited is displayed

			Dynamic Navigation iView	Assigned To
○			Room Member List	portal_content/com.sap.ip.collaboration/TemplateContent/TemplateSpecific/MeetingTemplate/Pages/con

Similar to dynamic navigation, you can enhance the pages with related links. These related links appear as hyperlinks directly below DTN. Again SAP places the *Room Member List* in several templates in the related links area.

In our example, the resulting workset has the following structure:



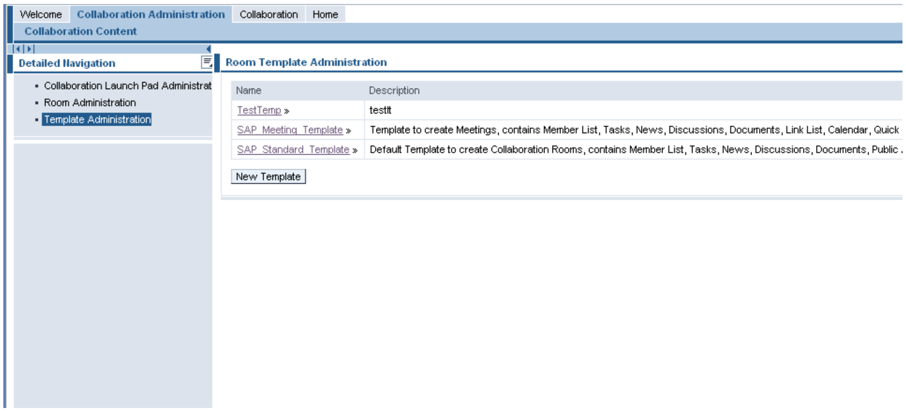
After this step, you can continue defining the Room template using *Room Template Administration*.

3.5 Steps in Room Template Administration

3.5.1 Room Template Administration and Template Creation

After completing the workset that is the basis of each Room template, you define the technical attributes and properties of the new Room template. You can build several Room templates based on the same workset.

First, enter the Room Template Administration. This tool supports the easy creation and maintenance of Room templates. You must be assigned to the appropriate portal role in order to access *Room Template Administration* (see 3.3). Room Template Administration can be found using the portal navigation area: *Collaboration Administration* → *Collaboration Content* → *Template Administration*. *Room Template Administration* initially provides a list of all available templates:



Name	Description
TestTemp »	testit
SAP_Meeting_Template »	Template to create Meetings, contains Member List, Tasks, News, Discussions, Documents, Link List, Calendar, Quick
SAP_Standard_Template »	Default Template to create Collaboration Rooms, contains Member List, Tasks, News, Discussions, Documents, Public .

You can edit or delete a listed Room template using its context menu.

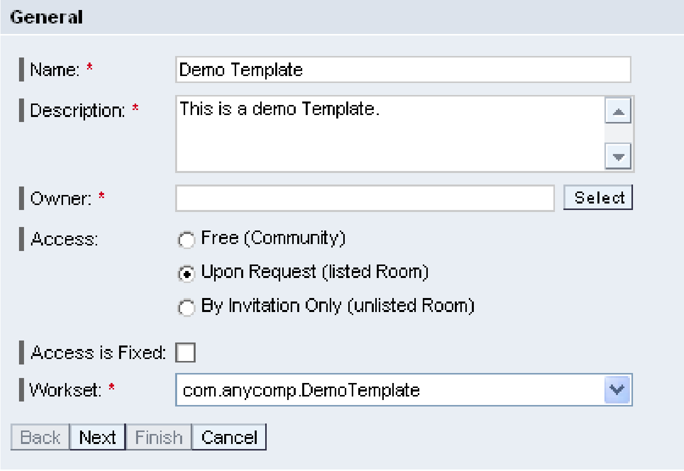
To create a new Room Template, choose *New Template*. A Wizard guides you through the necessary steps. These steps are described below.

3.5.2 Define General Attributes

The first wizard screen requests basic template information. The following information is requested:

- Name *(mandatory)*
  The name of the new template. The following characters are permitted:
  'a-z', 'A-Z', '0-9', '_', '%', '~', '-', '(', ')', 'äöüÄÖÜß', SPACE
- Description *(mandatory)*
  Description of the template.
- Owner
  The owner of the template. This field is optional. If you use it, keep in mind that the ID of the user is saved.
- Access *(mandatory)*
  Describes how the Room can be accessed and its visibility after creation:
  - Free
    The Room has no access control. This means any portal user can enter it. This is especially useful for open communities. The Room is listed on the *Communities* tab in the Room Directory.
  - Upon Request
    The Room can only be entered by members. The Room is listed on the *My Rooms* tab for its members, and in the *All Rooms* tab for all portal users (listed Room). Any user can request membership of it.
  - By Invitation Only
    The Room can only be entered by members. It is not listed on the *All Rooms* tab but is listed on the *My Rooms* tab for its members (unlisted Room).
- Access is Fixed
  If this checkbox is marked, the creator of a Room cannot change the access mode. Otherwise, the creator can change the access mode for each Room instance.
- Workset *(mandatory)*
  All worksets in the PCC folder mentioned above are listed here. The Room Template uses the selected workset as a basis.

In our example, we used the following values:



General

Name: * Demo Template

Description: * This is a demo Template.

Owner: *

Access:
○ Free (Community)
● Upon Request (listed Room)
○ By Invitation Only (unlisted Room)

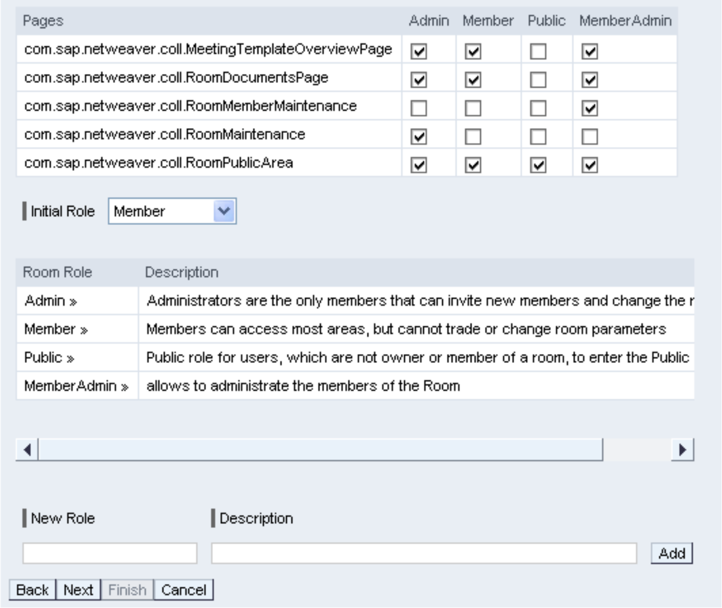
Access is Fixed: ☐

Workset: * com.anycomp.DemoTemplate

3.5.3 Define Room Roles and Access Permissions to the Room's Pages

It is now necessary to define which Room roles should be available to the Rooms based on the template. The Room role is a mechanism for defining viewing permissions (on a page level) to the portal content of the Room. These Room roles must be assigned to the members of the Room later on.

The graphic below shows the setting for our example:



Pages	Admin	Member	Public	MemberAdmin
com.sap.netweaver.coll.MeetingTemplateOverviewPage	☑	☑	☐	☑
com.sap.netweaver.coll.RoomDocumentsPage	☑	☑	☐	☑
com.sap.netweaver.coll.RoomMemberMaintenance	☐	☐	☐	☑
com.sap.netweaver.coll.RoomMaintenance	☑	☐	☐	☐
com.sap.netweaver.coll.RoomPublicArea	☑	☑	☑	☑

Initial Role: Member

Room Role	Description
Admin »	Administrators are the only members that can invite new members and change the r
Member »	Members can access most areas, but cannot trade or change room parameters
Public »	Public role for users, which are not owner or member of a room, to enter the Public
MemberAdmin »	allows to administrate the members of the Room

If you want to add a new Room role, you have to enter the name and description in the lower part of the screen in the field *New Role* and *Description*. Choose *Add*. If you want to delete a Room role, use the relevant context menu.

The default settings show three roles: "Admin", "Member", and "Public". By selecting the access permissions, you define the semantics of the roles ("Public" has a predefined meaning). You can delete these roles if they are not necessary for the template:

- Admin
  Access to all pages including the Room Administration
- Member
  Standard member of the Room
- Public
  The Public Role is a special Room role and can be used with the room type "Listed Rooms". It must not be assigned to a user at all. It controls which pages of the Room are to be available to every portal User (public area). In the standard scenario, this page contains an iView that lists the published documents of the Room. However, template designers can decide to make other or additional pages available to the public.

The initial role controls the default room access for an invited room member. The room owner can change the default room access in the Room Maintenance again.

Finally, it is necessary to define the access permissions of the Room roles to the pages of the workset. The pages of the selected workset are therefore automatically listed together with the available Room roles.

In our example (see figure above), we have added the Room role "MemberAdmin", whose task is to administrate the members of a Room. The initial role is "Member", which gives access to all pages except to the Room Administration page. The role "Admin" can access every page (except for Member Maintenance, which is a subset of Room Administration). The MemberAdmin role can access the same pages as the Member role plus Member Maintenance. All other portal users (→ public) can only access the Public Area.

3.5.4 Define and Create Content Areas

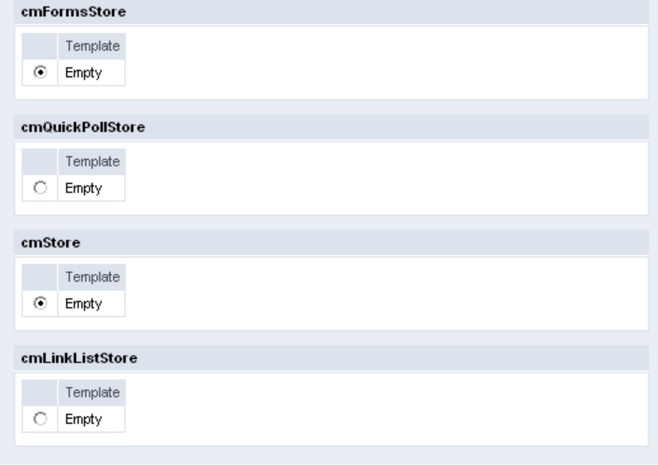
In the next screen, you can define which content stores are to be used in the Room. A content store is capable of being the persistency area for dedicated applications. The most common example is the "cmStore", which is needed for using documents in the Room.

The information about the stores available in the Room can later on be mapped to the iViews running. These iViews can then use the stores to persist their data.

By default, SAP provides four content stores:

- cmStore
  To be selected if the template is to support the *documents* application.
- cmFormsStore
  To be selected if the template is to support the *news* application.
- cmQuickPollStore
  To be selected if the template is to support the *quick poll* application.
- cmLinkListStore
  To be selected if the template is to support the *link list* application.

For simple scenarios, this is all you need. Select the *empty* radio button for each store that you want to use in the template. The stores that you need depend on the iViews used in the workset. If, for example, you want to support documents in the Room, select the "cmStore". In our example, we have selected iViews for documents and news items. We therefore include both content stores:



cmFormsStore

	Template
●	Empty

cmQuickPollStore

	Template
○	Empty

cmStore

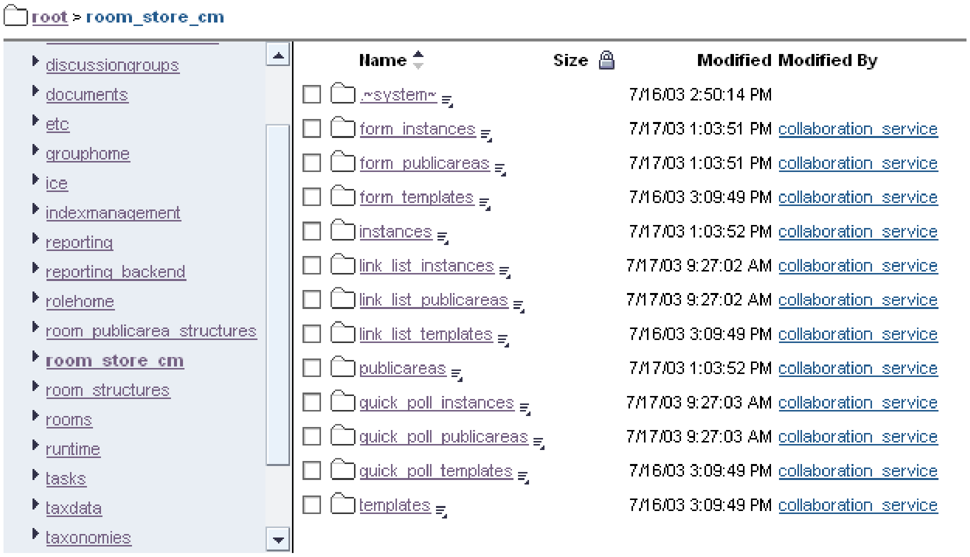
	Template
●	Empty

cmLinkListStore

	Template
○	Empty

3.5.4.1 Provide Predefined Content for Rooms

We can also provide predefined content for Rooms. In addition to an empty store, initial content is copied during Room creation. For example, you can define that each Room instance based on a specific template is to contain a help document and a specific folder structure. To use this, you need to understand the structure of the content store persistency. In our scenarios, all stores are located inside the Content Management system:



Name	Size	Modified	Modified By
☐ ~system~		7/16/03 2:50:14 PM	
☐ form_instances		7/17/03 1:03:51 PM	collaboration_service
☐ form_publicareas		7/17/03 1:03:51 PM	collaboration_service
☐ form_templates		7/16/03 3:09:49 PM	collaboration_service
☐ instances		7/17/03 1:03:52 PM	collaboration_service
☐ link_list_instances		7/17/03 9:27:02 AM	collaboration_service
☐ link_list_publicareas		7/17/03 9:27:02 AM	collaboration_service
☐ link_list_templates		7/16/03 3:09:49 PM	collaboration_service
☐ publicareas		7/17/03 1:03:52 PM	collaboration_service
☐ quick_poll_instances		7/17/03 9:27:03 AM	collaboration_service
☐ quick_poll_publicareas		7/17/03 9:27:03 AM	collaboration_service
☐ quick_poll_templates		7/16/03 3:09:49 PM	collaboration_service
☐ templates		7/16/03 3:09:49 PM	collaboration_service

To access the CM Navigator, the user must be assigned to the relevant portal role (see 3.3). Each store is based on three folders:

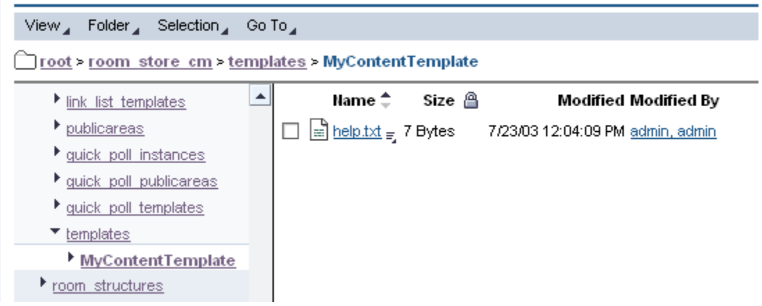
- Template folder (for example, form_templates)
  This folder contains content structures that can be used to prepopulate a Room instance. This is described below.

- Instance folder (for example, form_instances)
  If a store was chosen for a Room template, the system creates a persistency area for each Room. Therefore, this folder contains the different persistency areas for all Room instances that use this store.

- Public folder (for example, form_pulicareas)
  If, for example, a document is published in a Room, it is copied to this public area. The security permissions allow read-access to all portal users. You can therefore find a subset of the information stored in the related instances folder here. There is a 1:1 relationship between the Room instance specific area in the instances folder and the public folder.

If you want to add initial content to a Room, you have to create a new subfolder in the relevant content store. The name of your folder is the name of the content store specific content template. This content template can later on be selected in *Room Template Administration*.

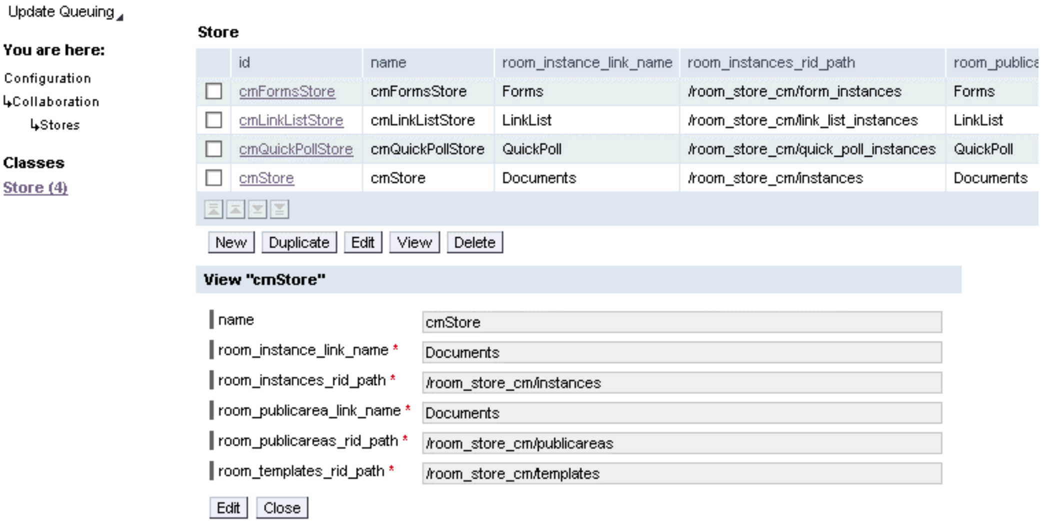
This graphic shows that we added a content template for the document-specific “cmStore”. The content template is named “myContentTemplate” and contains one document:



Name	Size	Modified	Modified By
☐ help.txt	7 Bytes	7/23/03 12:04:09 PM	admin, admin

3.5.4.2 Create and Modify Content Stores

Content stores are handled through configuration. A link for configuring content stores is located in the portal at: *Collaboration Administration → Collaboration Content → Related Links → Configure Room Content Stores*.



Store

	id	name	room_instance_link_name	room_instances_rid_path	room_publica
☐	cmFormsStore	cmFormsStore	Forms	/room_store_cm/form_instances	Forms
☐	cmLinkListStore	cmLinkListStore	LinkList	/room_store_cm/link_list_instances	LinkList
☐	cmQuickPollStore	cmQuickPollStore	QuickPoll	/room_store_cm/quick_poll_instances	QuickPoll
☐	cmStore	cmStore	Documents	/room_store_cm/instances	Documents

View "cmStore"

name	cmStore
room_instance_link_name *	Documents
room_instances_rid_path *	/room_store_cm/instances
room_publicarea_link_name *	Documents
room_publicareas_rid_path *	/room_store_cm/publicareas
room_templates_rid_path *	/room_store_cm/templates

To configure a store, you have to insert the RID (Resource ID) paths of the template, instance, and public folder. Additionally, two link names are required, under which the store is to be displayed in the Room.

The availability of a repository manager is a prerequisite for configuring a store. In our example, we reused the preconfigured `room_store_cm` repository manager. However, an application can define and use its own repository manager.

When a store has been created or modified, the changes take effect for new templates and for Rooms created with these new templates only. The new store cannot be added to old rooms.

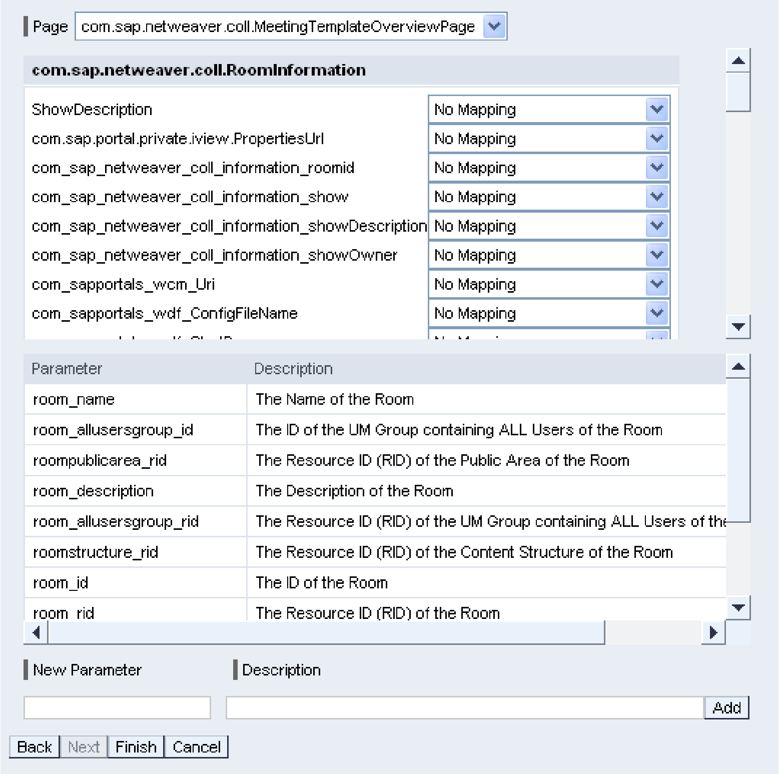
3.5.5 Define Mapping Parameters to the Corresponding iViews

The iViews that are part of the selected workset partially need additional information to run. For example, some iViews need a persistency, which we have already defined (see 3.4.2 and 3.4.3). Other iViews need the ID of the Room in order to obtain context-specific information. It is therefore possible to provide Room parameters for the iViews. Those parameters must be mapped to the corresponding iView properties. In general, we have three different kinds of Room parameters:

- Automatically generated parameters
  These parameters are automatically generated by the Room infrastructure. Examples are the Room ID or the ID of the corresponding user groups in user management
- Parameters generated by the selection of content stores
  For each selected content store, two parameters are generated automatically. One represents two RIDs for each Room: the private RID to the private area and the public RID to the public area. The private RID must be used for iViews in the private area within a Room, the public RID must be used for iViews located on the public pages within the Room.

– Parameters defined by the template designer
  These parameters can be defined and are requested from the Room creator in the creation wizard. A good example is the cost center, which can be related to a Room. This cost center ID can be requested from the Room creator and can be mapped to a BW iView showing reports about the cost center in question.

The graphic below shows the wizard screen:



Page: com.sap.netweaver.coll.MeetingTemplateOverviewPage

com.sap.netweaver.coll.RoomInformation

ShowDescription	No Mapping
com.sap.portal.private.iview.PropertiesUrl	No Mapping
com_sap_netweaver_coll_information_roomid	No Mapping
com_sap_netweaver_coll_information_show	No Mapping
com_sap_netweaver_coll_information_showDescription	No Mapping
com_sap_netweaver_coll_information_showOwner	No Mapping
com_sapportals_wcm_Uri	No Mapping
com_sapportals_wdf_ConfigFileName	No Mapping

Parameter	Description
room_name	The Name of the Room
room_allusersgroup_id	The ID of the UM Group containing ALL Users of the Room
roompublicarea_rid	The Resource ID (RID) of the Public Area of the Room
room_description	The Description of the Room
room_allusersgroup_rid	The Resource ID (RID) of the UM Group containing ALL Users of th
roomstructure_rid	The Resource ID (RID) of the Content Structure of the Room
room_id	The ID of the Room
room_rid	The Resource ID (RID) of the Room

If you want to have a self-defined parameter, you have to enter it with a description in the lower part of the screen and choose the *Add*. It then appears in the table in the center of the screen. This table contains all available parameters. It is also possible to delete self-defined parameters from their context menu here.

The following automatically generated parameters are available:

room_name	The name of the Room
room_allusersgroup_id	The ID of the UM group containing ALL users of the Room
roompublicarea_rid	The resource ID (RID) of the public area of the Room
room_description	The description of the Room
room_allusersgroup_rid	The resource ID (RID) of the UM group containing ALL users of the Room
roomstructure_rid	The resource ID (RID) of the content structure of the Room
room_id	The ID of the Room
room_rid	The resource ID (RID) of the Room
private_store_cmFormsStore	The private folder in the store cmFormsStore
public_store_cmFormsStore	The public folder in the store cmFormsStore
private_store_cmStore	The private folder in the store cmStore
public_store_cmStore	The public folder in the store cmStore
private_store_cmQuickPollStore	The private folder in the store cmQuickPollStore
public_store_cmQuickPollStore	The public folder in the store cmQuickPollStore
private_store_cmLinkListStore	The private folder in the store cmLinkListStore
public_store_cmLinkListStore	The public folder in the store cmLinkListStore

The *Page* drop-down box shows all pages available in the workset. Below you can see the iViews of the currently selected page with all the relevant iView properties. In the drop-down box next to each iView property, all Room parameters are listed. To define mappings, simply select the correct value in the drop down box.

A list with the most common mappings can be found in the appendix.

3.5.6 Test the Room Template

We recommend that you create a new Room in order to test the new Room template. You can enter this by default using the navigation path *Collaboration → Room Creation*. To access the creation wizard, the user must be assigned to the relevant portal role (see 3.3). Select the new template in the creation wizard and do not forget to assign yourself to the Admin Room role so that you can access *Room Maintenance*.

As soon as the Room has been created, you can access it from the creation wizard (directly after the creation process with the link *Enter Room*) and from the *Room Directory*. To access the creation wizard, the user must be assigned to the relevant portal role (see 3.3). If you want to delete the Room, you use the *Room Administration*. You can reach this iView using the navigation path *Collaboration Administration → Collaboration Content → Room Administration*. To access *Room Administration,* the user must be assigned to the relevant portal role (see 3.3).

4 Appendix

4.1 iView Mapping Rules

iView - Technical Name	iView - Property	Mapping Parameter
com.sap.netweaver.coll.RoomInformationSpacer	com_sap_netweaver_coll_information_roomid	room_id
com.sap.netweaver.coll.RoomNews	StartUri	private_store_cmFormsStore
	path	private_store_cmFormsStore
com.sap.netweaver.coll.RoomInformation	com_sap_netweaver_coll_information_roomid	room_id
com.sap.netweaver.coll.RoomTasks	RelatedId	roomstructure_rid
	path	roomstructure_rid
com.sap.netweaver.coll.RoomMemberList	StartUri	room_allusersgroup_rid
	path	room_allusersgroup_rid
	roomrid	room_rid
com.sap.netweaver.coll.RoomDocuments	StartUri	private_store_cmStore
	path	private_store_cmStore
	roomId	room_id
com.sap.netweaver.coll.RoomDiscussion	path	roomstructure_rid
com.sap.netweaver.coll.RoomMaintenance	com_sap_netweaver_coll_maintenance_roomid	room_id
com.sap.netweaver.coll.MemberMaintenance	com_sap_netweaver_coll_maintenance_roomid	room_id
com.sap.netweaver.coll.RoomInformationExtended	com_sap_netweaver_coll_information_roomid	room_id
com.sap.netweaver.coll.RoomInformationExtendedSpacer	com_sap_netweaver_coll_information_roomid	room_id
com.sap.netweaver.coll.RoomPublicDocumentNavigator	StartUri	public_store_cmStore
	path	public_store_cmStore
com.sap.netweaver.coll.RoomCalendar	com_sap_netweaver_coll_gwui_roomrid	room_rid
com.sap.netweaver.coll.RoomLinkList	StartUri	private_store_cmLinkListStore
	path	private_store_cmLinkListStore
com.sap.netweaver.coll.RoomQuickPoll	Path	private_store_cmQuickPollStore
com.sap.netweaver.coll.RoomQuickPollMaintenance	Path	private_store_cmQuickPollStore
	PathtoDocuments	private_store_cmStore
	PathtoImages	private_store_cmStore

Copyright