2
CHAPTER

SAP NetWeaver Enterprise Portal Building Blocks

Before we look at the portal architecture from a hardware standpoint, we'll address it from a software components point of view. As discussed in the first chapter, a portal consists of a basic *portal platform*, a *Knowledge Management component*, and a *Collaboration platform*. While a basic portal platform is mandatory in any portal implementation, whether the Knowledge Management and Collaboration components are installed would depend upon the business requirements in the project.

In this chapter, we take a deeper look at the software architecture of Enterprise Portals. The Enterprise Portal platform is deployed on the SAP Web Application Server (AS) Java and serves as the front end of the SAP NetWeaver platform. The other components, such as Knowledge Management, Collaboration, and Guided Procedures, are nothing but applications deployed within the portal platform.

Portal Platform

The portal platform consists of the following components (as shown in Figure 2-1):

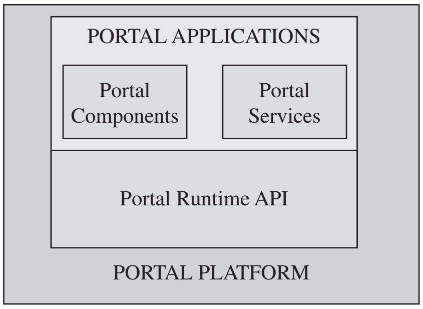
- Portal runtime
- Portal applications
  - Portal components
  - Portal services

A portal platform is basically a portal runtime that is composed of Java libraries. The two primary functions of the portal platform framework are to provide the necessary runtime environment so as to run iViews and to administer iViews.

Portal Runtime

The *portal runtime* is a virtual environment that provides the runtime and development environment in which the portal applications run. We can compare it with the Java runtime, which provides the runtime environment for Java applications. The portal runtime is made

Figure 2-1 Building blocks of a portal platform



Copyright by SAP AG

up of a Java API known as the *portal runtime API,* which is a collection of portal components and services. The portal components and services provide the basic core functionality to help the portal run efficiently. Portal components and services can be either provided by SAP or custom-built.

Info *A set of predefined portal components and services are loaded by the portal runtime at portal startup.*

Portal Components

Portal components are Java Server Pages (JSPs) or Java classes that produce HTML output, which is displayed on the client browser when a page is rendered. These core portal components include the page builder and administration tools. The portal components are responsible for creating the necessary content for display in the iViews.

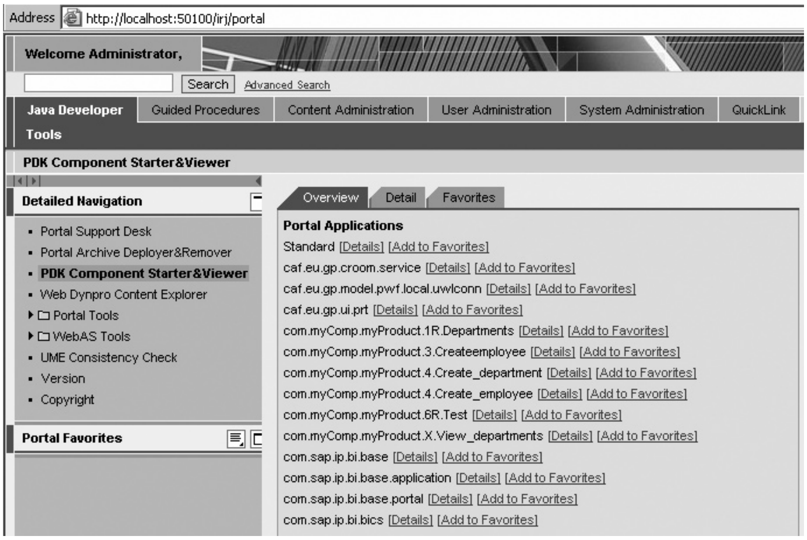
Tip *More than one portal component may be involved in generating the page, filling the iView with content, or defining the page layout.*

A portal component can do the following:

- Call other portal components, if needed
- Respond to other events
- Detect another component's profile and properties

A good example of a portal component is a *portal builder component*, which is responsible for the following:

- Receives HTTP requests from the client
- Forwards the incoming HTTP request to the relevant portal components and services
- Receives the response back from the portal server
- Builds the portal page
- Sends the page response back to the client



Portal Applications

Standard [Details] [Add to Favorites]
caf.eu.gp.croom.service [Details] [Add to Favorites]
caf.eu.gp.model.pwf.local.uwlconn [Details] [Add to Favorites]
caf.eu.gp.ui.prt [Details] [Add to Favorites]
com.myComp.myProduct.1R.Departments [Details] [Add to Favorites]
com.myComp.myProduct.3.Createemployee [Details] [Add to Favorites]
com.myComp.myProduct.4.Create_department [Details] [Add to Favorites]
com.myComp.myProduct.4.Create_employee [Details] [Add to Favorites]
com.myComp.myProduct.6R.Test [Details] [Add to Favorites]
com.myComp.myProduct.X.View_departments [Details] [Add to Favorites]
com.sap.ip.bi.base [Details] [Add to Favorites]
com.sap.ip.bi.base.application [Details] [Add to Favorites]
com.sap.ip.bi.base.portal [Details] [Add to Favorites]
com.sap.ip.bi.bics [Details] [Add to Favorites]

FIGURE 2-2 List of portal applications, components, and services

Copyright by SAP AG

Figure 2-2 displays the list of portal applications, components, and services deployed on the portal. This iView is available only if you have the Java Developer role.

Portal Services

Portal services are Java classes that provide functionality that can be accessed by other portal components and services. By functionality, we mean *data, procedures*, and *other resources*. A portal service is often called from the portal component to carry out some basic tasks such as searching for user-related information in the User Management Engine (UME) database.

TIP *Portal runtime (PRT) services are deployed on each server node and are loaded before other applications because they are necessary for the proper functioning of the portal.*

A portal service differs from a portal component in the following respects:

- While portal components have views, portal services do not have views.
- A portal service cannot be directly called by the client.
- Unlike a portal component, a portal service is not tied to a request.

Portal services can be classified as PRT services and external services. Examples of PRT services are caching services, notification services, iView services, application repository services, system landscape services, and services for portal content objects such as roles, pages, and worksets. External services extend the functionality of the PRT. Examples of external services are client eventing, the logger, the URL generator, HTMLBusiness for Java (HTMLB), and the Java Connector (JCO) client service.

Portal Content Directory

We will take a more detailed look at the *Portal Content Directory* (PCD) in a later chapter, but here is a little introduction. The PCD is the central storage mechanism in the portal used for storing content objects such as roles, worksets, pages, folders, and system landscapes. The top-level node is called *portal_content*. The PCD can interact with the UME, access control lists (ACLs), messaging notifications for invalidating caches, and transport mechanisms. The PCD can connect to a single database in a distributed portal scenario that involves several portal servers running on several machines.

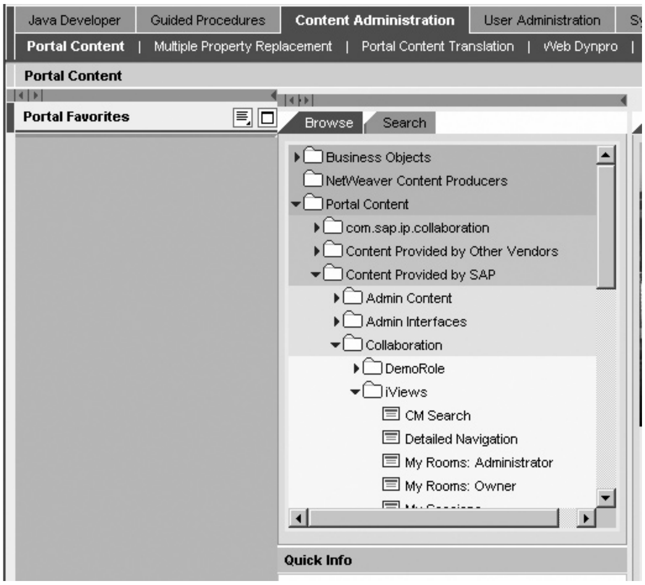
TIP *The nice thing about the PCD is that the portal content objects are stored in the form of a tree structure, which enables easy browsing when developing or administering content.*

Figure 2-3 displays the PCD structure, which illustrates the tree structure of the content objects.

Unification

The portal unification feature is based upon the *drag and relate* iView functionality. The unification feature facilitates *object-based navigation* between objects in the backend systems such as Customer Relationship Management (CRM), Business Information Warehouse (BW), and SAP R/3 as well as database systems such as Oracle, SQL, and so on. To implement the drag and relate functionality of the portal, you need to implement *unifiers*. The choice of the unifier will depend on the type of databases and SAP R/3 systems that are used for providing the functionality.

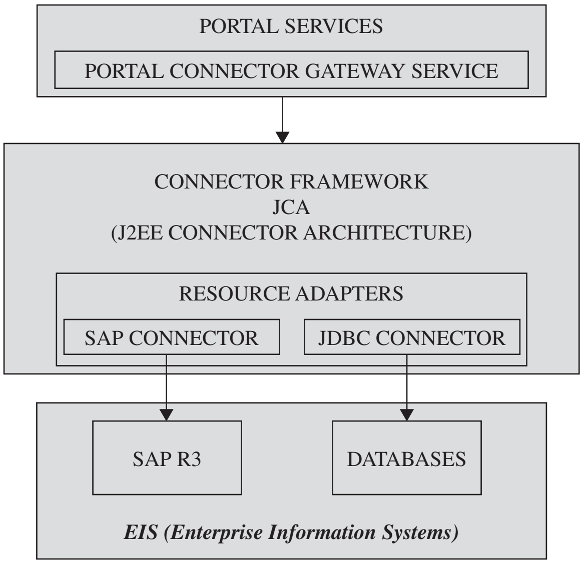
FIGURE 2-3 Portal Content Directory tree structure



Copyright by SAP AG

Connector Framework

The *Connector framework* of the portal resides on the Web Application Server (WAS) J2EE engine and provides a set of adapters called *connectors.* The connectors enable the portal to be connected to the backend *Enterprise Information Systems* (EIS). The Connector framework is based on the J2EE Connector architecture (JCA). Because the connectors have been created in line with the JCA specifications, any backend system-related issues such as the connectivity parameters and the protocol to be used are taken care of by the connector. The JCA specification is a framework provided by Sun for third-party vendors to follow when creating their adapters. As per JCA terminology, the connectors are referred to as *resource adapters* and the backend systems as *Enterprise Information Systems.* The various components of a connector framework are displayed in the following illustration.


PORTAL SERVICES
PORTAL CONNECTOR GATEWAY SERVICE
CONNECTOR FRAMEWORK
JCA
(J2EE CONNECTOR ARCHITECTURE)
RESOURCE ADAPTERS
SAP CONNECTOR
JDBC CONNECTOR
SAP R3
DATABASES
EIS (Enterprise Information Systems)

Copyright by SAP AG

The JCA API is available in the com.sapportals.connector.* package. For the connector functionality to work properly in the portal, the following JAR files are required:

- GenericConnector.jar (comes with Portal)
- Extended Connector.jar (part of the J2EE engine that contains additional functionality for GenericConnector.jar)
- Connector.jar (contains the JCA 1.0 API)
- JTA.jar
- JAAS.jar

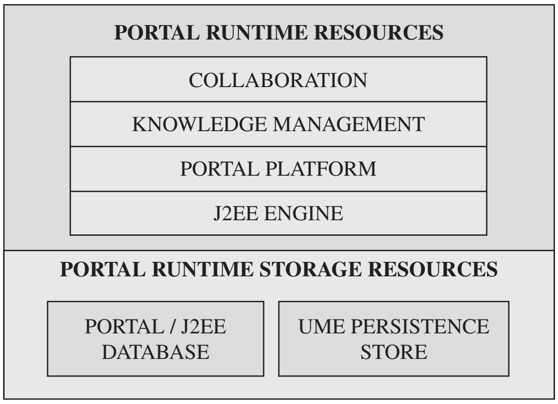
Assuming that the *system object* (which represents the backend system) has been defined in the portal system landscape, we can use the *Portal Connector Gateway Service* to call the JCA-compliant connectors for connecting to the backend system. The SAP NetWeaver Portal comes with standard connectors for connecting to a JDBC system, an SAP system, and a web service.

Tip *The standard connectors are automatically included by default in the portal during the initial installation.*

You will learn more about how to create system objects and connect to a backend system using the connectors in Chapter 18. iViews can be created that use these connectors to connect to a backend system using a wizard. The *SAP connector* can be used to connect to the SAP backend systems and the *JDBC connector* can be used to connect to JDBC databases.

Portal Runtime Storage Resources

So far you have seen the *portal runtime resources*, namely, the portal platform, the J2EE engine, the Knowledge Management (KM), and the Collaboration components. The portal runtime resources use the data that are stored in the *runtime storage resources* for rendering the portal framework, the pages, and the iViews during runtime. The following illustration displays the various portal runtime components and the portal runtime storage resources.



Copyright by SAP AG

The runtime storage resources are the *Portal/J2EE database* and the *UME persistence store.* The portal database stores the portal data such as the portal objects (iViews, roles, pages, worksets) and portal applications. The user persistence data store refers to the user-related data such as roles, users, and groups that could be stored in one or more repositories such as a database, an SAP system, or an LDAP system.

UME Architecture

UME stands for the *User Management Engine*, which, as the name implies, is responsible for managing objects such as users, user accounts, roles, and groups. The portal provides a UME service that can be used to call the appropriate UME API for executing the necessary activity such as create, change, or delete user management objects.

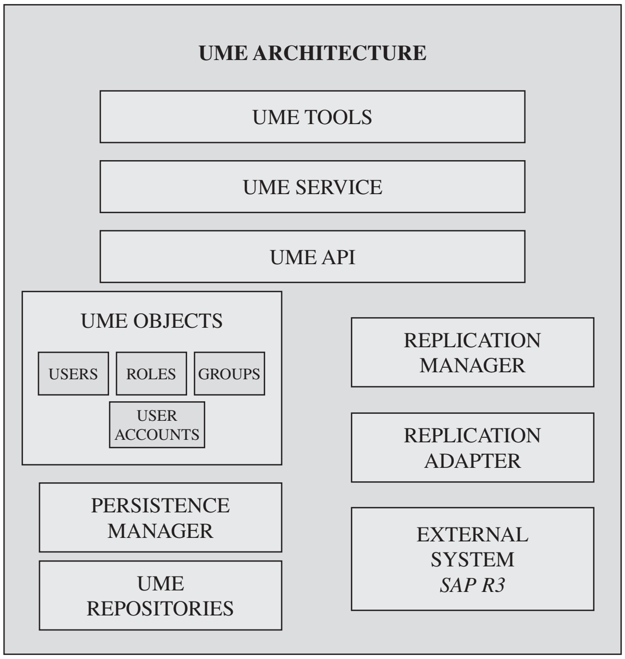
Tip *The advantage of the UME is that all of the different portal applications can share the same user management objects, thus reducing the maintenance effort.*

UME Components

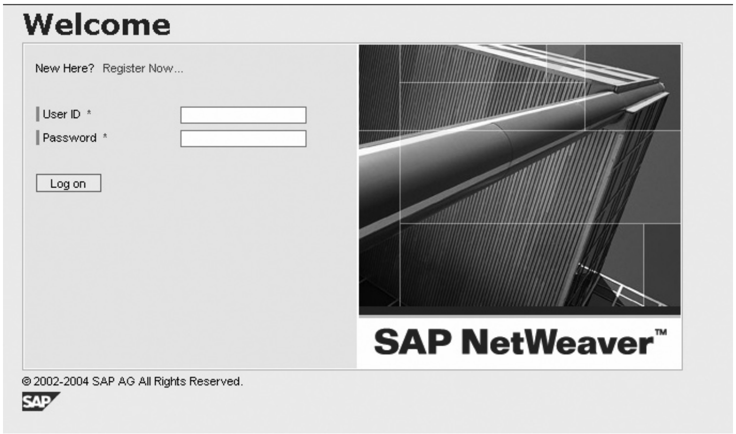
The components of the UME can be classified into the following broad categories:

- **UME tools** Consists of the user interface for creating, changing, or deleting user objects.
- **UME service** Responsible for calling the correct API when executing user administration–related activities.
- **UME API** Actual source code that is responsible for creating the user objects. The UME API calls the persistence manager.
- **Adapter** Also known as connector, which is used for connecting to the backend system.
- **Replication manager** Comes into play when we configure the J2EE engine to use the replication adapter for creating users in an external system such as SAP R/3 every time a user is created on the portal.
- **Persistence manager** Responsible for coordinating with the different UME repositories that are available for storing the UME data. Also responsible for taking care of all the connection details involved when connecting to a repository, thus absolving the programmer from having to know those details.
- **UME repository** Used for storing the user management objects. Can be an LDAP, an SAP R/3 system, or a database.

The following illustration displays the various components of the UME. For more information on UME, refer to Chapter 35.



Copyright by SAP AG



Welcome

New Here? Register Now...

User ID *

Password *

Log on

© 2002-2004 SAP AG All Rights Reserved.

FIGURE 2-4 Portal logon screen

Copyright by SAP AG

Portal User Interface

The *portal user interface* is composed of a number of parts, each of which has a specific purpose.

Logon Screen

Figure 2-4 displays a portal logon screen that appears when the portal is installed out of the box. Depending on the portal version that was implemented, the Register Now link may or may not exist. If the version of the portal that was implemented is 7.0, the Register Now link may be missing.

TIP *To activate the Register Now link, certain configuration steps must be taken in the J2EE engine. This is addressed in Chapter 25.*

After you enter the user ID and password and click Log On, the Welcome screen, shown in Figure 2-5, appears.

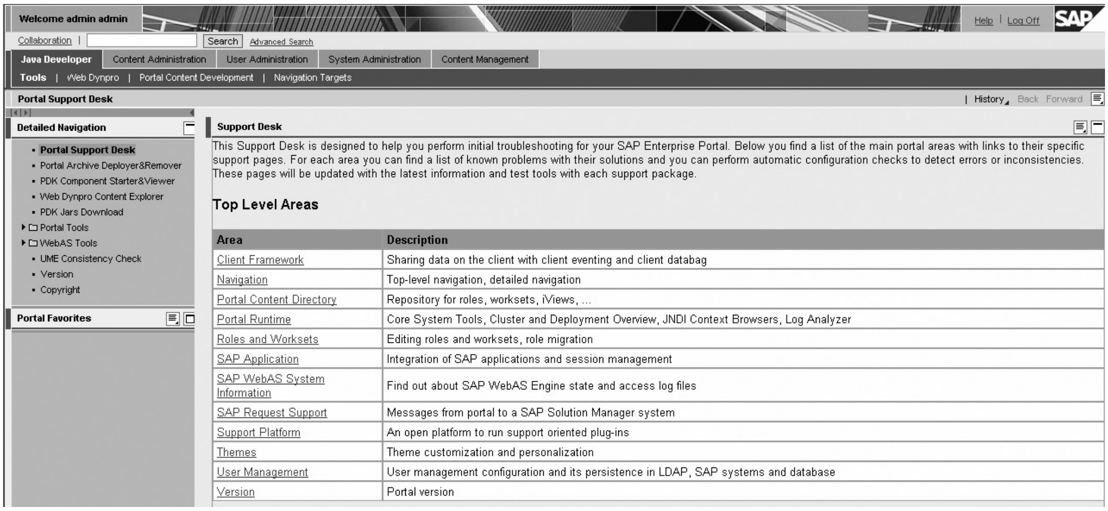
Top-Level and Detailed Navigational Menus

Depending on the roles that are assigned to the user, different first-level menus and second-level menus including detailed navigation menus will appear, as shown in Figure 2-6. The top-level menu includes both the first-level and the second-level menus. What appears on the top-level menus depends on the roles that have been assigned to this user, which in this case happens to be the Java Developer role, the Super Administrator role, and the Content Management role, as shown earlier in Figure 2-5.

Super Administrator Role and the Top-Level Menus

The *Super Administrator* role is a combination of three different roles:

- Content Administration
- User Administration
- System Administration



Welcome admin admin

Help | Log Off

Collaboration | Search Advanced Search

Java Developer | Content Administration | User Administration | System Administration | Content Management

Tools | Web Dynpro | Portal Content Development | Navigation Targets

Portal Support Desk

| History Back Forward

Detailed Navigation

- Portal Support Desk
- Portal Archive Deployer&Remover
- PDK Component Starter&Viewer
- Web Dynpro Content Explorer
- PDK Jars Download
- Portal Tools
- WebAS Tools
- UME Consistency Check
- Version
- Copyright

Portal Favorites

Support Desk

This Support Desk is designed to help you perform initial troubleshooting for your SAP Enterprise Portal. Below you find a list of the main portal areas with links to their specific support pages. For each area you can find a list of known problems with their solutions and you can perform automatic configuration checks to detect errors or inconsistencies. These pages will be updated with the latest information and test tools with each support package.

Top Level Areas

Area	Description
Client Framework	Sharing data on the client with client eventing and client databag
Navigation	Top-level navigation, detailed navigation
Portal Content Directory	Repository for roles, worksets, iViews, ...
Portal Runtime	Core System Tools, Cluster and Deployment Overview, JNDI Context Browsers, Log Analyzer
Roles and Worksets	Editing roles and worksets, role migration
SAP Application	Integration of SAP applications and session management
SAP WebAS System Information	Find out about SAP WebAS Engine state and access log files
SAP Request Support	Messages from portal to a SAP Solution Manager system
Support Platform	An open platform to run support oriented plug-ins
Themes	Theme customization and personalization
User Management	User management configuration and its persistence in LDAP, SAP systems and database
Version	Portal version

Copyright by SAP AG

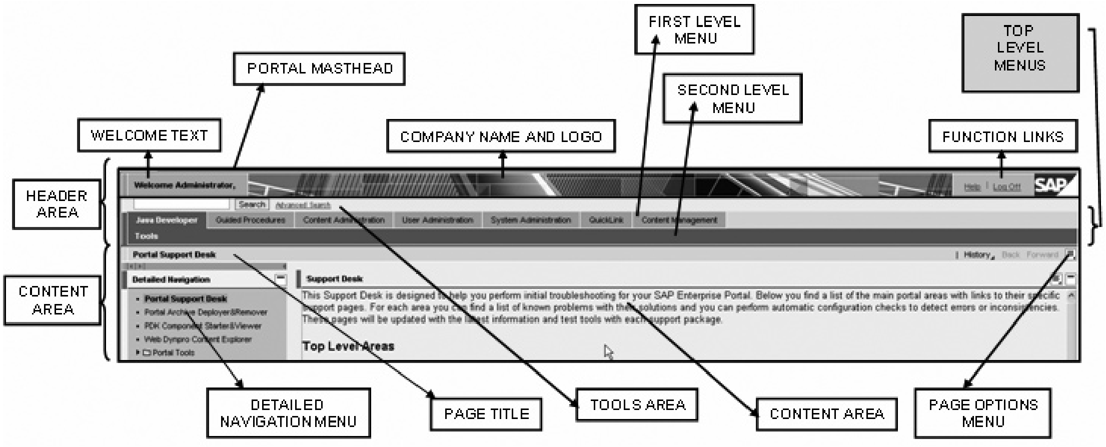
FIGURE 2-5 Welcome screen after successful logon

As a result of the Super Administrator role, three top-level menus appear, one each for the roles that make up the Super Administrator role.

Second-Level Menus

By assigning content to the roles, we can build the *navigational structure* of the portal. For example, by assigning content to the Java Developer role, the corresponding *second-level menus* and *detailed navigational menus* will be created.

In general, the portal user interface can be broadly divided into the following three areas: header area, navigation area, and content area.



Copyright by SAP AG

FIGURE 2-6 Parts of a portal desktop

The *header area* consists of the following:

- **Portal masthead** Contains welcome text for the user who has logged in; company name and logo; and function links such as the Help, Personalize, and Log Off links.
- **Tool area** Contains the search area and a link for opening the collaboration launch pad (if the necessary configuration for collaboration was carried out).
- **Top-level navigation** The leftmost, collapsible *navigation area* under the page title bar consists of the following:
  - Navigation iViews like the detailed navigation iView for third-level navigation
  - Dynamic navigation iViews for database access
  - Drag and relate iViews (if Unification functionality is implemented)
  - iViews with related links that open on new window (related links are predefined for the iView and page displayed in the content area)

The *content area* on the right consists of pages and iViews that provide information for the user. The *title bar* contains the title for the page that is being displayed. A History link, Back and Forward links, and a Page Options menu is available on the title bar. The Page Options menu provides links for opening the page in a new window, refreshing the page, personalizing the page, and adding the page to favorites.

Knowledge Management Architecture

Knowledge Management is very useful for organizations that store vast amounts of data among various types of media. It provides for controlled management of a document's life cycle and fosters collaboration between groups and communities.

Knowledge Management deals with making *unstructured information* available to the right audience. Unstructured information is content that is stored across various datasources such as text, web servers, file servers, intranet, mail servers, and database systems. It allows an end user to search and find information or knowledge very quickly, irrespective of where it is physically stored.

KM provides the following functions:

- *Integrates repositories* such as web servers, file servers, and notes database so that unstructured information is available at a central point in the portal.
- Ability to *navigate the folders* across all integrated repositories.
- Ability to access the documents in these repositories based on available *user permissions.*
- Ability to *search documents* in all integrated repositories.
- Ability to *create taxonomies* to classify documents.
- Ability to *store documents* based on classification. *Classification* could be based on content, organization, or other criteria; ensures that the information can be found efficiently.

- Ability to *create documents* based on user permissions.
- Ability to *publish documents* based on an *approval workflow.*
- *Provides Knowledge Management services* that allow internal support activities like the following:
    - Creating direct *feedback* on a document.
    - Creating *subscriptions* for notifying about any changes made to a resource.
    - Creating *reviews*, *ratings*, and *notes.*

Knowledge Management Components

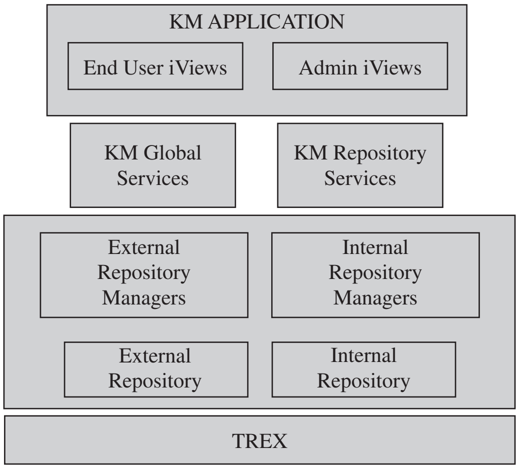
The Knowledge Management component consists of *content management and search functionality.* The Knowledge Management functionality is provided by the following components (see Figure 2-7) through the portal:

- **KM applications** Such as Navigation and Search applications
- **KM global services** Such as index management and audit service
- **KM repository services** Such as subscription and publishing services
- **KM Repository framework** Such as repository managers for file system, HTTP, and WebDAV
- **TREX** The search engine

KM Application

The KM application consists of *end user* and *administrator iViews* for carrying activities such as navigating through folders and documents, searching for documents, and carrying out file operations such as copying, deleting, and so on.

FIGURE 2-7 Knowledge Management components



Copyright by SAP AG

The following iViews are available for end users:

- KM Classification
- KM Content Exchange subscriber
- KM Documents
- KM Navigation
- KM Quick poll
- KM Search
- KM Subscriptions

The following administrator iViews are available:

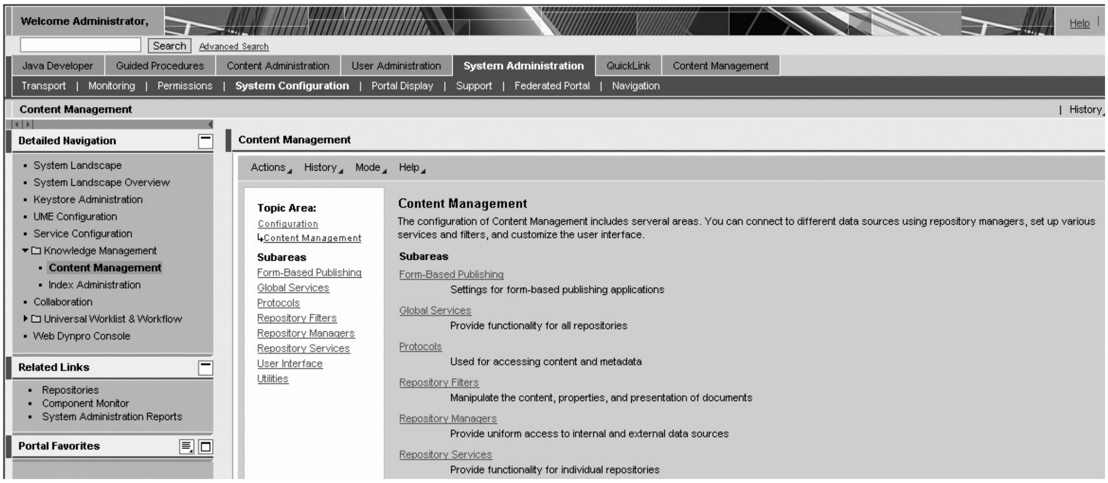
- KM Configuration
- KM Index administration
- KM TREX monitor

NOTE *To access the KM functionality for conducting activities such as changing/configuring data sources and KM services, choose System Administration | System Configuration | Knowledge Management | Content Management.*

Figure 2-8 displays the portal screen used for configuring Content Management.

KM Global Services

KM services can be classified as *global* and *repository* services. To manage the KM global services, choose System Administration | System Configuration | Knowledge Management |



Content Management

The configuration of Content Management includes serveral areas. You can connect to different data sources using repository managers, set up various services and filters, and customize the user interface.

Subareas

Form-Based Publishing
Settings for form-based publishing applications

Global Services
Provide functionality for all repositories

Protocols
Used for accessing content and metadata

Repository Filters
Manipulate the content, properties, and presentation of documents

Repository Managers
Provide uniform access to internal and external data sources

Repository Services
Provide functionality for individual repositories

Copyright by SAP AG

FIGURE 2-8 Portal screen for configuring Content Management

Content Management | Global Services. KM global services are available to all repository types in Content Management. Following are examples of KM global services:

- **Audit logs service** Used for auditing changes to documents and folders.
- **Cache service** Used for managing caches.
- **Index management service** Used for indexing and classification of documents and folders in KM repositories so that they are available during search.
- **MIME handler service** Used for identifying MIME types of documents so that the relevant icons are displayed against the document in the folder view.
- **Notificator service** Used for sending notifications to users when changes are made to subscribed resources and documents.
- **Virus scan service** Used for scanning documents for viruses.

Tip *The KM services require the TREX to function effectively.*

KM Repository Services

The next layer in the KM architecture is the KM services that are used by the various KM repositories. Unlike the global services, the repository services are tied to individual repositories and hence should be registered with the repositories. This is done by maintaining the *Repository Services* parameter in the repository definition. To change/configure the repository services, choose System Administration | System Configuration | Knowledge Management | Content Management | Repository Services.

Examples of repository services follow:

- **Application property service** Used by services or applications to store application properties, which is especially useful for classifying documents.
- **Subscription service** Used for sending notifications to users when changes are made to documents, folders, discussions, and so on.
- **Time dependent publishing service** Used to determine how long the document will be visible to the users.
- **Collaboration services** Enables functions such as feedback, notes, discussions, comments, ratings, and attachments.
- **Status management service** Used for managing documents based on their status in approval workflows.

KM Repository Framework

The Repository framework consists of

- Repositories for storing documents
- Repository managers for managing those repositories

The KM repository framework provides the functionality for storing documents in a physical storage location. It helps to perform some basic functions such as deleting, copying, and reading files. Documents and folders can be stored in the form of a virtual hierarchy and namespace, in both the internal and external repositories.

KM Repositories

When the repositories are used purely for KM purposes, they are known as *internal repositories,* and if they are used by other components such as BW or other Content Management solutions, they are known as *external repositories*. The repositories could be Windows-based file systems, HTTP servers, WebDAV systems, and so on. To create/configure a repository manager, choose System Administration | System Configuration | Knowledge Management | Content Management | Repository Managers.

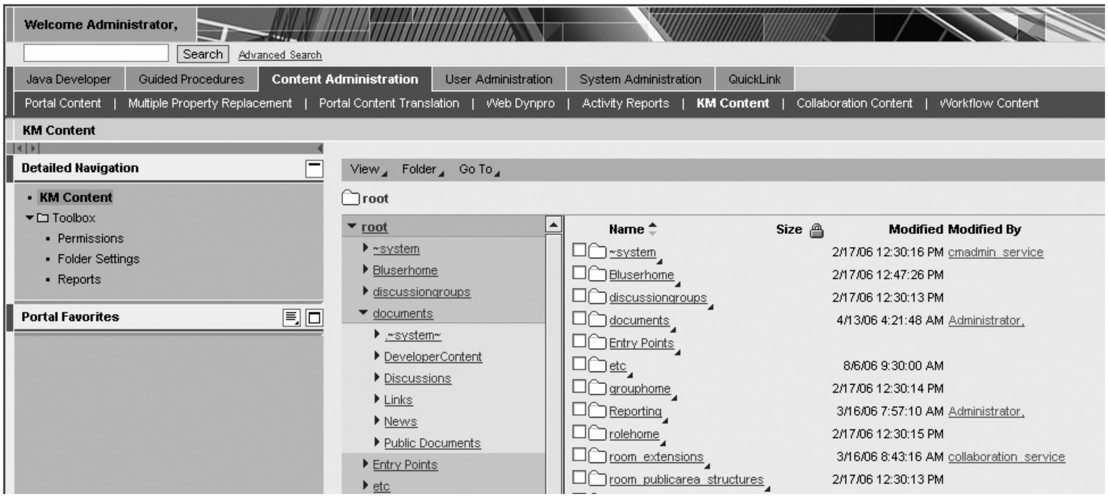
Figure 2-9 displays the folder and documents for a KM repository. Here we can carry out basic activities like:

- Creating new documents, folders, and links
- Searching for documents

KM External Repository Managers

External repositories are included so as to manage information that is stored external to Content Management. Those included in the standard delivery of the portal follow:

- **File System repository manager** Provides read and write access to the contents in a folder hierarchy in a file system
- **Lotus Notes repository manager** For read access to Lotus Notes database



	Name	Size	Modified	Modified By
□	~system		2/17/06 12:30:16 PM	cmadmin_service
□	Bluserhome		2/17/06 12:47:26 PM	
□	discussiongroups		2/17/06 12:30:13 PM	
□	documents		4/13/06 4:21:48 AM	Administrator,
□	Entry Points			
□	etc		8/6/06 9:30:00 AM	
□	grouphome		2/17/06 12:30:14 PM	
□	Reporting		3/16/06 7:57:10 AM	Administrator,
□	rolehome		2/17/06 12:30:15 PM	
□	room_extensions		3/16/06 8:43:16 AM	collaboration_service
□	room_publicarea_structures		2/17/06 12:30:13 PM	

Copyright by SAP AG

FIGURE 2-9 Folders and documents in KM Repository

- **WebDAV repository manager** For read/write access to information stored in IIS Services
- **Web repository manager** For read access to the contents of a website

KM Internal Repository Managers

Following are some types of internal repository managers:

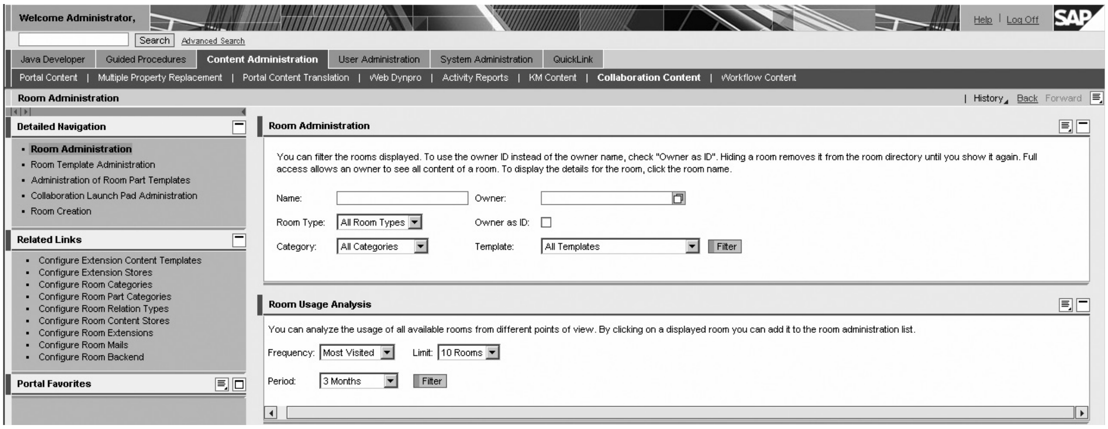
- **CM repository manager** Manages repositories that are used by CM to store documents and folders. Examples of CM repositories are
  - **/documents** for managing text documents
  - **/collaboration** for managing collaboration content
  - **/discussion groups** for managing discussion groups
- **User Management repository manager** Makes available user information to CM. An example of UM repository is /um.
- **PCD repository manager** For searching portal objects like roles and iViews in the Portal Content Directory (PCD) in KM. The repository is */pcd* in the standard delivery.

Note that this repository managers list contains just a few of the many repository managers that exist on the portal.

Collaboration Architecture

As the name implies, collaboration in the portal is all about bringing together individuals, teams, or groups to work together closely. This is achieved by using a common set of collaboration tools and services under the portal platform. Although a number of standalone collaboration tools, such as e-mail, project management tools, and application-sharing tools, are available in the market, bringing them all together under a common platform is a huge undertaking.

Figure 2-10 displays a sample screen for administering collaboration functionality.


Welcome Administrator,
Search
Advanced Search
Help | Log Off
SAP
Java Developer
Guided Procedures
Content Administration
User Administration
System Administration
QuickLink
Portal Content | Multiple Property Replacement | Portal Content Translation | Web Dynpro | Activity Reports | KM Content | Collaboration Content | Workflow Content
Room Administration
| History Back Forward
Detailed Navigation
Room Administration
Room Template Administration
Administration of Room Part Templates
Collaboration Launch Pad Administration
Room Creation
Related Links
Configure Extension Content Templates
Configure Extension Stores
Configure Room Categories
Configure Room Part Categories
Configure Room Relation Types
Configure Room Content Stores
Configure Room Extensions
Configure Room Mails
Configure Room Backend
Portal Favorites
Room Administration
Room Usage Analysis
Filter
Filter

Room Administration

You can filter the rooms displayed. To use the owner ID instead of the owner name, check "Owner as ID". Hiding a room removes it from the room directory until you show it again. Full access allows an owner to see all content of a room. To display the details for the room, click the room name.

Name: Owner:
Room Type: All Room Types Owner as ID: □
Category: All Categories Template: All Templates

Room Usage Analysis

You can analyze the usage of all available rooms from different points of view. By clicking on a displayed room you can add it to the room administration list.

Frequency: Most Visited Limit: 10 Rooms
Period: 3 Months

Copyright by SAP AG

FIGURE 2-10 Portal screen for administering collaboration functionality

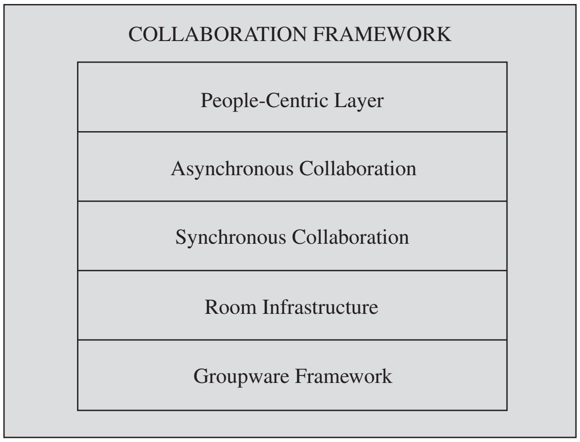


FIGURE 2-11 Collaboration framework components

Copyright by SAP AG

The *collaboration framework* consists of the following components (refer to Figure 2-11):

- **People centric layer** Consists of iViews for selecting people and displaying information regarding people.
- **Room infrastructure component** Allows you to create, delete, and change *collaboration rooms* (used by project teams to share data and information).
- **Asynchronous tools and components** Allows asynchronous collaboration between users using functionalities such as *discussion groups, feedback, rating*, and so on.
- **Synchronous collaboration framework** Used for real-time collaboration functionalities such as *application sharing* and *instant messaging.*
- **Groupware framework** Allows integration of *e-mail systems* such as Lotus Notes and Microsoft Exchange; helps synchronize the task lists and the calendar between the two groupware systems.

Asynchronous Collaboration Framework

Asynchronous collaboration tools and *components* can be used for online discussions, tasks, and documents and provide services such as feedback, notes, reviews, and ratings.

- **Rating** Can be applied to either documents or folders. Helps you evaluate the content available in a document or folder. The rating scale depends on the company's configuration. Multiple ratings can be maintained by a user for a given document. In such cases when the system calculates the average rating for all users, it includes the latest evaluation rating only for calculation.
- **Review feature** Allows you to make comments regarding documents.
- **Feedback** Can be created on documents and folders. To create a feedback, you should have read access to that object and also be able to use the feedback interface.

TIP *You cannot maintain ratings for your own documents.*

Synchronous Collaboration Framework

The *Synchronous collaboration framework* allows real-time collaboration through

- **Application sharing** Used for product demos, presentations, and customer support
- **Chat** Used for multiple users to communicate with each other in real time
- **Instant messaging** Used for one-to-one communication

Room Infrastructure

The *room infrastructure component* allows you to manage the life cycle of team rooms. *Collaboration rooms* are virtual workspaces with all the necessary tools, services, and information available in one place. They can be integrated with real-time collaboration services and other third-party groupware as well. The room templates allow you to define the structure, content, and access controls for the room.

Collaboration Launch Pad

The *Collaboration Launch Pad* is a central point of access to contacts, e-mails, documents, discussions, appointments, rooms, tasks, application sharing, and instant messaging.

Third-Party Services

Third-party services such as WebEx can also be integrated into the portal. This allows portal users to communicate with non-portal users.

J2EE Architecture

The next system to be addressed is the *J2EE architecture,* because the portal is nothing but a software application that is deployed on an *SAP Web application server*. What it really boils down to is that, from an architectural standpoint, when we speak of the portal architecture, we are really referring to the J2EE architecture. Thus, concepts such as load balancing, high availability, and security ultimately depend upon the J2EE engine. This topic is addressed in detail in Chapter 6.

Summary

This chapter discussed the building blocks of the portal platform from a software architecture standpoint. You learned at a high level the various components of the portal platform, the Knowledge Management component, and the Collaboration platform. A good understanding and knowledge of these topics will help you when you're designing the technical infrastructure as well as gathering the business requirements during the portal implementation planning process.