Room API

SAP NetWeaver Product Management

THE BEST-RUN BUSINESSES RUN SAP

Content

This document will cover the following topics:

- **Examples of Integration of Collaboration Rooms into SAP Applications**
- **Collaboration Room Infrastructure**
  - **Collaboration Room framework**
  - **Different layers of Collaboration Rooms**
  - **User management**
- **Collaboration Room APIs**
  - **Introduction**
  - **How to use the Collaboration Room API**
    e.g. to create Rooms, delete Rooms, add Room parts

© SAP AG 2006, Room API

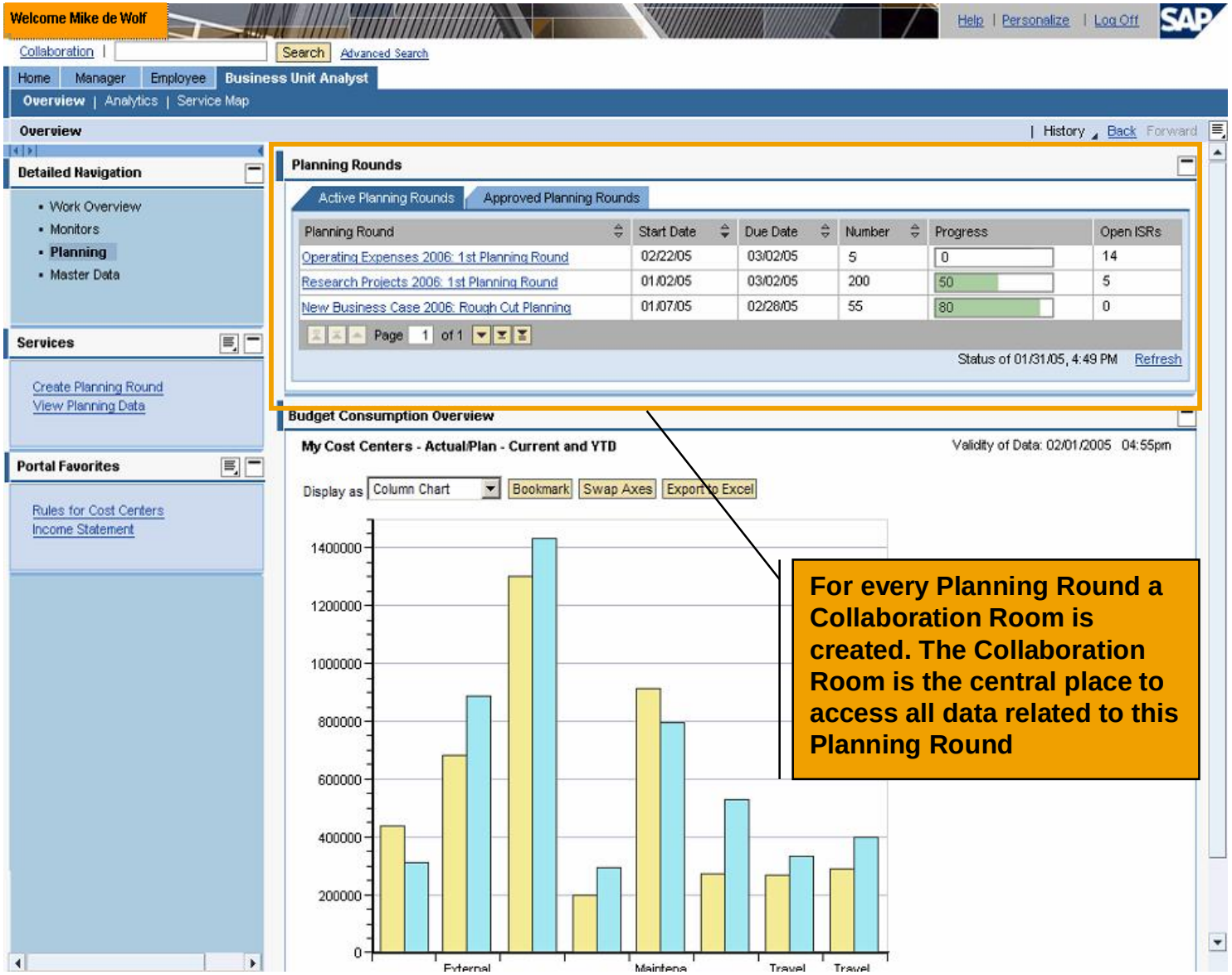
Examples of Integration into SAP Applications

Collaboration Room Infrastructure

Collaboration Room APIs

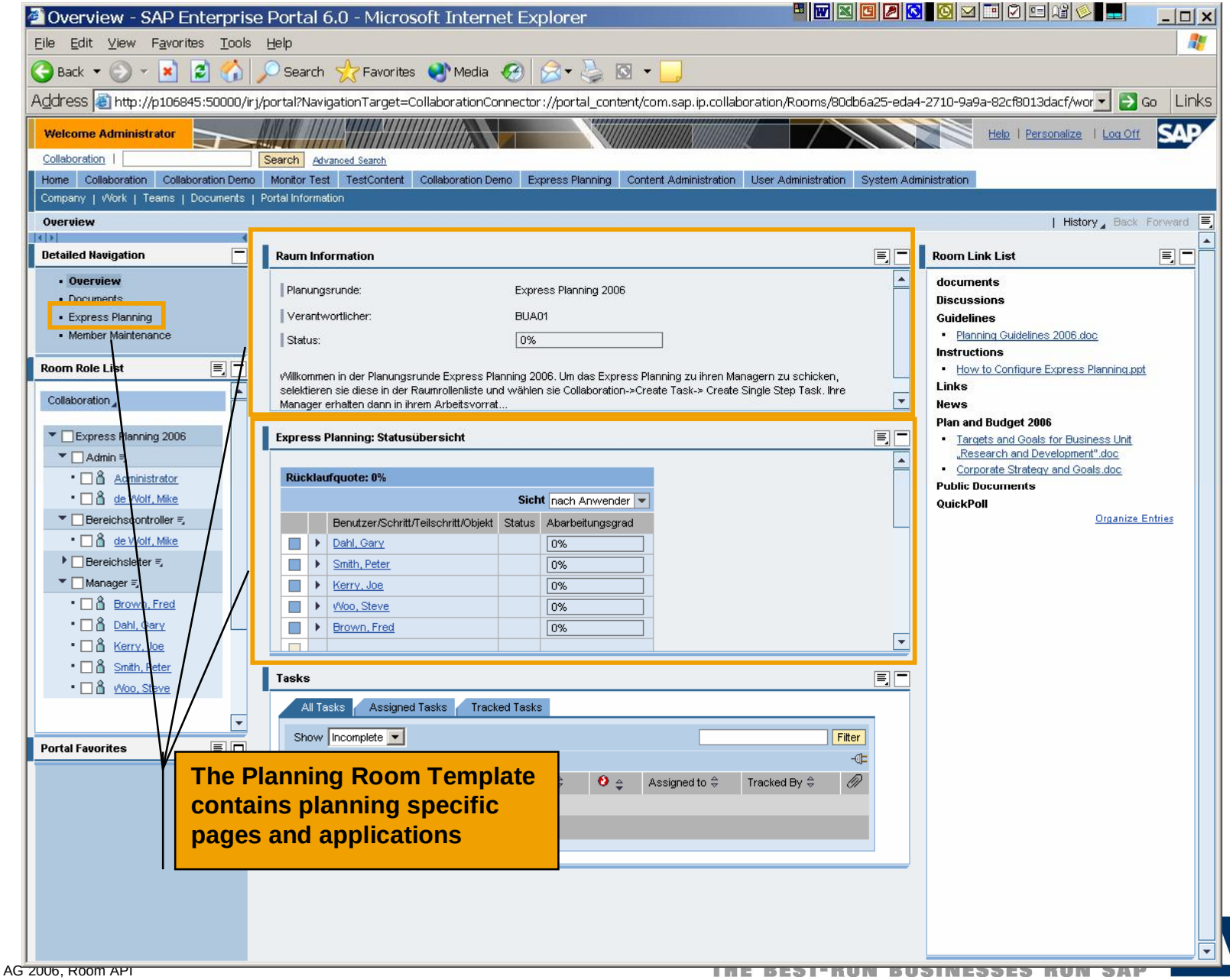
THE BEST-RUN BUSINESSES RUN SAP

Embedding Collaboration Rooms via Web Services API
(Planning Work Center of Business Unit Analyst in ERP2005)



© SAP AG 2006, Room API

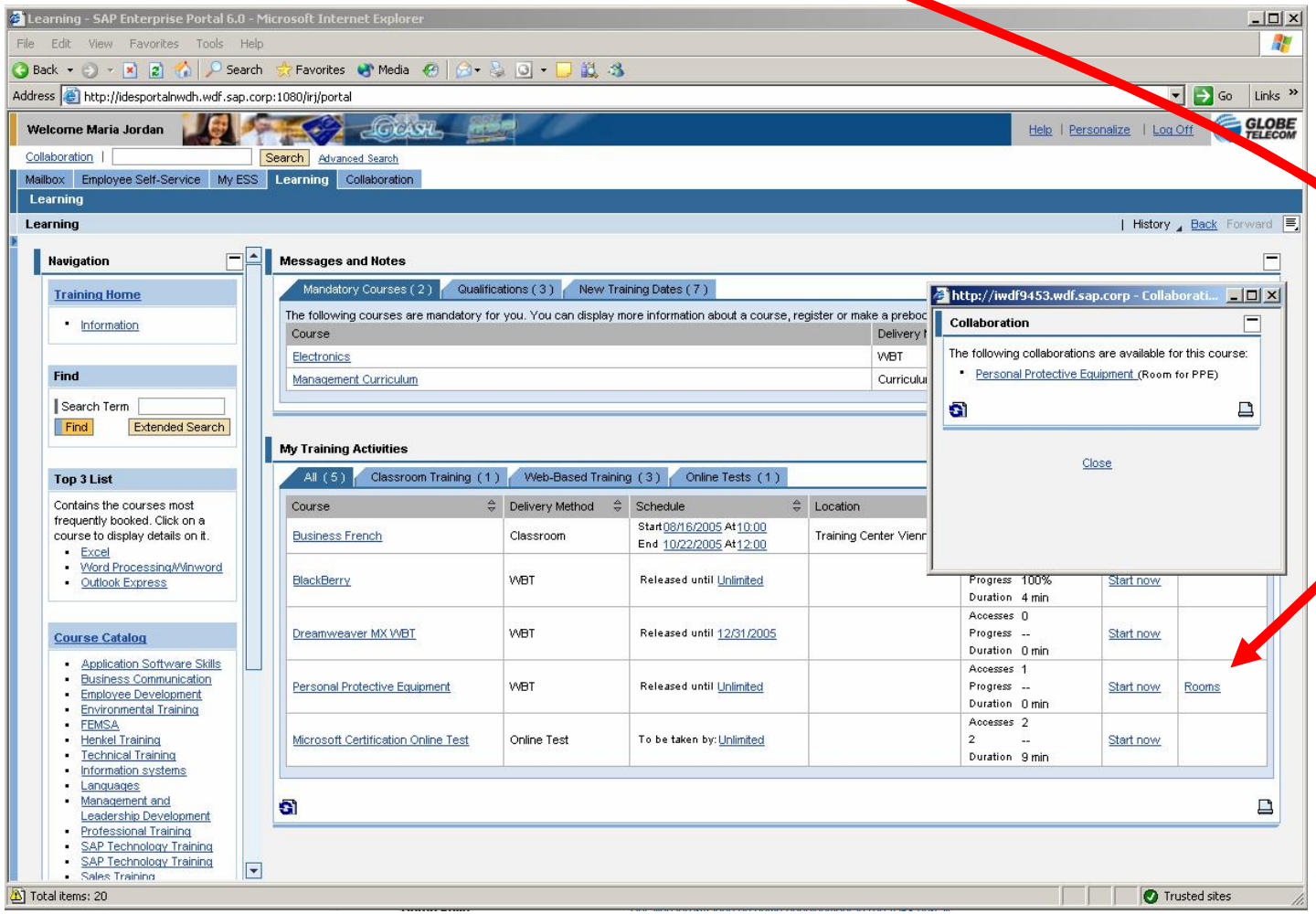
Collaboration Room for Planning: Overview Page



© SAP AG 2006, Room API

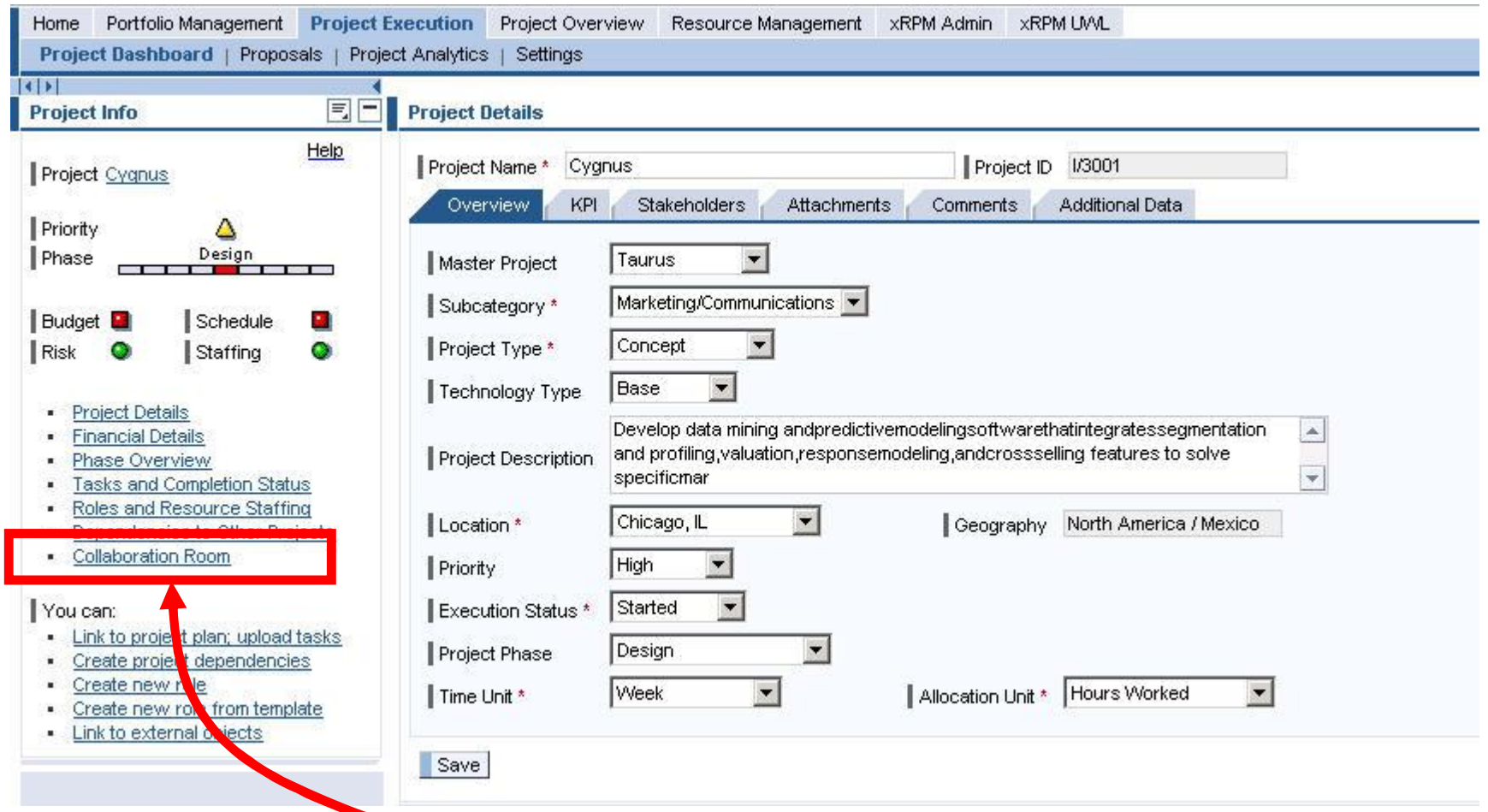
Example:
SAP Learning Solution and Collaboration Rooms

SAP Learning Solution enables you to set up virtual learner communities where students can speedily and efficiently exchange information with each other



© SAP AG 2006, Room API

Example (xRPM): Collaboration Rooms Integrated into Project Planning

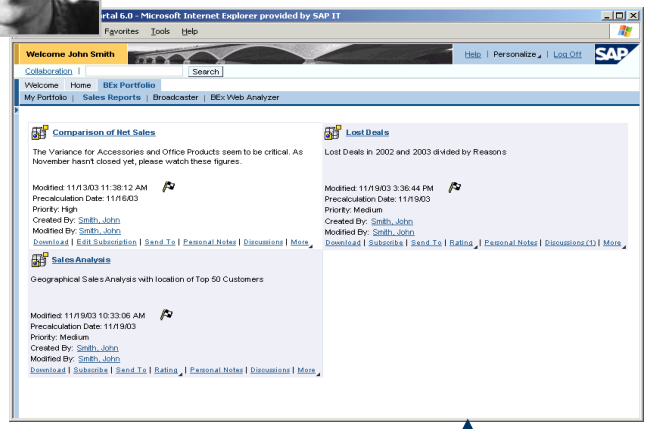


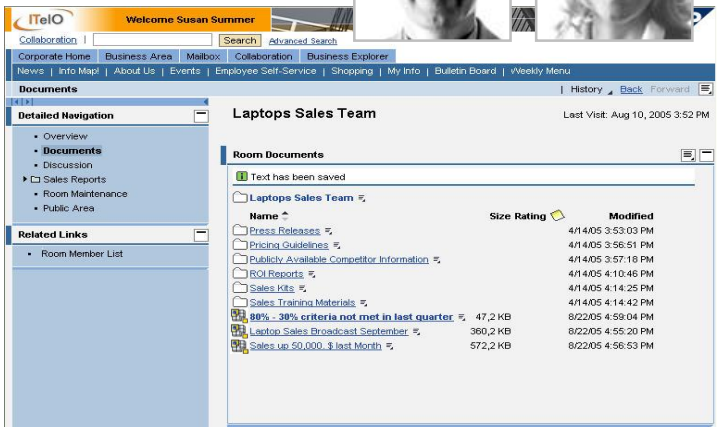
Once you have found the right project resources with xRPM, invite them into their future virtual working space

© SAP AG 2006, Room API

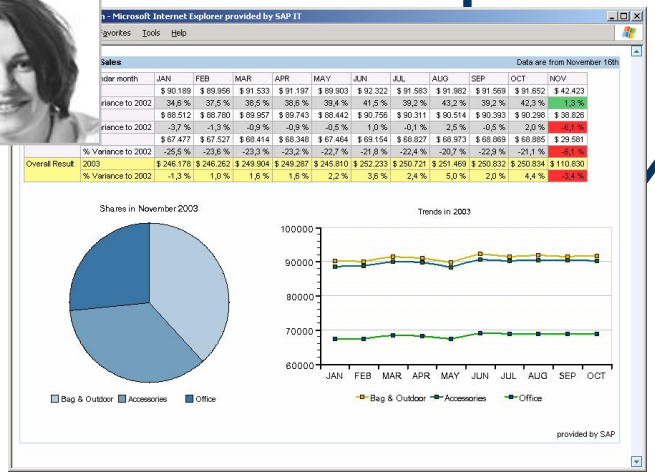
INFORMATION TO ACTION – INFORMATION BROADCASTING

Subscribe
Feedback
Notes
Search
...





Broadcast into KM
(ad-hoc, scheduled, event-driven)



Broadcast into KM
Folders within
Collaboration Rooms

© SAP AG 2006, Room API

Examples of Integration into SAP Applications

Collaboration Room Infrastructure

Collaboration Room APIs

THE BEST-RUN BUSINESSES RUN SAP

Collaboration Room Infrastructure

- **Collaboration Room Framework combines the functional capabilities of**
  - **Portal and**
  - **Knowledge Management**
- **Collaboration Room API**
  - **Provides abstraction**
  - **Offers scenario-oriented access to room functionality**

© SAP AG 2006, Room API

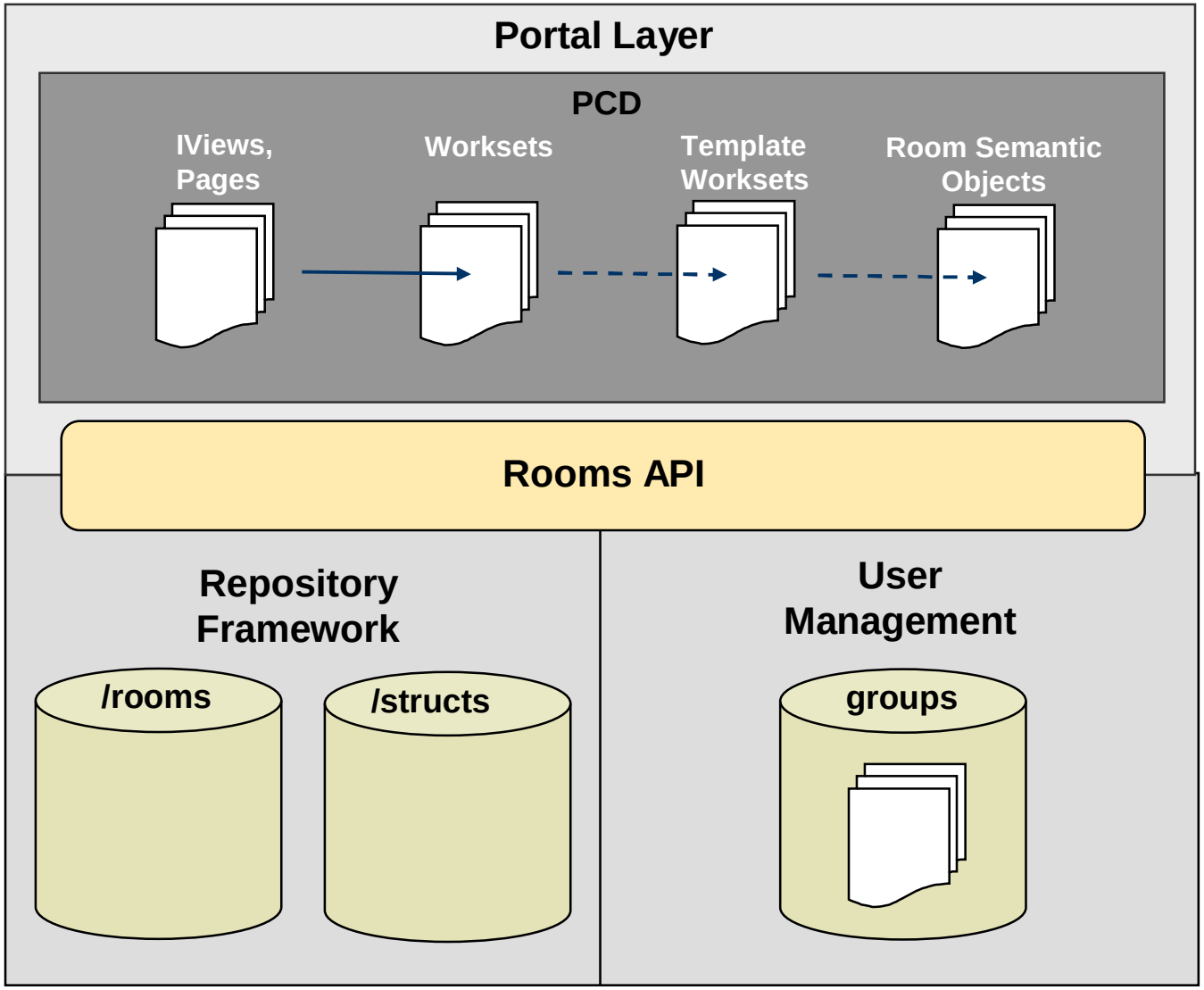
Collaboration Room Infrastructure

Three main parts of SAP NW Portal and KM framework are used to store rooms and room-related information

- **Portal Content Directory (PCD)**
  - **Used to store the portal-related parts of existing room instances such as worksets, pages, and iViews.**
  - **Also used as a store for the room templates that make the room creation process much easier for end users.**
- **User Management**
  - **Used to store user-to-room and user-to-room role assignments.**
- **Repository Framework (RF)**
  - **Used to create a semantic object for each room.**
  - **These room objects hold all room information that is not stored in one of the other two layers.**
  - **This includes the name and description of the room, and other basic information.**

© SAP AG 2006, Room API

Architecture Overview: Portal/RF/UM



© SAP AG 2006, Room API

Collaboration Room Infrastructure – PCD Objects

Location

- portal_content/com.sap.ip.collaboration/worksets
  - Worksets which are used when creating a Collaboration Room Template
- portal_content/com.sap.ip.collaboration/TemplateWorksets
  - Worksets of Collaboration Room Templates
- portal_content/com.sap.ip.collaboration/room_part_worksets
  - Worksets which are used when creating a Collaboration Room Part Template
- portal_content/com.sap.ip.collaboration/RoomPartTemplateWorksets
  - Worksets of Collaboration Room Part Templates
- portal_content/com.sap.ip.collaboration/rooms
  - pcd objects of instantiated collaboration rooms (not visible)

© SAP AG 2006, Room API

Collaboration Room Infrastructure – PCD Objects

Permissions

Configuration permissions guide

- **https://www.sdn.sap.com/irj/sdn/howtoguides ➔ SAP NetWeaver 7.0 -> User Productivity -> Running an Enterprise Portal -> How to Configure Initial Permissions for Initial Content in SAP NetWeaver.**

8. Set permissions to Collaboration content.

This section is only relevant if you installed Collaboration in your portal.

Set the permissions as follows:

- Folder: `/Portal Content/Content Provided by SAP/Collaboration`
  - Role: **Super Admin** [Owner/EU-on]
  - Role: **Content Admin** [Full Control/EU-on]
  - Role: **System Admin** [Read-Write/EU-on]
  - Group: **Everyone** [None/EU-on]

If you have integrated Collaboration Rooms into your portal, do the following as well:

- Folder: `/Portal Content/com.sap.ip.collaboration`
  - Group: **Everyone** group [None/EU-on]

  Note:
  - The `Rooms` sub-folder should inherit this permission also. This permission setting is necessary for users to be able to navigate within Collaboration-based rooms and view their content at runtime.
  - If the rooms are available to authenticated portal users only, you can instead add the **Authenticated Users** group with *EU-on* permission.
- Folder: `/Portal Content/Content Provided by SAP/Admin Content/Collaboration_Administrators`
  - Role: **Super Admin** [Owner/EU-on]
  - Role: **Content Admin** [None/EU-on]

—

THE BEST-RUN BUSINESSES RUN SAP

Collaboration Room Infrastructure – RF

Repositories

- /rooms
  - Storage of Collaboration Rooms with properties like description, roles, etc.
- /room_structures
  - Storage of structures of the room e.g. which folders belong to a room.
- /room_extensions
  - Standard repository which is used when creating storages for rooms.
- /room_stores
  - Former store concept storage. Should not be used anymore.

© SAP AG 2006, Room API

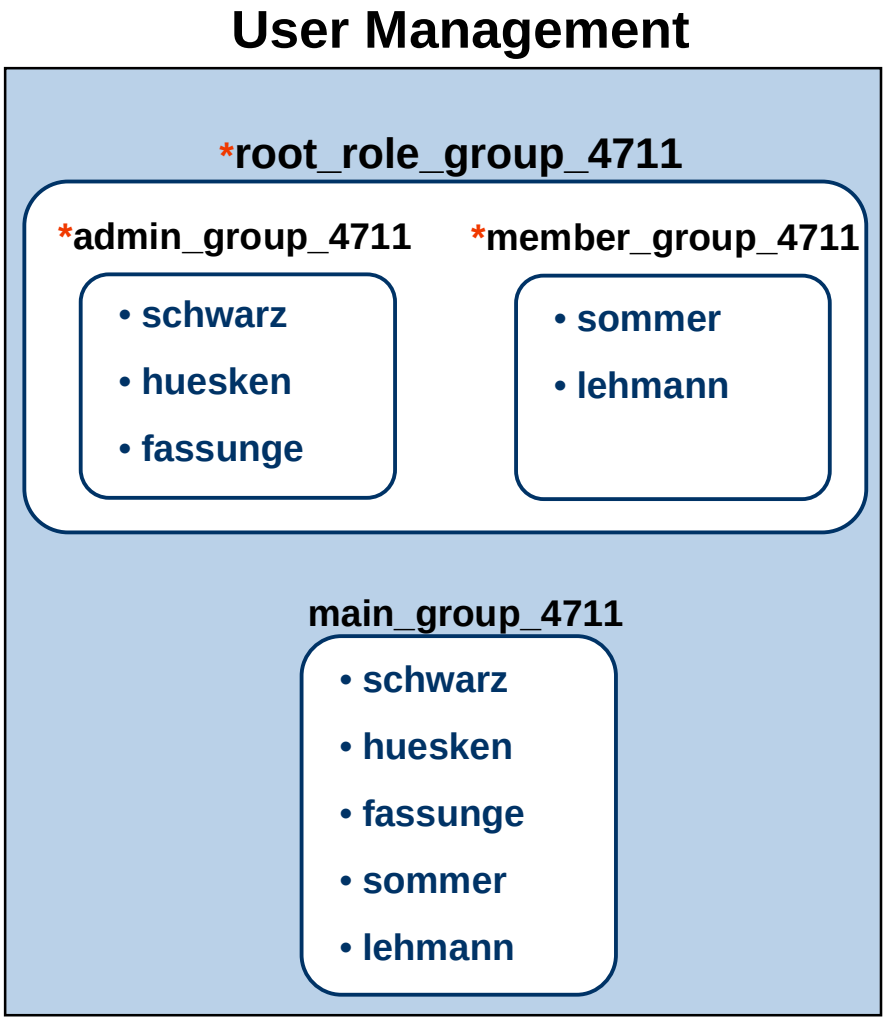
Collaboration Room Infrastructure – UME

Example Room 4711

RoomRole	User
owner	schwarz
admin	huesken
admin	fassunge
member	sommer
member	lehmann

* hidden group

(not visible in normal portal navigation)



© SAP AG 2006, Room API

Collaboration Room Infrastructure – Config Framework

Configuration Framework stores information about templates, categories, extensions, etc.

- **System Administration**
  - **➔ System Configuration**
    - **➔Knowledge Management**
      - **➔ Collaboration**

© SAP AG 2006, Room API

Collaboration Room Infrastructure – Room creation

What happens when a Collaboration Room is being created? – Steps performed by the Collaboration Room framework:

1. **Application calls**
   - **The Rooms API method createRoom() and**
   - **Passes all relevant data for the new room as a parameter**
2. **The Collaboration Room backend calls the User Management API to**
   - **Create a user group containing all users of the new room**
   - **Create a user group for each room role containing the users assigned to the role in question**
   - **Create a so called root role group - a virtual role group which contains all other role groups as children**

© SAP AG 2006, Room API

Collaboration Room Infrastructure

3. **Repository Framework API is called to**
   - Create a new folder resource in a special repository that represents the room itself
   - Set the name, description, and so on of the room as properties of the new room resource
   - Create a link from the room resource to the room *Structure* folder
   - Set ACL permissions for the room resource
4. **The Portal Connector API is called to**
   - Copy the template workset (including pages and iViews) into a special PCD folder
   - Map all room parameters (including the hidden parameters such as room ID and so on) to the iView parameters according to the mapping rules defined in the template
5. **Groupware API is called to**
   - Send invitation e-mails to all users of the new room
6. **Status Engine API is called to**
   - Create a status event for the room creation
7. **At each extension point during the room creation process**
   - Extension framework calls the extensions defined in the template to enable the connection to backend systems and to add additional functionality to the new room

© SAP AG 2006, Room API

Examples of Integration into SAP Applications

Collaboration Room Infrastructure

Collaboration Room APIs

THE BEST-RUN BUSINESSES RUN SAP

Overview - Features of the Rooms API

Actions

- Create new rooms
- Check for existence of rooms
- Retrieve existing rooms by ID
- Retrieve rooms for a given user
- Retrieve rooms with a given privacy
- Retrieve most/least visited rooms
- Create or delete typed relations between rooms
- Search for typed relations between rooms
- Delete rooms

© SAP AG 2006, Room API

Overview - Features of the Rooms API

Room-Instance-Based Information Functionality

- Room name
- Room description
- Creation date of the room
- Version of the room object
- Date of the last modification of the room
- Date when the room was last entered by a given user
- Date when the room was last entered by any user
- Internal/external room access URL
- Unique ID of the room
- Internal/external room parameters
- Room categories
- Template name
- Template description

© SAP AG 2006, Room API

Overview - Features of the Rooms API

Room-Instance-Based Information Functionality (cont.)

- Room privacy
- Lock state
- Hidden state
- Room owner
- Room users
- Room roles
- Initial room role
- Admin room role
- Members of a given room role
- Room roles of a given user
- Check whether a user is room member
- Check whether a room has public content
- Usage information for the room
- Room events
- Room parts
- Position of room parts in the *Detailed Navigation* iView

© SAP AG 2006, Room API

Overview - Features of the Rooms API

Modifications of a Room instance

- Room name can be changed
- Room description can be changed
- Room owner can be changed
- Room privacy type can be changed
- Room categories can be changed
- Room parameters can be changed
- Application properties can be set at the room instance
- Hide/unhide room
- Lock/unlock room
- Invite new members to a room
- Remove users from a room

© SAP AG 2006, Room API

Overview - Features of the Rooms API

Modifications of a Room instance (cont)

- Change a user's roles in a room
- Add a user to a room role
- Remove a user from room roles
- Register/deregister a user in a community
- Add/remove room parts
- Change the order of room parts in the *Detailed Navigation* iView

© SAP AG 2006, Room API

Rooms API Structure - Rooms API

Description

- Entry point for other applications to access the functionality provided by the Collaboration Room framework.

Accessibility

- as a KM service and
- as a portal service in the portal environment.
- It does not work on an instance level, but on a factory level that provides access to specific room instances.

© SAP AG 2006, Room API

Prerequisites – Compile Time

- **NW Developer Studio - Include the following jar file in your project**
  - ***coll.shared.roomobject_api.jar***
- **Located under in a NetWeaver Portal**
  - ***...\WEB-INF\portal\portalapps\com.sap.netweaver.coll.shared\lib***
- **Example in a NW Development Studio**
  - ***Project ->Properties ->Java Build Path ->Libraries***

© SAP AG 2006, Room API

Prerequisites – Runtime

- **Add a reference in the portalapp.xml**

... <application-config>

... <property name="ServicesReference„

value="...,com.sap.netweaver.coll.shared,..."/>

... </application-config> ...

© SAP AG 2006, Room API

Rooms API Structure - Rooms API

Example with a KM Service

```
import com.sap.ip.collaboration.room.api.IRooms;
import com.sapportals.wcm.service.ServiceFactory;
IRooms roomsAPI =
(IRooms) ServiceFactory .getInstance().getService(IRooms.KM_SERVICE_ID);
```

Example with a Portal Service

```
import com.sap.ip.collaboration.room.api.IRooms;
import com.sapportals.portal.prt.runtime.PortalRuntime;
IRooms roomsAPI =
(IRooms) PortalRuntime .getRuntimeResources() .getService(IRooms.PORTAL_SERVICE_ID);
```

© SAP AG 2006, Room API

Rooms API Structure - RoomObject API

Description

- **Provides access to the functionality of a specific room instance.**
- **E.g. invite new users to a room.**

Example

```
//Get the Rooms 6.0 API
Portal Service IRooms roomsAPI =
(IRooms) PortalRuntime.getRuntimeResources().getService(IRooms.PORTAL_SERVICE_ID);

//Get a Room
IRoom room = roomsAPI.getRoom("4711");

//throws RoomInstantiationException
```

© SAP AG 2006, Room API

Rooms API Structure - RoomInfo API

Description

- Implemented by the RoomInfo object.
- Handles parts of the instance data of a RoomObject.
- Similar to a partial snapshot of a related RoomObject and can be used to create new rooms.
- RoomInfo API provides
  - mandatory field validation.
  - consistency checks for the contained data.
- RoomInfo object is also passed to room extensions to provide them with information about the room.

RoomInfo objects do not contain

- Portal-related information (worksets, pages, and so on).
- Room content (documents, business objects, and so on).

© SAP AG 2006, Room API

Rooms API Structure – RoomUsers API

Description

- Provides access to the user-related functionality of a room instance
  - E.g. changing the user-role assignments.
- Implemented by a RoomUsers object
  - Contains the user-role assignments for a specific room.

Reasons why the user-related functionality has been separated from the RoomObject API

- Keep rarely used methods together, but separated from the RoomObject API methods.
- Changes to the RoomUsers object do not affect the corresponding room immediately.

© SAP AG 2006, Room API

Rooms API Structure – RoomUsers API

RoomUsers API is coupled with

- **Status Engine**
  - **All changes to the user-role assignments are tracked by the Status Engine.**
- **The e-mail messaging features of the Collaboration framework.**
  - **For example, when a new user is assigned to a room, an invitation e-mail is sent to the user.**

Room Users and Extension engine

- **An instance of the RoomUsers interface is also passed to room extensions at certain extension points to provide a set of affected user-role assignments to the extension (for example, added room users or removed room users).**

© SAP AG 2006, Room API

Rooms API Structure - RoomUsers API

Example

```
public void changeUserRoles() throws RoomInstantiationException, UMException
{
  //Get the Rooms 6.0 API Portal Service
  IRooms roomsAPI = (IRooms)
  PortalRuntime.getRuntimeResources().getService(IRooms.PORTAL_SERVICE_ID);

  //Get a Room
  IRoom room = roomsAPI.getRoom("4711");

  //Get the initial role of the room
  IRoomRole newRole = TemplateDataFactory.
                createRoomRole(room.getInitialRoleName());
```


THE BEST-RUN BUSINESSES RUN SAP

Example (cont.)

```
//Get the Room Users
IRoomUsers roomUsers = room.getRoomUsers();
IUser newUser = UMFactory.getUserFactory().getUserByLogonID("MyName");
roomUsers.addUserRole(newUser.getUniqueID(), newRole);
String[] userIds = roomUsers.getAllUsers();

//iterate over all room users
for(int i=0; i<userIds.length; i++)
{
  IRoomRole[] userRoles = roomUsers.getUserRoles(userIds[i]);
  //... do something with the user's roles
}

//Apply all changes to the Room and send no notification emails
room.setRoomUsers(roomUsers, false);
```

© SAP AG 2006, Room API

Rooms API Structure – Room Relations API

Description

- Implemented by the room relation manager
- Can be obtained from the Rooms API
- Provides methods for creating or deleting relations between rooms
- Each relation has a specific relation type and relates a source room with a target room

Provides methods for searching existing relations by

- Source room
- Source room and relation type
- Target room
- Target room and relation type

Remark

- No semantic connection (e.g. no role inheritance, content sharing)
- Only a link between two rooms that makes navigation between the two rooms easier

© SAP AG 2006, Room API

Rooms API Structure - Rooms Relations API

Example

```
import java.util.List;
import com.sap.ip.collaboration.room.api.IRooms;
import com.sap.ip.collaboration.room.api.relman.IRelationManager;
import com.sapportals.portal.prt.runtime.PortalRuntime;
import com.sapportals.wcm.repository.ResourceException;

public void relateRooms() throws ResourceException
{
   String sourceRoomId = "4711";
   String targetRoomId = "4712";

   //Get the Rooms 6.0 API
   Portal Service IRooms roomsAPI = (IRooms)
      PortalRuntime.getRuntimeResources().getService(IRooms.PORTAL_SERVICE_ID);
```

© SAP AG 2006, Room API

Example (cont)

```
  //Get the RelationManager
  IRelationManager relMan = roomsAPI.getRelationManager();

  //Retrieve a list of possible relation types
  List types = relMan.getAllAssociationIds();

  //Relate two rooms with the first available relation type
  relMan.relateRoom(sourceRoomId, targetRoomId, (String) types.get(0));

  //Search for target rooms of room "4711"
  List targetRooms =
     relMan.getRelatedRoomsFromRoom(sourceRoomId, (String) types.get(0));

  String roomId = (String) targetRooms.get(0); //roomId should be "4712" now

  //Delete relation
  relMan.unrelateRoom(sourceRoomId, targetRoomId, (String) types.get(0));
}
```

© SAP AG 2006, Room API

Rooms API Structure – Room Status Event API

Description

- Provides access to the room-related status events that are automatically logged during system operation

The following functionality is provided

- Retrieval of status events related to a certain room
- Retrieval of room usage data

Status events

- Can be retrieved using the RoomObject API or
- Can be retrieved using the Rooms API (specified by its ID)
- Possibility to filter events by
  - time stamp,
  - status action (defined in RoomStatusAction), and
  - user who caused the events.

© SAP AG 2006, Room API

Rooms API Structure – Room Status Event API

Room usage data

- **Room specific usage data can be accessed using the RoomObject API and the Rooms API.**
- **Non room specific usage data regarding the most visited or least visited rooms is accessible using the Rooms API only.**

© SAP AG 2006, Room API

Rooms API Structure - Room Status Event API

Example

```
//imports for Collaboration Rooms
import com.sap.ip.collaboration.room.api.IRoom;
import com.sap.ip.collaboration.room.api.IRooms;
import com.sap.ip.collaboration.room.api.IRoomStatusEvent;
import com.sap.ip.collaboration.room.api.RoomStatusAction;

//imports for Portal
import com.sapportals.portal.prt.runtime.PortalRuntime;

public void retrieveUserChangeEvents() throws RoomInstantiationException
{

  //Get the Rooms API
  Service IRooms roomsAPI = (IRooms) PortalRuntime
     .getRuntimeResources().getService(IRooms.PORTAL_SERVICE_ID);

  //Get a room
  IRoom room = roomsAPI.getRoom("4711");
```

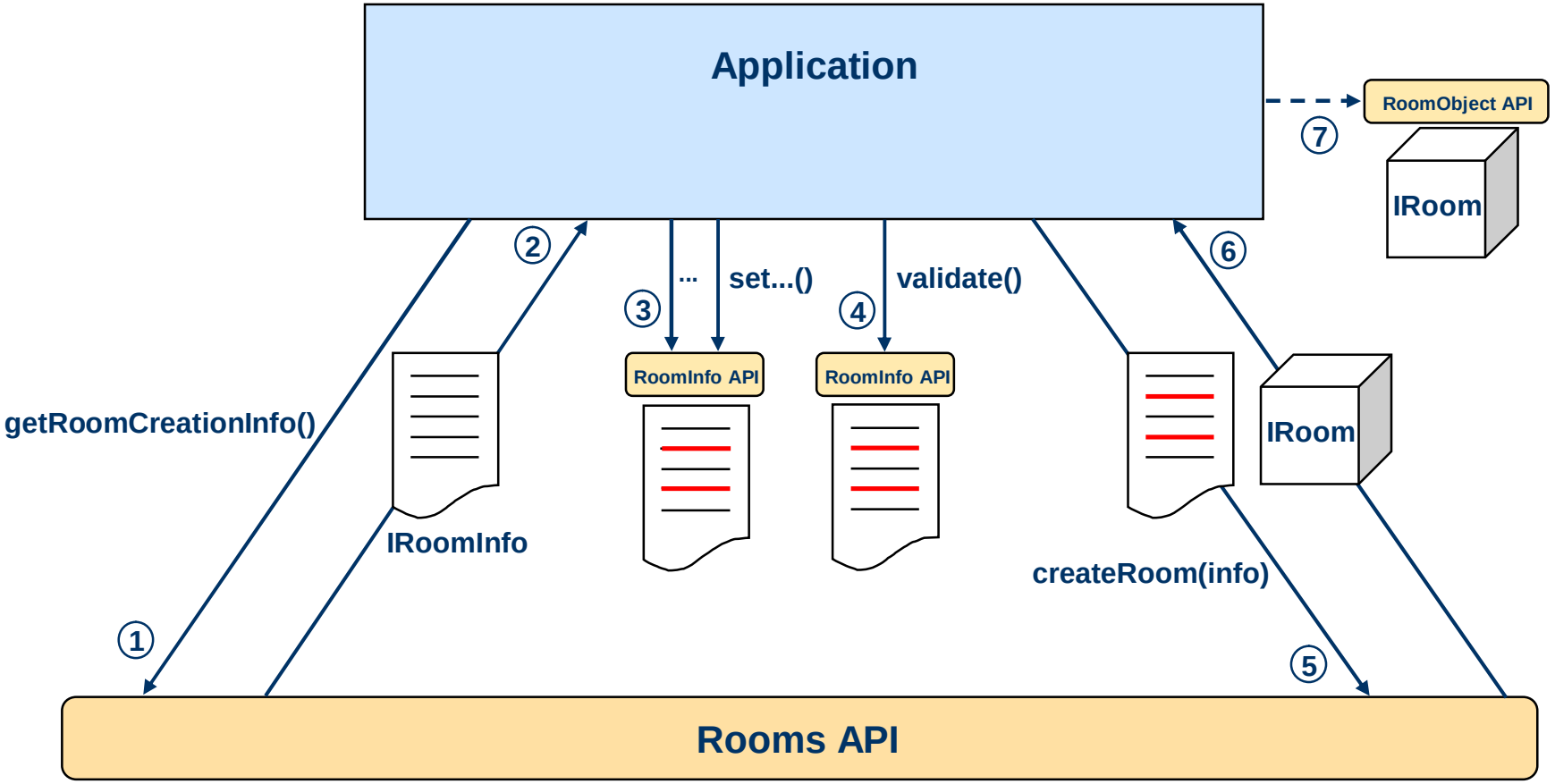

THE BEST-RUN BUSINESSES RUN SAP

Example (cont)

```
        // get all user related status events for the room
        IRoomStatusEvent[] roomEvents = room.getRoomEvents(
                0, // no time restriction = since room creation
                new RoomStatusAction[] {
                        RoomStatusAction.ROOM_ADD_USER,
                        RoomStatusAction.ROOM_REMOVE_USER,
                        RoomStatusAction.ROOM_CHANGE_USER_ROLE
                },
                null, // all users 50 // at most 50 events
        );

        for ( int i = 0; i < roomEvents.length; i++ )
        {
                // process events
                // ...
                // for instance: print message
                System.out.println( roomEvents[i].getMessage() );
        }
}
```

© SAP AG 2006, Room API

Rooms API Structure – Room Creation



Collaboration Rooms

© SAP AG 2006, Room API

Rooms API Structure – Room Creation

1. **Get RoomInfo Object**
   - Needed to create a new room
   - Call Rooms API method getRoomCreationInfo()
2. **Rooms API returns an empty RoomInfo object**
   - Contains no data
3. **Application calls at this RoomInfo object instance methods like**
   - setName(), setDescription() and other set...() to set the required data
   - Otherwise default value are taken from the room template (called by setTemplateName())
4. **RoomInfo method validate()**
   - Check for missing or inconsistent data in the RoomInfo object
   - Returns true, if everything is ok
   - If it returns false, a further call to getValidationErrors() provides more information about the error

© SAP AG 2006, Room API

Rooms API Structure – Room Creation

5. **Rooms API method createRoom()**
   - **RoomInfo object is passed as a parameter**
6. **The IRoom Object is created and a reference is passed back to the application**
7. **Room creation process**
   - **If creation in the rooms backend was successful, an instance of IRoom is returned**
   - **If there were any errors during the room creation process, a RoomCreationException is thrown. The exception contains more information about the error itself, or the causing exception as a nested exception.**
8. **The application uses the returned instance of IRoom for further tasks (RoomObject API).**

© SAP AG 2006, Room API

Rooms API Structure – Room Creation Example

```
public void createRoom() throws        TemplateNotFoundException,
                                       NoSuchRoleException,
                                       UMException,
                                       RoomCreationException

{
        String templateName = "Template1";

        String roleName = "Role1";

        //Get the Rooms 6.0 API Portal Service
        IRooms roomsAPI = (IRooms)
        PortalRuntime.getRuntimeResources().getService(IRooms.PORTAL_SERVICE_ID);

        //Get Template Related Infos
        ITemplate template = roomsAPI.getRoomTemplate(templateName);

        IRoomRole role1 = template.getRoomRole(roleName);

        //Get an empty RoomInfo object
        IRoomInfo roomInfo = roomsAPI.getRoomCreationInfo();
```

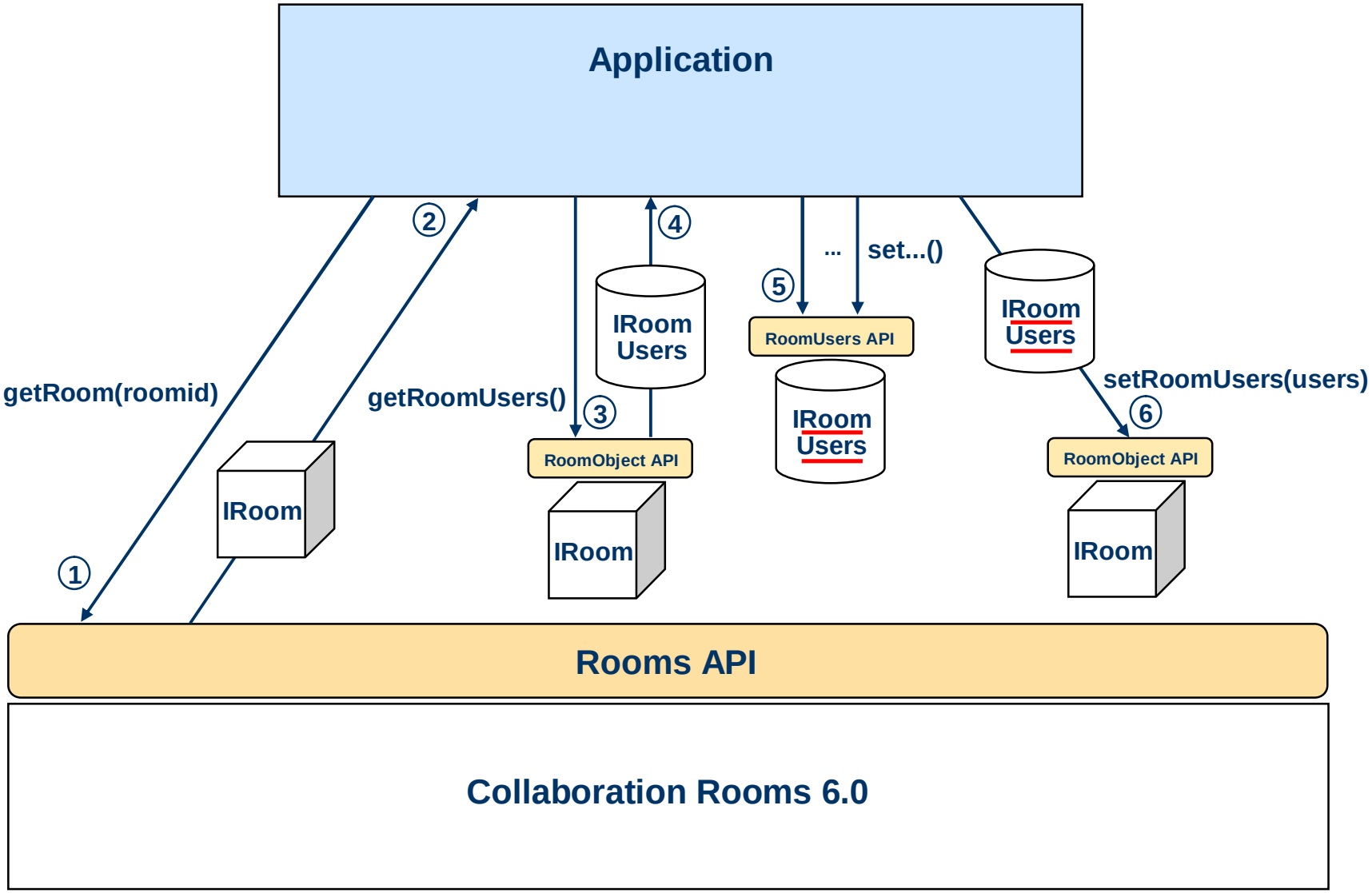

THE BEST-RUN BUSINESSES RUN SAP

Rooms API Structure – Room Creation Example

```
//Fill RoomInfo object with values
roomInfo.setName("New Room Name");

roomInfo.setDescription("New Room Description");

roomInfo.setTemplateName(templateName);

roomInfo.addRoomParameter("ParamName", "ParamDesc", "ParamValue", false);

//Set User Role Assignments
IUser owner = UMFactory.getUserFactory().getUserByLogonID("SmithJ");

roomInfo.setOwnerId(owner.getUniqueID());

roomInfo.addUserToRole(owner.getUniqueID(), role1);

//Finally Create the Room
if(roomInfo.validate())

    roomsAPI.createRoom(roomInfo, true);

}
```

© SAP AG 2006, Room API

Rooms API Structure – Change a Room



© SAP AG 2006, Room API

Rooms API Structure – Change a Room

1. **Call to the Rooms API method getRoom()**
   - **Passing the room ID of the room as a parameter**
2. **Rooms API returns an instance**
   - **If a room with the given ID exists**
3. **Room instance method getRoomUsers()**
4. **An instance of IRoomUsers is returned**
   - **Holds the current user-room role assignments**
5. **RoomUsers API**
   - **To modify user-room role assignments**
   - **E.g. addUserRole() to invite a new user to the room**
6. **Room instance method setRoomUsers()**
   - **Passes RoomUsers object as a parameter**
   - **Applies all changes to the room**

© SAP AG 2006, Room API

Rooms API Structure – Change a Room

```
public void changeUserRoles() throws RoomInstantiationException, UMException
{
    //Get the Rooms 6.0 API Portal Service
    IRooms roomsAPI = (IRooms)
    PortalRuntime.getRuntimeResources().getService(IRooms.PORTAL_SERVICE_ID);

    //Get a Room
    IRoom room = roomsAPI.getRoom("4711");

    //Get the initial role of the room
    IRoomRole newRole = room.getInitialRoomRole();

    //Get the Room Users
    IRoomUsers roomUsers = room.getRoomUsers();

    IUser newUser = UMFactory.getUserFactory().getUserByLogonID("SmithJ");

    roomUsers.addUserRole(newUser.getUniqueID(), newRole);

    String[] userIds = roomUsers.getAllUsers();
```


THE BEST-RUN BUSINESSES RUN SAP

Rooms API Structure – Change a Room

```
//iterate over all room users
for(int i=0; i<userIds.length; i++)
        {
                IRoomRole[] userRoles = roomUsers.getUserRoles(userIds[i]);
                //... do something with the user's roles
        }
//Apply all changes to the Room and send no notification emails
 room.setRoomUsers(roomUsers, false);
}
```


THE BEST-RUN BUSINESSES RUN SAP

Further Information

Public Web

- **NetWeaver Developer‘s Guide:**
  http://www.sdn.sap.com/irj/sdn/devguide2004s ➔Enabling User Collaboration

- **Commented Coding Samples:**
  http://help.sap.com/saphelp_nw04s/helpdata/en/7d/c69c42d706c66ae10000000a155106/frameset.htm

SAP Service Marketplace

- **Role-specific Learning Maps (available for purchase):**
  http://service.sap.com/rkt-netweaver ➔Available OKPs for SAP NetWeaver

© SAP AG 2006, Room API

Copyright 2006 SAP AG. 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.
- Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.
- IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, and Informix are trademarks or registered trademarks of IBM Corporation in the United States and/or other countries.
- Oracle is a registered trademark of Oracle Corporation.
- UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.
- Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.
- HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.
- Java is a registered trademark of Sun Microsystems, Inc.
- JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.
- MaxDB is a trademark of MySQL AB, Sweden.
- SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver 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 in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.
- These materials are subject to change without notice. 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.