CURE in a Nutshell

A "short" introduction for using the CURE system.

Till Schümmer
Cooperative Systems Department
FernUniversität in Hagen
Tuesday, 27 February 2007

Content

1. Login

When you connect to the CURE server, you will be prompted for a login with the following login screen (Figure 1):

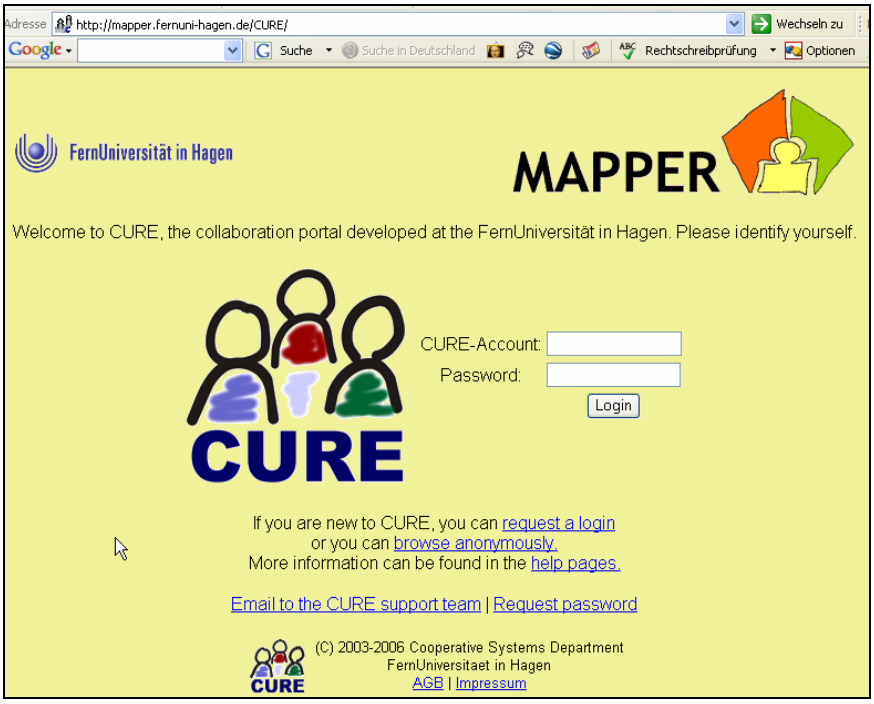

Figure 1: CURE Login Screen

Please enter your cure account here. If you don't have an account yet, you can request it by following the link "request a login" and filling out the following form (note that you might be prompted for a pass phrase that you may have received from the system administrator):

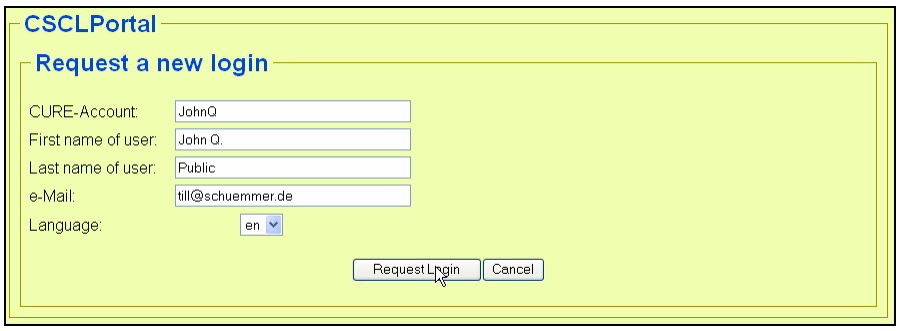

Figure 2: Creating a new account

Please remember the account name since you will need this to log in. The password will be mailed to the provided eMail address. Please check your mail and continue with the next chapter.

2. The Hall

After entering the system, you will be located in the Hall of the CURE system (Figure 4).

This is now the time to familiarize with some basic CURE concepts: CURE is based on the metaphor of virtual rooms and virtual keys. Room metaphors (Greenberg and Roseman, 2003, Pfister et al., 1998, Schümmer and Fernandez, 2005) have been widely used to structure collaboration. The room metaphor uses the room as the representation of a virtual place for collaboration. Figure 3 shows the abstractions that are offered by CURE. Users enter the collaboration space via an entry room that is called *Hall*. Rooms can contain pages, communication channels, e.g. threaded mail, and users. Users can also access all pages that are contained in the room. Changes of these pages are visible to all members in the room. The concept of a virtual key (Schümmer and Fernandez, 2005, Haake et al., 2004) is used to express access permissions of the key holder on rooms The access permissions, e.g., distinguish rights to enter a room, create sub rooms, edit pages, or to communicate within the room. Rooms with public keys are accessible by all registered users of the system.

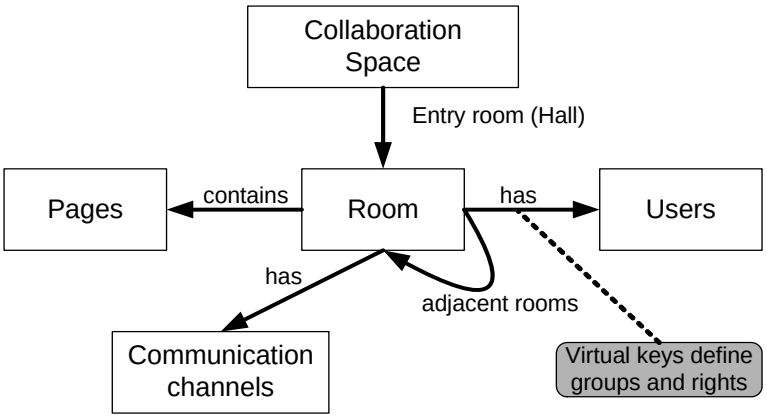

Figure 3 CURE abstractions

Users can enter a room, whereby they can now access the room's communication channels and may participate in collaborative activities. Users can also create and edit pages in the room. Pages may either be directly edited using a simple WIKI-like syntax (Leuf and Cunningham, 2001), or they may contain binary documents or artefacts. In particular, the syntax supports links to other pages, other rooms, external URLs or mail addresses. The server stores all artefacts to support collaborative access. When users leave the room, the content stays there to allow users to come back later and continue their work on the room's pages.

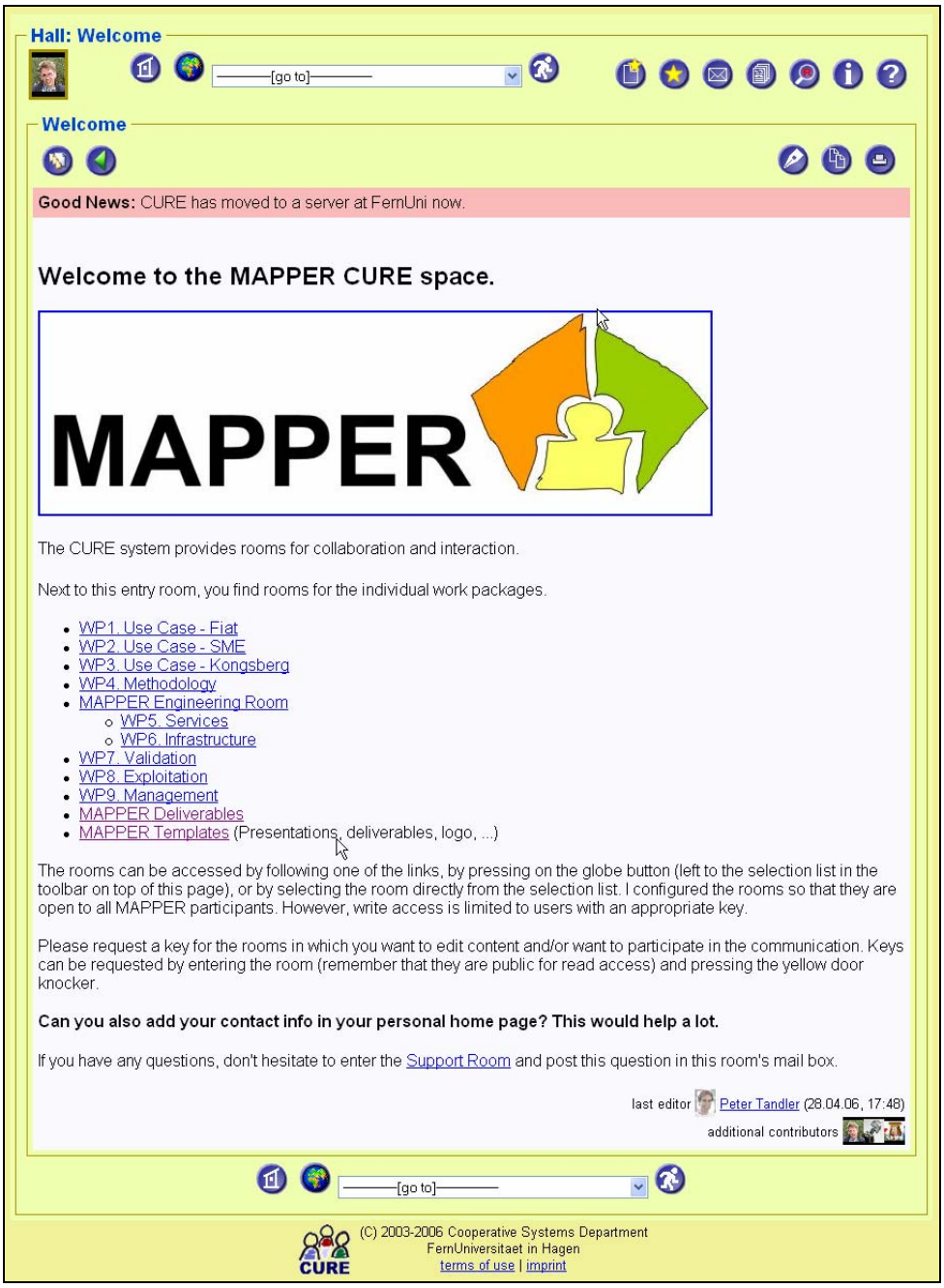

Figure 4: The Hall as it was in April 2006 (note that the content will change over time)

3. Introducing Yourself

After this excursus, it's now time to introduce yourself. Please edit your user profile and add a nice picture and some words explaining your role in the project. Remember that the CURE system has the goal of fostering collaboration and establishing links between the participants. So, this step is quite important in the beginning.

To edit your user profile please press your user button as shown in Figure 5.

Figure 5 Accessing your user profile

You will be directed to your personal home page in CURE.

Figure 6: The personal page of a CURE user

As any page in CURE, this page is editable. Pressing the pen icon (framed in Figure 6) will switch the page into the edit mode where you can modify the data. As a first try, please press the edit pen and continue to enter some data about you as shown in Figure 7.

You should change the description by entering some text explaining your role in the project. It is also good if you provide a picture. Select the picture by entering a path below the dummy picture (the blue person). The picture should be a JPG image with approx 240*320 pixels. However, larger pictures will be automatically scaled down and different aspect ratios will be compensated with black bars.

Finally, you should provide a new password by entering it in the password field and confirming it. Scroll the page to the end and press "set" to store your updated user profile.

Note that the button to your user profile will now have changed to the picture that you provided. This picture will be used in many places in the system as you will see in the next sections.

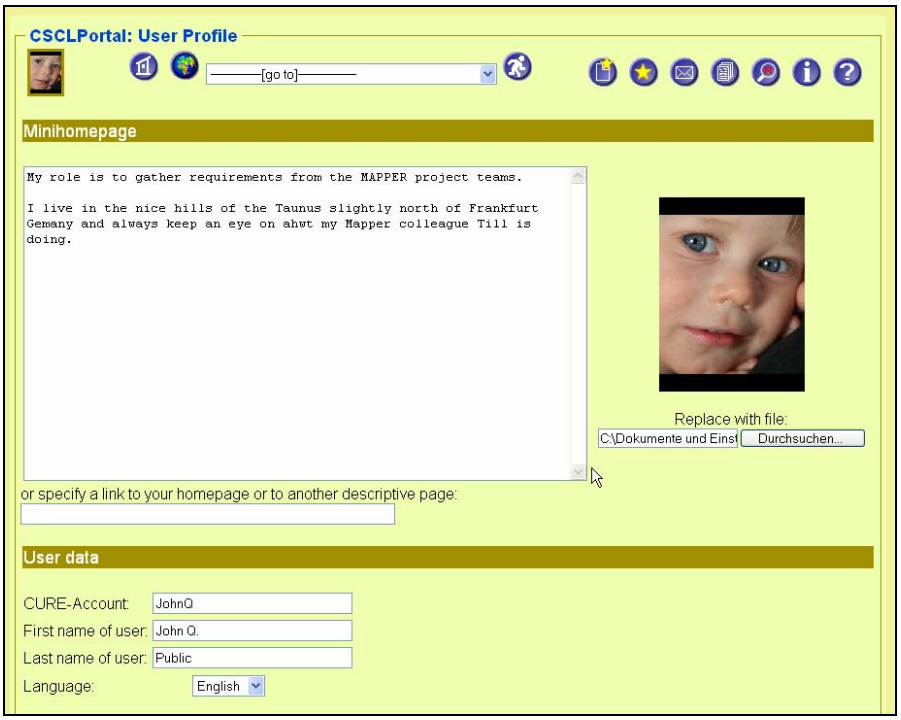

Figure 7: User profile in edit mode

4. Explore the content of CURE

As mentioned in chapter 2, CURE organizes content in rooms as pages. Each room has a special page (the start page) that is displayed when a user enters the room.

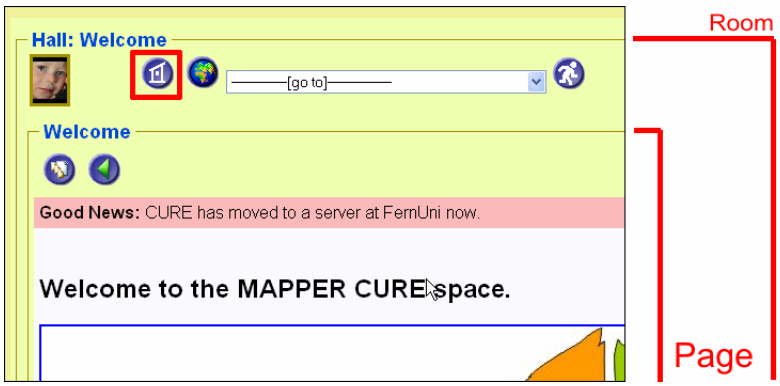

Figure 8: Recognizing rooms and pages

Figure 8 shows this default page for the room named Hall. The layout of CURE includes two frames that help to orient in the space of rooms. The outer frame is used to show the room. In the case of Figure 8, this is the room named *Hall*. The label of the room also includes the name of the currently shown page (in this case *Welcome*). The page starts in the inner frame. In the example, this frame is labelled with the name of the page (*Welcome*). However, different page types may fill this frame differently.

You can move back to the start page of a room by pressing the button with the small house highlighted in Figure 8 (some people mention that the house would look a little bit like an outhouse).

You can navigate through pages by following the hyperlinks on the page. The author of the page can add hyperlinks to other pages in the same room or to pages in other rooms. Thus, when following a hyperlink, it may happen that you move to another room. CURE will check your keys and see if you have a matching key to enter the room. If not, it will present you an error page with further instructions how to ask for a key for that room (more details on keys will follow in chapter 9).

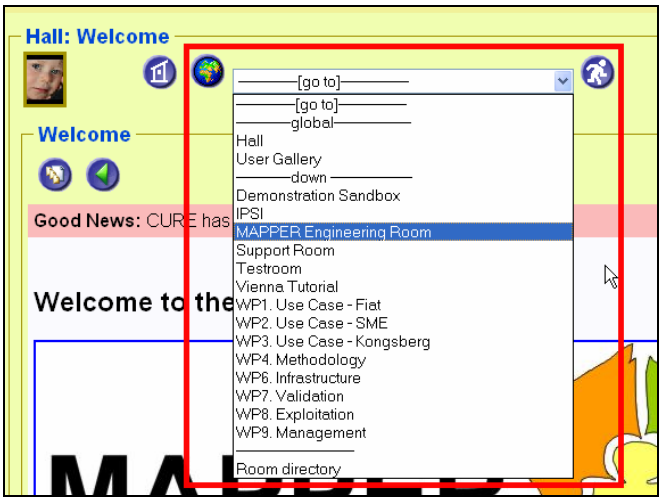

Figure 9: The navigation bar

The second way to move from one room to another is to use the room navigation bar (Figure 9). The drop down list in the centre of the navigation bar includes a list of adjacent rooms.

Selecting a room from the list will move you to this room. The button right of the drop down list is the exit button. Pressing this button will end your CURE session and point your browser to the login screen. Finally, the button left of the drop down list will open the room directory shown in Figure 10.

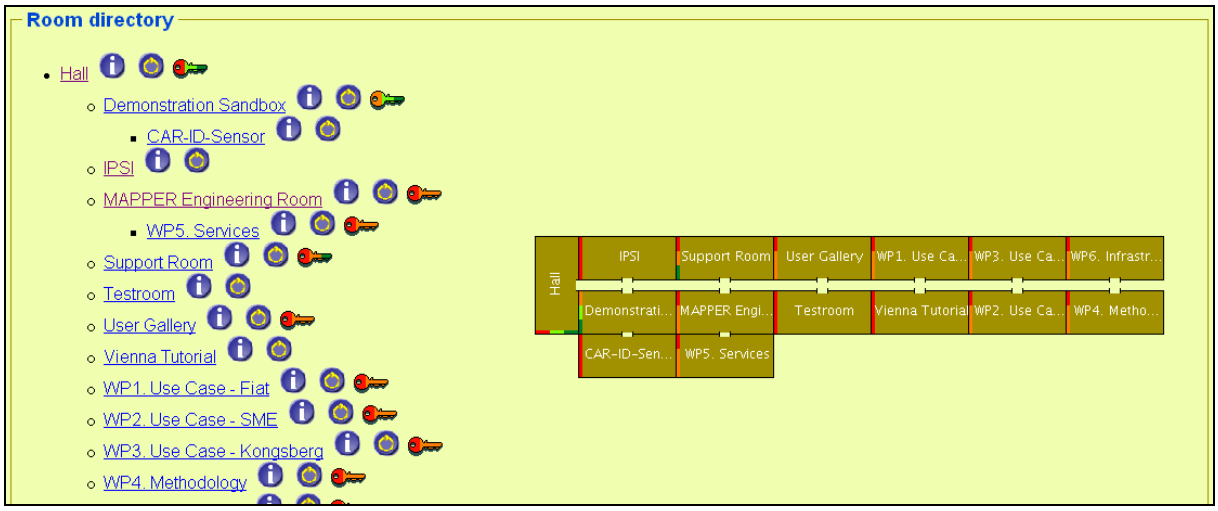

Figure 10: The room directory

The room directory lists all rooms to which you have access or can request access. This includes all rooms for which you have a personal key, that are open to the public, or that are adjacent to a room for which you have personal or public access. Rooms are shown in two ways: the textual room list lists the rooms together with the rights that you have in this room (the key), a button for getting more powerful keys, and a button that links to the information page of the room. The graphical map (right part of Figure 10) shows how the rooms are related in the collaboration environment. For now, you don't have to bother about the room rights, just try to remember that a missing key or a red bar in the graphical map indicates that you are not allowed to enter the room (the room named IPSI is an example for such a protected environment).

You can enter the room by clicking the hyperlinked room name or pressing on a room in the graphical map.

5. Edit a page in CURE

To try out the editing facilities in CURE, you may navigate to the room named "Demonstration Sandbox". Pressing the pen button on the start page will switch this page to the edit mode. You can use the WIKI syntax described in Table 1 to mark up your content.

Rendered page	The source text for the page

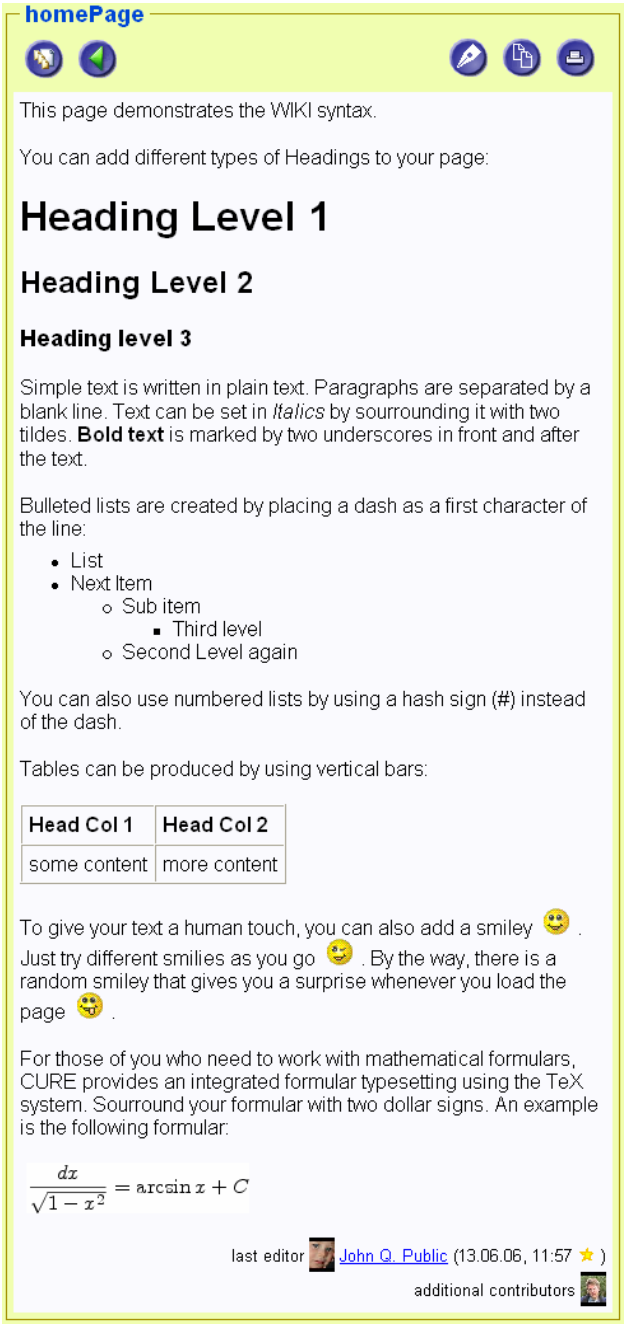	`This page demonstrates the WIKI syntax.`
	
	`You can add different types of **headings** to your page:`
	
	`! Heading Level 1`
	`!! Heading Level 2`
	`!!! Heading level 3`
	
	`Simple text is written in plain text. Paragraphs are separated by a blank line. Text can be set in ~~**Italics**~~ by surrounding it with two tildes. __**Bold** text__ is marked by two underscores in front and after the text.`
	
	`**Bulleted lists** are created by placing a dash as a first character of the line:`
	`- List`
	`- Next Item`
	`-- Sub item`
	`--- Third level`
	`-- Second Level again`
	
	`You can also use **numbered lists** by using a hash sign (#) instead of the dash.`
	
	`**Tables** can be produced by using vertical bars:`
	
	`|__Head Col 1__|__Head Col 2__|`
	`|some content  | more content |`
	
	`To give your text a human touch, you can also add a **smiley** :-). Just try different smilies as you go ;-). By the way, there is a random smiley that gives you a surprise whenever you load the page :?).`
	
	`For those of you who need to work with **mathematical formulas**, CURE provides an integrated formula typesetting using the TeX system. Surround your formula with two dollar signs. An example is the following formula:`
	
	`$$\Int{\frac{dx}{\sqrt{1-x^2}}=\arcsin x + C$$`

Table 1: The CRE wiki syntax

In addition to these rules, you can also switch the Wiki formatting temporarily off. Using three percentage characters (%%%) will switch to a **preformatted mode**. In this mode, the text will be shown exactly as it was inserted in the edit field (containing all line breaks). You can leave this mode by entering three % characters again. An example looks as follows:

```
%%%
//Source Code should not be translated ;-).
for (i=1;i++;i<10)
     System.out.println(i);
%%%
```

In the same way, you can switch to the **html mode** but now use `%html%` as a delimiter. In this mode, the text is interpreted as HTLM source code.

```
%html%
<div style="padding:5px; background-color:#fababa">
<h3 align="center">IMPORTANT NOTICE</h3>
There will be a maintanance day at the 13th of June 2005. Please don't
use the system at this day.
</div>
%html%
```

Pages can be linked. The easiest way to create a hypertext is to create the pages as you go. Imagine that you would like to link the page edited in the previous example to a page explaining links. You would append a text `**Links**` and press the save button. CURE will try to find a page named Links and create a hyperlink to this page. However, this page did not exist in the current room so far, so CURE will open the **link assistant** shown in Figure 11.

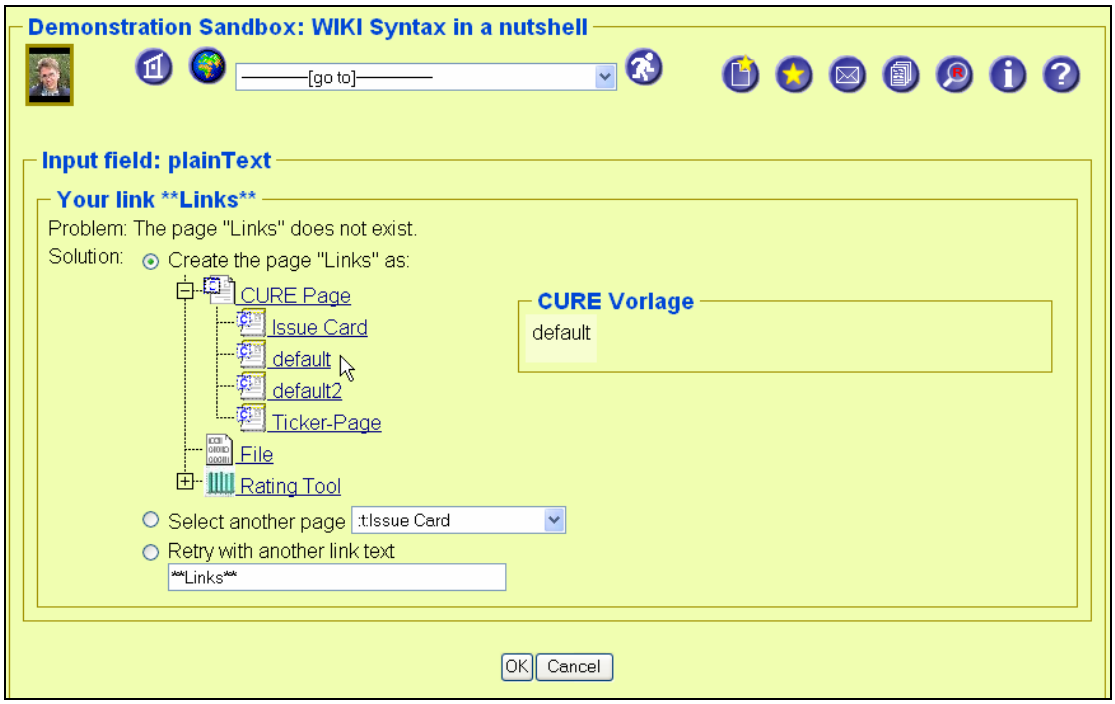

Figure 11: Link assistant

You can now decide what CURE should do. In most cases, you will accept the default selection which is the creation of a new default CURE page. You can also select a different page template (cf. section 10). In the example of Figure 11, there is, e.g., a page type named *Issue Card* that can be used to track issues in a development process. Pressing OK will create a new page named *Links*. Pressing Cancel will bring you back to the edit view of the page.

In addition to the implicit creation of pages, you can use the *Create Page* button to create a new page. This will open a dialog prompting for a page type (comparable to that shown in Figure 11).

CURE further distinguishes between **CURE Pages** and **File Pages**. CURE pages are Wiki pages that can be edited inline (as shown in the previous example). File pages are used to store any file in a CURE room. Their content cannot be edited inline but CURE pages can link to file pages to embed them in the hypertext.

There are different types of links that you can use in your documents:

Links to pages: Links to another page is the default way of links. You type the page name enclosed by two stars. If you want to link to the page *My Page*, you would need to type `**My Page**`.

Links to images: If the destination page is an image, the image will be automatically inlined. You can specify the size of the image by preceding it to the link body. If you want, e.g., to link to an image that is stored in the page *Beach*, you can insert it in the page as **Beach** which will result in a full-sized insertion or `**[100,100]>Beach**`, which will resize the image to 100 times 100 pixels. You can also leave one of the dimensions blank (put a * instead of the number). If you want to specify only the width of the picture Beach, you can use the link `**[100,*]>Beach**`, if you want to restrict the height only, you can use `**[*,100]>Beach**`.

Links to file pages: File pages are in general linked the same way as normal CURE pages. Clicking on the link will navigate to the description page of the file. You can configure the link in a way that the connected resource will be opened directly (without the intermediate info page). In this case, your link has to start with a `:d:` escape sequence. Linking directly to the file stored in the Beach file page would, e.g., be done with the link `**:d:Beach**`.

Links to rooms: You can link to other rooms by using the `:r:` escape sequence. If you want to link, e.g., to the room named *CAR-ID-Sensor* that is a child room of the *Demonstration Sandbox* room (cf. Figure 10), you can use the link `**:r:CAR-ID-Sensor**`. If you want to link to specific pages in another room, you can do so by adding the page after the room name separated by the `:p:` escape sequence. In the previous example, the link `**:r:CAR-ID-Sensor:p:Implementation**` would point to the page *Implementation* in the room *CAR-ID-Sensor*.

Links to users: You can link to any user by means of the `:u:` escape sequence. The link `**:u:John Q. Public**` would, e.g., point to the personal page of the User *John Q. Public*.

E-Mail links: You can link to any e-mail address using the `:m:` escape sequence. `**:m:cure@fernuni-hagen.de**` will create a link that opens a mail composition view for the receiver address cure@fernuni-hagen.de.

Links to external resources: You can create links to any web page by preceding the `:e:` escape sequence. `**:e:http://www.pi6.fernuni-hagen.de/~schuemm**` would, e.g., point to the internet page at the address *http://www.pi6.fernuni-hagen.de/~schuemm*.

6. Staying aware of other users

While using CURE, it is important that you stay aware of the activities done by other users. This so-called *group awareness* that Dourish and Bellotti (1992) described as *"an understanding of the activities of others which provides a context for your own activities"* is supported by CURE in different ways.

The User List

All users that are currently situated in the same room are shown in the user list of the room.

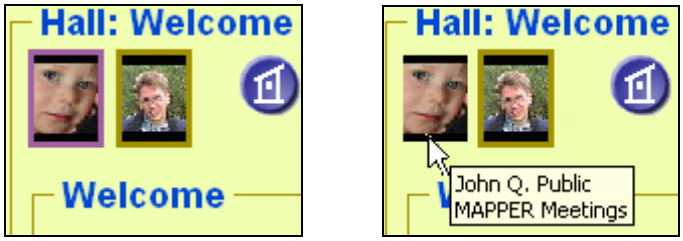

Figure 12: User list

Your own user picture will always be shown at the end of the user list. If users are at the same page, their picture will be decorated by a frame (Figure 12-left). If the users are viewing another page, their picture will not have a special frame but a tool tip will provide additional information on the user's current location (Figure 12-right shows that the user *John Q. Public* is currently looking at the page *MAPPER Meetings*).

The List of Contributors

For each page, CURE visualizes who has been contributing to the page. The last author of the page is shown with his picture and his name while the other authors are only shown with their picture.

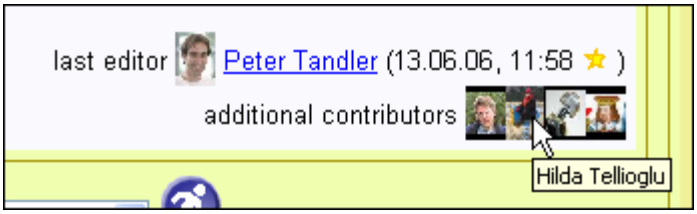

Figure 13: List of Contributors

Consider the *Welcome* page of the *Hall* as an example. This page has been edited by several users as shown in Figure 13 that shows the lower right part of the page. *Peter Tandler* was the last author of the page. Other authors include *Hilda Tellioglu*. You can use the tool tips again to find out more about the users.

The Daily Report

CURE can update you frequently on changes in the system. If activated, it well send you a daily e-mail including a list of activities performed by other users (you may have to enable the report by editing your user profile).

You can also access the information by pressing the news button (framed in Figure 14). This will open a list of all changed artifacts and describe what your colleagues did with the artifact. Note that the report does not include activities that you already saw or activities that you performed yourself. If, e.g., you had already browsed the page MAPPER Deliverables after

Peter had changed it on 12.06.2006 (cf. Figure 14), this page would no longer appear in the report.

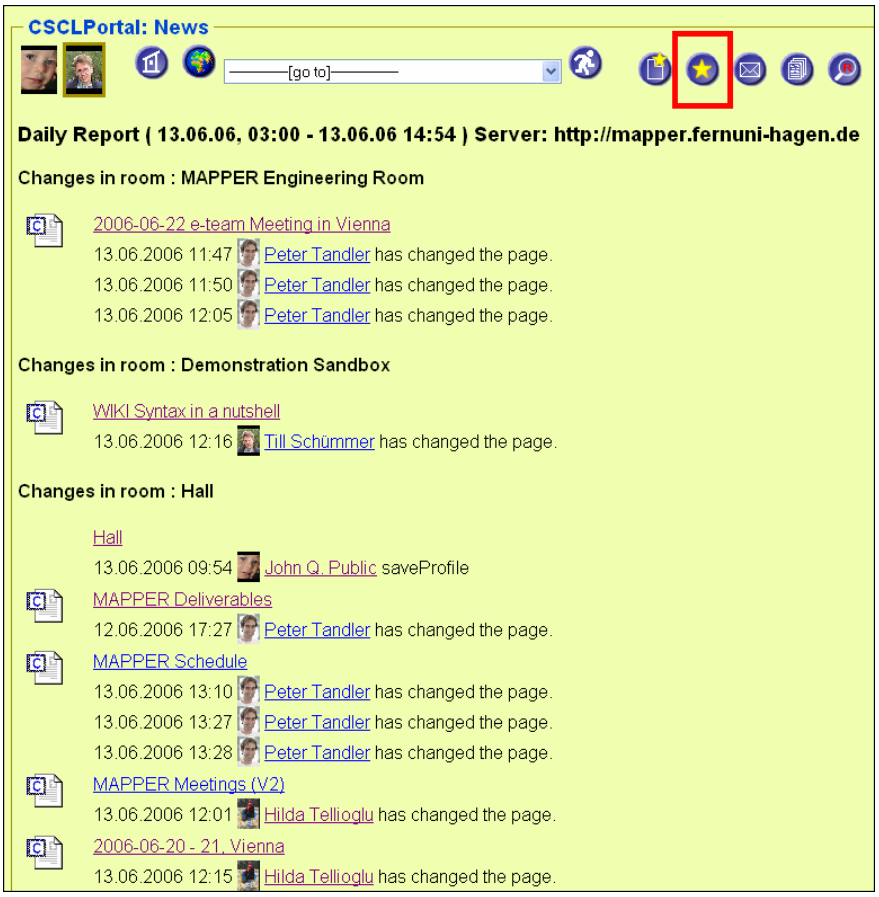

Figure 14: The Daily Report

You may wonder why more than one activity appears for each page. The reason for that is that you should be able to understand the evolution of the page. The list of all activities also helps to judge, which pages have been under massive revisions.

You can configure the report on a room basis. The required user interface for that is located in your user profile. Select or deselect rooms for that you want or do not want update notifications.

The Version History

If you want to find out more about the changes and the history of the document, you can open the page's version history by pressing the version list button (the first button in Figure 15).

Figure 15: Navigating through versions

This will open a list of versions as shown in Figure 16. The list shows who has edited a specific page at what time. Clicking on the title of the page will navigate to the selected version directly.

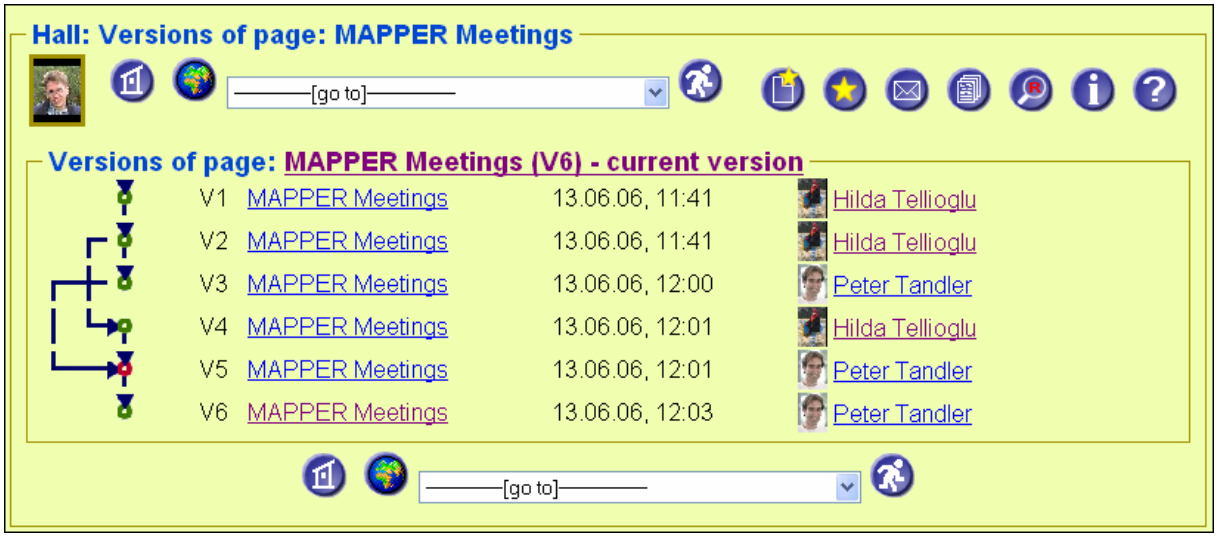

Figure 16: A Version History

7. Versioning

CURE is based on the idea of Wikis. The term Wiki was coined by Ward Cunningham after the "wiki wiki" or "quick" shuttle buses at Honolulu Airport. This means that all users should be encouraged to quickly edit pages and contribute to the community's evolving knowledge. However, this has the danger that users by accident or intentionally damage a page. Therefore it is common to Wikis that all pages are versioned. Whenever you save the page, the system creates a new version. This version is shown when the page is requested by a user.

As mentioned in the previous section, you can browse through the versions using the version navigation buttons (the three right buttons of Figure 15) or by looking at the list of versions. You can also make an old version of the page the current version. Pressing the second button (from left) in Figure 15 would make *V3* the current version of the page *Mapper Meetings*.

Figure 17: A version conflict

When two users edit the same page at the same time, there can be a version conflict. The version history of Figure 16 shows one conflict at version 5. Actually, the user Peter looked at the page in version 2 and started to edit it. Hilda also looked at version 2 and started to edit it. At 12:00, Peter saved his changes and thereby created version 3. One minute later, Hilda also saved her changes creating version 4. since both version 3 and version 4 were created from version 2, they are considered on par. But what would happen if a user now wants to see the most recent version of the page? In CURE, this cannot be decided and therefore both versions are shown together in a merge version (version 5 - Figure 17).

It is up to the author to resolve the conflict manually. To do this, press the edit button of the page and look at the source.

```
==VERSION==4
- **Vienna Meeting 20.21.6.2006**
==ENDOFVERSION==4

==VERSION==3
On this page, all overall MAPPER meetings are listed; the work packages
have their own list of internal meetings.

- **2006-06-20 - 21, Vienna**
- **2006-04-03 - 04, Turin, Review**
- ...
==ENDOFVERSION==3
```

It shows both versions of the page. To merge these versions, you should manually compare the text and keep what's best. The version delimiters (e.g., `==VERSION==4`) have to be removed before you save the page again.

8. Communicating with others

Up to now, we have discussed two of the three Cs in the 3C classification of groupware systems: Cooperation and Coordination. We will now look at the third C: Communication.

Each CURE room can have a built-in mailing list. Whether or not a concrete room has a mailing list depends on the room properties that the creator of the room defined when creating the room (or later on - more details on creating rooms will follow in the next chapter). To access the mailbox, press the button with the small envelope.

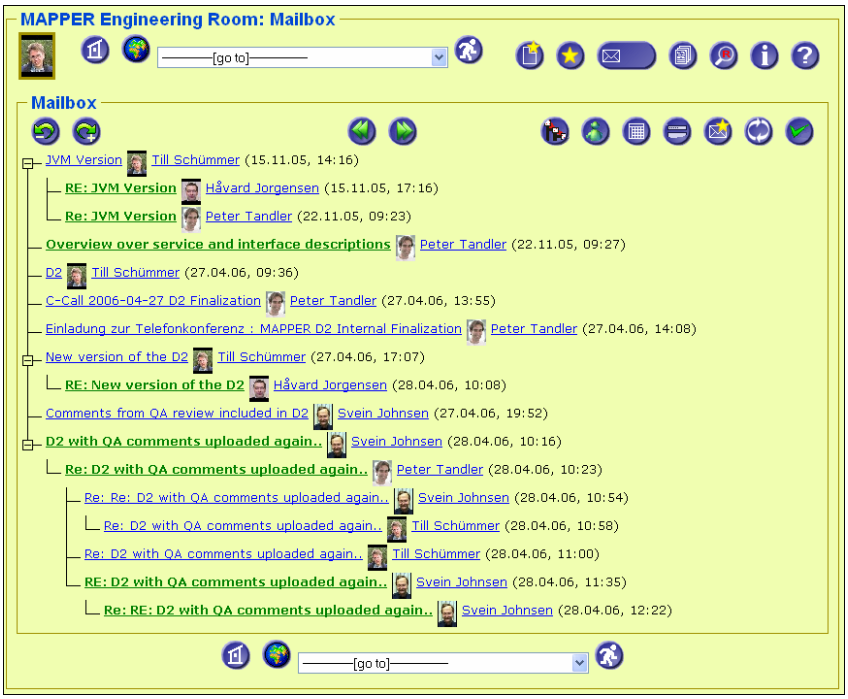

Figure 18: A room's mailbox

The mailbox overview (Figure 18) lists all messages that were written to the room. Messages are threaded which means that you can see which message was sent as a reply to another message. Clicking on the message opens the detail view for the selected mail (Figure 19).

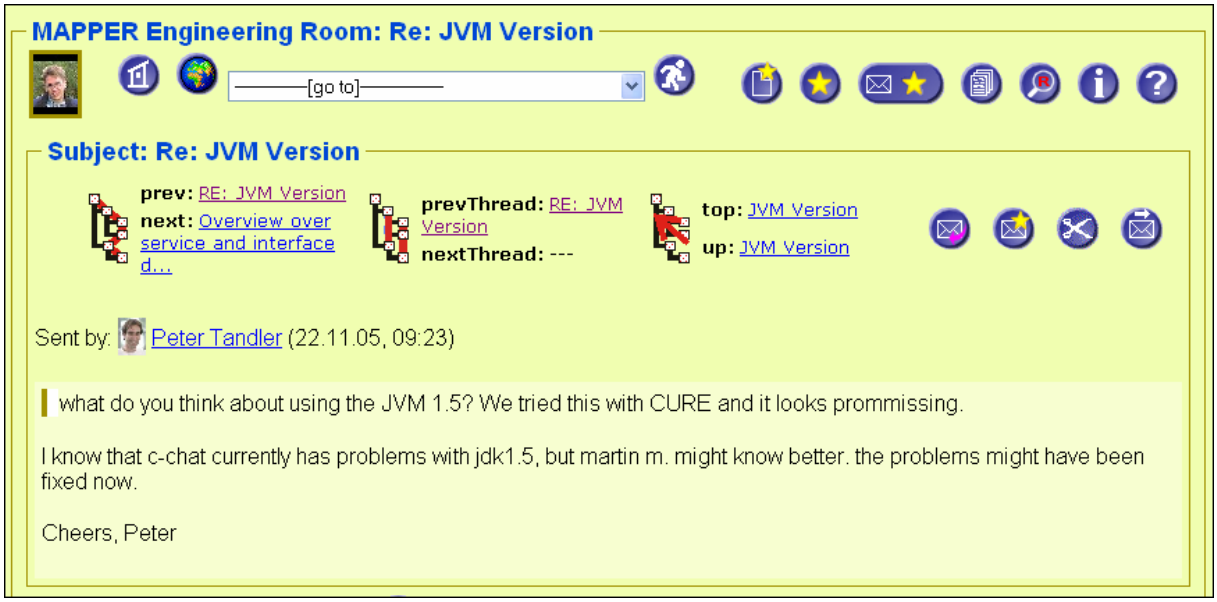

Figure 19: A detail view of a mail

You can navigate the threads in the mailbox using the navigation links.

Navigation section	Description
	Navigate to the next or previous message in the message thread. This will traverse the tree, as it is shown in the mailbox overview (Figure 18).
	Navigate to the next or previous thread on the same level. As an example, look at the message "Re: Re: D2 with QA comments uploaded again.." written at 10:54 in (Figure 18). Moving to the next thread would navigate to the part of the message tree written at 11:00 by Till.
	Navigate to the parent message or the top parent.

Table 2: Navigation links

To write a new mail, you can either reply to the currently shown e-mail () or create a new thread (). Mails are also Wiki pages which means that you can use all ways of linking that can be used for normal pages. This is especially useful when linking to pages of the wiki. If you want to point the room users to the agenda page of a meeting room, you could, e.g., send the following mail.

```
Dear all,

please find the preliminary agenda in CURE at the page named **2006-06-20
- 21, Vienna** and send your comments or modify the agenda in CURE
directly.

Thanks and regards

Hilda
```

After saving the mail, it will be stored in the room's mail box and delivered to the users who activated e-mail-receiving for the room in their profile. Note that all room users will be informed about the mail in the daily report.

You can check who is receiving the mail for a specific room by looking at the room info page (accessible by means of the  button). Those users having a small coloured letter next () to their user picture will receive the mail in their external mailbox.

When receiving a mail from a cure room in your external mail client, you can also use this client to reply to the room. Simply press reply and leave the last part of the subject line unchanged (the number in the square brackets). This number is required by CURE to assign the mail to the right thread. Please note that your reply will be sent to a CURE mail address (e.g., 6@mapper.fernuni-hagen.de). The mail address has the room number as a local part and the server address of the CURE server as the server part.

You can also add attachments to the mail. These will be converted to file pages in CURE and not directly sent to the users. Instead, CURE will insert a link to the attachment at the end of the mail.

When using your external mail client, you should avoid to send mail footers since these often include characters that are hard to parse by the CURE system (the mail will look ugly in the web-based mail archive). You should also -- if possible -- send the mail in plain text. This will give the best results.

9. Creating your own room and inviting members

Before we will go into details of how to create and administrate rooms in CURE, it is now time to revisit the concept of virtual rooms:

Normally, you will start your collaboration in a room that someone else (probably the person who invited you to collaborate in the CURE system) created for you. However, there might also be the need for creating a new room. The basic rule for creating a new room is that it should discuss a different topic than the existing rooms and/or involve different people.

After you identified the need for creating a new room, you should think about a place where this room will be located. Again, this depends on how accessible the room should be. If the room is of general interest to all users of the system, it makes sense to create them as an adjacent room to the entry Hall. In cases where the room is used by a sub-team of an existing team, you should place the room next to the collaboration space of the existing room. Finally, if you want to create a room for a newly initiated team and decide to keep this room hidden from the public, you should create it as a sub-room of a private room.

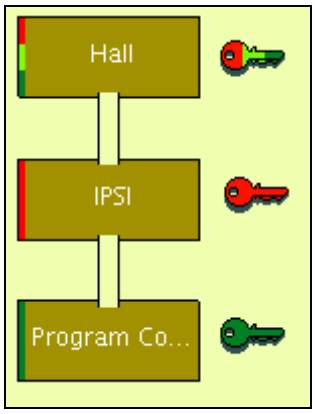

Figure 20: An example room structure

As an example consider the room structure in Figure 20: The users created a room for the group collaborating at the IPSI institute. This room is private but should be reachable from the Hall. The reason for this is that people interested in the IPSI should see that it has a room in the CURE system. However, the access to this room is restricted to members of the IPSI. In the example, the user does not have access to this room (which can be seen by the red key).

A sub-team of the IPSI created a room for a *Program Committee* of an event hosted in the IPSI. The reason for placing this room next to the IPSI room is that it should only be visible to members of the IPSI. Strangers using the CURE system will only see a locked IPSI room and cannot explore the room structure that is inside the IPSI room. But as it can be seen in the Figure, there is an exception to this rule. If a user decided to pass on a key for the *Program Committee* to a user who is not member of the IPSI room, this user will be able to see the *Program Committee* room. He will be able to enter the room but he will not be able to navcigate to the room using the room navigation facilities (navigating from the Hall to the IPSI room and moving on to the *Program Committee* room). You should therefore remind such users to set a bookmark to the newly created room.

Creating and deleting a room

To create a room, you have to move to the desired parent room and open the information page of the room (using the room's -button). Depending on the rights that you have in the room, you will see a subset of the following buttons:

Button	Function
	Go to the room's start page. This is equivalent to using the -Button in the navigation bar.
	Create an adjacent room. The system will create a new room, link this top the current room and show the new room's property page.
	Delete the room. This will remove the room and all other rooms that are sub-rooms of the deleted room. All pages in the deleted room will be lost. There is no way to restore the pages after the room has been deleted (actually, your system administrators could find a way to restore some content but this will cost them a lot of effort and you some drinks to the system administrator). So be careful when using this operation.

Table 3: Room manipulation buttons accessible on a room's info page.

Understanding the room properties

Now that you have seen how to create rooms, you have already seen the property page of a room. This page provides additional information on the room's capabilities.

General buttons and navigation facilities that can be found in all CURE views.
Properties of the room including a description. If you have the appropriate rights, you will also find here an edit button ( ) that can be used to edit the room's description. Note that WIKI syntax can be used here
This section lists the users who can interact in the room. In the upper section, user groups are shown such as the group of all registered users that includes all users who have an account in the system. The lower part shows individual users.
A room can provide a set of page templates. How these are created and used will be discussed in chapter 10.
Finally, a room can offer different features that will be discussed in the remaining part of this section.

Figure 21: The property page of a room.

Mailbox: If activated, the room will provide mail communication facilities as described in Chapter 8.

Mail delivery to users of the room: If activated, the mails can be delivered to the user's external mail accounts. This normally speeds up communication since users are more often checking their mail than they check the room in CURE for new messages.

Allow anonymous posts: If activated, the CURE system does not verify the sender of the message. Consider, e.g., a room in which two people, Alice and Bob, collaborate. If the option of anonymous posts is deactivated, only Alice and bob are allowed to send messages to the room using their external mail client (cf. Chapter 8). When Alice sends a mail from their external client, the system checks if the sender address of Alice's mail is the same as the mail address that Alice provided in her user profile (cf. Chapter 3). If the address matches, the mail is assigned to Alice. If not, it is rejected. Activating the *"Allow anonymous posts"* option will prevent the rejection of mails. The unrecognized mail will be added to the mailbox and a dummy sender will appear as sender of the mail (called *Mail System*).

Chat: You can activate an internal chat in each CURE room. The chat is persistent and appears on the bottom of each page that is located in the CURE room.

Awareness: You can decide if users should see who else is currently present in the same room. If the Awareness option is activated, co-present users are shown in the user list as described in Chapter 6.

Inviting other users to the collaboration space

As we mentioned before, the rights that a user has in a collaboration space are defined by means of virtual keys. If you have an appropriate key, you can create copies of the key or pass your key on to other users. The key management features can be accessed by visiting the room's info page. Depending on your rights in the room, the following buttons will become visible:

Button	Function
	*Create a new key using your own keys as master keys.* The system will show a sequence of forms in which the rights and the receiver of the new key can be defined. The new key can be equipped with as powerful rights as the key from which it was copied. If you, e.g., have a key that allows you to copy the key and to create an adjacent room, you can create a key with the rights to return the key and to enter the room. More details on rights will be provided right after this table.
	*Pass the key on to another user.* The system will show a dialogue that allows you to select the key that you want to pass on and a receiver. This can, e.g., be used to transfer the administrator key (green key) for a room to another user.
	Delete the key. As in the previous cases, the system will show you a form i which you can select one of your keys. The selected key will then be destroyed. Note that you can only destroy a master key if another user still owns a master key for the room.
	Give back the key. The selected key will be added to the room's set of free keys.
	Take a free key. The key that was placed on the room can be taken by any user pressing the green door knocker button. Free keys are an easy way to manage rooms with a limited number of participants. Imagine that you want to assign team members to work groups and that you want to have three groups with four members each. If it is not important who is participating in which group, you can create three rooms and place four free keys at each room.
	Retract a key. Users holding a master key can retract keys held by other users in the room. This is important when a user should be excluded from a group (e.g., because

	of inadequate behaviour).
	Request a key. If your key does not have sufficient rights to perform an operation, you can request a key. The system will show you the dialogue for defining your desired key rights and ask you for a reason why you want to have the key. Your arguments will then be passed on to all users who have an administration key (a green key). These administrators can then decide whether or not you will receive the desired key.

Table 4: Key manipulation buttons accessible on a room's info page.

As mentioned above, the system will prompt you to define key rights and other options for the key. The following illustrations capture these dialogues. You can switch between the different tabs by pressing the next button or by clicking on the captions of the tabs. Note that some tabs will only be available if the previous dialogues were filled appropriately.

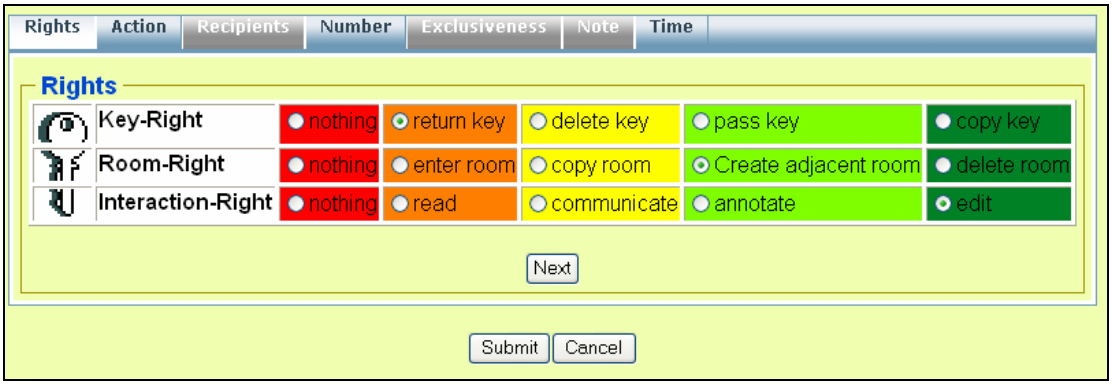

Figure 22: Selecting the appropriate rights.

The first step in the key definition process shows a matrix of available rights in which you can select the desired right.

As shown in Figure 22, CURE offers three different classes of rights: **key-rights** defining what the user can do with the key, **room-rights** defining whether or not a user can enter a room or change the room structure, and **interaction-rights** specifying what the user can do in the room. Rights are ordered from left (most restrictive) to right (less restrictive). A less restrictive right always *includes* all more restrictive rights. A key offering maximal rights in all three classes is called a *master key* for the respective room.

Key-rights specify what the key holder may do with his key. If the key holder does not need his key for the room anymore, he may return the key to the room the key belongs to, delete his key, or pass it to another person. If the user wants to keep his key but wants to propagate his key to other users, the key needs to have the key-right “copy key”. Having the “copy key”-right, the user can create copies of this key with potentially restricted rights and give this copy to other users. To grant access rights only for a certain time frame, keys may be equipped with *temporal restrictions*, e.g. being valid from a certain point in time until a certain point in time.

Room-rights of a key specify what the user can do with the room the key matches. Users may enter a room if the key comprises the “enter room”-right. They need the "copy room"-right to create a second room that has the same content as the original room (using the -Button on the room's info page) or may create an adjacent room to the room the key matches, if the key features the “create adjacent room”-right. Finally, they may delete the room if the key offers the “delete room”-right.

The **interaction-rights** granted with a key are applicable uniformly to all documents in the room. You can define whether a user can see the documents and the content (read), participate in the room's communication, or modify the room's pages.

After selecting the desired right and pressing the next-button, you can select one of three actions:

1. **Keep them as own keys** so that you can pass them on later,

2. **Place them as free keys in the room** so that other users can grab the key and join you in the room, or

3. **Pass them on to other users**.

The third option is most frequently used since it is the way how to invite other users to a collaboration space. The system will present the list of users shown in Figure 23. You can select one or more "normal" users from the left-hand list. To select more than one user press the Ctrl-key and click on the user's name. If the list of users is long, it can be helpful to search for the specific user. You can enter any part of the user's name and press the filter button. All unselected items of the list that do not include the provided search term will then be hidden. To show all users again, simply remove the search term and press filter again.

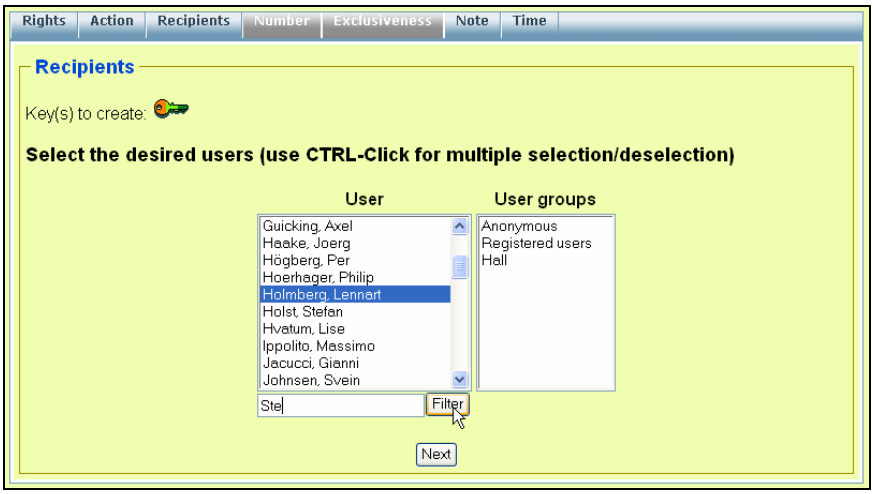

Figure 23: Interface for selecting the receivers of a key.

Besides the "normal users", you can also select user groups. These are defined either by properties of the user's account (whether or not they registered an account) or by membership in other rooms. In Figure 23, there is, e.g., the user group named Hall. This group includes all users who have a personal key for the room Hall. By using the room keys, you can thus base your access rights on the rights provided in the parent room.

Please note: Room keys still have a small technical problem which means that the system may refuse to issue a room key. This problem will be fixed in the next release of CURE.

The additional tabs can be used to limit the validity of the key to a specific date range and to add a note for the receiver of the key.

10. Tailoring page types

Until now, we have shown how you can create pages in your collaboration space and link these pages to a collaborative hypertext. Using these techniques, you could for instance create a set of introduction pages in which you ask your team members to provide some personal information on their role in the group.

One way to support this is that you as a room owner create a first page and ask all other team members to copy this page and add their personal information. Such a page could have the following content:

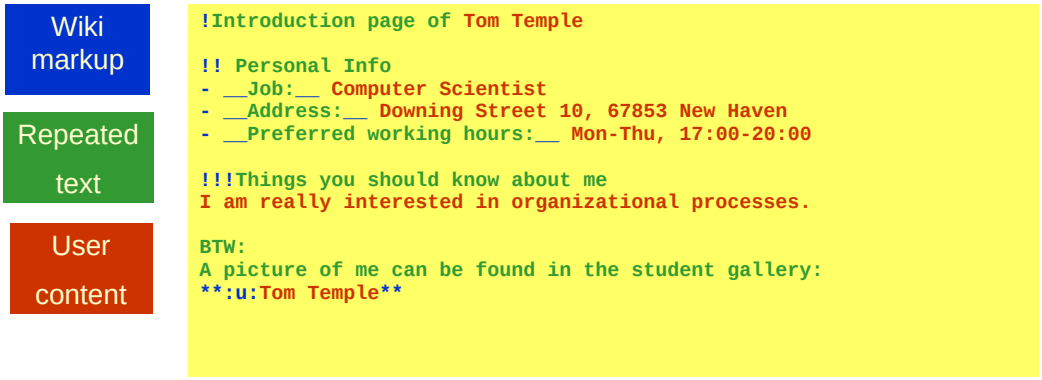

Figure 24: Using a seeding page to collect structured pages.

Such an approach is often references as *Seeding Page Approach* (cf. Leuf and Cunningham, 2002). Although this approach is quick and may work for disciplined team members, it does not ensure that the structure of the introduction pages stays the same for all users. Especially, it does not reflect structure changes for pages that were already created. Imagine that you as a room owner noticed that an important bit of information (e.g., an instant messaging address) was missing in the initial seeding page. You could add this information to the page but this will not affect the page structures of all pages that were created so far.

The problem is that the seeding page combines structural information (repeated text) with formatting information and content provided by the user. The formatting information and the structural information should stay consistent for a larger collection of pages while the content provided by the user can vary for each page.

The Concept of Page Templates

As a solution to this problem, CURE introduced the idea of page templates. It distinguishes between a display template that defines how the content provided by the user should be shown to other users an edit template that defines input fields for manipulating the content. Applied to the abovementioned example, the first part of the templates would look as follows:

Figure 25: Templates and pages in CURE.

As you can see, the edit template contains of a normal wiki page that is enriched with special xml tags (the text embraced by <>). The most important field of such a template is the <wikiTextInput> tag. This is used to create a placeholder for user content. As a template designer, you have to provide this tag with an id. This id will be used to store the user content in the actual page instance.

The third line of the example edit template, e.g., defines a *<wikiTextInput id="pageTitle" />*. This will create an input field for the field named *pageTitle*. When saving a page, the content is stored in the fields of a page. The example defines four different fields inthe page (pageTitle, job, address, and hours). When displaying the page, these fields are inserted at the corresponding place holders in the display template. For the example, this would mean that the page in combination with the display template would produce a page comparable to that of the seeding page shown before.

Creating a new Page Template

In Figure 11, we have already seen that CURE allows the user to create pages of different types. As a room owner, you can extend the set of available page types so that the room can support different domain-oriented representations of wiki pages. First navigate to the room's information page (using the room's -button) and then press the new template button as shown in Figure 26.

Figure 26: Creating a new CURE template.

You will see a form that prompts you for the name of the template, the display template, and the edit template (Figure 27).

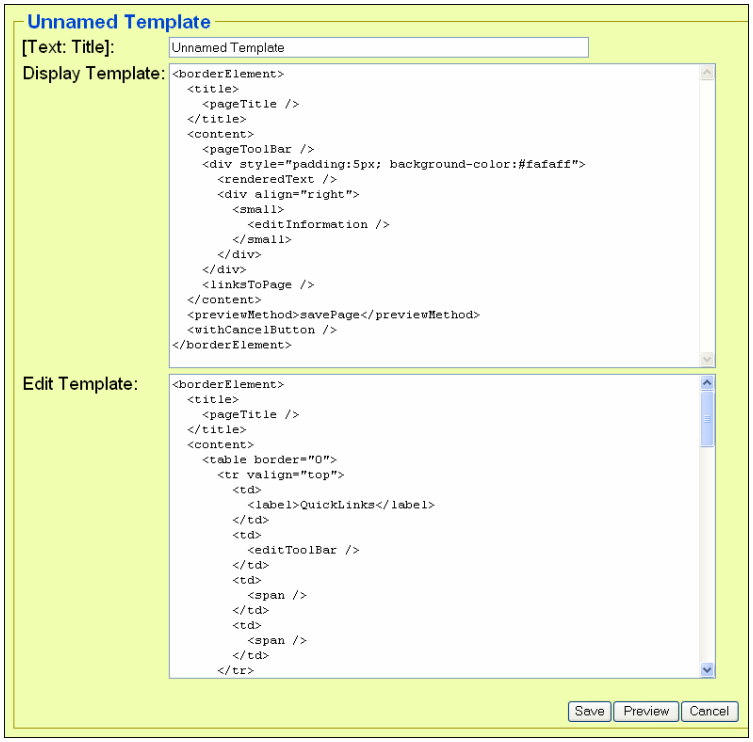

Figure 27: Input form for entering templates.

After saving the template, you can create new pages using this template or change the templates of existing pages so that they are adapted to the new template (only for page templates that allow a later template change).

The Template Markup Language

As mentioned before, you can use all wiki formatting rules in templates as well. In addition, you can add template commands (in brackets). This section provides an overview on available template commands.

Commands for editing and showing content

Command	`<renderedText id="..." />`
Purpose	Show the content of the text field identified by the id attribute. The content is rendered according to the WIKI syntax.
Attributes	**`id`** The name of the field that should be shown. If no id attribute is provided, the default text field is used. Several pre-defined fields exist in every page: */pageTitle* The name of the page. *description* The page's description. This is especially used in file pages but all other pages can also have a description. *lastEditTime* The time at which the page was saved.
Use in	Display Template
Example	`<renderedText id="hobbies" />`

Command	`<unrenderedText id="..." />`
Purpose	Show the content of the text field identified by the id attribute. The content is directly pasted to the page without applying any rendering.
Attributes	**`id`** The name of the field that should be shown. If no id attribute is provided, the default text field is used. See `<renderedText ... />` for additional information on predefined fields.
Use in	Display Template
Example	`<unrenderedText id="job title" />`

Command	`<wikiTextInput id="..." />`
Purpose	Create an input field for an element of a page.
Attributes	**`id`** The name of the field that can be edited in the input area. If no id attribute is provided, the default text field is used. See `<renderedText ... />` for additional information on predefined fields but notethat the lastEditTime field cannot be edited using a `wikiTextInput`. **`lines`** Number of lines that are reserved for the input field. This affects the visual layout of the input field, not the potential length of the content. If no lines attribute is provided, CURE creates an input field with one line. **`cols`** Number of columns that are reserved for the input field. The use of this attribute is comparable to the lines attribute. The default value is 70 columns. **`template`** If provided, this attribute controls the default selection of page types for pages that will be created with the link assistant. Using this attribute, you could, e.g., define that links in the input field named *hobbies* will by default link to pages that are based on the *hobby* page template.
Use in	Edit Template
Example	`<wikiRenderedText id="hobbies" lines="5" cols"80" template="hobby" />`

Command	`<wikiAppendText id="..." />`
Purpose	Create an input field that can be used to add content to a field. The behavior of the field is comparable to that of a log book. Whenever a user enters some text in this field, this text is appended to the text field referenced by the id tag. Application areas are BLOGs or Commenting tasks. Note that each display template may only have one `wikiAppendText` field. Only users who have the right to comment pages will see this field.
Attributes	**`id`** The name of the field to which text should be appended. **`lines`** see `<wikiRenderedText ... />` **`cols`** see `<wikiRenderedText ... />` **`rememberAuthor`** If set to "yes", the author of the comment is stored in the text as well. This can be suppressed to allow anonymous comments by setting the attribute to "no" (note that the author history will still show the author, which means that you would need to allow the anonymous user to comment pages in the room if you really want to have anonymous comments).
Use in	Display Template (may not be used in an edit template)
Example	`<wikiAppendText id="hobbies" lines="5" cols"80" rememberAuthor="yes" />`

Command	`<wikiSelectUser id="..." />`
Purpose	Creates a selection list in which users can be selected. The id of the user is stored in the field identified by the id. The set of users from which one user can be selected can be restricted by providing filter attributes as described below. In general, the filters control which rights users need to have to appear in the list of potential users.
Attributes	**`id`** The name of the field in which the user's id should be stored. **`scope`** The name of a room. If present, the attribute uses the provided room to find users who have access to this room. By default, this value is set to `"room:."` which references the current room. If the attribute would, e.g., be set to `"room:.."`, all users of the parent room would be listed. If it would be set to `"room:MyChildRoom"`, all users who have access to the room named `myChildRoom` would be shown. **`keyRights`** Only users who have a specific key right are shown. Possible values are: "0"=no rights, "1"=return key, "2"=delete key, "3"=pass key, or "4"=copy key **`roomRights`** Only users who have a specific room right are shown. Possible values are: "0"=nothing, "1"=enter room, "2"=copy room, "3"=create adjacent room, or "4"=delete room **`interactionRights`** Only users who have a specific interaction right are shown. Possible values are: "0"=nothing, "1"=read, "2"=communicate, "3"=annotate, or "4"=edit **`restrictToPersonalKeys`** If set to "yes", only users who show up personally in the room's information page are considered. If set to "no", also users who have access to the room by means of a room key or a public key are shown.
Use in	Edit Template
Example	`<wikiSelectUser id="main editor" interactionRights="4" />`

Command	`<wikiShowUser id="..." />`
Purpose	Visualizes a link to a user stored in the field identified by the passed id. This tag corresponds to the `<wikiSelectUser ../>` tag. The Link contains the user's name as well as a small picture of the user (this is the same visualization as it is created by `**:u:Name**`).
Attributes	**`id`** The name of the field in which the user's id is stored.
Use in	Display Template
Example	`<wikiShowUser id="main editor" />`

Command	`<wikiSelectPages id="..." />`	
Purpose	Creates a selection list of available pages. The result is stored as a wiki link to the selected page (i.e., as `**My Page Name**`) in the field identified by the passed id. This means that a link to the page can be shown using the `<renderedText ... />` tag with the corresponding id. The set of shown pages can be restricted using the attributes shown below.	
Attributes	**`id`**	The name of the field in which the link to the page is stored.
	`search`	A search string that will be used to find the pages that are shown in the list. The string can include a `*` as a wildcard at the beginning or at the end of the string. If the room contains the three pages `"Till's page"`, `"Till's ideas"`, and `"Liam's page"`, a search phrase `"Till*"` would find the first two pages, while a search phrase `"*page"` would find the first and the last page. By default the search string is set to `"*"` which means that all pages are listed.
	`case`	Defines if searches are case sensitive. If set to `"yes"`, the pages' names have to match the case. The default value is `"no"`.
	`folder`	Restricts the search for pages to the specified folder. A "*" wildcard can be used to include any subfolder. folder="/resumes/*" would, e.g., restrict the search to pages that are stored in the resumes folder or any of its subfolders.
	`sort`	Defines a sort criteria for the pages. By default, pages are sorted by name. Other options are: "1"=sort by date, "2"=sort by author, and "3"=sort by page type
Use in	Edit Template	
Example	`<wikiSelectPages id="specification" folder="/specifications/*" sort="1"/>`	

Command	`<templateSelector />`
Purpose	Creates a drop-down list from which a user can select the desired page type of a specific page.
Attributes	**`---`**
Use in	Edit Template
Example	`<templateSelector />`

Command	`<folderSelector />`
Purpose	Creates a selection list with available folders. The content of the list shows the folder in which the page is currently located. If changed, the page is moved to the new selected folder (cf. Chapter 11).
Attributes	**`---`**
Use in	Edit Template
Example	`<folderSelector />`

Command	<wikiSelectString id="..."> <option title="..."> <option title="..."> </wikiSelectString>
Purpose	Creates a customized selection list. The selected value is stored in the field identified by the passed id. Possible options are provided in the option tags.
Attributes	**id** The name of the field in which the selected text is stored.
Use in	Edit Template
Example	<wikiSelectString id="priority"> <option title="*" /> <option title="* *" /> <option title="* * *" /> </wikiSelectString>

Commands for sending content to the server

Command	<submitButton />
Purpose	Creates a button used to store the page.
Attributes	---
Use in	Edit Template
Example	<submitButton />

Command	< cancelButton />
Purpose	Creates a button that will terminate the edit process and return to the formatted version of the page.
Attributes	---
Use in	Edit Template
Example	<cancelButton />

Command	<resetButton/>
Purpose	Reverts the fields of the form to the values that were present when the user started editing the page.
Attributes	---
Use in	Edit Template
Example	<resetButton />

Command	<previewButton method="savePage"/>
Purpose	Shows a formatted version of the page without actually changing the page on the CURE server. This can be used to avoid storing a page with formatting errors.
Attributes	---
Use in	Edit Template
Example	<previewButton method="savePage"/>

Commands for polishing up the formatted version of the page

Command	`<box style="...">`
	`  <boxTitle>...</boxTitle>`
	`  ...`
	`<box>`

Purpose	Creates a rectangular area in the page that can contain any content. If the <boxTtle> tag is provided, the content of this tag is used as a title of the area.
Attributes	**`Style`** A string containing a HTML style attributes list. By default, this is set to "border-width:1px; background-color:#ffffff; border-color:#000000".
Use in	Display Template

Example	`<box style=" border-width:1px; background-color:#d6d6f1">`
	`  <boxTitle><unrenderedText id="pageTitle" /></boxTitle>`
	`  <renderedText />`
	`<box>`

Command	`<linksToPage />`
Purpose	Show a list of pages that reference the current page
Attributes	`---`
Use in	Display Template
Example	`<linksToPage />`

Command	`<pageToolBar />`
Purpose	Show buttons for navigating between page versions and for manipulating the page.
Attributes	`---`
Use in	Edit Template
Example	`<pageToolBar />`

Command	`<editInformation />`
Purpose	Show the last author, the time of the last change, and a list of pictures of additional authors.
Attributes	`---`
Use in	Edit Template
Example	`<editInformation />`

The Default Template

Now that we have discussed the possible commands in a wiki template, we will show the default template as an example. Figure 28 shows the display template that is used to visualize default pages.

Figure 28: The default display template.

It displays the page's title and the page's content as a rendered wiki text. Note that the template makes use of the HTML div-tag in order to right-align the edit information.

The edit template is almost as simple as the display template (Figure 29). As in the display template, the border of the page is defined using two nested box commands. In the inner box, the content is arranged using a wiki-formatted table. The left column provides labels for the input fields. The fields are generated in the right column.

Finally, at the end of the page, a collection of buttons is inserted (as a centred line).

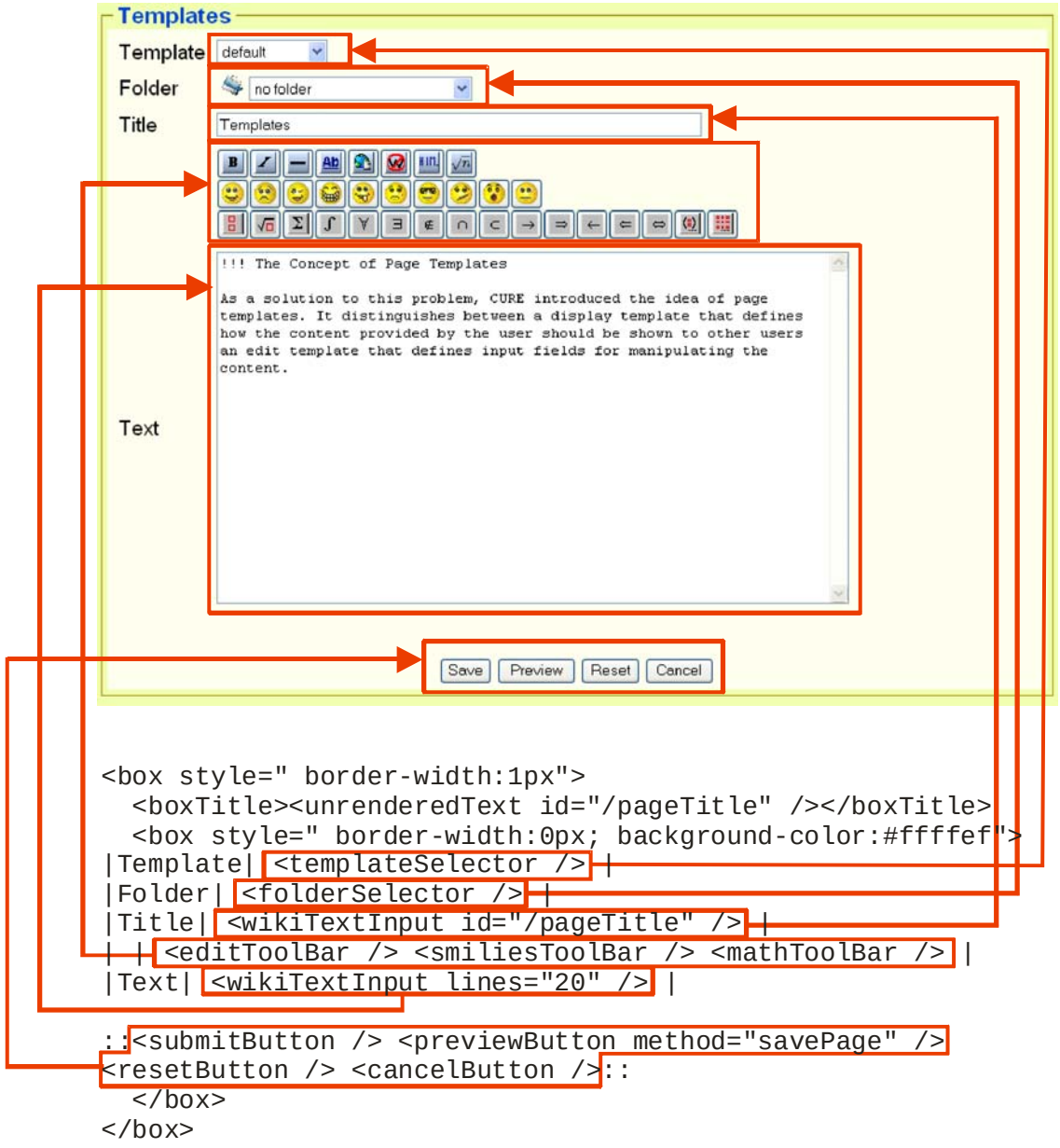

Figure 29: The default edit template.

Inheritance of Templates

Templates are inherited from a parent room to a child room. If your room is created as a sub-room of the Hall, it will by default provide all templates that exist in the Hall. If you add new templates in your sub-room, these will only be visible in your sub-room and in all sub-rooms of the sub-room. When looking for a template (for displaying or editing a page), CURE first looks in the page's room. If it does not find a matching template, it looks in the parent room. This is recursively continued until the template is found. If even the Hall does not provide a matching template, then the default template is chosen for visualizing the page.

The list of available templates is shown on the room's information page in the section labelled Activated Page Types (Figure 30).

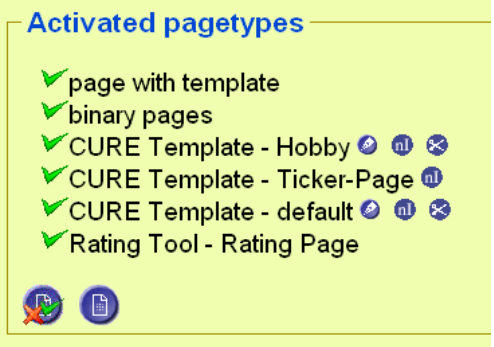

Figure 30: Activated Page Types

You can control the availability of templates in the following ways:

1. **Overriding a template:** By pressing the button, the template is replaced with a new template. This means that all pages that were using a template from the parent room will now use the new version of the template.
2. **Removing a template:** Templates can be removed in the same way as pages (by pressing). If the removed template did override a template from the parent room, the pages will again rely on the template of the parent room.
3. **Blocking a template:** By pressing the button, you can hide a template that was inherited from a parent room. The system will present you a list of all possible page types and you can add/remove templates that should be hidden.

11. Searching and organizing content

Traditionally, wikis keep the organization of content simple. This means that all pages can be accessed by only using the page's name. While this is true for CURE as well, CURE adds an additional organization mechanism as it is known from file systems. Pages can be placed in folders. Placing a page in a folder makes it easier to find that page when using the list of pages view (accessed by the button).

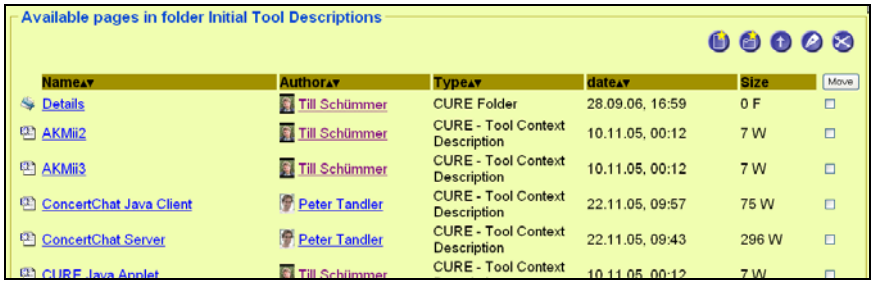

Figure 31: A folder in CURE.

The list shows the pages and folders organized in different columns. The first four columns can be used to sort the pages and folders. The size of a page is shown according to its type. If it is a wiki page, CURE shows the number of words that are on the page. In case of a folder, CURE shows the number of contained files. Finally, if it is a File page, the size represents the file size in bytes.

Pages can be moved to other folders by selecting the page (using the checkbox in the last column) and then pressing the move button. CURE will prompt you for a destination folder and move the pages.

Additional actions can be performed by using the document list buttons:

Button	Action
	Create a new page in the currently shown folder.
	Create a new folder in the currently shown folder.
	Show the content of the parent folder.
	Edit the folder (rename it).
	Remove the folder and its content.

A second way for finding content is to use the full-text search of CURE. Pressing the -button will open a search form in which different fields of the pages can be searched. You can limit the search to the current room or search the whole system.

References

Dourish and Bellotti (1992). Awareness and coordination in shared workspaces. In: Conference proceedings on Computer-supported cooperative work, pp. 107-114, ACM Press, New York.

Greenberg, S. and M. Roseman (2003). Using a Room Metaphor to Ease Transitions in Groupware. In M. Ackermann, V. Pipek, and V. Wulf, editors, Sharing Expertise: Beyond Knowledge Management, pages 203–256. MIT Press, Cambridge, MA, USA.

Haake, J. M, A. Haake, T. Schümmer, M. Bourimi, and B. Landgraf (2004b). End-User Controlled Group Formation and Access Rights Management in a Shared Workspace System. In CSCW ’04: Proceedings of the 2004 ACM conference on Computer supported cooperative work, pages 554–563, ACM Press.

Leuf, B. and Cunningham, W. (2001). The WIKI way. Addison-Wesley, Boston, MA, USA.

Pfister, H.-R., Schuckmann, C., Beck-Wilson, J., Wessner, M. (1998). The Metaphor of Virtual Rooms in the Cooperative Learning Environment CLear. In Norbert Streitz, S. Konomi, and H. Burkhardt, editors, Cooperative Buildings - Integrating Information, Organization and Architecture. Proceedings of CoBuild’98, LNCS 1370, pages 107–113, Springer-Verlag Berlin Heidelberg.

Schümmer, T. and Fernandez, A.: Patterns for Virtual Places, Proceedings of the European PLoP conference (EuroPLoP'05), 2006.