MULTIMEDIA COLLABORATION

User Manual
for SunOS
(MMCS Release 2.7)

Jürgen Dittrich, Jens Krähmer
Hartmut Kühn, Miroslav Vodsloň

March 14, 1995

BERKOM-Project: "Multimedia Teleservices (MMTS)"

Project Leader: G. Schürmann
Gesellschaft für Mathematik und Datenverarbeitung mbH (GMD)
Forschungszentrum für offene Kommunikationssysteme (FOKUS)
Hardenbergplatz 2, D – 10623 Berlin
Tel.: (030) 25499–200, Fax: (030) 25499–202

This description will be supplemented and modified according to useful requirements
during the implementation phase.

Contents

1 Introduction

Multimedia collaboration (MMC) allows people to hold multimedia teleconferences. In such a conference, several persons can talk and see each other without leaving their office desks.

Participants in a MMC conference can also work together using computer applications. These are ordinary, off-the-shelf applications such as editors, CAD programs or games. The MMC system allows to "share" one or more such applications. Such a shared application runs on one machine but its user interface (windows) is visible on the computer screens of all the participants in a conference. Participants take turns in controlling the application, i.e., in providing input to the application by typing at it, clicking in the menus etc. Most popular applications can be made "sharable" in this way. No special "sharable version" of such an application is necessary, only a special manner of starting it - which is provided by the MMC software.

This manual describes the MMC software implemented by **GMD-FOKUS**.

If you have ever used a personal computer (such as PC or Macintosh, but a Unix workstation with X-Window is best) with a mouse and a keyboard, then you should be able to use the system described in this manual.

If you find some places in this manual too technical, just skip them. Only your MMC administrator must know about technical details. Ask her how to start MMC on your system and concentrate on the description of the user interface.

A Glossary at the end of the manual explains some of the technical terms used.

Multimedia Teleservices in Berkom

This MMC software was developed by **GMD-FOKUS** within the BERKOM (Berliner Kommunikationssystem) project, an R&D project funded by the German PTT (Telekom) and supervised by DeTeBerkom. Among the aims of BERKOM is the development of two "multimedia teleservices": The Multimedia Mail Service (MMMS) [MMM_MANUAL] and the Multimedia Collaboration Service (MMCS). Several industrial and academic research centers cooperate in these projects. Both services are designed as electronic communication protocols, using existing international standards wherever possible.

2 Overview

This overview of the service's "architecture" will help you to understand some of the terms we shall use later on.

The MMC service is implemented as a heterogeneous distributed application. "Distributed" means that several hardware and software components (computers and programs), located in different places, are connected by a network and work together to provide the service. "Heterogeneous" means that computers and programs coming from various vendors can be used interchangeably.

At most one instance of each kind of software component can run on a given machine.

The MMC service consist of the following components:

- The Conference Directory (CD) where other components get static information about users and groups of users.
- At least one Conference Manager (CM) and one Audiovisual Manager (AVM), running both on the same machine. They manage dynamic information about currently running conferences.
- At least one Invitation Broker (IB). It brings invitations (calls) for currently running conferences to MMC users.
- At least one Application Sharing Component (ASC) enabling all participants in a conference to view and control applications.
- One Conference Interface Agent (CIA) per user, running on the user's personal workstation. Through the CIA, users manage conferences: they join a conference or create a new one, leave or close a conference, start and share applications, manage input to shared applications.
- One Audiovisual Component (AVC) per user, also running on the user's personal workstation. The AVC drives the user's audio and video hardware. It sends and receives AV data to/from AVCs running on other machines. The AVC is controlled by AVM.

Each component is known to the other components only by the name of the machine where it runs; therefore, at most one instance (process) of each component's program should run on a given machine.

In the present version of the service, each user can participate only in one conference at a time.

For a detailed description of the MMC architecture, see [MMC-SPEC].

3
Basic Procedures

The purpose of this chapter is a basic guide for the beginning user through the first steps of handling the MMC service. The following sections will show you how to start and stop a conference, how to join and finally leave one. These are the main features a user will use while communicationg with other conference members. If at any point you feel like you missed something or you fail to understand the meaning of certain elements, please first finish this chapter and then go on to the next chapter “Advanced Topics”. There you will find a closer look at the windows and other elements of the MMC service with a detailed description of those features only an advanced user will use.

3.1 Starting And Stopping MMCS

With some luck, your MMC administrator will have rigged your computer so that you only have to activate an entry in a menu to start the MMC service on your machine and another entry to stop it.

If you don’t find any hints in your window menus, simply type

```
mmc
```

to your shell and the MMC service will be started automatically.

To stop it, type

```
mmc.kill
```

to your shell and the MMC service with all its components will be stopped. In case of any errors resulting from these commands, please contact your MMC administrator or if you have some trust in your own skills and feel like an enhanced computer user, please refer to the chapter “Installation and Administration” for further details.

A window should open after a short while, this is the *Main Window* of the Conference Interface Agent (see Figure 3.1). The Conference Interface Agent is controlled exclusively by means of the left mouse

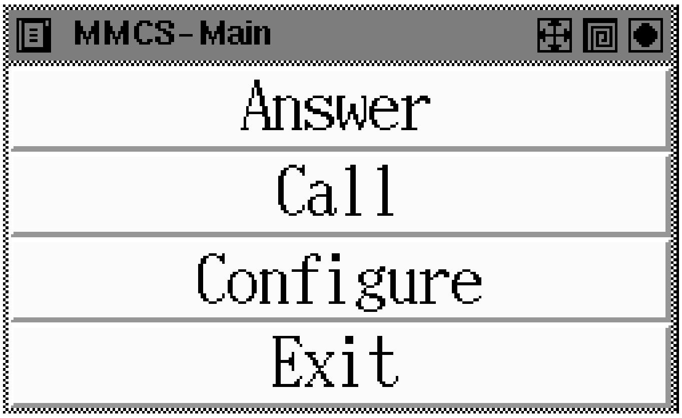

Figure 3.1: Main Window

button and the keyboard. To activate a command in the main window, simply click on the corresponding control field. For example, to choose the command *Call* from the main window, place the mouse cursor above the text and press the left mouse button.

From now on, you are able to start and stop conferences and other fascinating things. The following sections will guide you through the actions you are about to do by following the paths of the *Main Window*.

Note: If you share a machine with other users, do not forget to stop the service before logging out from your machine, as it is currently not possible for another user to stop the service !

3.2 Starting a Conference

Usually there are two types of actions that start a conference, you can call someone or you are called by someone. In this section you will call someone, means creating your own new conference, in the next section you will join an existing conference.

By clicking the *Call* button of the main window, a second window will open, it is called the *Conference Parameter Box* (see Figure 3.2).

Here you see a number of different control elements, some are important to you at this level, others you will use if you step to more advanced topics.

These are the elements you see:

- A list of all users of the MMC service found in the Conference Directory (List title: **Invite:**). (For a detailed description of all elements refer to the section “A Closer Look” in the chapter “Advanced Topics”.)

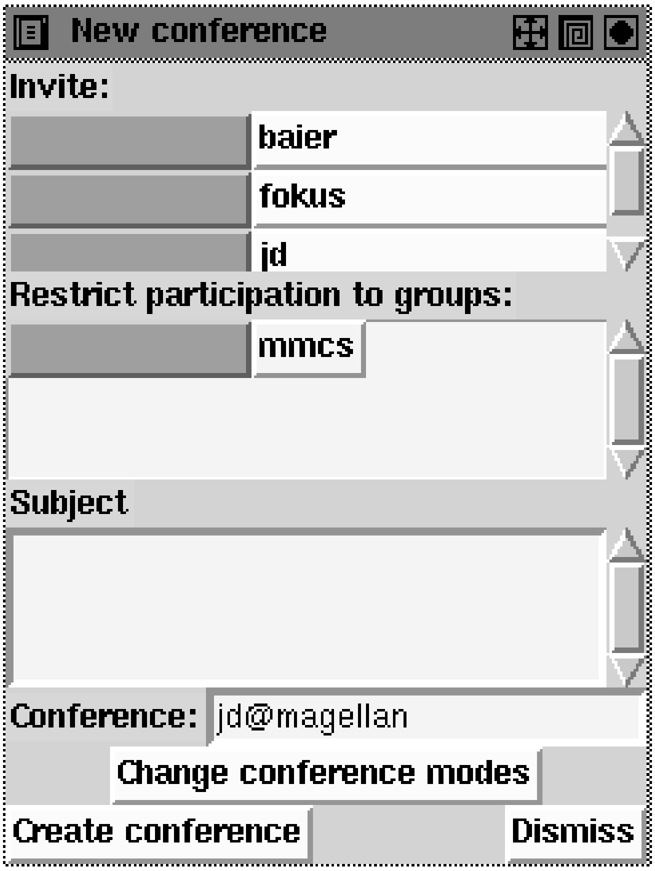

Figure 3.2: Conference Parameter Box

- A list of all user groups of the MMC service found in the Conference Directory (List title: **Restrict participation to groups:**).
- A text field where you can (but need not) describe the subject of the conference (List title: **Subject**).
- The name of the proposed conference. By default, this will be your username combined with the name of your machine. This name can be altered, why and how you will learn in the next chapter. There is no need for doing it now (List title: **Conference:**).
- A button to change conference modes. This is an advanced topic, the default settings will suit your purposes for a normal conference.
- Two buttons to start the conference or to abandon the *Conference Parameter Box*.

For the purpose of starting a simple conference, only the user list and the last two buttons are important to you. Now you can choose your wished participants for the upcoming conference by clicking in the colored field left to the user's name. The first click will make the user an active participant, a second click will make him a passive observer. A third click will return him to the initial state of not being invited.

After you chose your participants, all you have to do is click on the button titled *Create conference*. This will make you the leader of your new conference (the chair) and a new window will follow. This is the *Conference Window* (see Figure 3.3).

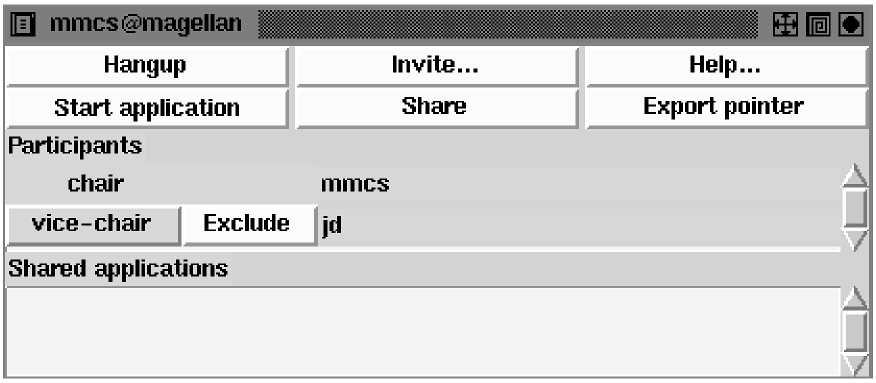

Figure 3.3: Conference Window (for the chair)

In this window you will find:

- 6 control buttons in the first two rows:
  - *Hangup*: activation of this command will make you leave the conference. You are the current chair, this means, it will also close the whole conference regardless of how many other participants are in the conference.
  - *Invite*: this button allows to invite additional users to the conference. It pops up a list of all the users found in the conference directory, which you can invite in the same manner described for the *Conference Parameter Box*.
  - *Help*: will open a help window.
  - *Start application*: this is an advanced topic. Please refer to the chapter "Advanced Topics".
  - *Share*: this is also an advanced topic.
  - *Export pointer*: you guessed it, this is an advanced topic, too.
- A list of participants in the conference. You are represented by the topmost row. Every entry in this list (except you of course) has two control buttons nearby, which allow you (as the chair) to change the role of every participant in the conference. Let's assume, that for this first simple conference you will stay the leader and all other participants keep their role as initiated.

- A list of as many rows as there are shared applications in the conference. Because we have not yet started shared applications, this place will remain empty, until we step to the next chapter.

If nothing went wrong and your participants have started their MMC services succesfully, too, you should find as many video windows as there are participants (of course only if your participants have their video cameras attached) and you should be able to commincate with them via audio.

Your first simple conference has started.

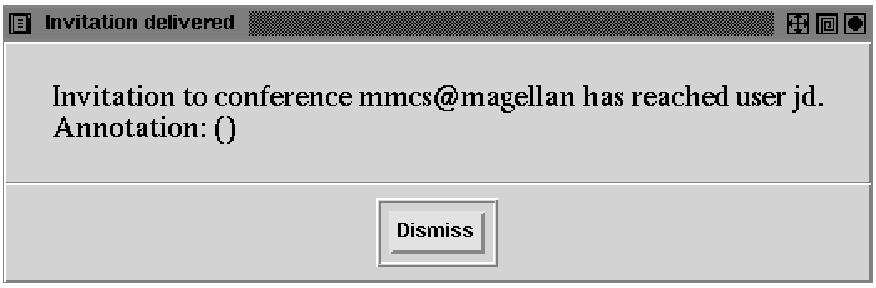

Figure 3.4: Note for a succesful invitation

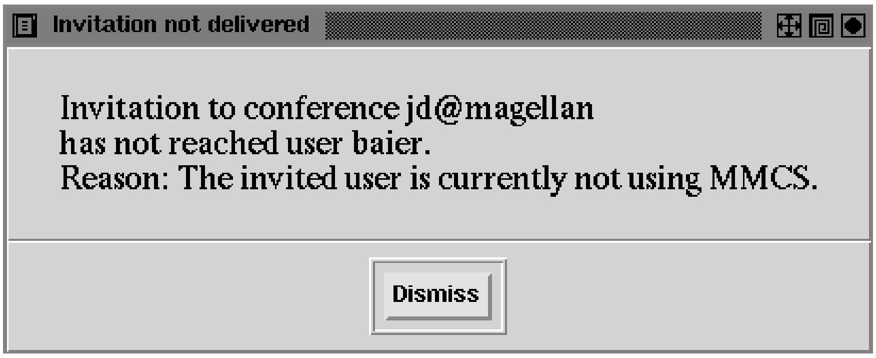

Figure 3.5: Note for an unsuccesful invitation

Note: Usually you will get a note that you have reached a particular user which looks like Figure 3.4. If any of your wished participants has not started the MMC service succesfully, you will get a note like Figure 3.5.

3.3 Joining a Conference

The second important basic situation is an invitation for you to join a conference. If someone invites you, a window will open showing you the name of the conference (or conferences) you are invited to

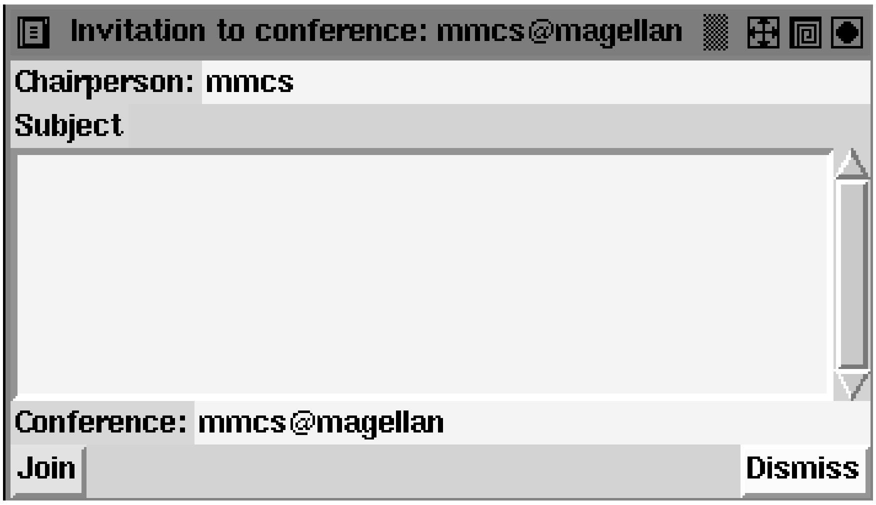

Figure 3.6: An Invitation

(see Figure 3.6) as well as the chairperson and the subject of the conference.

You can now choose to *Join* the conference or *Dismiss* the window. If you choose to dismiss the window, it doesn't mean that the invitation is gone, you simply do not participate in this conference at this time.

If you join the conference, you will see a window similar to the one in the last section, the only difference is your role in the conference (see Figure 3.7). As you can see, now you are not the chair person anymore, you are a normal participant who cannot alter the role of another participant.

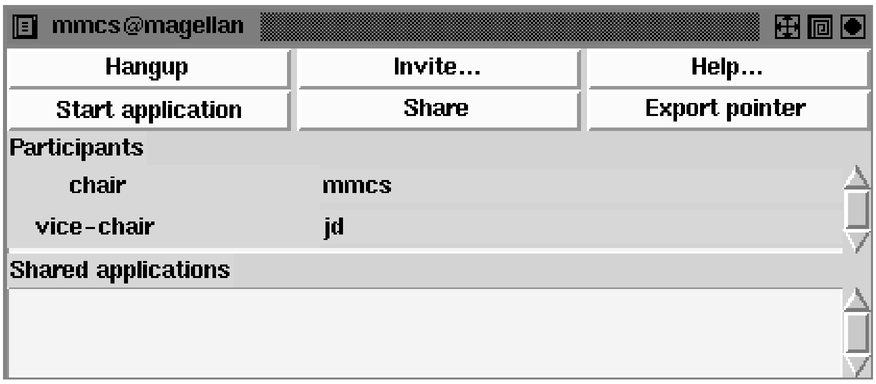

Figure 3.7: Conference Window (for non-chair persons)

If you chose to dismiss the invitation, you always can change your mind by clicking on the *Answer* button of the *Main Window*. If no invitations are active for you, you see a window like Figure 3.8. Otherwise you see something like Figure 3.9. Just click on the button with the name of the desired conference to join in.

Figure 3.8: Invitiation window (empty)

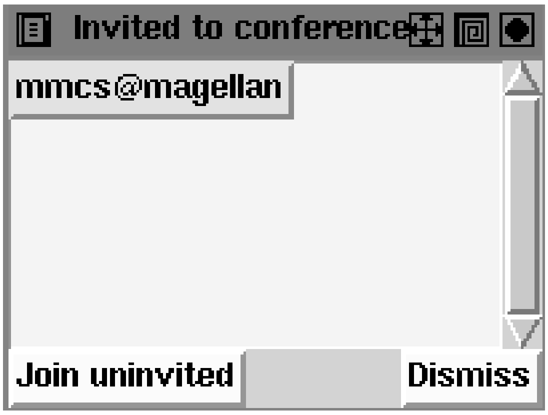

Figure 3.9: Invitation window (with active invitiation)

3.4 Leaving a Conference

In the *Conference Window*, to leave the conference, activate the *Hangup* command. This will unshare (make private to yourself) any shared applications (see "Advanced Topics") you own in the conference and notify all other participants that you have left the conference.

If you were not the (actual) chairperson of the conference, you may join the conference again.

If you are the chairperson of the conference, activating the *Hangup* command has the effect of closing the conference altogether, even if there were still other participants in the conference. Each of these remaining participants will cease to see and hear the others; any shared application (except those he owns) will disappear from his screen. Also, all users who were invited to the conference will be notified that the invitation is cancelled.

Leaving a conference as the actual chairperson is the most drastical way of ending it. In many cases it will fit your purposes, because you want to close the whole conference. If you only want to leave the conference as the chairperson without closing it down, refer to the chapter "Advanced Topics". There you will find how to pass the chair privilege over to another participant. By doing so, you can leave the conference in the hands of the remaining participants, who then can decide for themselves when to end it.

4
Advanced Topics

In this chapter the user will learn about the more advanced topics of the MMC service. Of course these topics are based on the knowledge given in the previous chapter.

Here you will find a closer look at the windows and other elements of the MMC service with a detailed description of those features only an advanced user will use. You will learn how to handle shared applications, export your mouse pointer, changing user roles and customizing the MMC service.

4.1 A Closer Look

This section gives detailed descriptions of the windows you will see while working with the MMC service.

4.1.1 Details of the Conference Parameter Box

The *Conference Parameter Box* pops up when the *Call* command is activated in the *Main Window*. Its purpose is to set up parameters of a conference before creating it (see Figure 3.2).

It features:

- A list of all **users** of the MMC service found in the Conference Directory.

  Each user's name is tagged with a multistate button showing the role you wish to bestow on this user in the future conference (refer to the section "Starting a Conference"). Clicking on the button once or several times will both select the user for the future conference and give him a role.

- A list of all **user groups** of the MMC service found in the Conference Directory.

  Similarly, each group name is tagged with a button. Initially, the button displays no text, meaning that the group is not to be selected. Click on the button to select the group. Click again to unselect it.

  - If no group is selected, and the conference access mode is open (the default, see below), then everybody will be allowed to join.

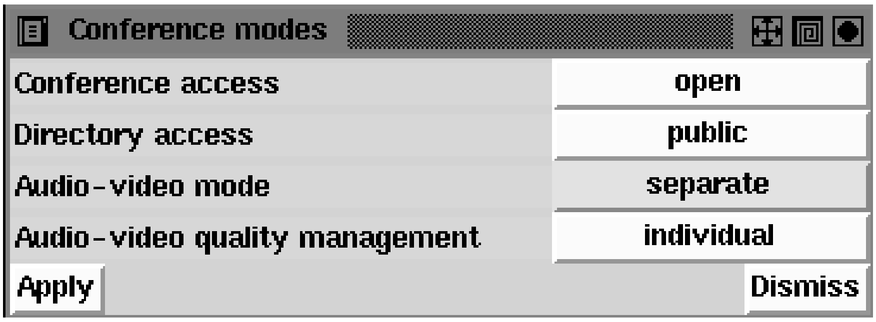

Figure 4.1: Conference Modes

  - If some groups are selected, then only users who are members of one of the selected groups will be able to join the conference uninvited.

- A text field where you can (but need not) describe the subject of the conference.

- The name of the proposed conference. It consists of a conference name proper (by default, this is your user name), followed by the character @, followed by the name of the machine where your conference manager runs. For example: `Smith@hal-atm`. The default name is allright most of the time, sou you need not touch it.

  You may omit the conference manager part, and type, e.g., only `Smith`, or `My-Conference`. In this case, the CIA will use the "default Conference Manager". See information about startup parameters, in another section of this manual.

- A button to change conference modes. It pops up a window to change the conference access mode and other parameters for the future conference (see Figure 4.1). However, most of the time, the default mode parameters will suit your purpose well so you won't have to worry about them.

- Commands to start the conference or to abandon the *Conference Parameter Box*.

4.1.2 The Conference Modes Window

The *Conference Modes Window* is activated by the command *Change conference modes* in the *Conference Parameter Box* (for the next conference to create) or in the *Editor of Preferences* (to set default values for all future conferences in this MMC session). The *Editor of Preferences* will be described later.

Here is an explanation of the conference modes and their values.

- Conference Access Mode:
  - Open (the default): every user of the MMC service will be able to join the conference.
  - Joinable: every user will be able to join in, subject to approval by the actual conference chairperson.
  - Close: only invited users will be able to join.
    For flexibility, choose the Open access mode.
- Directory access mode: This parameter has currently no effect.
- Audiovisual Mode:
  - Separate (the default): use this if you wish to make all participants in the future conference visible and hearable to one another, while transmitting audio and video in separate data streams.
  - No AV: use this for a conference where the only means of communication is through shared applications, e.g., taking turns typing into an editor program.
  - Audio only: use this if you do not want to incur the overhead of video processing and transmission.
  - Video only: for silent demonstrations, maybe.
  - AV: interleave audio and video in one data stream.
- Audiovisual Policy:
  - Individual best (the default): the service will try to make the most of the capabilities of each participant's workstation. E.g., people with digital video equipment will transmit and receive video streams, people without will not.
  - All worst: the service will downgrade all participants to the capabilities of the least equipped one. If only one participant has no digital video equipment, nobody in the conference will transmit or receive video streams.
  - Strict: the conference will be created only if all participants' equipment is up to the required mode of operation. E.g., if the audiovisual mode is "separate" (meaning: we attempt to establish an audiovisual communication) and one of the workstations has no digital video equipment, the conference will not be created.

4.1.3 The Conference Window

It pops up when a conference is started either from the *Main Window* or from the *Conference Parameter Box*. It shows the current state of the conference and allows the user to change it (see Figure 3.7).

It features:

- Several commands:
  - Hangup: activation of this command will make you leave the conference. If you are the current chair, it will also close the conference regardless of how many other participants there are.
  - Invite: this button allows to invite additional users to the conference. It pops up a list of all the users found in the conference directory, for you to select 0 or more of them to invite and assign roles in the same manner as in the *Conference Parameter Box*.
  - Start application...: this pops up a window to start a sharable application. The contents of this window can be customised by setting the startup parameter `SHARABLE_APPLICATIONS` in one of the configuration files or on the command line, or in in the *Editor of Preferences*.
  - Share: to share an application, activate this command then click in the window of the application you wish to share. Only applications started using the Start application window are guaranteed to be sharable.
  - Export pointer/Withdraw pointer: by activating this command, you can make your mouse pointer visible to all participants in the conference when it is in one of the windows of a shared application. By activating the command again, you make your mouse pointer visible only to yourself again.
- A list of participant-control rows, one for each participant in the conference. You are represented by the topmost row. Each *participant-control row* contains
  - the name of the participant, as registered in the Conference Directory;
  - a button bearing indication of its current role, which is one of the following:
    1. "observer", i.e., never allowed to provide input into shared applications;
    2. "active", i.e., allowed to provide input into shared applications (if and when it is his turn, or the application is currently in the so-called free mode); this participant cannot become the actual chairperson;
    3. "vice-chair", i.e., in addition to the rights of an active participant, this user can become (but is not) the actual chairperson; if you are the chairperson you can make this user the next chairperson by clicking on this button.

4. “chair”: this participant’s role is the “actual chairperson”, i.e., in addition to the rights of an active participant, he has the exclusive right to do one of two things:
   * make some potential chairperson to actual chairperson by clicking on his role button;
   * stop the conference by activating the Hangup command.

   Every conference has exactly one actual chairperson, initially the initiator of the conference.

If you are the current chairperson, you get a menu by pushing mousebutton 1 on the role button of any participant (except yourself). From this menu, you can

* exclude the participant from the conference (i.e., force him or her immmediately out of the conference) or
* make the participant the next chairperson, if his role was “vice-chair”. If you do this, you lose your chairpersonship and become only a “vice-chair” yourself. Why you should want to do this is explained in another section of this manual.

- A list of as many rows as there are shared applications in the conference. Each *application-control row* shows:
  - a number assigned to the application by the Conference Manager program; this number identifies a shared application in a conference;
  - an application title;
  - a Release/Request command, which is either
    1. labeled “Pass to...”, and followed by the name of the first user who requested the token from you (if you are the token holder for that application); or
    2. labeled “Request from...”, and followed by the name of the user who currently holds the token (if you are not the token holder for that application);
  - a mode button, which shows the token-passing mode of this application; it is one of the following:
    1. Explicit: means you have to be the tokenholder to provide input to the application;
    2. Free: means every participant can provide input to the application.

  By pushing mousebutton 1 on the mode button, if you brought the application into the conference, you get a menu allowing you to set the token-passing mode to free or explicit or to unshare the application. “Unshare” means that you withdraw the application from the conference and make it private to yourself again. This does not end the application.

4.2 Sharable Applications

This section will show you how to start and share applications, how to handle input to those applications and how you to export your mouse pointer.

4.2.1 Starting a Shareable Application

In the *Conference Window*, to start a sharable application, activate one of the commands in the window that pops up when you activate the button *Start Application*. You can start a private or a shareable application.

After a short while, a window should appear on your screen (see Figure 4.2), offering you examples of applications you can start.

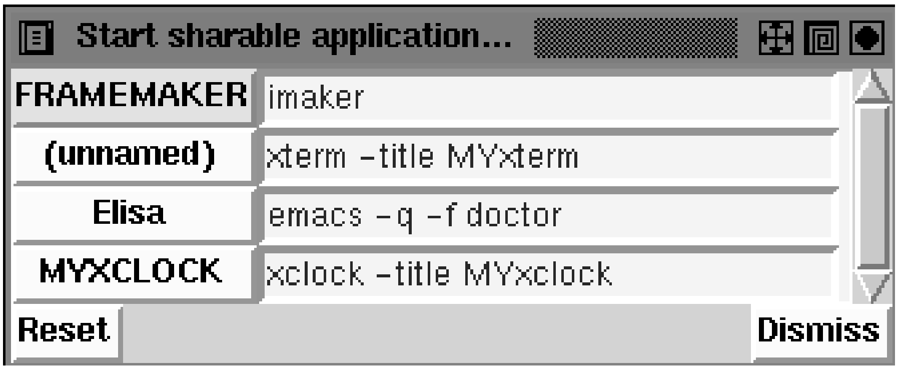

Figure 4.2: Start shareable applications

If you wish to start an application which is not on offer, you are allowed to change this window using the *Editor of Preferences* (see below). Most applications can be started more than once, e.g., you can have more than one instance of a given editor running, including several sharable instances.

Once successfully started, the application is still private to you, i.e., visible on your screen only. However, it was started in a special way which makes it sharable. Please take note that a sharable application will terminate automatically if the MMC service is stopped. (More precisely, the application will stop as soon as your ASC component of the service is stopped.)

4.2.2 Share und Unshare Applications

To share a sharable application among all the participants of the conference, first make sure that the window of the sharable application is visible on your screen, at least in part. Then activate the *Share* command in the *Conference Window*. Your mouse cursor becomes a crosshair. Move the crosshair into the window of the sharable application (NOT on its border or other "decoration" which does not really belong to the application); then click.

As a result, the MMC service posts copies of any and all windows of this application to the screens of all the conference participants. The application becomes shared (see Figure 4.3).

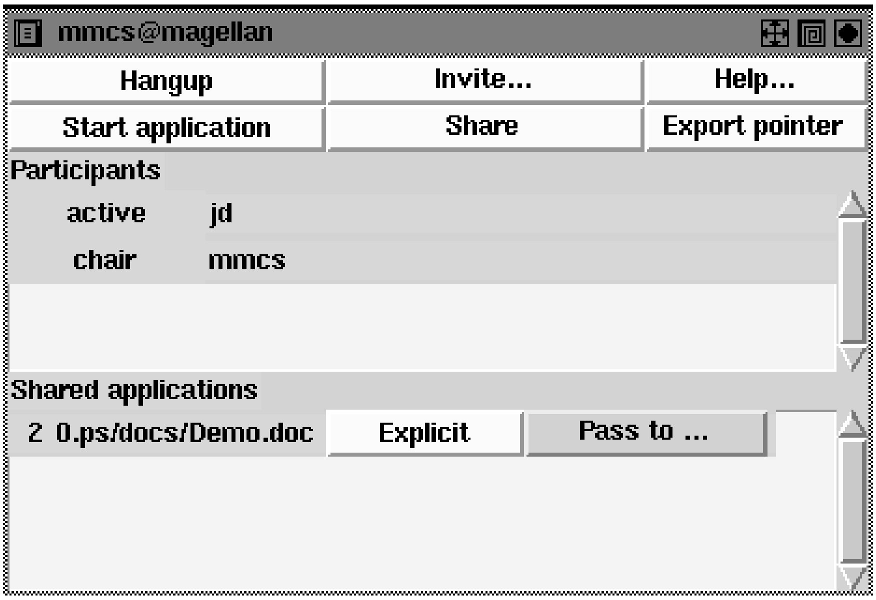

Figure 4.3: A Shared Application

This is also reflected in the conference windows of all participants who use the **GMD-FOKUS** Conference Interface Agent: a new *application-control row* appears in the list of shared application.

4.2.3 Controlling Input to Shared Applications

The application windows are now visible to all other participants, but the owner still controls the application. We say that he "holds the token" for that application. Any input to the application windows (by keyboard or mouse) by another participant is ignored. This is because the application is, at first, in the so called *explicit token-passing mode* and the owner of an application is always the first token holder.

In the *explicit token-passing mode*, those participants who are not mere observers can request the token, i.e, the right to provide input to the shared application. They can do so by activating the *Request* command in the *application-control row*. Such a token request is not automatically granted, however. The immediate effect is only that the current token-holder will be notified that you have requested the token.

It is up to the current token holder of the application to release the token and make it available to the first requestor. This is done by activating the *Pass to...* command in the *application-control row*. The next token holder will be the earliest requestor or, if there is none, the application owner.

At any time, the application owner can activate the *Unshare* command in the mode menu of the *application-control row* in order to withdraw the application from the conference. This will make disappear all the windows of the application from the screens of all the other participants and make the application private to the owner again. The application remains sharable: the *Share* command can be used again to share the it.

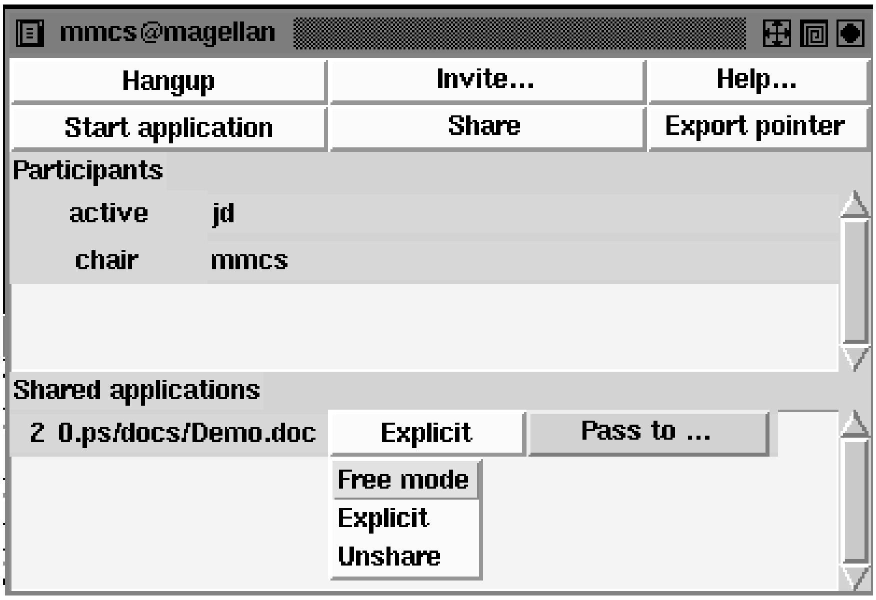

Figure 4.4: Altering a Shared Application

The application owner can also use this menu to set the token-passing mode for that application. The explicit mode (the default) was just described. In the free mode, the token is not used. Everybody can type and mouse-click at the application (see Figure 4.4).

4.2.4 Export and Withdraw Telepointer

Sometimes it is useful that the participants of a conference see what you are doing with your mouse pointer. A teaching lesson could be a good example for that. To export your mouse pointer to all participants, click on the button labeled *Export pointer* in the *Conference window* (see Figure 4.5). A note will come up, telling all members that you have exported your mouse pointer.

You will also see the button text changing to *Withdraw pointer*. If at any time you wish to withdraw your pointer, click on the same button again (see Figure 4.6). This means, that your mouse pointer will only be visible to you again.

4.3 Customisation (The Editor of Preferences)

The *Editor of Preferences* pops up when the command *Startup Parameters* is chosen from the *Configure* button of the *Main Window*. It is usually not needed if the MMC software was installed properly on your system. Its purpose is to change the values of some startup parameters of the CIA (see Figure 4.7).

It features:

- A text editor window which contains initially the current values of all the startup parameters. Each parameter is shown on a line by itself, in the form: attribute = value.
- Commands:
  - Apply: Try and change the state of the CIA in accordance with current contents of the text window.
  - Discard changes: Throw away the text, replace it with a text reflecting exactly the current state of the CIA.
  - Dismiss: Pop down the *Editor of Preferences* without changing anything.

4.3.1 I Am Somebody Else

In this and the next few sections we list some changes to startup parameter values which can be useful in some situations. You can skip these chapters on first and perhaps all subsequent readings of this manual.

One useful parameter to modify is your MMC user name and password, values of the parameters `THISUSER` and `MY_CIA_PASSWD`.

Unless otherwise specified in the file `$HOME/MmcUser.def` or on the command line when starting the CIA, the values are your login name and '`mmc`'. However, it is possible that you are expected to use the MMC service under another name or with another password than your login name or the name specified in `$HOME/MmcUser.def` or on the command line and you do not wish to restart the CIA.

Just change the values to the right of '`THISUSER =` ' and '`MY_CIA_PASSWD =` ' in the editor window, then activate the Apply command to make the change take effect.

Please do not change these values while in a conference.

4.3.2 This Component Runs Elsewhere

The *Editor of Preferences* is also useful should it happen that the machine, named A, where the conference directory usually runs is down for the moment and the conference directory now really runs on some other machine, B, but nobody changed the configuration files accordingly for you and nobody even told you before you started the CIA. The CD is now unreachable for you and you cannot create a new conference, but you do not wish to restart the CIA.

The *Editor of Preferences* can help you if you now know where the CD really runs. Just replace the value A in '`CDHOST = A`' by B.

You can do the same for the default conference manager. The parameter is named `CMHOST`.

You can set both `CMHOST` and `CDHOST` to the same value B by adding AT THE END of the text in the text window, the line `BACKBONEHOST=B`, then activate the Apply command to make the change take effect.

Please do not change these values while in a conference.

4.3.3 Bells and Whistles

The *Editor of Preferences* can also be used to change aspects of the user interface. All relevant parameters may be changed while in a conference and will take effect immediately.

E.g., when you receive an invitation to a conference, the CIA will play a sound-file on your loudspeaker or in your headset. The parameter `INVITATION_SOUND_NAME` specifies the name of the sound-file.

The value is `<NULL>` by default (meaning: play nothing), but your configuration files may specify some sound to play. Change the value to the right of '`INVITATION_SOUND_NAME=`' to change this sound. For silence, change it to `<NULL>`.

Similarly, the parameter `CANCEL_INVITATION_SOUND_NAME` tells the CIA which sound-file to play when a conference to which you were invited is closed. You can change this value, too.

A more important user interface feature is the *Start application window* which pops up from a command in the *conference window*, and is used to start sharable applications. A startup parameter, `SHARABLE_APPLICATIONS`, specifies the contents of this window.

Its default value is `<NULL>`, meaning an empty window. A non-empty window must be specified as a string between double quotes, as in the following example:

```
"FRAMEMAKER     imaker
xterm -title MYxterm
PSYCHO  emacs -q -f doctor"
```

Each line specifies one row in the Start application window. In this example, there are three sharable applications, Framemaker (an editor), xterm (a terminal emulator) and Psychoanalyst (a game).

The first word in each line, if followed by a TAB character, is the name of the command button, e.g., FRAMEMAKER and PSYCHO. The rest of the line, e.g., imaker and `emacs -q -f doctor` are actual command lines to be passed to the command interpreter (shell) on your system in order to start the application.

In the second line of our example, the first word (xterm) is not followed by a TAB character. In this case, a default name of the command button in the Start application window will be chosen automatically. (You may specify it as an X resource.)

Instead of typing the TAB character, which is invisible, you may use the two characters '\t', as follows:

```
"FRAMEMAKER\timaker
xterm -title MYxterm
PSYCHO\temacs -q -f doctor"
```

Instead of putting each application on a separate line, you may separate them (or only some of them) by the two characters '\n', as follows:

```
   "FRAMEMAKER\timaker\nxterm -title MYxterm\nPSYCHO\temacs
-q -f doctor"
```

4.4 Permanent Startup Parameters

In the *Editor of Preferences* you can change several parameters of your CIA, but these changes remain in effect only as long as the CIA runs.

To set some of these parameters permanently, record them in a file named `MmcUser.def` in your home directory. Each of the parameters mentioned in the chapters describing the *Editor of Preferences* can be specified in `MmcUser.def`, using exactly the same syntax as described in those chapters.

4.5 Audio and Video Hints

A good public-domain audio-control program, x_gaintool+, is delivered with the software. Start it and you will be able to switch on/out your loudspeaker, headset and microphone as well as control the gain levels, using your mouse. In practice, x_gaintool+ is indispensable.

A gain of 25/100 is usually about right, in both directions.

If any participant in a conference has both his loudspeaker and his microphone switched on, all participants will hear an echo. To help suppress the echo, either use a headset or make it a habit to switch your microphone off when not speaking.

The recommended microphone is the original Sun device.

An important parameter of a video conferencing system is the video encoding scheme used to send video data over the network.

The **GMD-FOKUS** MMC software currently supports two video encoding schemes:

- JPEG, which is an international standard for the encoding of still pictures (moving pictures are encoded as a sequence of still pictures). This encoding affords true colour. It is supported by hardware (the Parallax board), allowing you to have a conference of three participants with frame rates of 20 frames per second and still do useful work on your Sparcstation.
- SMP (Software Motion Pictures), which is a product by Digital. It is not supported by hardware, making colour video impractical. Even with grey-scale video, SMP will eat up at least twice as much processing power of a Sparcstation as JPEG.

The recommended camera is FLEXCAM, which allows flexible positioning of the camera in front of you.

The camera should be the nearest possible to the video windows of the other participants on your screen. This will give your partners the best approximation of eye contact, i.e., they will have the impression that you look at them when talking to them.

Figure 4.5: Export pointer

Figure 4.6: Withdraw pointer

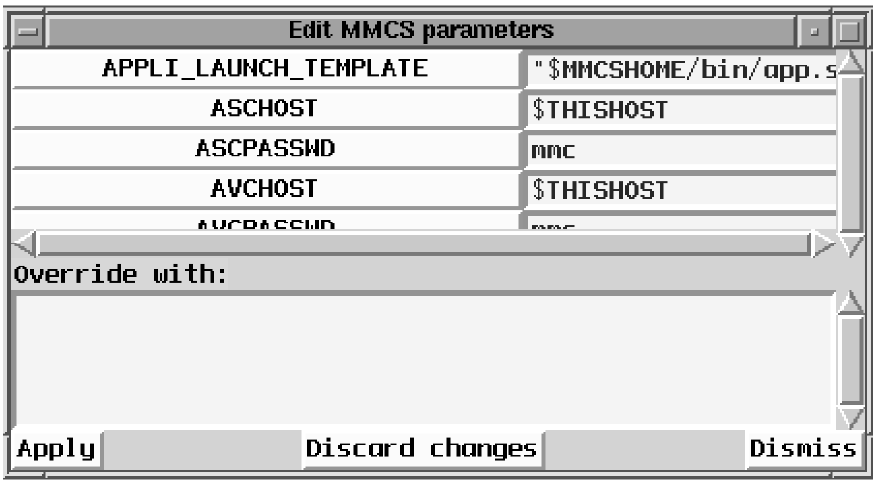

Figure 4.7: The Editor of Prefences

5
Installation and Administration

This chapter is of interest to the MMC administrator only.

GMD-FOKUS provides its own implementation of all MMC components, except the ASC.

The ASC must be licenced from Digital. Details are available at the following address:

- E-mail: `altenhof@nestvx.enet.dec.com`
- Mail: CEC-DIGITAL, attn. Michael Altenhofen, Vincenz-Priessnitzstr. 1, D-76131 Karlsruhe
- Phone: +49 (0) 721 69 -0232 (voice), -6816 (fax).

5.1 Requirements and Availability

GMD-FOKUS MMC software requires the following:

- Hardware:
  - SparcStation 2 or 10/20 or 10/30 or 10/40 or 10/50
  - ParallaxBoard (XV-24SV, PowerVideo or compatible)
  - Sun Microphone (either external or built into a Headset)
  - (recommended) Headset
  - free disk space: about 30 MB for binary, about 35 (respectively 130) MB for uncompiled (respectively compiled) source distribution
- Software (binary and source distribution)
  - SunOS 4.1.3 (Solaris 1.1) (Copyright Sun)
  - Parallax XVideo Software, X11R5 Version 1.1 (Copyright Parallax)
  - MIT X11R5 (including Xt) (Copyright MIT)
  - OSF Motif 1.2 (Copyright OSF)
  - MMCS Application Sharing Component (ASC, programs ascfe/shXbridge) (Copyright Digital)

- Software (additional, source distribution)
  - GNU gcc Compiler/Debugger (Copyright Free Software Foundation)
  - Software Motion Pictures (SMP) Package (Copyright Digital)
  - The ISODE package, either
    * ISODE Version 8.0 (Public Domain) or
    * ISODE Consortium Release 1.0 (Copyright ISODE Consortium)

  The following must be noted:

  - The SunOS 4.1.3 kernel must be built with the shared memory option enabled. See the Sun Administrator Manual for details.
  - The ISODE library (Consortium Release) must be patched. Holders of a valid licence from ISODE Consortium get, an arleady patched binary version of the library as part of the distribution of the **GMD-FOKUS** MMC software. In other cases, you may need to apply our patches to your own ISODE library. The patches are to be found in the file `$MMCSHOME/Isode.Patch`. If you have further questions, please let us know.

The **GMD-FOKUS** implementation of the CD, CM, IB and CIA components has been ported to Hewlett-Packard workstations by the Technical University of Berlin.

5.2 Contact Address

For additional details and conditions of distribution, please contact Gerd Schuermann at the following address:

- E-mail: `gs@fokus.gmd.de`
- Mail: GMD-Fokus, Multimedia Dept. (OVMA), attn. Gerd Schuermann, Hardenbergplatz 2, D-10623 Berlin,
- Fax: +49 30 25499 202.

5.3 Installation

The MMC administrator creates a directory where the software should reside and unpacks the distribution file (usually on tape) in that directory, using the tar program. We shall refer to this directory as `$MMCSHOME`.

The **GMD-FOKUS** MMC software is distributed either in binary form or in source form. The name of the archive file reflects this and the version of the software. It is:

```
mmcs-NNN-SPECIFIER.tar
```

where `NNN` is the version number, and `SPECIFIER` is either bin or src.

In both cases, un-tar your distribution file:

```
tar xfp mmcs-NNN-SPECIFIER.tar
```

Having done this, the administrator has on-line access to a detailed installation manual, to the description of various tools as well as to up-to-date release and specialised troubleshooting information. The first file to read is the file named `$MMCSHOME/README.FIRST`.

To compile the source distribution, do the following

- `cd $MMCSHOME`
- edit the file `src/make.common.macros`
- enter the following command:

```
make it
```

This will generate the binaries of all the **GMD-FOKUS** MMCS programs.

Next, the following must be done to install the software:

1. Make sure that the file `$MMCSHOME/etc/isode/isoentities` is writable for all.
2. Decide on which machine(s) in your network the "backbone" components (see Glossary) of the service will run. These components are not resource-hungry, so you might wish to have them running all on one machine.
3. Edit the site-runtime-configuration file `$MMCSHOME/MmcSite.def` to record your decision about the backbone. This file contains comments (starting with '#' and ending at the end of line) and pairs of keyword and value, separated by white space. There

can be also one '=' sign between a keyword and the corresponding value. Each keyword stands for one or more startup parameters. The meaning of each parameter is explained in the file itself. If your backbone machine is `yyy`, you need to find the line beginning with `#BACKBONEHOST`, uncomment it (i.e., delete the leading '#') and make the whole line look like this:

```
BACKBONEHOST yyy
```

If the CM and the CD run on different machines, the keywords to use are `CMHOST` and `CDHOST`, instead of `BACKBONEHOST`.

4. Prospective MMCS users must be registered with the conference directory each time the service is started. In order to have this registration happen automatically, you must enter a record for each user into one of MMCS system files, the users-registry.

   To do so, make sure the CD component and the X server are running, then call:

   ```
   cdtool
   ```

   `cdtool` has a self-explanatory user interface.

5. Encourage each user to set up his/her environment as described above, in chapter SETTING UP A MMCS USER.

Also part of the **GMD-FOKUS** MMCS implementation are tools which start, terminate and monitor the service.

The scripts `mmc` and `mmc.kill` (described in chapter "STARTING AND STOPPING MMCS" above) start all MMC components. Although this does no harm and ensures that the user's CIA will always be able to access the necessary backbone components to create a conference, it is something of an overkill.

You (the MMC administrator) may choose, instead, to provide users with scripts to start/stop the service which will affect only the non-backbone components. The following tools will help you to do that, and more:

- `mmc` : Starts a given single MMCS component, or the backbone, or all MMCS components. Type `mmc -help` to learn about details of usage.
- `mmc.kill` : Stops a given single MMCS component, or all of them. Type `mmc.kill -help` to learn about details of usage.
- `mmc.track` : Shows which MMCS components are running on your machine and which network addresses they use.
- `mmc.clean` : removes any temporary files (in directory /tmp) left behind **GMD-FOKUS** MMC software.

- `mmc.show` : shows the values of startup attributes for this user and for a given **GMD-FOKUS** MMC program. These are "raw" values, exactly as they appear in configuration files. Type `mmc.show` to learn about details of usage.
- `mmc.showeval` : like `mmc.show`, but any variable references in the values are resolved.
- `mmc.play` : plays a given audio file on a given AVC. It can be used, e.g., to ring a bell to draw the attention of another participant in a conference, or to listen to your favourite song (possibly one you recorded with `mmc.record`, see below) when bored during a conference. To play file `ff` on host `hh`, type

  ```
  mmc.play -f ff -avc hh
  ```

  Either or both of `-f ff` and `-avc hh` can be omitted. The startup parameters `AVC_File` and `AVCHOST` will be taken instead of missing parameters.
- `mmc.record` : this program listens to a given AVC (by default: the one running at your own machine) and records its audio output in a given file. Usage:

  ```
  mmc.record filename [-host avchostname]
  ```

  This will pop up a control window with buttons to start recording, stop recording and terminate.

5.4 Setting Up A MMCS User

Please make sure that in the shell environment, the variable MMCSHOME is set to the directory where the **GMD-FOKUS** MMC software was installed. If it is not, and assuming that the MMC software is installed in /usr/local/mmcs, then set it by typing, for example (in a C-shell):

```
setenv MMCSHOME /usr/local/mmcs
set path = ($path $MMCSHOME/bin)
```

To avoid doing this often, include these lines in the file .cshrc in the user's home directory.

Also, if you plan to use video, make sure that you have the Parallax hardware installed in the computers in question and that the X-Window server is the one delivered by Parallax.

5.5 Startup Parameters and Customisation

One manner to start all the necessary MMC programs is by typing

```
mmc -allhosts '$THISHOST' -cd xxx
```

where `xxx` stands for the name of the machine where the Conference Directory (CD) program runs.

This will start all MMC components, including CD, on a machine and tell all components which need to know that all other components also run on your machine, except that the CD runs on machine `xxx`.

For best results, all MMC components that need to talk to the CD should access the same CD. All users should therefore agree (better: you, the MMC administrator should decide) on which machine the CD runs.

Parameters like `-allhosts`, `-cd` and several others are called startup parameters. Each can be specified either on the command line as above, or in so-called configuration files. In fact, the latter is a better idea, because it leaves less room for error. With suitable contents of configuration files, users should need only to type `mmc` to start the service.

Users usually need not concern themselves with these details, but in some cases it helps to know how all **GMD-FOKUS** MMCS components get startup parameters. First, they look if the parameter is on the command line. If it isn't, they look in the file `$HOME/MmcUser.def`. If it is not there either, they look in the file `$MMCSHOME/MmcSite.def`. Finally, if still not found, every parameter has a reasonable default value.

For instance, `-allhosts '$THISHOST'` is one such default value, so specifying it in startup files or on the command line is unnecessary.

Another useful default value is that of the parameter `THISHOST`. It is the name of the machine where the program runs.

The default values are chosen such that very few parameters need to be set by the installer of the software in the file `$MMCSHOME/MmcSite.def`.

The file `$HOME/MmcUser.def` can be often omitted altogether.

The mechanism has one important exception: any specification of the parameter `THISUSER` in `$MMCSHOME/MmcSite.def` is ignored. Instead, the login name of the user is taken, unless otherwise specified in `$HOME/MmcUser.def` or on the command line.

A utility program, `mmc.show`, which is part of the distribution, shows the current values of the startup parameters resulting from this mechanism, for any MMCS component implemented by **GMD-FOKUS**.

To learn more about it, type `mmc.show` to your shell.

Detailed information about **GMD-FOKUS** MMCS startup parameters can be found in the sample configuration file `MmcSite.def` which is part of the distribution of this software. It is also appended to this manual.

This manual will mention only those startup parameters which influence the behaviours being described.

Further customisation issues concern the look of the Conference Interface Agent's windows which are described below.

You can change the values of some X resources in the file .Xdefaults in your home directory to change the look of the CIA for. A list of useful resources is provided with the distribution.

6
References

[MMCSPEC] The Berkom Multimedia Collaboration Service (Release 3.1).

[MMM_MANUAL] The Berkom Multimedia Mail User Manual.

7 Appendix

7.1 Glossary

Application owner: The conference participant who shared a sharable application in a conference.

Backbone: The set of MMC service component instances which must be up and running in your network for the service to be available. They are: one CD, at least one CM, at least one IB. One AVM must be co-located with each CM.

Command (button): A small window in the user interface. You can get something done by clicking in a command button with the left mouse button.

Command line: A line of text input into a shell. Most command lines start an application, whose name is the first word. The MMC service on your machine is started using a command line which begins with the word `mmc` and continues with zero or more startup parameters.

Configuration file: An editable file which specifies zero or more values of startup parameters. There are two configuration files: `$HOME/MmcUser.def` and `$MMCSHOME/MmcSite.def`. The first is personal to one user and overrides the second which is common to all users using an installation of the MMC software implemented by GMD-FOKUS.

Editable file: A file whose contents can be viewed and modified by the standard text viewing and editing programs in your computer.

Frame rate: the number of video images (also called frames) displayed each second.

Grey scale: A picture is grey-scale if it has no other colours than shades of grey.

Home directory: A place on your computer's disk where you can store data in files. Your home directory's name is the value of the shell variable HOME.

Initiator of a conference: The MMC user who creates a conference.

JPEG: An international standard for the encoding of still pictures. Encoding of moving pictures (video encoding) is achieved by encoding a sequence of still pictures, one by one.

Login name: The name by which your computer "knows" you.

MMC administrator: The person who installs and manages the MMC software on your computer.

Sharable application: An application which has been started on a machine in such a way that it can subsequently be shared.

Shared application: A sharable application which is visible to all participants in a conference. They can take turns to control the application with their mice and keyboards.

Shell environment: A set of named attributes maintained by a shell.

Shell variable: One of the named attributes in a shell environment. The MMC software by **GMD-FOKUS** uses two shell variables: HOME, which is set automatically by your system and MMCSHOME which you must set in the manner described in the chapter "STARTING AND STOPPING MMCS".

Shell: A program in your computer, also called a command interpreter. It allows to start applications. C-Shell and Bourne shell are two shell programs available on almost every Unix computer.

SMP: Software Motion Pictures, a video encoding scheme and product by Digital.

Startup parameter: One of a fixed set of named attributes used by MMC components implemented by **GMD-FOKUS**. Each startup parameter can be specified like this: name=value. Example: `CDHOST=mymachine`. Some startup parameters can be specified using alternate names. Example: `-cd=mymachine`. The '=' can be replaced by spaces, like this: `-cd` mymachine. Startup parameters can be specified in configuration files, on the command line when starting the MMC service and in the CIA's *editor of preferences*.

Token holder: The conference participant who has currently the right to control a shared application.

Video encoding scheme: A manner of transforming moving pictures from a camera into a stream of data which can be sent over a digital network to another location.

Widget: An object in a graphic user interface such as MOTIF. Each widget belongs to a class of similar widgets. E.g., MOTIF has a class of command buttons named XmPushButton. Each widget has a number of so-called resources, i.e., attributes which have names such as "background" (which specifies the background colour of the widget). Some resources can be changed in order to change the look of an application.

7.2 Configuration Files

Here are the contents of the site-wide startup file `MmcSite.def`:

```
# $MMCSHOME/MmcSite.def $Id: MmcSite.def,v 1.10 1994/12/11 21:00:18 vod Exp $
#
# Common MMCS configuration for all users accessing the same
# $MMCSHOME.
#
# Any attribute's value can be overridden by a specification for the
# same attribute either later in this file, or in $HOME/MmcUser.def or
# else on the command line.
#
# This is only useful in rare circumstances like debugging etc.
# In the typical case it is recommended *not* to use the
# $HOME/MmcUser.def file.
#
# For the typical case this file may be left unchanged.
#
# Synopsis: ValueName
#           Description: ...
#           Default Value: ... the "hard coded" value
#           Used from:     ... which components "read" or refer to this value
#
#=============================================================================
#THISHOST dont_set_it
# Description:   The default hostname for all hosts.
# Default Value: Value of uname -n.
# Used from:     CIA. Potentially ALL components.
#=============================================================================
#THISUSER        guest
# Description:   The name of the user.
# Default Value: USER_NAME=login name (= `whoami`); PASSWD="mmc".
# Used from:     CIA. Potentially ALL components.
#=============================================================================
#ISODE_ETCPATH   $MMCSHOME/etc/isode/
# Description:   The directory path, under which the ISODE files (isoentities,
#                isotailor, etc.), used for RPC communication, reside.
# Default Value: $MMCSHOME/etc/isode/
# Used from:     ALL components.
#=============================================================================
#CDHOST    $THISHOST
# Description:   The name of the host on which the CD is running
#                CM and CIA will try to bind to the CD during their start-up.
# Default Value: $THISHOST
# Used from:     CIA, CM
#=============================================================================
#BACKBONEHOST    $THISHOST
# Description:   A synonym for CDHOST and CMHOST (see below) entries.
# Default Value: $THISHOST
```

```
#==============================================================================
#CDUSERSFILE         $MMCSHOME/etc/users
# Description:   Text file describing users to be registered in the CD.
# Default Value: $MMCSHOME/etc/users
# Used from:     file2cd
#==============================================================================
#CD_ROOT_NAME        su
#CD_ROOT_PASSWORD    mmc
# Description:   The name/password pair of the administrator of the CD.
# Default Value: su, mmc
# Used from:     file2cd(=CDUA), CD
#==============================================================================
#CD_CM_NAME          ConferenceManager
#CD_CM_PASSWORD      mmc
# Description:   The name/password pair used by the locally started CM
#                to acess the CD.
# Default Value: ConferenceManager, mmc
# Used from:     CD, CM
#==============================================================================
#AVCHOST    $THISHOST
#AVCPASSWD  mmc
# Description:   The name/password pair to access your AVC. Given by the CIA
#                to the CM when opening/joining a conference.
#                This pair MUST MATCH your user entry as registered in
#                the CD, i.e. as written in the file $CDUSERSFILE.
# Default Value: $THISHOST, mmc
# Used from:     CIA, AVC (AVCPASSWD)
#==============================================================================
#ASCHOST    $THISHOST
#ASCPASSWD  mmc
# Description:   The name/password pair to access your ASC. Given by the CIA
#                to the CM when opening/joining a conference.
#                This pair MUST MATCH your user entry as registered in
#                the CD, i.e. as written in the file $CDUSERSFILE.
# Default Value: $THISHOST, mmc
# Used from:     CIA, ASC (ASCPASSWD)
#==============================================================================
#ASC_BRIDGE_HOST    $THISHOST
# Description:   Tells the ASC where the X-sharing pseudo-server runs.
# Default Value: $THISHOST
# Used from:     ASC
#==============================================================================


##############################################################################
#                                                                            #
#     C  I  A    Specific Parameters. Used from the CIA only.                #
#                                                                            #
##############################################################################
#MY_CIA_PASSWD   mmc
# Description:   $THISUSER and $MY_CIA_PASSWD form the name/password pair
#                used by the CIA to access the CD and the CM.
#                This pair MUST MATCH your user entry as registered in
#                the CD, i.e. as written in the file $CDUSERSFILE.
# Default Value: USER_NAME=login name (= `whoami`); PASSWD="mmc".
#==============================================================================
#CMHOST    $THISHOST
# Description:   The default CM hostname used by the CIA to
#                open/join a conference,
#                when user gives no explicit CM hostname.
# Default Value: $THISHOST
# Used from:     CIA
#==============================================================================
#CIA_INITIAL_CONFERENCENAME
# Description:   Conference name and conference manager to be used when the
#                user gives none.
# Default Value: "$THISUSER@@$CMHOST"
#==============================================================================
#MMCSXPMPATH      $MMCSHOME/etc/photos
# Description:   Directory with photographs of users.
```

```
#               Each photo is in an xpm image file. The name of the
#               file is <username>.xpm, such as smith.xpm
# Default Value: $MMCSHOME/etc/photos
# MMCSXPMPATH is OBSOLETE in version 2.7
#============================================================================
SHARABLE_APPLICATIONS "FRAMEMAKER\timaker\nxterm -title MYxterm\n\
PSYCHO\temacs -q -f doctor\nMYXCLOCK\txclock -title MYxclock\nMYMAZE\tmaze"
# Description:   Menu of sharable applications.
#                Value syntax:
#                   "@{[button_name\t]command\n@}*". Default for button_name is
#                   command. button_name can be used in the Xt resource file
#                   to label the button in the menu of sharable applications.
# Default Value: <NULL> (meaning: empty menu)
#============================================================================
#APPLI_LAUNCH_TEMPLATE "DISPLAY=$ASCHOST:2\nexport DISPLAY\necho %s\n%s &"
# Description:   Specification for how to start a sharable application.
#                The value string may contain at most 10 substrings "%s"
#                which will be replaced by identical copies of one of the
#                commands specified in SHARABLE_APPLICATIONS.
#  The above template is an alternative to the default (below).
#  app.start currently tries to start the ASC if none runs.
# Default Value: "$MMCSHOME/bin/app.start %s &"
#============================================================================
#PLAY_SOUND_TEMPLATE "mmc.play -avc  $AVCHOST -f %s &"
# Description:   How to play a sound file. The %s substring will be
#                replaced by the name of the sound file to play.
# Default Value: "mmc.play -avc $AVCHOST -f %s &"
#============================================================================
INVITATION_SOUND_NAME $MMCSHOME/etc/sounds/ring.au
# Description:   Sound to play when Invitation Broker brings a new invitation.
# Default Value: <NULL> (meaning: keep quiet)
#============================================================================
CANCEL_INVITATION_SOUND_NAME $MMCSHOME/etc/sounds/bark.au
# Description:   Sound to play when an invitation was cancelled.
# Default Value: <NULL> (meaning: keep quiet)
#============================================================================
#TP_RGB "60000   0 0"
#TP_ANNOTATION $THISUSER
# Description:   Colour (red green and blue intensities) and text to be used
#  for telepointers exported by this participant.
# Default values: as shown.
#============================================================================



###########################################################################
#                                                                         #
#     A  V  C    Specific Parameters. Used from the AVC only.             #
#                                                                         #
###########################################################################
#
##############################  PARALLAX  #################################
#
#AVC_PlxVideoInput        0
# Description:   Set PlxVideoInput to 0 or 1 to select the video input port.
# Default Value: 0
#============================================================================
#AVC_PlxTopClipping       4
# Description:
# Default Value: 4
#============================================================================
#AVC_QFactor              120
# Description:  Determines the quality of the JPEG compression algorithm.
#               The value should be between 25 and 1000. Small value=good
#               image quality, large frames. High value = poor image quality,
#               small frames, fast processing. 120 (default) is a good
#               compromise.
# Default Value: 120
#============================================================================
#
######################   VIDEO CODING PREFERENCES   ########################
```

```
#
#AVC_SmpOnlyMode 0
# Description:   Determines if the local AVC should use SMP as the only
#                video coding.
#                A value of "1" will do so, a value of "0" will result
#                in the use of other codings (i.e. M-JPEG) as well.
# Default Value: 0
#=============================================================================
#
###########################   FRAME RATES   ################################
#
#AVC_JpegFrameRate        25
# Description:   Maximal sent fps for JPEG only source.
# Default Value: 25
#=============================================================================
#AVC_SmpFrameRate         10
# Description:   Maximal sent fps for SMP only source.
# Default Value: 10
#=============================================================================
#AVC_ReducedJpegFrameRate 10
# Description:   Maximal sent JPEG-fps for JPEG/SMP source.
# Default Value: 10
#=============================================================================
#AVC_ReducedSmpFrameRate  5
# Description:   Maximal sent SMP-fps for JPEG/SMP source.
# Default Value: 5
#=============================================================================
#
###########################  WINDOW PLACEMENT  ###############################
#
# Video windows are placed by the video server from left to right (and top
# to bottom). A simple slot scheme with fixed-size slots is used.
# Set these values to determine size and arrangement of the slots.
#
#AVC_MaxSlots             12
# Description:   Maximum number of video windows that can be placed.
# Default Value: 12
#=============================================================================
#AVC_SlotsInRow           3
# Description:   Maximum number of columns.
# Default Value: 3
#=============================================================================
#AVC_SlotWidth            330
# Description:   Horizontal dimension of a video-window slot.
# Default Value: 330
#=============================================================================
#AVC_SlotHeight           268
# Description:   Vertical dimension of a video-window slot.
# Default Value: 268
#=============================================================================
```