



Reliant UNIX *ONLINE Documentation*

Reliant UNIX 5.45

Reliant UNIX Operation

RM200, RM300, RM400, RM600

Edition September 2000

Copyright © 2000: Fujitsu Siemens GmbH

Identification: U42086-J-Z915-2-76

Copyright and Trademarks

All rights reserved. □

Delivery subject to availability; right of technical modifications reserved.

All hardware and software names used are trademarks of their respective manufacturers.

1 Preface

UNIX® (UNIX is a registered trademark of The Open Group) is a very powerful operating system used in computers in different performance classes. Fujitsu Siemens has extended this operating system and adapted it specially to suit the hardware used in its own systems. The operating system therefore has a special name: Reliant® UNIX®.

This manual describes how to operate Reliant UNIX. It applies for all RM models, i.e. the RM200, RM300, RM400 and RM600.

The manual also explains the administration of the Reliant UNIX operating system. However, only information that is absolutely essential is provided here. A detailed introduction can be found in the manual "[System Administrator's Guide](#)".



Reliant UNIX is a licensed product. It may only be installed on computers for which a license has been acquired.



You will find a *readme* file on the hard disk as well as on the CD-ROM supplied. This file contains important information and changes that were not available when the manuals went to print. You should therefore make sure to read this *readme* as soon as possible.

1.1 Target group

This manual is intended primarily for individuals who are involved in the administration and support of the computer (generally referred to as system administrators). You only require a limited knowledge of UNIX for most of the sections in the manual. If extensive knowledge of the system is required, this will be indicated in the relevant sections.

1.2 Changes since the last version of this manual

The documentation has been updated to take into account the software version Reliant UNIX 5.45B.

1.3 Requirements

The following requirements must be met before you begin any of the tasks that are described in this manual:

- The computer hardware must be properly installed (see the Operating Manual).
- All hardware devices and controllers must be properly installed.
- You should know how to switch the computer on and off.

1.4 Summary of contents

The [Chapter "Basic information about Reliant UNIX"](#) explains the individual boot phases on the different RM systems. It also describes how to log on to and off from the system, how to configure the environment and how to use the online help. You will also find a short description of the *SYSADM* and *WebSysAdmin* user interfaces.



You must read the [Chapter "Basic information about Reliant UNIX"](#) **before** you begin working with Reliant UNIX.

The [Chapter "Operating Reliant UNIX"](#) explains how to perform the most important administration tasks. This chapter requires extensive knowledge of the UNIX operating system.

The [Chapter "Processing software with SYSADM"](#) describes how to install software that is not part of the Reliant UNIX system software (or not part of the basic system to be more exact). It also describes how to deinstall software and how to get an overview of the installed software.

The [Chapter "Node administration with WebSysAdmin"](#) introduces readers to the new *WebSysAdmin* user interface. Its *VConfig* component replaces *XConfig*. The most important operating elements are also explained.



This manual reproduces all input and output (console dialog) which is typical of the respective situation. The output on the screen may differ somewhat from the text that is reproduced here. This may be because your computer is configured differently or because the installation procedure was slightly modified.

1.5 Notational conventions

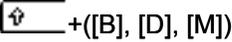
The following notational conventions are used in this manual:

Italics Names of files, programs, commands, variables, and options, as well as screen references in the main body of text, such as input fields, text fields and menus.

Command (number) The number following a command indicates the chapter in which the command is described in one of the following manuals:

- "Commands. User's Reference Manual"
- "System Administrator's Reference Manual"
- "Networking Reference Manual"

The brackets and number are not part of the command, i.e. they must not be entered.

Bold	Used for highlighting purposes in the main body of text.
Typewriter text	System output (error messages and other messages) and file entries.
Bold typewriter text	User input in examples, with the exception of variables (see above).
"Quotes"	References to other chapters or manuals. □
	System output or user input in screen masks. □
[D] □ □ [Ctrl] □ □ Ú	Keys on the keyboard.
[HELP] □ □ [ENTER]	Function keys in the mini operating system and in SYSADM
[Ctrl]+[C]	Key combination: A plus sign between a series of keys means that the keys must be pressed at the same time.
[#], [Q], Ú	Key sequence: A comma between a series of keys means that the keys must be pressed one after the other.
	If a key sequence is part of a key combination, it is enclosed in parentheses. Example: Press and hold the  key, and then press [B] followed by [D] followed by [M].
	Actions performed by the user.
	Additional information, notes and tips.
	Warning that must be heeded.

2 Basic information about Reliant UNIX

This chapter contains the basic information you need to know for administration of Reliant UNIX. Additional information on operating Reliant UNIX can be found in [Chapter "Operating Reliant UNIX"](#).

It is beyond the scope of this manual to provide a comprehensive description of the basics of a complex operating system like UNIX®. For further information, please refer to the recommended documentation.

2.1 Boot phases

When the computer is switched on, various control programs and utilities kick into action before the operating system is booted. Several phases are run through in this process. [Figure "System startup phases on the RM200 - RM400"](#) and [Figure "System startup phases on the RM600 system"](#) provide an overview.

Loading or booting refers in general to a process whereby a loader routine loads an operating system completely into main memory, handing over control to it. The various phases that are run through during the boot process are described below separately for the RM200 - RM400 and RM600. You will find a detailed description in the [Chapter "Operating Reliant UNIX"](#).

2.1.1 Boot phases on the RM200, RM300, RM400

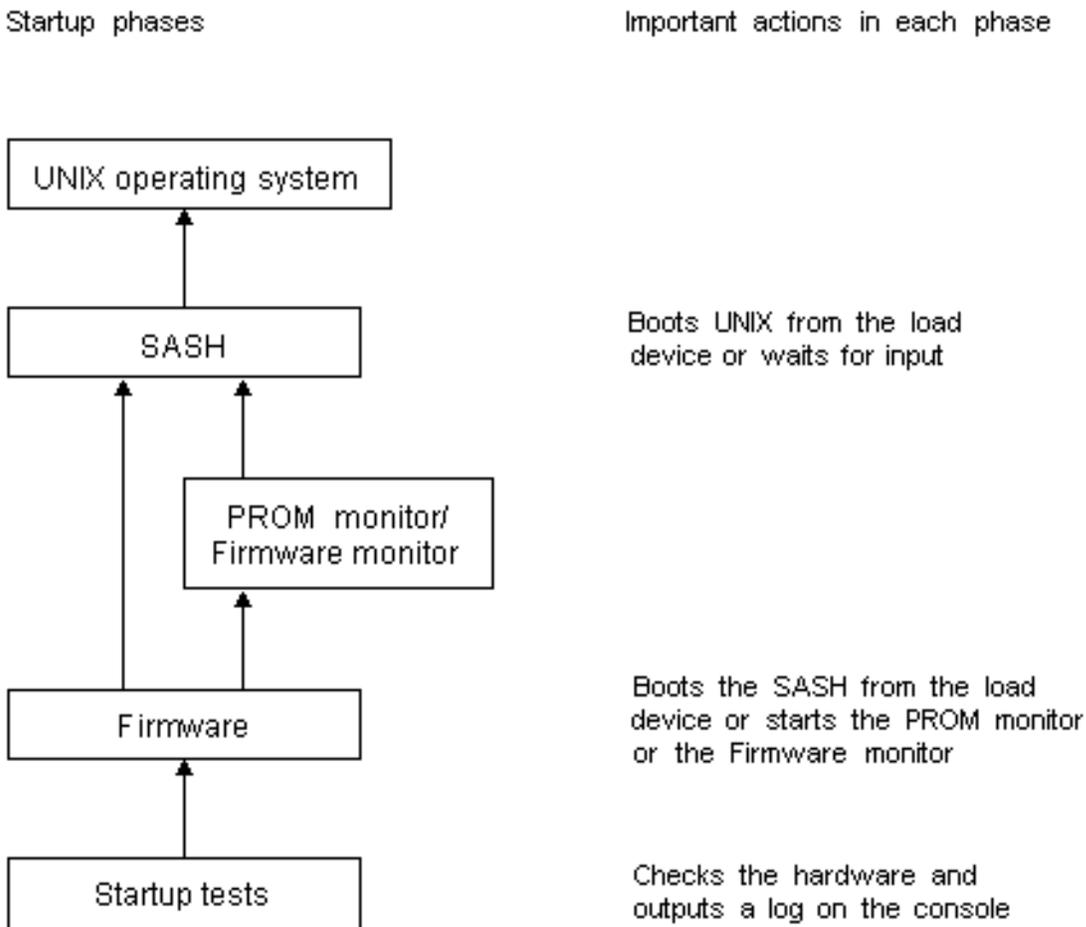


Figure 1: System startup phases on the RM200 - RM400

Firmware

The firmware is started automatically when the computer is switched on. It checks the hardware and logs the results on the console. The listing contains a test status for each module tested. If an error is detected in a module, the firmware stops the computer from starting up. In this case, switch off the computer and switch it on again after a few minutes. If the same error is displayed again, please notify the Fujitsu Siemens Service department.

PROM monitor (computer without graphics card)

The PROM monitor (PROM = Programmable Read Only Memory) is stored as firmware in the PROM. It is started when autoboot is switched off or if an error occurs during the system startup test (boot mode *d*, *e* or *m*; for the RM200 and 200c, with alphanumeric consoles, boot mode *s*, with graphics consoles, boot mode *c*). It waits for user input with the >> prompt. Different environment variables can be set in the PROM monitor.

Information on how to switch boot mode from the PROM monitor and how to start the SASH manually can be found in the [Section "PROM monitor \(RM200 - RM400\)"](#). Further details about the PROM monitor can be found in the manpage for *prom*(8) (see also the [Section "Online display"](#)).

Firmware monitor (computer with graphics card)

On a computer with a graphics card, a firmware monitor appears instead of the PROM monitor. This is a menu-based user interface, from which all PROM monitor commands can be started. Information on how to switch to boot mode from the firmware monitor and how to start the SASH manually can be found in the [Section "Firmware monitor \(RM200 - RM400\)"](#).

SASH

The SASH (stand-alone shell) is a system interface that operates independently of the operating system. The SASH is located in a specific area of the system disk (in partition 10). It is loaded by the firmware into main memory and started. Depending on the boot mode set, it either loads the Reliant UNIX operating system automatically or waits for user input with the *sash:* prompt if the system is started manually.

The commands that are available in the SASH are listed in the [Section "SASH \(RM200 - RM400\)"](#). A reference list of all SASH commands can be found in the manpage for *sash*(8) (see also the [Section "Online display"](#)).

Reliant UNIX operating system

A system kernel is loaded into main memory when the operating system is "loaded" or "booted". The kernel resides as a single file on the system disk. By default, this file is called */stand/unix* or */unix* (hard link).

2.1.2 Boot phases for the RM600-xxx, RM600 E

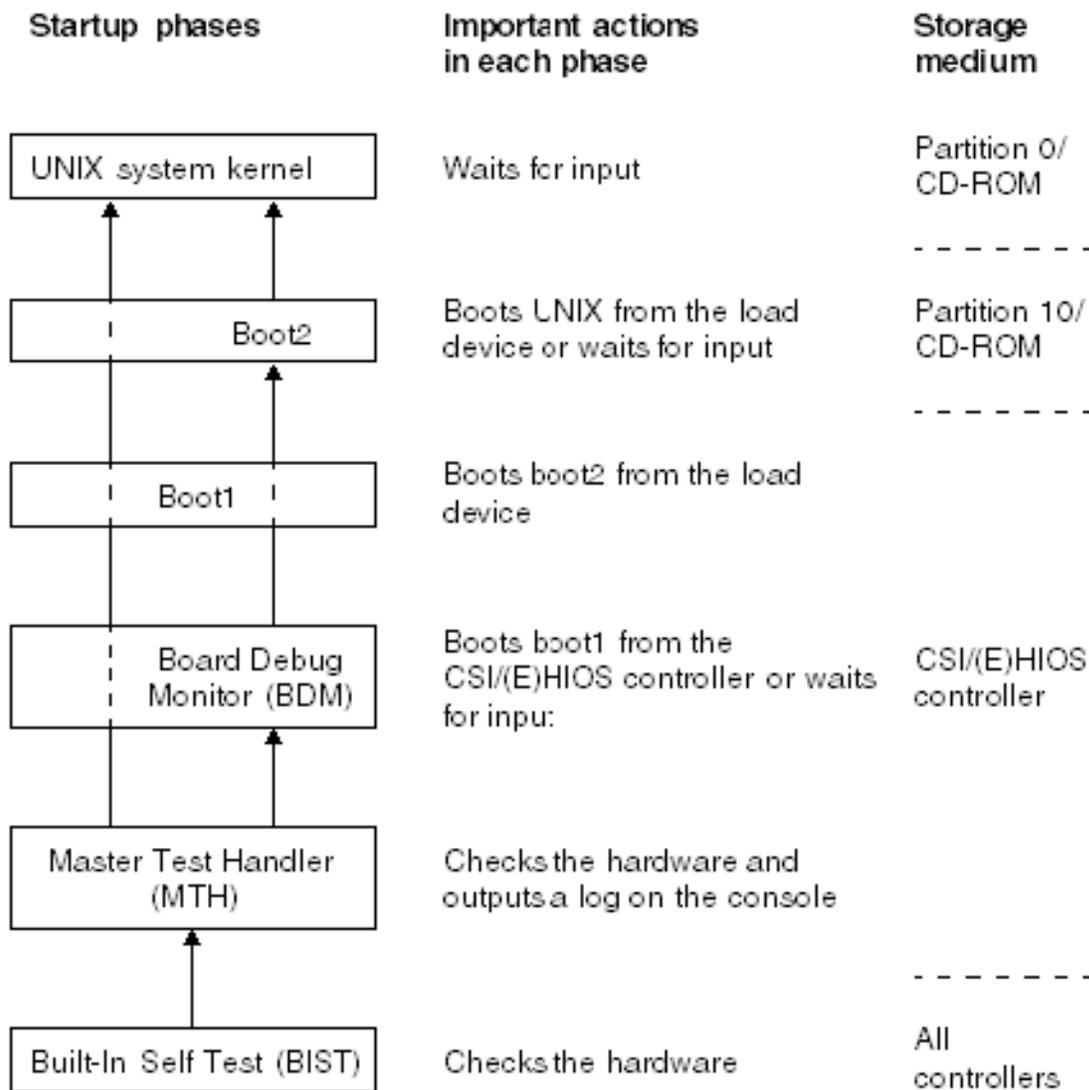


Figure 2: System startup phases on the RM600 system

System startup parameters

The load parameters required for system startup are stored on a non volatile storage medium, the NVRAM (Non Volatile Random Access Memory). The NVRAM resides in a special controller:

RM600-xxx:

in the CSI controller (CSI = Central Services & Interfaces)

RM600 E:

In the (E)HIOS ((E)HIOS = (Enhanced) High Performance Input Output System) basic controller (hereafter referred to as (E)HIOS for short)

The (E)HIOS basic controller refers to the (E)HIOS controller which is located in a specific slot in the SP bus (SP bus = Synchronous Pipelined Bus) system. This slot has the following slot ID:

Model	Slot ID	Controller type
E20/E60	□ 8	HIOS
E30/E4x	10	EHIOS

E70/E8x	14	EHIOS
---------	----	-------



See the Operating Manual for the RM600 E.

Built-in self tests

The built-in self tests (BIST) are started automatically when the computer is switched on. They check all of the controllers and store the results in the CSI or (E)HIOS controller. These results are then queried by the master test handler.

Master test handler

The master test handler (MTH) starts working automatically once the built-in self tests have checked the controllers. It queries the results of these tests, checks other hardware modules and outputs a log on the console. This log contains a test status for each module that was checked. If a module contains an error, it cannot be accessed subsequently. If this is the case, reboot the system. If the same error is displayed again, please notify our Service department (see the [Chapter "Telephone and Fax Numbers of the Support Center"](#)).

Board debug monitor

The board debug monitor (BDM) is a monitoring program on the CSI/(E)HIOS board to which the console and possibly also the system disk are connected. It enables you to use commands to access various system parameters, even if Reliant UNIX or the CPU (Central Processing Unit) is not available. The BDM starts working automatically once the MTH has completed its tests correctly. Depending on the boot mode that you set, the BDM either automatically starts boot1 or waits for user input.

The BDM outputs the following prompt:

```
RM600-xxx:      BDM(MBII□0□t)>
RM600 E20/E60: BDM(SPB□□8□t)>
RM600 E30/E4x: BDM(SPB□10□t)>
RM600 E70/E8x: BDM(SPB□14□t)>
```

For information on how to switch the boot mode from the BDM or how to start boot2 manually, please refer to the [Section "Setting the boot mode"](#).

Boot1

Boot1 (first-level boot) loads boot2 from the load device (hard disk or CD-ROM) into main memory and starts it.

Boot2

Boot2 (second-level boot) is a system interface that functions independently of the operating system. It is located in partition 10 of the system disk. It is loaded into main memory and started by boot1. Depending on the boot mode that was set, it either loads the Reliant UNIX system kernel automatically, or waits for user input with the prompt boot2:.

Boot2 has a similar function to the **SASH** (stand alone shell) on the RM200, RM300, RM400 and RM600-xx systems. With the RM600-xxx and RM600 E, the term SASH is occasionally used as a synonym for boot2.

The commands that are available in boot2 are listed in the [Section "Boot2 \(RM600\)"](#). A reference list of all boot2 commands can be found in the manpage for *boot2(8)* (see also ...).

Reliant UNIX system kernel

A system kernel is loaded into main memory when the operating system is "loaded" or "booted". The kernel resides on the system disk as an executable file. By default, this file is called */stand/unix* or */unix* (hard link).

2.2 Logging on and off

Various steps are required in order to log on to and off from the system. The steps involved depend on

whether or not you are using a graphics console (possible on the RM200 - RM400 or PC).

2.2.1 Logging on to computers without a graphics console

Logging on to the system

As soon as the boot process is complete, the following message appears:

```
The □system □is □ready.  
Console □Login: □_
```

This is how you log on to the computer:

▶ Enter your login name.

▶ Confirm the entry by pressing **Ú**.

You are then prompted to enter your password.

▶ Enter your password.

To ensure data protection, the characters you enter for the password are not displayed on the screen.

▶ Confirm your entry by pressing **Ú**.

The system only accepts your login if

- your login name has been entered in the */etc/passwd* file,
- the password that you entered is correct,
- the user's *\$HOME* directory exists.

The **system administrator** is the user who is logged on under the login name *root*. Only the system administrator is authorized to execute certain administration tasks. The rights required for this are also referred to as root rights.

2.2.2 Logging off from computers without a graphics console

Before you log off, you must ensure that all active applications have been terminated correctly so that you do not lose any data.

Perform one of the following steps to log off from the system:

▶ Start the `exit(1)` command

or

▶ Press [Ctrl] + [D].

The `stty eof` parameter must not have been redefined.

The following message is displayed again:

```
Console Login: _
```

2.2.3 Logging on to computers with a graphics console (RM200 - RM400)

If SINIX/windows User Environment has been installed and started on the computer, a login window is displayed.



You can switch to ANSI mode at any time using the key combination [Alt], [S-Abf] + H] (German keyboard) or [Alt] [SysRq] and [H] (English keyboard). Input and output then correspond to a computer without graphics. For further information on this topic, please refer to the [Section "Logging on to computers without a graphics console"](#).

You can switch back to graphics mode using the key combination [Alt], [S-Abf] + [F1] (German keyboard) or [Alt] [SysRq] and [F1] (English keyboard).

The individual components of the login window as well as the operation of SINIX/windows are described in the manual "SINIX/windows User Environment V3.0 (CDE) - User Guide". You should refer to this manual if you would like to learn about SINIX/windows in detail; some brief information is given below. If you are familiar with the operation of Microsoft® Windows™, you will find it very easy to use SINIX/windows.

Work through the following steps to log on to the computer:

- ▶ Double-click on the icon containing your login name. The name is then displayed in the *Login:* input field. The *Password:* input field is then activated.
- ▶ Enter your password via the keyboard and confirm your entry by pressing **Ú**.
To ensure data protection, the characters you enter for the password are not displayed on the screen. The system only accepts your login if
 - your login name has been entered in the */etc/passwd* file,
 - the password that you entered is correct,
 - the user's *\$HOME* directory exists.



If you cannot log on, check that all of the requirements listed above have been fulfilled. The **system administrator** is the user who is logged on under the login name *root*. Only the system administrator is authorized to execute certain administration activities. The permissions required for this are also referred to as root rights.

If your entries are correct, the login window is closed and the graphical user interface with the Control Panel appears after a short time.

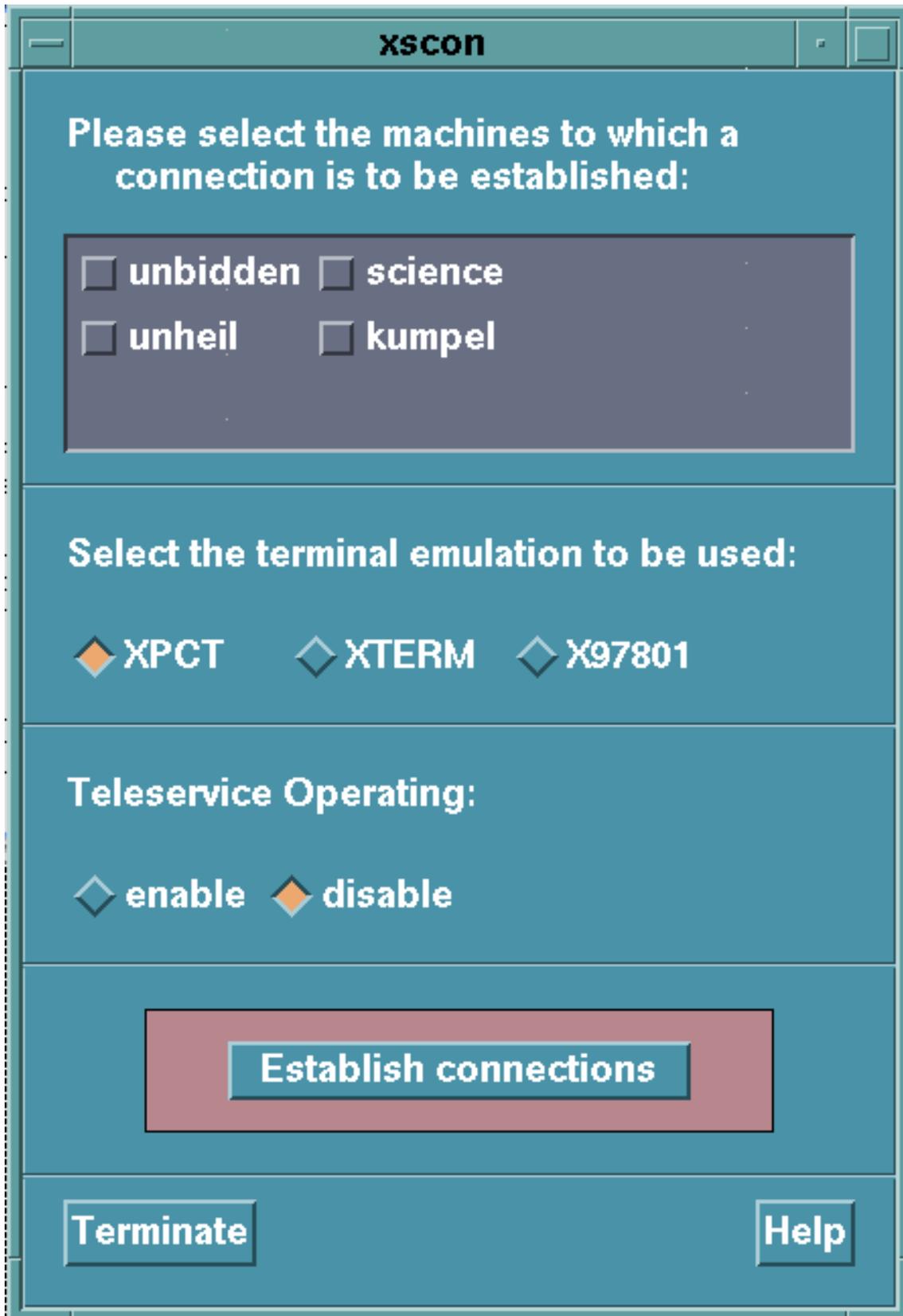
2.2.4 Logging on to computers with a graphics console (RM600)

If you are using an RM200 or RM300 as a console and want to administer one or more RM600 systems, you can display their console screens on your graphics terminal (refer to the RM200 or RM300 Operating Manual).

To do this, you have to:

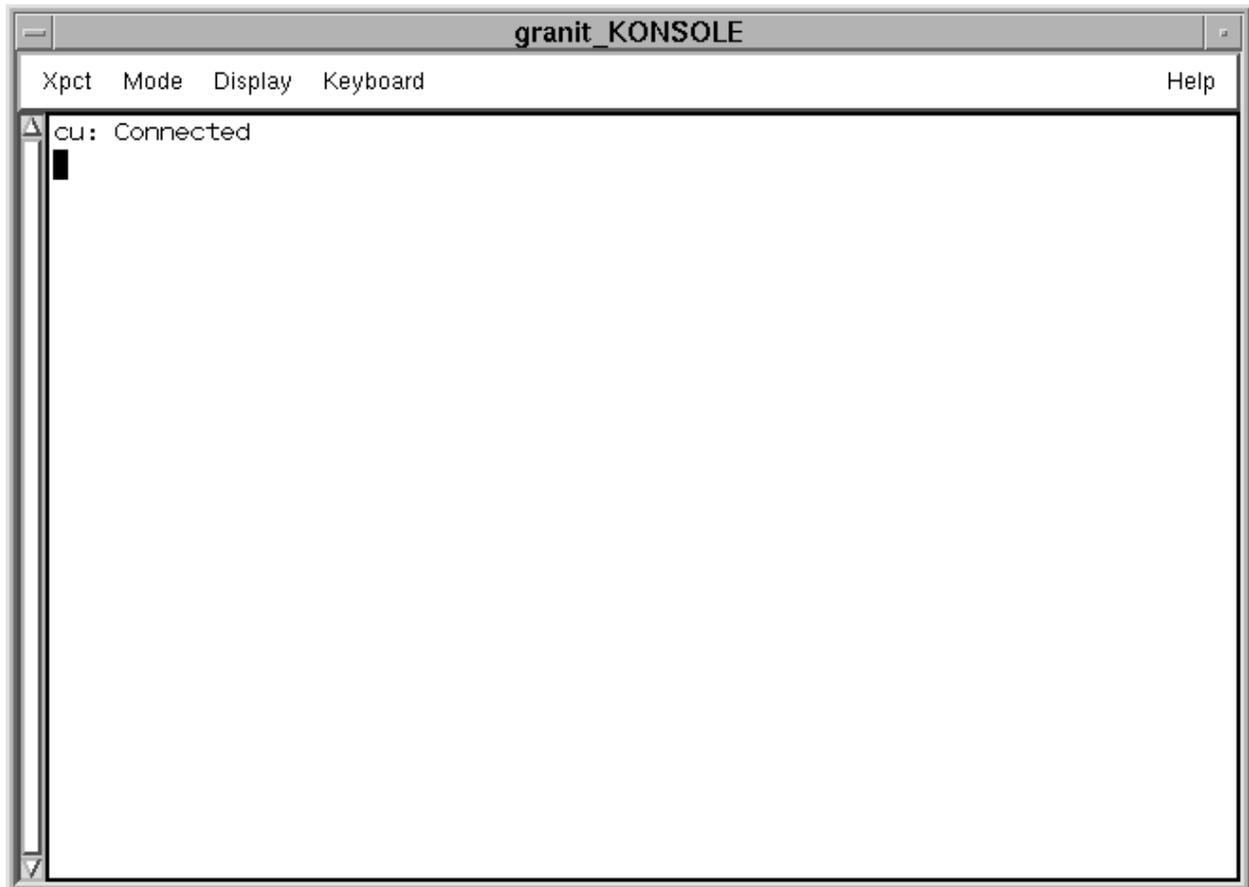
- ▶ Start SINIX/windows User Environment.
- ▶ Start a terminal emulation.
- ▶ Start the *xscn* program.

The following window is opened:



- ▶ Select one or more computers from the top section of the window.
To do this, click on the appropriate check box(es) with the mouse.
- ▶ Select the desired terminal emulation.
- ▶ Activate Teleservice mode in the center area of the window if required.
- ▶ Click on *Establish connections*.

The connection to the selected computers is established with the `cu(1)` command. The `xscn` program is then terminated immediately.



- ▶ Press `Ú`.

You are then prompted to log on to the system (see the [Section "Logging on to computers without a graphics console"](#)).

To ensure that `xscn` can establish the `cu` connection to the desired computers, appropriate entries must exist in the `/etc/uucp/Systems` and `/etc/uucp/Devices` files. The consoles must be entered using the alias `SconX` ($X = 0 \dots n$) in this case.

If a connection cannot be set up (e.g. because the console of an RM600 is already connected to another terminal), you will receive the error message

```
cu: Connect failed: DEVICE LOCKED
```

instead of the message

```
cu: Connected
```

shown above. In this case, the window is closed automatically after 10 seconds.

For further information, please refer to the manpage for `cu(1)`.

2.2.5 Logging on to the system using a computer with a LAN console

If you have a LAN console and want to use this to manage one or more RM systems, you can display the console screen of these systems on your graphical terminal (see the Operating Manual for the LAN console).

To enable this you must carry out the following operations:

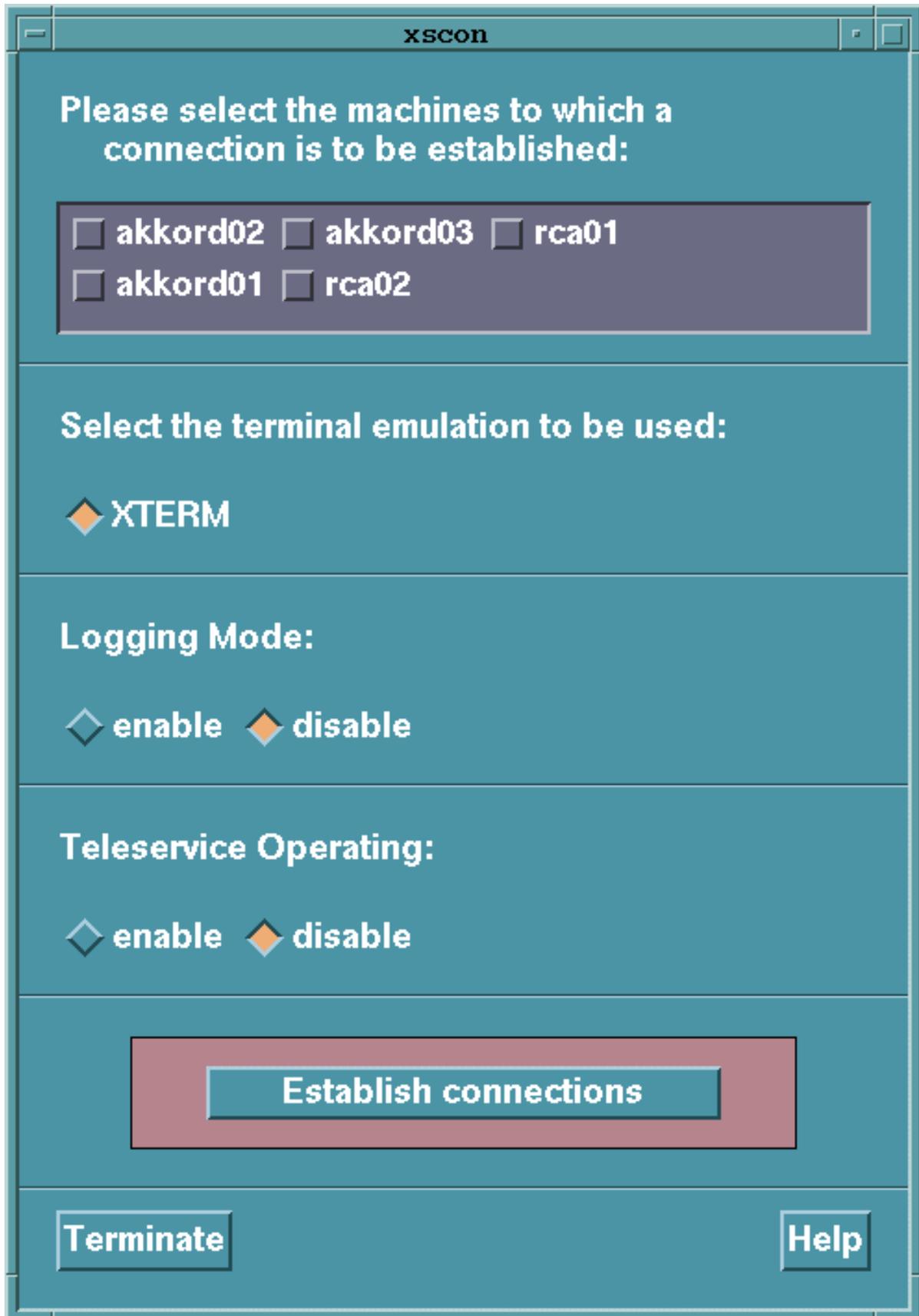
- ▶ Start a terminal emulation, if one is not already started.



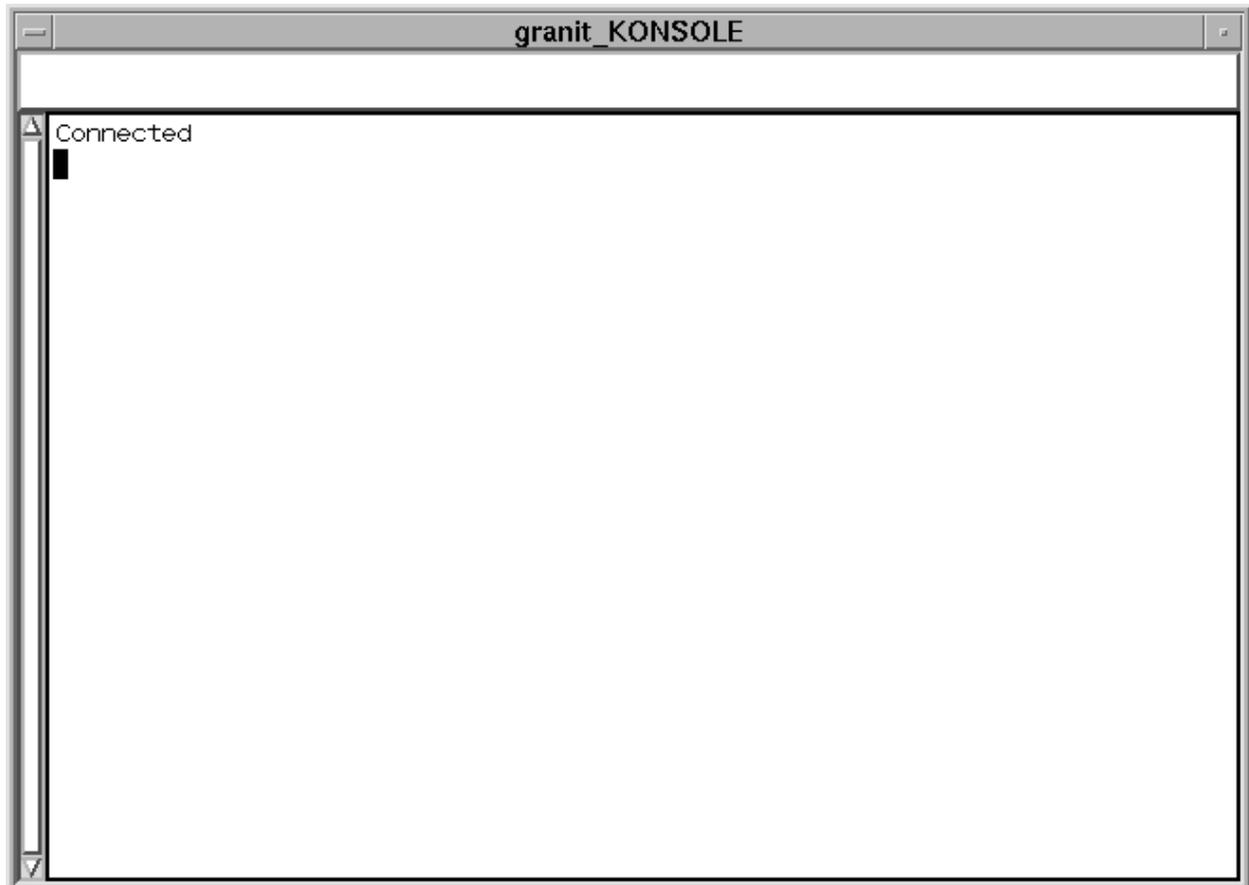
It is also possible to call up the *xscn* interface via the Solaris Common Desktop Environment. This has a corresponding entry in the "Application Manager".

- ▶ Log on as root.
- ▶ Start the *xscn* program.

This opens the window shown below:



- ▶ In the upper part of the window, select one or more computers.
To do this, use the mouse and click on the appropriate buttons.
- ▶ If required, activate logging mode.
You can use the menu command *Logging mode* to select whether the outputs from the console are to be logged. Further information about this can be found in the LAN console manual.
- ▶ If required, activate the Teleservice operating option in the center of the window.
- ▶ Click the *Establish connections* button.
Connection to the selected computers is carried out in the same way as for *cu(1)*. The *xscn* program is then terminated immediately.



- ▶ Press the Ú key.
You will then be asked to log on to the system (see the [Section "Logging on to computers without a graphics console"](#)).
- If a connection cannot be established (for example, because the console of an RM600 is already connected to another terminal), you will not be asked to log on to the system;
- Connected
- you will instead receive the following error message
- Connect failed:
- If this happens the window will close automatically after 10□seconds.

2.2.6 Logging off from computers with a graphics console (RM200 - RM400)



Before you log off, you must ensure that all active applications have been terminated correctly so that you do not lose any data.

Work through the following steps:

- ▶ Click on the relevant control element (CDE Desktop) with the left mouse button (in the middle of the control panel).
- ▶ Click on the icon in the top left corner of the control panel with the left mouse button. A pull-down menu appears.
- ▶ Select the menu item *Logout...*
- ▶ Use the right mouse button to click on a free area in the workspace. The *Workspace Menu* appears.
- ▶ Select the menu item *Logout...*

In all cases, a window appears in which you are prompted to confirm that you want to terminate SINIX/windows.



The following applications are **not** backed up **automatically**:

- Applications that are run in terminal emulation windows.
- Non-desktop applications that do not perform an automatic status backup during log off.

You must therefore close these applications before you log off.

- ▶ Click on the *Continue Logoff* button.

The current SINIX/windows configuration is backed up. When you log on again, your work environment is the same as it was when you left it.

All windows are then closed automatically. After a few seconds, the login window for SINIX/windows User Environment is displayed again.

2.2.7 Logging off from computers with a graphics console (RM600)



Before you log off, you must ensure that all active applications have been terminated correctly so that you do not lose any data.

There are several ways of shutting down the *cu* connection to a console:

- ▶ Enter: `~.Ú`
or
- ▶ Click on the *Xpct* or *Xterm* item in the menubar of the window.
A pull-down menu appears.
- ▶ Click on the *Exit* menu item.
or
- ▶ Click on the control menu box.
A pull-down menu appears.
- ▶ Click on the *Close* menu item.

The terminal emulation window is then closed.

2.2.8 Logging off from computers with a graphical LAN console



Before you log off, you must ensure that all active applications have been terminated correctly so that you do not lose any data.

There are several ways of shutting down the connection to a console:

- ▶ Enter: `~.Ú`
or
- ▶ Use the mouse to click on the window menu button.
A pull-down menu appears.
- ▶ Click on the *Close* menu item.

The terminal emulation window is then closed.

2.3 Important environment variables

Before the # prompt is displayed, the shell sets up the "environment" for you. To do this, it assigns values to certain shell variables that are defined in the `/etc/profile` file and in the `$HOME/.profile` file, if this exists. These shell variables are also called environment variables for this reason.

Important environment variables are listed below:

HOME

"Home" directory, default: login directory

You access this directory by invoking the `cd` command with no arguments

LANG

Language in which the system messages are displayed

LOGNAME

Login name under which you logged on to the system

PATH

Names of all the directories through which the shell is to search in order to find programs that you can invoke

PS1 Prompt characters

TERM

Terminal type

TZ Time zone

Displaying environment variables

If you want to know the value of an environment variable, invoke the `set` command:

```
# set
```

Changing environment variables

You can change the value of environment variables in either of two ways: You can change the variables temporarily, i.e. only for the current session, or permanently, i.e. the change remains valid for all future sessions. The procedure for changing environment variables is described here using the *LANG* variable as an example. Set this variable to English.

temporary:

Enter the following commands:

```
# LANG=En_US.88591
# export LANG
```

permanent:

Add an appropriate definition either to the */etc/profile* file or to the *\$HOME/.profile* file.

```
...
LANG=En_US.88591 export LANG
...
```

The difference between the two files is as follows:

- Definitions in the */etc/profile* file apply to all users.
- Definitions in the *\$HOME/.profile* file only apply to one user.



You have to add the following line to the *.profile* file so that German umlauts can be displayed correctly in graphics mode:

```
# stty cs8 -istrip echoe
```

The changes only take effect the next time you log on to the system. However, you can also execute the modified file as follows:

```
# . $HOME/.profile
```

2.4 Online display

The Reliant UNIX operating system contains numerous commands, most of which are described in the manuals "

Commands. User's Reference Manual" and the " System Administrator's Reference Manual". The command descriptions can also be displayed online. The *man*(1) command is provided for this purpose.



The online command descriptions (also termed manpages) will no longer be generated automatically as of Version 5.44 A00. You can, however, install the manpages as an optional product following installation. Please refer to the [Section "Installing software products"](#) for further information.

First of all, you should find out how to use the online display. To do this:

- ▶ Select the language in which the information is to be displayed. To do this, you must assign the desired value to the *\$LANG* variable. For example:

```
# LANG=En_US.88591 export LANG
```

- ▶ Enter the following command:

```
# man man
```

Information on the *man*(1) command is then displayed. This information is for the most part identical to the corresponding pages in the manual "

Commands. User's Reference Manual". This is why the online display of a command is also termed a "manual entry", "reference page" or, as already mentioned, a "manpage".



If there are differences between the online text and manual text, the online text should be regarded as the latest version.

The entire manpage comprises several pages each containing 56 lines, where 23 lines can be displayed at any one time. There is also a command line at the bottom of the screen. This line contains a few guidelines for using the online display. A cursor at the end of the command line waits for further input.

```
man: press 'h' for help, 'q' to skip, or <space> to continue: _
```

- ▶ Read the manpage for the *man(1)* command.

- ▶ Then press one of the following keys:

- [H] key

- Displays instructions for using online help.

- When you press the space bar, the program displays additional instructions. If you press [Q] and Ú in succession, you return to the manpage.

- Space bar

- Continues the display of the manpage.

- Ú key

- Allows you to scroll line by line through the manpage or in the online help.

- [Q] key

- Quits the display of the manpage.

Since additional manpages were found in this case for the keyword *man*, these are displayed in succession. If you are not interested in this information, press [Q] a number of times until the online display is terminated and the # prompt appears.

If you require information on any *command*, call the online help as follows:

```
#□man□command
```

Online display on a graphical screen

- ▶ Invoke the application manager.

- ▶ Double-click on the *Desktop applications* folder with the left mouse button.

- ▶ Double-click on the *Display window for manpages* icon with the left mouse button.

- ▶ Type in the required command.

2.5 DOCetc – Online documentation

DOCetc® is the online documentation system from Fujitsu Siemens. This modern server program offers numerous functions for retrieving information and is characterized by the following features in particular:

- The online documentation for Reliant® UNIX® and its add-on products can be viewed with any HTML3.2-enabled browser from any system that has access to the URL of the *DOCetc* server program.
- A built-in search engine makes it easier to locate words, expressions or phrases in all installed *DOCetc* manual collections (A *DOCetc* manual collection is a collection of *DOCetc* online manuals.).
- The collections contain structured documents implemented with full text. These are converted on the fly (= during loading) to HTML for any Web browser and appear in the appropriate output format on the screen.

Installing the *DOCetc* online documentation

To use the online documentation, you should first install the *DOCetc* server software and then the *DOCetc* manual collections.

The *DOCetc* server software is supplied with the "CD-SYS-MI" CD-ROM and installed with *cdinstor* under *Sysadm*. It includes the programs that prepare the online manuals for on-screen display as well as the *DOCetc* administration, a web-based menu-driven program that you can use to install manual collections and create a user menu.



The description of how to install the *DOCetc* server software can be found in the readme file for *DOCetc* server software on the "CD-SYS-MI" CD-ROM.

Using the *DOCetc* server software

The *DOCetc* user interface can be used intuitively and is largely self-explanatory. The integrated help function also provides detailed explanations. You will find a comparable environment on the online documentation server of Fujitsu Siemens under the URL <http://manuals.mchp.siemens.us>

Administration of server software and manual collections

Various functions are available for managing the *DOCetc* server software as well as the manual collections that are located on the server. Refer to the appropriate sections in the readme file for the server software on the "CD-SYS-MI" CD-ROM.

2.6 Readme file

The readme file contains important information and changes that were not available when the manuals went to print. The readme file is stored under the name *readme*

- in the */opt/readme/sinix.GB* directory on the system disk
 - for RM200 - RM400: On the CD-ROM "Reliant UNIX" in the */cdrom/sinix_n/5_45b00/en* directory
 - for RM600: On the CD-ROM "Reliant UNIX" in the */cdrom/sinix_y/5_45b00/en* directory
- ▶ Read the readme file as soon as you have started up your computer or as soon as you have received a new revision version of Reliant UNIX.

You can display the readme file

- using the commands *pg(1)*, *more(1)*, *less(1)*,
- using an editor (e.g. *vi(1)*),
- using the *SYSADM* user interface.

Reading the readme file from the system disk (UNIX running)

You can use the *SYSADM* user interface to install the readme file on the system. A detailed description of this can be found in the [Chapter "Processing software with SYSADM"](#). You can then read the readme file as follows:

```
# pg /opt/readme/sinix.GB/readme
```

Reading the readme file from the CD-ROM (UNIX running)

Use the *SYSADM* user interface. A detailed description of this can be found on [Section "Installing software products"](#), an example see

You can read the readme file at shell level as follows:

RM200 - RM400:

```
# mount -F hs /dev/ios0/sdisk005s0 /cdrom
```

```
# pg /cdrom/sinix_n/5_45b00/en*
```

RM600-xxx:

```
# mount -F hs /dev/ios0/sdisk006s0 /cdrom
```

```
# pg /cdrom/sinix_y/5_45b00/en*
```

RM600 E:

```
# mount -F hs /dev/ios0/sdisk000s0 /cdrom
```

```
# pg /cdrom/sinix_y/5_45b00/en*
```

Reading the readme file from the CD-ROM (mini operating system running)

Select the menu item *Read Readme File* from the mini operating system main menu. The *pg(1)* command is called automatically with the readme file. If you type *q* after the colon, you return to the main menu of the mini operating system.

2.7 The SYSADM user interface

Although many of the administration activities can be carried out via the WebSysAdmin graphical user interface (see the [Section "WebSysAdmin user interface"](#)), there are still some things which are still more easily handled or which must be handled using the *SYSADM* user interface. This includes the configuration of LAN controllers for a new installation. This process can only be carried out using the *SYSADM* user interface, as *WebSysAdmin* requires an intact network connection.



A detailed description of how to use *SYSADM* is contained in the manual "[System Administration and Hardware Configuration Using the SYSADM User Interface](#)".

Various menu items in *SYSADM* are also described in the following chapters of this manual:

Menu item	Chapter
<i>software</i>	Processing software with SYSADM
<i>software_prod</i>	Processing software with SYSADM

Before you start *SYSADM*, you should make sure that the environment variables *TERM*, *LC_MESSAGES* and *LANG* have been set correctly. Their values depend on which terminal type and language you wish to use.



The user interface SINIX/windows User Environment and the terminal emulation x97801 were used for the screen dumps in this manual. The environment variable *TERM* is thus set to *97801*, and *LC_MESSAGES* and *LANG* to *En_US.ASCII*.

You must have **root authorization** in order to start *SYSADM*. If you are not the user *root*, you will be required to enter the *SYSADM* password.

Starting SYSADM (alphanumeric screen)

- ▶ To start *SYSADM* on an alphanumeric or graphical screen in ANSI mode, enter the following command:

```
#□sysadm
```

Starting SYSADM (graphical screen)

- ▶ Call the application manager.
- ▶ Double-click on the *System administration* folder with the left mouse button.
- ▶ Double-click on the *SYSADM* icon with the left mouse button.
- ▶ Alternatively, you can switch to the shell and start *SYSADM* from there (see above). You cannot use the mouse in *SYSADM*.

Using SYSADM

As soon as you start *SYSADM*, its main menu appears.

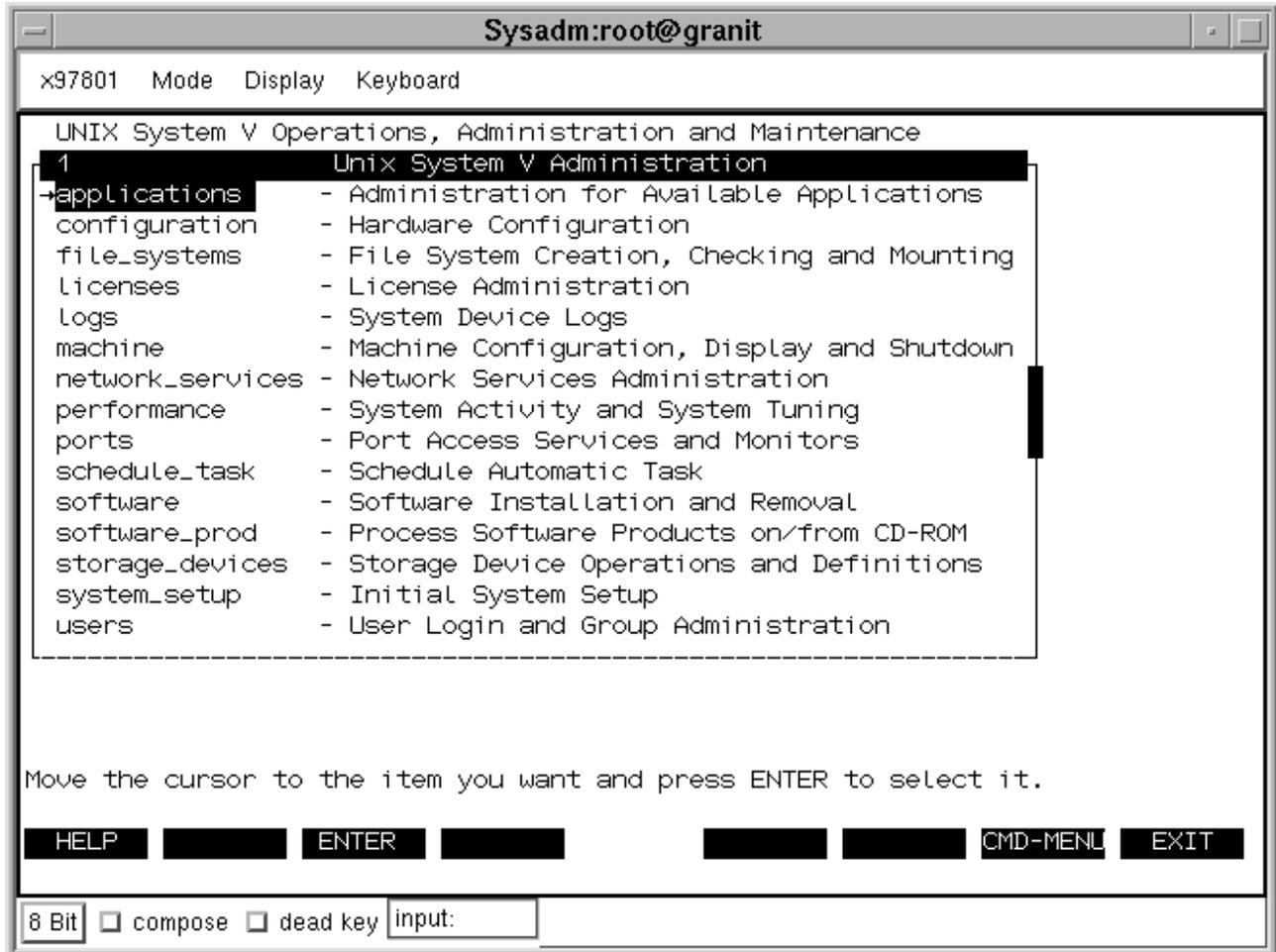


Figure 3: SYSADM main menu



Certain application programs generate additional entries in the *SYSADM* main menu.

The cursor points to the first menu item. When you select a menu item, additional windows are opened in succession. There are eight function keys in the last line of the screen. The assignment of the function keys depends on the program context.

- You can use the **→** and **[TAB]** keys to move the cursor to the next menu item or input field.
- You can use the **[HELP]** or **[F1]** function keys to display help text. The help text always refers to the input field in which the cursor is located.
- Use the **[CHOICES]** or **[F2]** function key to fill in the input fields. You can also enter data using the keyboard, but it is preferable to use the specified function key. When you use the function key, all permitted values are offered for selection.
- If you want to save the entries, press the **[SAVE]** or **[F3]** function key. If you do not want to change the default setting, press the **[CANCEL]** or **[F6]** function key instead.
- You can quit *SYSADM* using the **[CMD-MENU]** or **[F8]** function key.

If the function keys do not work, the relevant functions can be executed by pressing special key combinations:

- ▶ Press and hold down the **[Ctrl]** key.
- ▶ Then press **[F]**.
- ▶ Release both keys.

- ▶ Press the number key that corresponds to the function key whose function you want to execute. You must press the number key immediately after you have released the [Ctrl] and [F] keys.

Example: [Ctrl]+[F], [2] corresponds to [F2].

Printing information

In some windows you can also print the information displayed. This is always the case if the [PRINT] function key is offered. To do this, you must first define a suitable print command:

- ▶ Press [CMD-MENU].
The *SYSADM* command menu is displayed.
- ▶ Select the menu item *def-print*.

The following form appears:

The screenshot shows a terminal window titled "Sysadm:root@granit". At the top, it displays "x97801 Mode Display Keyboard". Below this is a menu titled "UNIX System V Operations, Administration and Maintenance" with a sub-menu "1-----Unix System V Administration-----". The sub-menu lists various system administration tasks, with "applications" selected and highlighted. Below the menu is a section titled "2 Define PRINT command" with a text input field containing "Print command: cat > /tmp/print1". Below the input field is the instruction "Fill in the form and then press SAVE." and a row of function keys: HELP, CHOICES, SAVE, CANCEL, CMD-MENU, and EXIT. At the bottom, there are checkboxes for "8 Bit", "compose", and "dead key", followed by an "input:" field.

- ▶ Enter a print command.

If you simply want to store the information in a file, you can use a command such as the one in the above example.

- ▶ Press the [SAVE] function key.

You can then print the information or redirect it to a file by pressing the relevant function key.

2.8 WebSysAdmin user interface

WebSysAdmin offers a web-based user interface for administration purposes. It allows efficient management and monitoring of both individual computers as well as entire administration domains.



WebSysAdmin comprises the following components:

- *VConfig* (hardware configuration)
- *Diagnostic* (node administration)
- *Accounts* (user administration)
- *Software* (software administration)
- *Tasks* (task definitions)

WebSysAdmin comes with a self-explanatory user interface, which allows intuitive handling in most cases. In addition, an online help system is available to provide information.

Refer to the [Chapter "Node administration with WebSysAdmin"](#) for a brief introduction to how *WebSysAdmin* works.

2.9 User profile (RM200 - RM400)

The Reliant UNIX operating system comprises numerous software packages, all of which can be installed on your computer's system disk if you so wish. In some cases, however, it makes more sense only to install some of the software. Reasons for this include:

- The system disk is less than 1 Gbyte.

- The applications that are to run on the computer only need some of the software.

The less system software you have installed on the system disk, the less space is required for it in the individual partitions. This saves space, which can then be used for other purposes.

You can choose between three different user profiles in order to help you decide which software to install:

- **Core System**

This user profile is designed for small systems. Only the minimum amount of software required to run the computer is installed. You can also decide whether or not to install the language supplement.

- **Application Client System – with developer support option**

This user profile is designed both for utility program users and software developers. You can also decide here whether or not to install the developer support and language supplement. However, **no server software** is installed.

- **Full System**

This user profile covers the complete Reliant UNIX software. It is designed for computers

- that have a sophisticated hardware configuration
- that can be used as servers
- on which a number of people can work



The software that is needed to operate a server is only installed when the *Full System* user profile is used!

In Table 1 you will find details of which software packages are installed, depending on the selected user profile:

Software packages	User Profile		
	Core System	Application Client System	Full System
Basic packages <input type="checkbox"/> – Commands <input type="checkbox"/> – <i>SYSADM</i> user interface <input type="checkbox"/> – Terminal support <input type="checkbox"/> – Environment for generating a <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> new system kernel <input type="checkbox"/> – System V spooler	yes	yes	yes
Compatibility packages	no	no	yes
Software for the LAN <input type="checkbox"/> – Drivers for Ethernet <input type="checkbox"/> – TCP/IP <input type="checkbox"/> – NFS – SNMP	yes	yes	yes
	no	yes	yes
X-Window system environment <input type="checkbox"/> (with a graphics console) <input type="checkbox"/> – X-Server <input type="checkbox"/> – Basic X-System <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> (Terminal emulators and <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> Window Manager) – Desktop <input type="checkbox"/> – WWW client	yes	yes	yes
	no	yes	yes
Font server for SINIX/windows User Environment	no	no	yes
European keyboard tables for ANSI consoles and x97801 emulation.	no	yes	yes
Diagnostic and configuration tools with user interface	no	no	yes
Support for <input type="checkbox"/> – Virtual and mirror disks <input type="checkbox"/> – High-availability <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> configurations <input type="checkbox"/> – CD jukeboxes	no	no	yes
NFS daemon for SINIX-PCs	no	no	yes
Developer support <input type="checkbox"/>	no	possible	yes

<ul style="list-style-type: none"> - Static libraries <input type="checkbox"/> - Header files <input type="checkbox"/> - Documenter's Workbench <input type="checkbox"/> - OSF/Motif and X clients <input type="checkbox"/> <input type="checkbox"/><input type="checkbox"/><input type="checkbox"/> (goodies) 			
Language supplement for installed packages	possible	possible	yes
Online manpages	no	possible	possible

Table 1: Software configuration of user profiles

If, while working on your computer, you find that you need more software from the Reliant UNIX basic system, you can subsequently install the necessary groups or packages using SYSADM by selecting *User Profile Management* under *software*. The same procedure can be used if you have installed too much software. A description of this is given in the [Section "Managing user profiles \(RM200 - RM400\)"](#).



It is only possible to install additional system software if the system disk partitions are larger than the minimum values specified in the manual "[Reliant UNIX Installation](#)".

2.10 System disk partitioning

Each hard disk is divided into a number of areas known as partitions. Several partitions can overlap. The current partitioning is stored in a partition table on the hard disk. For information on viewing this partitioning, refer to [Section "System disk partitioning"](#).

2.10.1 Approach (RM200 - RM400)

The hard disk containing the root file system (/) is called the system disk. This disk can be partitioned as follows:

- "Normal" partitioning

The system disk is generally divided into eight partitions. Six partitions contain the file systems */*, */proc*, */opt*, */usr*, */var* and */home*. One partition is occupied by SASH and another contains status information if the system disk is mirrored.

- One large partition

A special type of partitioning, which only contains the file systems */* (root) and */SAM*, was developed so that even relatively small hard disks (i.e. of less than 1 GB capacity) can be used as the system disk. The normal file system structure used in Reliant UNIX is generated here as follows:

1. Directories are created in the */SAM* file system (e.g. */SAM/home*).
2. Symbolic links to these directories are created from */* (e.g. *ln -s /SAM/home /home*).

In addition to this, there are two further partitions on the system disk whose position and standard size may not be changed, i.e. partitions 10 and 15.

1. Partition 10 contains the SASH which is required for booting the computer.
2. The status information is stored in partition 15 if the system disk is mirrored.

2.10.2 Approach (RM600)

The hard disk containing the root file system (/) is called the system disk. There are six partitions on the system disk that are numbered consecutively from 0 through 5. These partitions contain the file systems /, /swap, /opt, /usr, /var and /home. /swap refers to the primary swap area. It is used to temporarily swap out processes or parts of processes from main memory.

There are at least two other partitions on the system disk, whose position and standard size must not be changed, i.e. partitions 10 and 15.

1. Partition 10 contains the SASH, which is required for booting the computer.
2. The status information is stored in partition 15 if the system disk is mirrored.

Partition number	Device file	File system	Contents
0	/dev/ios0/sdisk011s0	/	root area
1	/dev/ios0/sdisk011s1	/swap	Primary swap area
2	/dev/ios0/sdisk011s2	/opt	
3	/dev/ios0/sdisk011s3	/usr	Commands, constant system data
4	/dev/ios0/sdisk011s4	/var	
5	/dev/ios0/sdisk011s5	/home	
6	/dev/ios0/sdisk011s6		
10	/dev/ios0/sdisk011s10		Boot2, header information from the hard disk
15	/dev/ios0/sdisk011s15		Status area for mirrored disk operation

Table 2: System disk partitioning



A partition 6 is available on disks with a size up from eight Gbytes with a size of four Gbytes (see also [Figure "Partitioning the system disk using the HD07 as an example"](#)).

Reliant UNIX can only be installed if the system disk partitions are a minimum size. Details can be found in the manual „Reliant UNIX Installation“.

2.11 Boot modes (RM600)

The operating system is booted when the RM600 is switched on (cold start) or restarted (warm start). Numerous operations are performed during the boot process. Load parameters can be used to determine which operations are executed. These include the bootflags and the bootstring. Detailed information on these parameters can be found in the [Section "Setting the boot mode"](#).

There are two modes for booting the operating system:

1. Logical boot mode: The logical boot mode is used by default:

RM600-xxx :	As of Reliant UNIX V5.43 and Firmware V2.42 for the CSI
RM600 E:	Generally

2. Physical boot mode: The **logical boot mode** offers the following **advantage**:

If the system disk is mirrored, its mirror disk can be connected to another controller in the system cabinet. This increases reliability as both controllers are unlikely to fail at the same time. The following configuration is recommended:

RM600-xxx :	System disk connected to any DCS, □ mirror disk to another DCS. □ To use the OLR (Online Replacement) facility for hard disks, (see operating manual and manpage for <i>olr(7)</i>), the system disk and mirror disk must be connected to different DCSs.
RM600 E:	Active system disk connected to any (E)HIOS controller, mirror disk to another (E)HIOS controller.

[Table ""](#) shows the attributes of both modes.

Attribute	Boot mode	
	Logical	Physical
Contents of the bootstring	log. device name, e.g. <i>sdisk011s0</i>	phys. device name, e.g. <i>sd(1,0,0)</i>
Possible boot media	Hard disk	Hard disk, CD-ROM
Use of the boot parameters	From the root partition list	From the <i>/etc/default/boot</i> file
Controller in the system unit to which the system disk can be connected		
<ul style="list-style-type: none"> — RM600-xxx: — RM600 E: 	CSI, DCS (E)HIOS	CSI (E)HIOS

Table 3: Attributes of the boot modes

Data belonging to the new system disk is entered in a root partition list during installation. This list is resident in the NVRAM (NVRAM = Non Volatile Random Access Memory) of the CSI or (E)HIOS basic controller. This ensures that the system can only be booted from a suitable system disk.

The load parameters that are set can be displayed as follows:

in the BDM: `BDM(SPB□□8□t)>ra`

in Reliant UNIX: `# bootflags -p`

The following message, for example, is then displayed:

```
Actual□reboot□values:
bootflags:□□0x00000200
bootstring:□sdisk011s0
```

Figure 4: Logical boot mode: Display of load parameters (example)

```
Actual□reboot□values:
bootflags:□□0x00000200
bootstring:□sd(1,0,0)
```

Figure 5: Physical boot mode: Display of load parameters (example)

If the bootstring is changed, the next time the system is booted it checks whether the new bootstring relates to the logical or physical boot mode. The boot mode is selected automatically based on this check.



If a root disk is replaced physically, the data on this root disk must be modified accordingly in the root partition list. The *disk_id* of the new root disk must be entered using the `boot2 root` command. The following steps must be performed:

- ▶ Boot the system to `boot2`
- ▶ Display the active root disk using the `root□-□l` command.
- ▶ Update the active root disk by entering `boot2:□root -n <entry> -p <phys_part>`

Where:

entry

is the number of the active root disk

phys_part

is the physical bootstring of the active root disk

This boot2 command enters the inquiry string of the new root disk as *disk_id* in the table (root partition list).

3 Operating Reliant UNIX

The interfaces available for node administration have already been explained in brief in the [Section "Boot phases"](#). The following table lists these again for the various systems:

RM600	RM200 - RM400
Board Debug Monitor	PROM/ Firmware Monitor (P-M/F-M)
Boot2	SASH
Reliant UNIX	Reliant UNIX

Table 4: Interfaces for node administration for the different systems

These interfaces are now described in detail below. The tasks that you can perform via the individual interfaces are also described. [Table ""](#) and [Table ""](#) provide an overview:

Task	Interface	Command on RM600
Boot the system manually after you have switched it on (cold start)	BDM□	<i>b</i>
	Boot2	<i>boot</i>
Modify the boot parameters	BDM□	<i>wf, wn</i>
	Reliant UNIX	<i>bootflags(8)</i>
Shut down the system and boot it again (warm start)	Reliant UNIX	<i>shutdown(1M)</i>
Shut down the system	Reliant UNIX	<i>shutdown(1M)</i>
Modify the run level	Reliant UNIX	<i>shutdown(1M), init(1M)</i>
Ascertain the active users	Reliant UNIX	<i>who(1), whodo(1)</i>
Inform the active users	Reliant UNIX	<i>wall(1M)</i>

Table 5: System administration tasks

Task	Interface	Command on RM200 - RM400
Boot the system manually again after you have switched it on (cold start, boot mode <i>d</i> or <i>m</i>)	P-M, F-M□	<i>auto, boot□</i>
	SASH	<i>boot -f <bootstring>unix</i>
Modify the environment variables for booting	P-M, F-M	<i>setenv bootmode (x)</i>
Shut down the system and boot it again (warm start)	Reliant UNIX	<i>shutdown(1M)</i>
Shut down the system	Reliant UNIX	<i>shutdown(1M)</i>
Modify the run level	Reliant UNIX	<i>shutdown(1M), init(1M)</i>
Ascertain the active users	Reliant UNIX	<i>who(1), whodo(1)</i>
Inform the active users	Reliant UNIX	<i>wall(1M)</i>

Table 6: System administration tasks (here: RM200 - RM400)



Please note that some of these tasks can only be executed

- if you are logged on to the system with the user ID 0 (i.e. with the login name *root*, *sysadm* or *admin*),
- if the system is operating in multiuser mode.

3.1 General guidelines

The tasks involved in system administration carry with them a high degree of responsibility. These tasks can affect the performance of the computer, but can also influence the work of each individual user.

For many administration tasks, it is necessary to exit "multiuser mode" run level and to switch to "single-user mode" (see the [Section "Overview of the run levels"](#)). However, an "ordinary" user cannot access the system at this run level. For this reason, you should only change the run level for your administration tasks if this does not interfere too much with the work of other users.

Inform other users well in advance of events that prevent use of the computer. When you shut down the system, you should also inform users of the time when you plan to boot the system again. To do this, use the *wall(1M)* command or the daily message (in the */etc/motd* file). Only under exceptional circumstances should you shut down the system without sending warning messages to the users.

If you want to exit multiuser mode, please observe the following recommendations:

1. Schedule tasks that reduce the performance level of the system for periods when demand on the system is as low as possible. Inform users of the periods when the system will not be accessible to them.
2. Prior to each administration task, check which users are active on the system. To do this, use the *whodo(1M)* and *who(1)* commands.
3. If other users are working on the system, you should give notice of planned modifications well in advance. If it is necessary to shut down the system without warning, you should send a corresponding message to all active users beforehand using the *wall(1M)* command. You should allow users sufficient time to finish their work properly and to log off from the system.
4. Note all administration tasks using system logs in a separate directory that will function as a "notebook". This can be extremely useful if you have to resolve a serious error in the system.

3.2 PROM monitor (RM200 - RM400)

When the computer is switched on, startup tests are performed (depending on the boot mode). The PROM monitor then responds in the following cases:

- if autoboot is switched off (boot mode = *d* or *m*),
- if an error was discovered during the startup test (boot mode is then automatically set to *e*).

The PROM monitor is a command interface for the firmware that is stored in the PROM. It responds with the prompt `>>`. You can then execute special system administration commands; in particular, you can set, modify or delete the environment variables that affect the boot process.

3.2.1 Switching to the PROM monitor

The procedure for accessing the PROM monitor is determined by two factors:

1. The current run level of Reliant UNIX.
2. Whether or not autoboot is switched on.

Switching to the PROM monitor after the computer is switched on

When the computer is switched on, the firmware executes a series of actions that correspond to the boot mode set. If autoboot has been switched off, you automatically gain access to the PROM or firmware monitor. If autoboot is switched on (boot mode *c*), you can then abort the boot process by pressing [Ctrl] [C] as soon as the following message appears:

```
Autoboot: Waiting to load dkncr(0,0,10)sash(Ctrl-C to abort, RETURN to expedite
```



dkpcs or *dkdpt* is output in place of *dkncr*, depending on the selected system disk.

Switching to the PROM monitor from Reliant UNIX

If the computer is already switched on and Reliant UNIX is running, you can access the PROM or firmware monitor by switching to run level 5. The boot mode setting is not important here.



If the system is currently in multiuser mode, you should use the `who(1)` command to check whether other users are logged on before shutting down the system.

```
# cd /  
# shutdown -i5 -g900
```

Switching to the PROM monitor from the SASH

Press [Ctrl] +[D] or issue the SASH `end` command.

3.2.2 Important commands for the PROM monitor

You can perform certain administration tasks in the PROM monitor. Various commands are available for this purpose. These commands are largely identical to those also available in the SASH. The table below provides an overview of the most important commands; all commands and options are described in detail in the manpage for `sash(8)`

Command	Function
<i>auto</i>	Loads the SASH, which in turn automatically boots Reliant UNIX. The command uses the <i>bootfile</i> environment variable to do this. This variable must contain both the device name and the file name of the SASH (e.g. <i>dkncr(0,0,10)sash</i>). If the variable does not contain a device name, the value of the environment variable <i>path</i> is placed in front of the file name.
<i>boot</i>	Loads the SASH. If a file name is not specified, the file named in the <i>bootfile</i> environment variable is loaded. If the variable does not contain a device name, the value of the <i>path</i> environment variable is placed in front of the file name.
<i>checkchid</i>	Checks the consistency of the values in NVRAM against those saved on the system disk.
<i>dkprint</i>	Returns the assignments of channel IDs to the numbers allocated by the firmware.
<i>help or ?</i>	Displays the syntax of commands.
<i>printenv</i>	Displays the values of environment variables.
<i>putchid</i>	Writes a backup of the channel IDs from partition 10 of the system disk into NVRAM.
<i>setenv</i>	Sets up a new environment variable or modifies the value of an existing environment variable.
<i>unsetenv</i>	Deletes an environment variable.

Table 7: Commands in the PROM monitor



Note that the *checkchid*, *dkprint* and *putchid* commands described in [Table ""](#) and the miniroot *rr* command are available only on C models and, as of 5.44B, also on E models.

The SASH can be booted from various media: from SCSI-I/O disks (IOS), from CD-ROM (DVD) drives, from tape drives and via the LAN. The *bootfile* environment variable defines the file to be booted when the *boot* (without parameters) or *auto* command is called. Both commands expect to find the SASH in partition 10 of the system disk. As of Reliant UNIX 5.44, the *bootfile* variable can have the following values, depending on the selected system disk:

bootfile	Boot medium
<i>dkncr(c,t,s)sash</i>	internal disk
<i>dkpcs (c,b,t,s)sash</i>	external disk (SNI SCSI controller)
<i>dkdpt(c,b,t,s)sash</i>	external disk (DPT SCSI controller)

The variables *c*, *b*, *t* and *s* stand for the following values:

Variable	Definition
<i>c</i>	number allocated by the firmware
<i>b</i>	SCSI bus number (always 0)
<i>t</i>	SCSI ID
<i>s</i>	partition number

If no value is specified, the variable is set to *0* by default.

The *bootfile* variable is automatically defined during installation when the system disk is selected. It can be determined with the *printenv bootfile* command.



You are strongly recommended to make a note of the value of the *bootfile* variable. If a system error occurs you may not be able to determine its value and it must then be entered explicitly (see also the [Section "Procedure in the event of errors \(RM600\)"](#)).

In almost all cases it is possible to exit the PROM monitor or to abort a command that has been issued. To do this, press [Ctrl] [D] or [Ctrl] [C]. The PROM monitor then waits for further user input.

3.2.3 Commands for loading the installation software

You can execute a number of tasks from the PROM monitor that are described in other chapters in the manual "[Reliant UNIX Installation](#)". These tasks are as follows:

- update the EEPROM firmware
- load the installation menu

The following additional commands are available for these tasks.

Command	Function
<i>install</i> □ <i>rinstall</i>	the installation menu is loaded from CD or□ remote installation
<i>update</i> ,□ <i>rpdate</i> ,	the EEPROM firmware is updated locally or□ remotely

Table 8: Commands for loading the installation software



If you want to execute one of the commands *install* or *update* you must provide the software required by the command on a CD-ROM. The CD-ROM (DVD) drive must have the SCSI address *5* (EISA bus) or *15*.

3.2.4 Environment variables of the PROM monitor

Most environment variables are stored in NVRAM (NVRAM = non-volatile RAM), which is battery-buffered. This enables the variables to obtain their value even if the computer is switched off. The table below indicates which function the environment variables have. Detailed information is contained in the manpage for *nvrnm(7)*

Environment variable	Default value	Function
<i>bootfile</i>	Dependent on the selected system disk	Name of the file that is booted when the <i>boot</i> command (without parameters) or the <i>auto</i> command is issued. Maximum length: 20 characters.
<i>bootmode</i>	'c'	Boot mode.
<i>console</i>	't'	Definition of the devices that are automatically activated in the PROM monitor.
<i>console_dev</i>	0	Description of the drivers for the console interface.
<i>cpuid</i>	0	Logical CPU slot numbers (multiple values are output for computers with multiple CPUs)

<i>ethernetaddr</i>	<mac_address>	Ethernet address.
<i>grafic_dev</i>	0	Type of graphics chip.
<i>keyboard</i>	'us'	Country variant of the keyboard (only affects the keyboard assignment in the PROM monitor or in the firmware monitor).
<i>netaddr</i>	'89.89.89.89'	Internet address of the computer.
<i>lbaud</i>	19200	Baud rate for communication with the tty device on the tty(0) (local) connection.
<i>lantype</i>	<i>ethernet</i>	Description of the LAN interface type.
<i>rbaud</i>	9600	Baud rate for communication with the tty device on the tty(0) (remote) connection.
<i>resetepc</i>	0x.....	Memory cell for the exception program counter (epc) in non-volatile memory.
<i>resetra</i>	0x1234567	Memory cell for the return address <i>ra</i> in non-volatile memory.
<i>tty552(0)_if</i>	'S1(108)/S2(105)/M5(109)'	tty interface signals.
<i>tty552(1)_if</i>	'S1(108)/S2(105)/M5(109)'	tty interface signals.
<i>TS_baud</i>	19200	Teleservice: Baud rate
<i>TS_bit/char</i>	8	Teleservice: Width of characters transferred
<i>TS_parity</i>	'no'	Teleservice: Parity
<i>TS_stopbits</i>	1	Teleservice: Stop bits
<i>TS_if</i>	'S1(108)/S2(105)/M2(106)/M5(109)'	Teleservice: Modem signals
<i>TS_flags</i>	'tse'	Teleservice: Flag bits
<i>version</i>	Dependent on the installed firmware version	Firmware version

Table 9: Environment variables of the PROM monitor



If the battery is faulty or low on power or if environment variables have invalid values, the relevant variables are automatically reset to the default values.

3.2.5 Setting the boot mode

The boot parameters that are set determine which actions are performed when the system starts up. These boot parameters include the boot mode and the bootstring. The boot mode is defined by means of the **bootflags**. The table below provides an overview of the bootflags available for selection and the actions performed in each case.

Option	Action
<i>c</i>	AutobootThe startup tests are performed. If an error occurs, boot mode is automatically changed to <i>e</i> . If no errors occur, the firmware automatically boots the file that is named in the <i>bootfile</i> environment variable, i.e. the SASH. The SASH then boots Reliant UNIX automatically.
<i>d</i>	The startup tests are skipped and the main memory is deleted. The PROM monitor waits for user input.
<i>e</i>	ErrorAn error was discovered in a previous execution of the startup test. This boot mode is not changed automatically to another value (<i>m</i> or <i>c</i>) even if the next start test is executed without an error.
<i>i</i>	InstallationThe operating system is installed locally after a reboot. The boot mode is reset to <i>c</i> before the PROM monitor starts the <i>install</i> utilities. This executes an autoboot with the first reboot after installation. This boot mode can also be set with the <i>nvr</i> command.
<i>m</i>	The startup tests are executed. If an error is discovered, boot mode is automatically changed to <i>e</i> . When the tests have concluded, the PROM monitor waits for user input.
<i>r</i>	Remote installationFollowing a reboot, the operating system is installed from an installation server. The boot mode is reset to <i>c</i> before the PROM monitor starts the <i>install</i> utilities. This executes an autoboot with the first reboot after installation. This boot mode can also be set with the <i>nvr</i> command.You will find additional information on preparing for a remote installation in the manual " Reliant UNIX Installation ".
<i>w</i>	Warm startWhen the computer is reset, a warm start is performed. The firmware searches the memory for a valid restart block and transfers control to the routine indicated by the block. If a valid restart block is not found, a cold start is performed.
<i>s</i>	AutobootBoot mode <i>s</i> for RM200/RM200C and alphanumeric consoles.

Table 10: Possible boot modes

The *bootmode* and *bootfile* variables are located in the non-volatile memory of the computer. They retain their value even when the system is switched off.

It is advisable to set boot mode to *c* or *s* for routine work. The SASH and Reliant UNIX are then booted automatically when the computer is switched on. For this reason, this boot mode is also referred to as autoboot. In the process, Reliant UNIX boots to the run level that corresponds to the *initdefault* entry in the */etc/inittab* file. Generally, this is run level 2 (multiuser mode).

When you first start up the computer, or if you have performed a new installation, autoboot is set by default.

Displaying and setting the boot mode:

When you have switched to the firmware monitor (see the [Section "Firmware monitor \(RM200 - RM400\)"](#)), you can display the *bootmode* variable using the *printenv* command and change it with the *setenv* command:

```
>>□setenv□bootmode□<boot mode>
```

3.2.6 Setting the country variant (PROM monitor)

In the PROM monitor, the character set for the keyboard country variant ENG/US is used by default. This

character set primarily affects the assignment of keys with special characters. If your keyboard has another country variant, you should define the *keyboard* environment variable and assign a relevant value to this variable.

```
>> setenv keyboard <country variant>
```

An overview of the country variants supported is given in the [Section "Setting the country variant \(firmware monitor\)"](#).



The *keyboard* environment variable affects the keyboard assignment "below" Reliant UNIX exclusively, i.e. in the PROM monitor, in the firmware monitor and in the SASH. As soon as you boot Reliant UNIX, the keyboard character set that you configured under Reliant UNIX is used. In the mini operating system, you are always asked which keyboard variant you want to use. For further information, please refer to the manual "[Reliant UNIX Installation](#)".

3.2.7 Exiting the PROM monitor and booting Reliant UNIX

When you have concluded your work in the PROM monitor, you can boot Reliant UNIX.

Booting Reliant UNIX in multiuser mode

Enter the following command:

```
>> auto
```

The firmware first boots the SASH and then boots Reliant UNIX automatically. In the process, the run level that corresponds to the *initdefault* entry in the */etc/inittab* file is selected. Generally, this is run level 2 (multiuser mode).

Booting Reliant UNIX in single-user mode

Enter the following command:

```
>> boot
```

The firmware boots the SASH. The sash: prompt is displayed and you can now boot Reliant UNIX manually. Specify the *initarg* argument so that Reliant UNIX goes straight into single-user mode, for example:

```
sash: dkncr(,,)unix initarg=-s
```

Enter *dkpcs* or *dkdpt* instead of *dkncr*, depending on the value of *bootfile* (see also ...).

3.3 Firmware monitor (RM200 - RM400)

The firmware monitor is a menu-driven user interface for the PROM monitor. It is available on computers that have a graphics card and a graphics screen. The most important commands of the PROM monitor can be started by selecting menu items here. The firmware monitor, like the PROM monitor, is stored as firmware in the PROM.

3.3.1 Switching to the firmware monitor

On computers with a graphics card and a graphics screen, you always access the firmware monitor when you switch to the PROM monitor (see ...). However, to give you an opportunity to read the hardware messages of the firmware beforehand (see ...), the firmware monitor only appears when you have confirmed the messages by pressing any key:

```
Hit any key to continue..
```

The main menu of the firmware monitor looks like this:

```
RM200 Firmware Version 4.2004
```

```
SINIX boot (auto)
```

```
SINIX boot single user mode (single_user)
```

```
Stand-alone shell boot (boot)
```

```
Standard installation (install)
```

```
Tape installation (tinstall)
```

```
Remote installation (rinstall)
```

```
Firmware update (update)
```

```
Remote firmware update (rupdate)
```

```
Switch to NT firmware (sinix_to_nt)
```

```
Environment modification
```

```
Prom monitor
```

You can use the keys \downarrow and \uparrow to move the cursor to the next menu item. Press \rightarrow to select a menu item. The table below shows you which functions are linked to the individual menu items and which commands are issued respectively.

Menu item	Command	Function
<i>SINIX boot</i>	<i>auto</i>	Loads the SASH, which in turn boots Reliant UNIX automatically
<i>SINIX boot single user mode (single user)</i>	<i>boot</i>	Boots in single-user mode
<i>Stand-alone shell boot</i>	<i>boot</i>	Loads the SASH
<i>Standard installation</i>	<i>install</i>	Loads the installation menu:
<i>Tape installation</i>	<i>tinstall</i>	— from the local system
<i>Remote installation</i>	<i>rinstall</i>	— from tape
	<i>all</i>	— from the server
	<i>update</i>	Updates the EEPROM firmware:
<i>Firmware update</i>	<i>rupdate</i>	— from the local system

<i>Remote firmware update</i>	— from a server
<i>Switch to NT firmware</i> <i>sinix_to_nt</i>	Activated if Windows-NT (up to V4.0) is to be installed Important: If you inadvertently select this menu item, you must reboot your computer
<i>Environment modification</i>	A submenu is displayed where you can set environment variables
<i>PROM monitor</i>	the graphical user interface is exited. The "normal" prompt of the PROM monitor ">>" appears. Press [Ctrl] [C] to return to the firmware monitor.

Table 11: Menu items in the firmware monitor

The menu items *auto* and *boot* are started by the commands of the same name in the PROM monitor.

3.3.2 Commands for loading the installation software

The menu items *install*, *tinstall*, *rinstall*, *update* and *rupdate* are only relevant if you want to execute one of the tasks listed in the table. The software required for this must be made available on CD-ROM (or on a tape for *tinstall*). The CD-ROM (DVD) drive must have the SCSI address *15* (see also ...).

3.3.3 Submenu for the environment variables

When you select *Environment modification* from the main menu, the following submenu appears:

```

Environment modification
Keyboard nationality (keyboard)
Internet address (net addr)
Local serial console baudrate (lbaud)
Remote serial console baudrate (rbaud)
Stand-alone shell name (bootfile)
Firmware startup mode (bootmode)
tty(0) modem signals (tty(0)_if)
tty552(0) modem signals (tty552(0)_if)
tty552(1) modem signals (tty552(1)_if)
Teleservice baudrate (TS_baud)
Teleservice bits per char (TS_bit/char)
Teleservice parity (TS_parity)
Teleservice stop bits (TS_stopbits)
Teleservice modem signals (TS_if)
Teleservice flags (TS_flags)
Return to main menu

```

The items in this menu can be used to set the environment variables that were described in [Table ""](#). Detailed information in this regard is contained in the manpage for *nvrnm(7)*. When you select a menu item, a command line appears below the menu showing the current value of the environment variables. The *Return to main menu* menu item brings you back to the main menu of the firmware monitor.

3.3.4 Setting the boot mode

You can display the *bootmode* environment variable by selecting *Firmware startup* from the *Environment modification* submenu. The command line shown below then appears:

```
New bootmode or ESC to exit: c_
```

If you want to modify the *bootmode* variable, delete its previous value. To do this, use the [~~-]~~ key or the backspace key. Then enter the desired value and confirm your entry by pressing RETURN. If you press [Esc] instead, you exit the command line without modifying the *bootmode* variable. Finally, return to the main menu of the firmware monitor by selecting the *Return to main menu* menu item.

3.3.5 Setting the country variant (firmware monitor)

In the PROM monitor, the character set for the keyboard country variant ENG/US is used by default. This character set primarily affects the assignment of keys with special characters.

In the firmware monitor, the keyboard is normally only used to position the cursor on a particular menu item using the arrow keys in the cursor block. The assignment of these keys is identical for all country variants. For this reason, the keyboard assignment is only relevant if you want to exit the firmware monitor and switch to the PROM monitor (by selecting *Prom monitor* from the main menu). In this case, you should define the *keyboard* environment variable and assign a relevant value to this variable.

To do this, select *Keyboard nationality (keyboard)* from the *Environment modification* submenu. The following submenu then appears:

```
Keyboard nationality selection
German (ge)
US english (us)
French (fr)
Belgian (fr)
UK english (uk)
Danish (dk)
Swedish (sw)
Swiss german (sg)
Swiss french (sf)
Italian (it)
Spanish (sp)
Return to main menu
```

The menu items in this submenu can be used to set the country variant of the keyboard. Use the *Ë* and *Ê* keys for this. To select a menu item, press the *Ú* key. You are returned to the main menu of the firmware monitor when you select the *Return to main menu* menu item.



The *keyboard* environment variable only affects the keyboard assignment "below" Reliant UNIX, i.e. in the PROM monitor, in the firmware monitor and in the SASH. As soon as you boot Reliant UNIX, the keyboard character set that you configured under Reliant UNIX is used. The keyboard country variant is always prompted in the mini operating system. For further information in this regard, please refer to the manual "[Reliant UNIX Installation](#)".

3.3.6 Exiting the firmware monitor and booting Reliant UNIX

When you have concluded your work in the firmware monitor you can boot Reliant UNIX.

Booting Reliant UNIX in multiuser mode

Select the *SINIX boot (auto)* menu item from the main menu of the firmware monitor. The firmware then boots the SASH and then boots Reliant UNIX automatically. In the process, the run level that corresponds to the entry *initdefault* in the */etc/inittab* file is selected. Normally, this is run level 2 (multiuser mode).

Booting Reliant UNIX in single-user mode

Select the *Stand-alone shell boot (boot)* menu item from the main menu of the firmware monitor. The firmware then boots the SASH. The sash: prompt appears and you can now boot Reliant UNIX manually. Specify the *initarg* argument so that Reliant UNIX goes directly into single-user mode, e.g

```
sash: dkncr(,.)unix initarg=-s
```

Enter *dkpcs* or *dkdpt* instead of *dkncr*, depending on the value of *bootfile* (see also ...).

3.4 Board debug monitor (RM600)

The board debug monitor (BDM) is a monitoring program in a special controller to which the console is connected:

RM600-xxx:

In the CSI controller

The BDM signs on with the prompt BDM(MBII□0□t)>.



In the examples in this chapter, the prompt is generally represented instead as it appears on an RM600 E60 model.

RM600 E:

In the (E)HIOS basic controller. Refers to (E)HIOS controllers located in the following slots in the system.

Model	Location
E20/E60 models	Slot 8
E30/E4x models	Slot 10
E70/E8x models	Slot 14

The BDM signs on with the following prompt on the RM600:

Model	Prompt
E20/E60 models	BDM(SPB□□8□t)>
E30/E4x models	BDM(SPB□□10□t)>
E70/E8x models	BDM(SPB□□14□t)>

The BDM is used to execute special system administration commands. In particular, it enables you to modify the parameters that affect the boot process. However, it also allows you to update the EEPROM firmware, for example.

3.4.1 Switching to the board debug monitor

The procedure for accessing the board debug monitor is determined by two factors: The current run level of Reliant UNIX and whether or not autoboot is enabled.

Switching to the board debug monitor after the computer is switched on

When the computer is switched on, the firmware executes a series of actions that correspond to the boot mode set. If autoboot has been switched off (the last number of the bootflags is 8 or 9), you automatically gain access to the board debug monitor.

If autoboot is switched on (the last digit of the boot flags is 0), you can then abort the boot process by pressing *Ctrl + C* as soon as the following message appears:

```
Autoboot:□Waiting□to□load□□sdisk011s0/unix□sd(1,8,0)unix□root=ios0/sdisk011s0□sw
ap=ios0/sdisk011s1
(Ctrl-C□to□abort,□RETURN□to□expedite)
```

You are then in boot2. You can then access the board debug monitor as follows:

RM600-xxx: [Ctrl]+[A]

RM600 E: [Ctrl]+[@], +([B],[D],[M])

Switching to the board debug monitor from boot2 or from Reliant UNIX

If the computer is already switched on and Reliant UNIX is running, the above key combinations also bring you into the board debug monitor.



If Reliant UNIX is running temporarily in the background (multiuser mode), not all actions can or may be performed in the BDM.

You should not use the *wreset*, *creset* or *b* commands! Otherwise, you risk losing data in the file systems and it will also not be possible to load other programs. Thus, the call `b 9 cd(0,0)flsh` produces no result and the flash tool cannot be loaded.

3.4.2 Important commands in the board debug monitor

You can execute certain administration tasks in the board debug monitor. Various commands are available for this purpose. [Table 12](#) provides an overview of the most important commands.

Command Type	Syntax	Function
Utilities	<code>h[lp],</code> <code>? [<help-level>]</code>	Displays a command overview
System	<code>cf</code> <code>cp</code> <code>cparam [1]</code> <code>breset<level></code> <code>rrrrrr[<slot>]</code> <code>wreset</code> <code>creset</code> <code>rl [<count>]</code> <code>wl</code>	Displays the configuration of the CPU board Reads the console parameters 1: Modifies the console parameters Resets the board at <i>slot</i> System warm start System cold start Reads the console buffer Modifies the console buffer
Teleservice	<code>tpasswd</code> <code>tparam [1]</code> <code>setdial</code> <code>dial [numbers]</code> <code>tele_off</code>	Sets the password Reads the Teleservice parameters 1: Modifies Teleservice parameters Dial_out number Dial_out Switches off the remote console
Boot	<code>ra</code> <code>wf <flags></code> <code>wn <x> <string></code> <code>b <flags> <string></code>	Reads the boot parameters Modifies the boot flags <i>x</i> =0: Modifies the bootstring <i>x</i> =1: Modifies the dump string Boots boot2

Table 12: Important commands in the board debug monitor



If the system is currently in multiuser mode, you must not execute the *wreset*, *creset* or *b* commands under any circumstances. If you do, the file systems will be inconsistent.

In almost all cases, it is possible to abort a BDM command. To do this, press [Ctrl]+[D] or [Ctrl]+[C]. The board debug monitor returns with its prompt and waits for the user to enter information.

Boot2 can be booted from various media, including SCSI I/O disks (IOS), CD-ROM.

3.4.3 Setting the boot mode

The boot parameters that are set determine which actions are performed when the system starts up. These boot parameters include the boot mode and the bootstring. The boot mode is defined by means of the **bootflags**. The table below provides an overview of the boot flags available for selection and the actions performed in each case.

Boot flags	Action
0000	Autoboot All boot phases are run through automatically: built-in self tests (BIST), master test handler (MTH), boot1, boot2 and UNIX (see ...). Unlike bootflag 0200, additional built-in self tests are performed in order to test communication between the controllers.
0001	Stop at boot2 Boot2 waits for user input.
0008	Stop at the BDM The board debug monitor waits for user input.
0200	Autoboot (default option following installation) All boot phases are run through automatically: built-in self tests (BIST), master test handler (MTH), boot1, boot2 and UNIX (see ...). Unlike bootflag 0000, only the controllers themselves are tested. Communication between the controllers is not checked.

Table 13: Possible bootflags

The specified bootflag values can also be combined, e.g. 209 for manual booting.

For daily routine work, it is advisable to set the bootflag 200. Boot2 and Reliant UNIX are then booted automatically when the computer is switched on. This boot mode is called **autoboot** for this reason. In the process, Reliant UNIX boots to the run level that corresponds to the *initdefault* entry in the */etc/inittab* file. Generally, this is run level 2 (multiuser mode).

When you first start up the computer, or if you have performed a new installation, autoboot is set by default.

Modifying the boot mode in the board debug monitor

You can display the current bootflags using the *ra* command.

For example:

```
bootflags:0000000200
bootstring:sdisk011s0
dumpstring:sd(1,8,1)
```

You can modify the bootflags using the *wf* command:

Switch on autoboot:

```
BDM(SPB008 t)>wf 200
```

Switch off autoboot:

```
BDM(SPB008 t)>wf 201
```

Modifying the boot mode in Reliant UNIX

You can also display the current bootflags while Reliant UNIX is running. To do this, use the *bootflags(8)* command.

For example:

```
Actual reboot values:
bootflags:0x00000200
bootstring:sdisk011s0
```

You can modify the bootflags using the *-f* option:

Switch on autoboot

```
# bootflags -p -f 200
```

Switch off autoboot

```
# bootflags -p -f 201
```

Further information can be found in the manpage for *bootflags(8)*.

3.4.4 Setting the bootstring

In addition to the boot mode, the boot parameters also include the bootstring. The bootstring determines the following:

- The device from which boot2 is to be loaded,
- Whether the boot mode is to be logical or physical.

The differences between logical and physical boot mode are described in [Section "Boot modes \(RM600\)"](#).

Bootstring for logical boot mode

The system can be booted in logical mode if the following software is installed:

RM600-xxx:

Reliant UNIX V5.43 or higher
and Firmware V2.42 or higher for the CSI

RM600 E:

General software

The bootstring has the following syntax:

```
<device>/<file>
```

<device> is the block-oriented device name of the boot medium (see manpage for *sdisk(8)*), *<file>* is the name of the system kernel to be loaded (normally *unix*).

Example: *sdisk011s0/unix*.

Only *<device>* is stored in the NVRAM. With autoboot, *<file>* is extended as follows: in boot1 *sash*, in boot2 *unix*. With manual booting, *<file>* must be specified explicitly.



Reliant UNIX 5.43 or higher must be installed on a system disk before it can be booted in logical mode. During installation, the disk parameters are automatically entered in the root partition list which is resident in the CSI or (E)HIOS-NVRAM. Logical boot mode is set automatically. Read the *boot2(8)* manpage for information on how to swap the active system disk.

Bootstring for physical boot mode

Physical boot mode is fully compatible with the previous ("old") boot mode. The bootstring has the following syntax:

```
device(<channel_id>,<scsi_id> * 8,<partition>)file
```

The individual parameters are explained in the following table.

Parameter	Meaning
<i>device</i>	Name of the load device The following names can be used: <i>cd</i> Load from CD-ROM <i>hd</i> or <i>sd</i> Load from SCSI disk <i>et</i> Load from Ethernet

<i>channel_id</i>	<p>Number of the SCSI bus (synonym: SCSI string, channel)</p> <p>RM600-xxx: This number is the same as the logical number of the controller itself. The CSI controller has only one SCSI bus with the number 0. The addresses of the SCSI buses on the existing DCS controllers are numbered sequentially in the order of the slot numbers (DCS = Disk Controller SCSI).</p> <p>RM600 E: See Figure "SCSI addresses in the system cabinet of the RM600 E"</p>
<i>scsi_id * 8</i>	<p>Number of the device on the SCSI bus multiplied by 8 (synonym: SCSI address)</p> <p>Example: If the CD-ROM (DVD) drive has a SCSI address of 6, the value 48 must be specified here.</p>
<i>partition</i>	<p>Number of the root partition The following number can be used:</p> <p>0 For standard partitioning</p>
<i>file</i>	<p>Name of the system kernel to be loaded. If no name is specified, a default name is used:</p> <p><i>sash</i> in the BDM for boot2 <i>unix</i> in boot2 for Reliant UNIX</p> <p>You can also specify the name of a system kernel that you created yourself.</p>

Table 14: Parameters of the bootstring

Booting from Ethernet

When booting from Ethernet, the physical bootstring has the following format – only for the EHIOS board:

et(a,b,c)[cip:][sip:][host:][file]

Where:

<i>et</i>	Driver ID for Ethernet (as specified in Table "")
<i>a</i>	Ethernet channel ID
<i>b</i>	Not evaluated when booting from Ethernet.
<i>c</i>	Parameter for LAN bootstring.
<i>cip</i>	IP client address (e.g. xxx.xxx.xxx.xx).
<i>sip</i>	IP server address (e.g. xxx.xxx.xxx.xx).
<i>host</i>	Name of bootp server (e.g. nizza:).
<i>file</i>	Contains the complete path or also the file name of the program to be booted (e.g. <i>/home/eth_file</i> or <i>eth_file</i>).

Ethernet channel D:

Similar to choosing a SCSI channel, this parameter is used in Ethernet to select the logical number of an Ethernet connection. This logical number can assume the following values:

E30/E70:

32 ... 63

Logical X86 channel ID plus offset 32. The Ethernet channels are numbered consecutively in ascending order of Ethernet controller slot IDs.

E4x/E8x:

64 ... 95

Logical X86 channel ID plus offset 64. The Ethernet channels are numbered consecutively in ascending order of Ethernet controller slot IDs.

b:

This parameter is not evaluated when booting from Ethernet.

c:

Parameter for the LAN bootstring (control flags for the LAN driver). The following decimal entries are possible:

0 ... The network speed is established from tests with 10 and 100 Mbit/s.

2 ... The network speed is established on the basis of auto-negotiation. If this fails, a fixed setting of 100 or 10 Mbit/s is used.

10 ... The network speed is set to a fixed value of 10 Mbit/s.

20 ... The network speed is set to a fixed value of 100 Mbit/s.

cip:

This parameter contains the client IP address (e.g. 129.103.156.26). This must only be specified if the Ethernet (MAC) address of the client is not entered on the server in the */etc/inet/bootptab* file or if the IP address of the server (*sip*) is specified in the bootstring. The IP address of the client specified here must be entered, however, in the */etc/inet/bootptab* file on the server.

sip:

This parameter contains the IP address of the server (e.g. 129.103.156.88). It only makes sense to specify the server IP address in conjunction with the IP address of the client (*cip*). The *host* parameter need not be specified in the bootstring in this case.

The IP address of the server in the bootstring was introduced to eliminate the need for the bootp protocol between server and client. The other advantage of this is that no further administration intervention is required on any router that may be involved in the network (e.g. release bootp between two subnetworks).

host:

This parameter contains the name of the bootp server (e.g. nizza:). This entry is only required if more than one server in the network has entered the client to be booted in its *bootptab* file and if there is no IP address for the server in the bootstring.

If neither the *host* parameter is specified nor the IP address of the server in the bootstring, the boot file is loaded in the network by the default bootp server.

file

This parameter contains the complete path or also only the file name of the program to be booted (e.g. */home/eth_file* or *eth_file*).

If this entry is missing from the bootstring, the default bootp file is loaded by the bootp server. The default bootp file as well as the default bootp directory are entered on the server in the *etc/inet/bootptab* file.

If the path name is omitted for the *file* parameter and only the file name is specified (e.g. *eth_file*), this file must exist in the server's default bootp directory.

If security mode was entered with *-s <pathname>* on the bootp server in the */etc/inet/inetd.conf* file for the *tftp* Internet service, *tftp* transmission is only possible for files that are contained in the specified directory.

In this case, it is possible to omit the path name on the client in the bootstring and simply specify the file name. To load files from other directories, the entry for security mode (*-s <pathname>*) must be deleted in the */etc/inet/inetd.conf* file.

Possible Ethernet bootstring variants (not E4x/E8x)

Syntax	Example
et(a,b,c)	et(32,0,0)
et(a,b,c)host:	et(32,0,0)nizza:
et(a,b,c)file	et(32,0,0)eth_file
et(a,b,c)host:file	et(32,0,0)nizza:eth_file
et(a,b,c)cip:	et(32,0,0)129.103.156.26:
et(a,b,c)cip:host:	et(32,0,0)129.103.156.26:nizza:
et(a,b,c)cip:file	et(32,0,0)129.103.156.26:eth_file
et(a,b,c)cip:host:file	et(32,0,0)129.103.156.26:nizza:eth_file
et(a,b,c)cip:sip:	et(32,0,0)129.103.156.26:129.103.156.88:
et(a,b,c)cip:sip:host:	et(32,0,0)129.103.156.26:129.103.156.88:nizza:
et(a,b,c)cip:sip:file	et(32,0,0)129.103.156.26:129.103.156.88:eth_file
et(a,b,c)cip:sip:host:file	et(32,0,0)129.103.156.26:129.103.156.88:nizza:eth_file

Table 15: Selection of possible bootstring variants when booting from Ethernet

Possible Ethernet bootstring variants (E4x/E8x only)

Syntax	Example
et(a,b,c)	et(32,0,0)
et(a,b,c)host:	et(32,0,0)nizza:
et(a,b,c)file	et(32,0,0)eth_file
et(a,b,c)host:file	et(32,0,0)nizza:eth_file
et(a,b,c)cip:	et(32,0,0)129.103.156.26:
et(a,b,c)cip:host:	et(32,0,0)129.103.156.26:nizza:
et(a,b,c)cip:file	et(32,0,0)129.103.156.26:eth_file
et(a,b,c)cip:host:file	et(32,0,0)129.103.156.26:nizza:eth_file
et(a,b,c)cip:sip:	et(32,0,0)129.103.156.26:129.103.156.88:
et(a,b,c)cip:sip:host:	et(32,0,0)129.103.156.26:129.103.156.88:nizza:
et(a,b,c)cip:sip:file	et(32,0,0)129.103.156.26:129.103.156.88:eth_file
et(a,b,c)cip:sip:host:file	et(32,0,0)129.103.156.26:129.103.156.88:nizza:eth_file

Table 16: Selection of possible bootstring variants when booting from Ethernet (E4x/E8x only)

Preparations for the bootp protocol

The server must be prepared for the bootp protocol, if this has not already happened. In addition, the server must be notified of the bootp clients with their Internet and Ethernet addresses. The Ethernet address (MAC address) of the Ethernet controller on the client is displayed with the *dv* command in the BDM.

Example:

```
BDM(SPB 10 t)>dv
HARDWARE: EHIOS   VERSION: 07   SERIAL NR: 000000001588
FIRMWARE: BOOT BANK VERSION: 01.30 DATE:   98/02/26
FIRMWARE: FLASH BANK VERSION: 94.38 DATE:   98/12/11
SOFTWARE: BOOTSTAGE VERSION: 03.08
```

Slot	Product-Code	Serialnumber	HW_Rev	special
00:	EIFC-HC.V4	000000001811	0001	0800060D8DE9
01:	SIFC-SE16	000000001999	0000	FFFFFFFFFFFF
02:	SIFC-SE16	000000001994	0000	FFFFFFFFFFFF
03:	SIFC-SE8	000000003260	0000	FFFFFFFFFFFF

BDM(SPB 10 t)>

Example (E4x/E8x only):

```

BDM(SPB 10 t)>b 0 @conf_ctrl
*** CPU activated - 1st bootstage is running
EHIOS Boot1 revision: 3.13
Boot flags: 0000 Boot string: "@conf_ctrl"
Get SCSI/LAN Controller Configuration
X86 "... PCI":---
+-----+
| X86 SCSI/LAN Controller Configuration in Main Cabinet of T5-MR-System |
+-----+
| SCSI | HIOS/EHIOS | X86 Controller | Back | Bootstring |
+-----+
| Log | HIOS/EHIOS | S | X86 | Chann | SCSI | T5 | Physical |
| Chan | Product | | Controller | 0..3 | Bus | SCSI | Bootstring |
| ID | Code | D | Product | singl | 0..5 | Bus | |
| | | | Code | diffe | | Bootstring | |
+-----+
| X 0 | EHIOS-V01 | A | SIFC-SE16 | 1 s16 | 2 | (2,x,0) | (32,x,0) |
| X 1 | EHIOS-V01 | A | SIFC-SE16 | 2 s16 | 1 | (1,x,0) | (33,x,0) |
| X 2 | EHIOS-V01 | A | SIFC-SE8 | 3 s08 | 0 | (0,x,0) | (34,x,0) |
+-----+
| PCI SCSI/LAN Controller Configuration in Main Cabinet of T5-MR-System |
+-----+
| SCSI | EHIOS | PCI Controller | Backp | Bootstring |
+-----+
| Log | EHIOS | Mothe | PCI | Chann | SCSI | T5 | Physical |
| Chan | SID | Board | Controller | 0..5 | Bus | SCSI | Bootstring |
| ID | | | Product | singl | 0..5 | Bus | |
| | | | Code | diffe | | Bootstring | |
+-----+
| P 0 | A | upper | PASD16 | 4 d16 | - | | (64,x,0) |
| P 1 | A | lower | PASD16 | 5 d16 | - | | (65,x,0) |
| P 2 | A | lower | PASD16 | 4 d16 | - | | (66,x,0) |
+-----+
| LAN | EHIOS | PCI Controller | MAC Address | Bootstring |
+-----+
| Log | EHIOS | Mothe | Controller | Chann | MAC Address | Physical | |
| Chan | SID | Board | Product | 0..5 | | Bootstring |
| ID | | | Code | | | | |
+-----+
| P 0 | A | upper | PENE | 5 lan | 00:c0:95:fa:d5:df | et(64,0,0) |
+-----+
Boot1:
    
```

Bootp actions on the server

- ▶ Make changes or additions in the following files:

File to be changed	Entry
<i>/etc/inet/hosts</i>	Input of Internet address and symbolic name of the Ethernet controller: □ □ Example: 129.103.153.23 fahne
<i>etc/inet/bootptab</i>	Examples of entering Ethernet controllers with their Ethernet and Internet address: .default:\□ □□□□□□:hn:\□ .subnet129:\□ □□□□□□:hn:\□ □□□□□□:ht=ether:\□ □□□□□□:sm=255.255.252.00:\□ □□□□□□:gw=129.103.161.2: fahne:\□ □□□□□□:tc=.subnet129:\□ □□□□□□:ha=08.00.06.0d.8d.e9:\□ □□□□□□:ip=129.103.153.23:\□ □□□□□□:hd=/Y5B0000/cdrom/BOOT:\□ □□□□□□:bf=sash_b20:
<i>/etc/inet/inetd.conf</i>	Release entries for <i>bootp</i> and <i>tftp</i> .



For more information, see the appropriate section in the "Reliant UNIX Installation" manual.

Finally, you have to restart the *inetd* daemon with the modified bootp configuration:

```
# ps -ef | grep inetd | grep -v grep
# kill -1 <inetd-pid>
```

SCSI addresses in the RM600 E

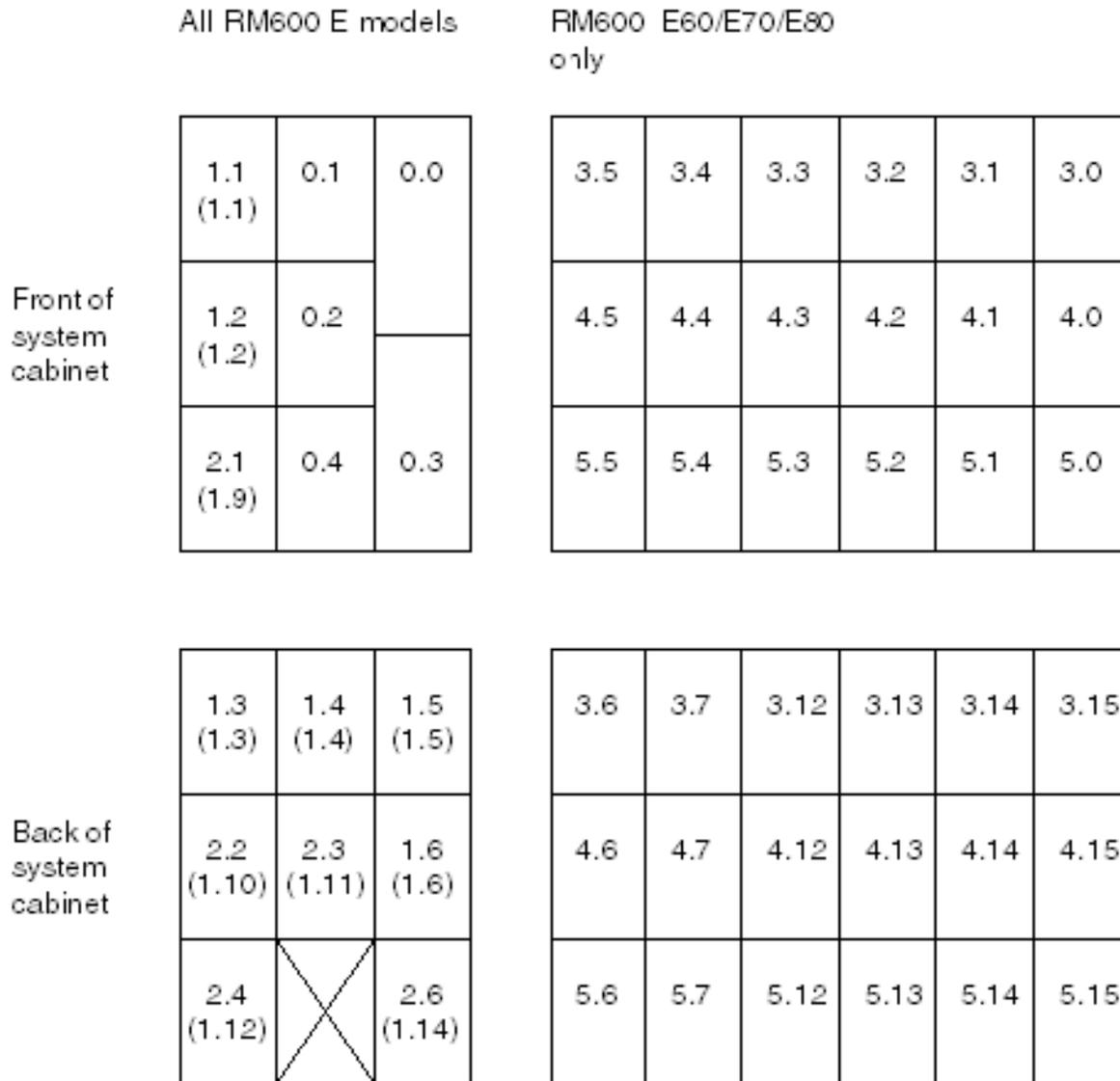


Figure 6: SCSI addresses in the system cabinet of the RM600 E

[Figure "SCSI addresses in the system cabinet of the RM600 E"](#) shows the allocation of hard disks in the system cabinet of the RM600 E. The upper row represents the view from the front, the lower the view from the back. The left-hand side of the figure applies for both RM600 E models, the right-hand side for the RM600 E60/E70/E80 only.

The meaning of the digits is as follows:

1st digit	<i>channel_id</i>	Logical number of the controller (00 ... 99) (synonym: SCSI string, channel)
2nd digit	<i>scsi_id</i>	Number of the device on the SCSI bus (synonym: SCSI address)

The numbers in brackets are valid if the two strings of the relevant controller are coupled via a bridge. The devices for data input and output are generally connected to controller 0. These devices have the following SCSI addresses:

0.0	CD-ROM/DVD drive
0.1	Disk drive
0.2	4 mm MTC (DAT)
0.3	¼" MTC or 8 mm MTC drive
0.4	if available: SM7 (encryption and decryption module)



The devices on controller 0 can have a different SCSI address than that specified. The number depends on the hardware settings of the relevant device (jumper). You can list the actual addresses under Reliant UNIX using *autoconf -l*.

If you want to access a device under Reliant UNIX, you have to enter the hexadecimal digits a ... f instead of the decimal digits 10 ... 15. The hard disk 5.15 then has the logical name *ios0/sdisk05f*.

Modifying the bootstring

There are two ways of modifying the bootstring:

In the board debug monitor:

```
BDM(SPB□□8 t)>wn□0□<bootstring>
```

In Reliant UNIX:

```
#□bootflags□-p□-b□<bootstring>
```

The *file* parameter (see [Table ""](#)) is not stored in the CSI-NVRAM. With autoboot, *file* is extended as follows: in boot1 *sash*, in boot2 *unix*. With manual booting, *file* must be specified explicitly.

If the bootstring is modified, the system checks whether the new bootstring belongs to the logical boot mode or the physical boot mode. The appropriate bootflag is then set automatically.



Change as of Reliant UNIX 5.43:

If a parameter of the bootstring is set to zero, its value must still be specified. For example, the string *sd(0,0,0)unix* can **no** longer be shortened to *sd(,,)unix*. This change will prevent the system being booted inadvertently from the wrong system disk.



Reliant UNIX 5.43 or higher must be installed on a system disk before it can be booted in logical mode. During installation, the disk parameters are automatically entered in the root partition list which is resident in the CSI/(E)HIOS-NVRAM. Logical boot mode is set automatically. Read the *boot2(8)* manpage for information on how to swap the active system disk.

3.4.5 Exiting the board debug monitor

When you have concluded your work in the board debug monitor, you can exit the monitor. There are several options here.

Returning to the active system

If boot2 or Reliant UNIX is already running, you can return to the active system by pressing [#], [Q], Ú.



If the system is currently in multiuser mode, you must not execute the *wreset*, *creset* or *b* commands under any circumstances. If you do this, the system is restarted immediately and all data in the main memory is then lost.

Booting boot2 and Reliant UNIX

If you wish to boot Reliant UNIX without further interruption, you must either modify the boot flags (see above) or override the current boot flags. Enter the following command (example of *sdisk000s0* in an RM600-xxx):

Logical boot mode:

```
BDM(MBII□0□t)>b□200□sdisk000s0/sash
```

Physical boot mode:

```
BDM(MBII□0□t)>b□200□sd(0,0,0)sash
```

The firmware first boots boot2 and then boots Reliant UNIX automatically. The run level that corresponds to the *initdefault* entry in the */etc/inittab* file is selected. Generally, this is run level 2 (multiuser mode).

Booting boot2

If the boot process is to end in boot2, you must either set the bootflags accordingly (see above) or override the current bootflags. Enter the following command (example of *sdisk000s0* in an RM600-xxx):

Logical boot mode:

```
BDM(MBII□0□t)>b□201□sdisk000s0/sash
```

Physical boot mode:

```
BDM(MBII□0□t)>b□201□sd(0,0,0)sash
```

The firmware boots boot2. The following prompt is then displayed.

boot2:

You can now enter the required command.

3.5 SASH (RM200 - RM400)

The stand alone shell (SASH) is loaded into main memory by the firmware of the computer. It performs the following tasks:

- If autoboot is switched on (boot mode *c*) or if the SASH was booted from the PROM monitor using the *auto* command, it automatically boots Reliant UNIX. To do this, it receives the necessary commands from the firmware and processes them. As soon as the boot process is complete, it transfers control to the Reliant UNIX operating system.
- If the SASH was booted from the PROM monitor using the *boot* command, it awaits user input with the *sash:* prompt. You can then perform system administration tasks.

3.5.1 Switching to the SASH

You must first switch to the PROM monitor or to the firmware monitor; please refer to the [Section "Switching to the board debug monitor"](#) for information on the procedure involved. Then start the *boot* command.

Prom monitor:

```
>> boot
```

Firmware monitor:

```
Stand-alone shell boot (boot)
```

3.5.2 Important SASH commands

You can perform certain administration tasks in the SASH. Various commands are available for this purpose. These are largely identical to those available in the PROM monitor or in the firmware monitor. [Table ""](#) provides an overview of these commands. A detailed description of all commands and options is contained in the manpage for *sash(8)*.

In contrast to the PROM monitor or the firmware monitor, you can use a few additional commands in the SASH:

Command	Function
<i>cp</i>	Copies the contents from one file to another.
<i>end</i>	Exits the SASH and returns control to the PROM monitor.

Table 17: Additional commands in the SASH



Please note also the following differences to the PROM monitor:

- The *bootfile* environment variable has the same value as in the PROM monitor (e.g. *dkncr(0,0,10)sash*). If you want to boot Reliant UNIX, you must specify the file to be booted explicitly (e.g. *dkncr(0,0,0)unix*). The *auto* command cannot be used.
You can retrieve the *bootfile* environment variable with the *printenv bootfile* command.
- You can modify the *bootfile* environment variable using the *setenv* command. However, this type of modification only affects the booting of Reliant UNIX. The change is **not** entered in the NVRAM.

In most situations it is possible to cancel the commands that have been issued. In this case, press [Ctrl] + [D] or [Ctrl] + [C]. The command interpreter is then reactivated and waits for user input.

The SASH can boot Reliant UNIX from different media. Normally, it loads the */stand/unix* file from the system disk. However, it is also possible to load an executable kernel from any other disk drive containing a ufs file system with a suitable kernel. The file system must not be mounted beforehand.

A kernel is normally only loaded from a CD-ROM or a tape device if an installation is to be performed. This procedure can also be useful, however, if a fault in the system disk prevents the kernel from being booted from there. Consequently, the SASH does not need to perform any read operations to the disk (see also ...).

3.5.3 Writing the SASH to the system disk

The SASH is required in order to boot the operating system. It is stored in a specific area of the system disk, i.e. in partition 10. If you have inadvertently overwritten this partition, it is no longer possible to boot the system. The only way to rectify this situation is to write the SASH to partition 10 again. You can use the *Write Sash to the System Disk* menu item in the main menu of the mini system (mini root) to do this.

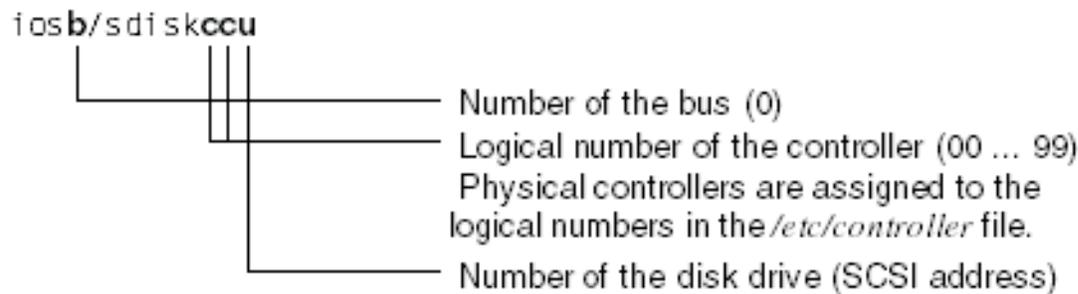
You must first specify which hard disk is the system disk. The dialog program suggests the first disk found. You can use the [CHOICES] function key to activate a selection menu. Disks that are suitable for use as system disks are marked with an asterisk. Only these disks are available to you.

Select the hard disk to which the stand-alone shell (SASH) is to be written. The window displays the preferred hard disk (abbreviation and disk type). Press the CHOICES (F2) function key if you want to select a different hard disk.

A submenu containing a list of all hard disks connected to your computer and found by the installation procedure is then displayed.

Possible system disks are marked with an asterisk (*).

The following syntax governs the names used for hard disks:



Further information about naming conventions can be found in the manpage for *sdisk(7)* and in the "System Administrator Reference Manual" (two volumes).



An error message is displayed if no system disk is found. If this is the case, please refer to the manual "[Reliant UNIX Installation](#)" to find out what to do.



Make sure that the CD-ROM in the CD-ROM (DVD) drive is the CD-ROM that was used to install the previous operating system. Only then can you be sure that the "correct" SASH, i.e. the SASH that corresponds to the installed Reliant UNIX revision version, is written to the system disk.

▶ Press [SAVE] to confirm the suggested hard disk.

The following message is displayed during the write operation:

```
Sash is written to the volume header. Please wait...
```

The mini operating system main menu then appears again.

3.5.4 Exiting the SASH

When you have completed your work in the SASH, you can boot Reliant UNIX.

Booting Reliant UNIX in multiuser mode

Enter the command in one of the following two formats, e.g.:

Long form: `sash: boot -f dkncr(,)unix`

Short form: `sash: dkncr(,)unix`

Enter *dkpcs* or *dkdpt* instead of *dkncr*, depending on the value of *bootfile* (see also ...).

Reliant UNIX then enters the run level that corresponds to the *initdefault* entry in the */etc/inittab* file. This is normally run level 2 (multiuser mode).

Booting Reliant UNIX in single-user mode

Enter the command in one of the following two formats, e.g.:

Long form: `sash: boot -f dkncr(,,)unix initarg=-s`

Short form: `sash: dkncr(,,)unix initarg=-s`

Enter *dkpcs* or *dkdpt* instead of *dkncr*, depending on the value of *bootfile* (see also ...).

The *initarg* argument causes Reliant UNIX to go directly into single-user mode.

Returning to the PROM monitor or the firmware monitor

Press [Ctrl] + [D] or issue the SASH *end* command.

3.6 RAID configuration via miniroot (RM200 - RM400)

Configuration of an OLR system mirror disk on a DPT RAID 1 system mirror disk is available on the RM400C and the RM300-E with an additional DTP controller. This is how you configure the system:

- ▶ First load the main menu of the mini operating system and start the standard installation.
- ▶ Select the language, console type and keyboard.
- ▶ Choose *Start DPT manager* from the menu and then choose the submenu item *Start DPT manager*.
- ▶ Use the RAID manager to configure a *DPT-RAID-1 disk* (see the "RAIDmaster" manual).

After configuration, the system shuts down automatically and is powered down.

- ▶ Reload the main menu of the mini operating system and start the standard installation.
- ▶ Again, select the language, console type and keyboard.
- ▶ Then reinstall the operating system on the DPT RAID 1 disk.

The root disk is then mirrored and can be replaced online if necessary.

3.7 Boot2 (RM600)

Boot2 is loaded into main memory by boot1. It performs the following tasks:

- If autoboot is switched on (the last number of the bootflags is 0) or if boot2 was booted accordingly from the board debug monitor, it automatically boots Reliant UNIX. To do this, it receives the necessary commands from the firmware and executes them. As soon as the boot process has concluded, it transfers control to the Reliant UNIX operating system.
- If boot2 was booted accordingly from the board debug monitor (the last number of the bootflags is 1), it awaits user input with the boot2: prompt. You can then perform system administration tasks.

3.7.1 Switching to boot2

Restart the system.

Autoboot switched on (last number of the bootflags set to 0)

If the startup tests ran without errors, boot2 announces that Reliant UNIX will be loaded after a brief waiting period (approximately 10 seconds):

```
Autoboot: Waiting to load sdisk011s0/unix sd(1,8,0)unix root=ios0/sdisk011s0 sw
ap=ios0/sdisk011s1
(Ctrl-C to abort, RETURN to expedite)
```

Use this waiting period to abort the boot process by pressing [Ctrl]+[C].

Autoboot switched off (last number of the bootflags set to 1)

If the startup tests ran without errors, boot2 is accessed automatically.

Autoboot switched off (last number of the bootflags set to 8 or 9)

If the startup tests ran without errors, you access the board debug monitor.

- ▶ Now load boot2 as follows (example for *sdisk000s0* in an RM600-xxx):

Logical boot mode

```
BDM(MBII0t)>b201sdisk000s0/sash
```

Physical boot mode

```
BDM(MBII0t)>b201sd(0,0,0)sash
```

3.7.2 Important commands in boot2

You can perform certain administration tasks in boot2. Various commands are available for this purpose. The following table provides an overview of these commands. A detailed description of all commands and options is contained in the manpage for *boot2(8)*.

Command	Function
General commands	
<i>boot</i>	Loads Reliant UNIX. Logical boot mode: Only a hard disk that is entered in the root partition list in the CSI/(E)HIOS-NVRAM (when the disk is installed) can be specified as the load device. Physical boot mode: You can specify one of the following media as the load device: □ – SCSI I/O disks (IOS), □ – CD-ROM.
<i>cp</i>	Copies the contents of one file to another file.
<i>help</i> or <i>?</i>	Displays the syntax of the boot2 commands.

<i>installrins tall</i>	Loads the mini operating system main menu from the local medium. □ The mini operating system main menu is loaded from the network.
<i>version</i>	Displays the boot2 version.
Commands for logical boot mode (available as of Reliant UNIX 5.43)	
<i>nv_backu p</i>	Calculates the checksums using the software parameter area of the CSI/(E)HIOS-NVRAM and saves the area and checksums to the disks of the system cabinet.
<i>nv_list</i>	Displays the checksums stored on the disks of the system cabinet.
<i>nv_restore</i>	Copies the checksums and the software parameter area from the specified hard disk to the CSI/(E)HIOS-NVRAM.
<i>root</i>	Displays the root partition list and allows the entries to be modified manually. This is only necessary if entries are incorrect and are to be corrected.
Commands for both boot modes	
<i>loadenv</i>	Loads the boot parameters from the <i>/etc/default/boot</i> file.
<i>printenv</i>	Displays the values of the boot parameters.
<i>setenv</i>	Creates a new boot parameter or modifies the value of an existing boot parameter.
<i>unsetenv</i>	Deletes a boot parameter.

Table 18: Important commands in boot2

You will require the *boot* and *copy* commands, in particular, for routine work. If you require information on the syntax of the commands, you can use the *help* or *?* commands.

In most situations, it is possible to abort a boot2 command. In this case, press [Ctrl]+[D] or [Ctrl]+[C]. Boot2 then waits for the user to input information with the boot2: prompt.

You can load the mini operating system main menu from boot2, for example. The *install* command is provided for this purpose. If you wish to execute this command, you must supply the software required by the command on a CD-ROM.

Boot2 can boot Reliant UNIX from various media. Normally, it loads the */unix* file from the system disk. However, it is also possible to load an executable system kernel from any other disk drive containing a ufs file system with a suitable kernel. The file system must not be mounted beforehand.

A kernel is normally only loaded from CD-ROM or tape device if an installation is to be performed. This procedure can also be useful, however, if a fault on the system disk prevents the kernel from being booted there. Consequently, boot2 does not need to perform any read operations on the disk.

3.7.3 Root partition list

As of Reliant UNIX Version 5.43, the logical boot mode is used by default (RM600-xxx: Firmware V2.42 and higher for the CSI). Up to 10 system disks can be administered in this mode. For that reason, a list of all system disks known to the system is stored in the NVRAM in the root partition list. All available hard disks contain a copy of the NVRAM.

```
line state log_part swap_part phys_part disk_id
0 active sdisk011s0 sdisk011s1 sd(1,8,0) SEAGATE ST15150W..
1 inactive sdisk030s0 sdisk030s1 sd(3,0,0) SEAGATE ST15150N..
2 undefined
```

```

3 undefined
4 undefined
5 undefined
6 undefined
7 undefined
8 undefined
9 undefined
State: 0 inactive 1 active 2
    
```

Figure 7: Sample root partition list

The columns in the root partition list have the following meaning:

<i>line</i>	Line number (0 ... 9)	
<i>state</i>	Status	
	<i>active</i>	Active system disk
	<i>inactive</i>	Inactive system disk
	<i>undefined</i>	Undefined or not assigned
<i>log_part</i>	Logical bootstring	
<i>swap_part</i>	Logical swap string	
<i>phys_part</i>	Physical bootstring	
<i>disk_id</i>	Unique ID name of the hard disk	

In normal mode, the root partition list is managed **automatically**:

- If Reliant UNIX is installed on a hard disk, the new system disk is automatically entered in the root partition list with the status *active*. This is the case for both new installations and update installations. The disk on to which the operating system was installed is then the active system disk.
- If the active system disk is mirrored, you can specify from which of the two disks the new start is to be executed using the *bootdisk(8)* command. This disk is automatically entered
 - in the root partition list
This overwrites the entry for the previously active system disk.
 - in the bootstring

The root partition list must normally only be changed **manually** if the system cannot be booted from the active system disk. The *root* command is provided in boot2 for this purpose. A description of this command can be found in the manual entry *boot2(8)* and in the "System Administrator's Reference Manual".

3.7.4 Exiting boot2

When you have concluded your work in boot2, you can boot Reliant UNIX. The boot parameters used here are described in the manual entry for *bootvariable(7)*. This manual entry also explains which boot parameters are currently used if you start the boot command without specifying any boot parameters.

Booting Reliant UNIX in multiuser mode

Enter one of the following commands (example of *sdisk000s0* in an RM600-xxx):

Logical boot mode

```
boot2: boot -f /unix
```

Physical boot mode

```
boot2: boot -f sd(0,0)unix
```

Short form

```
boot2: boot
```

Reliant UNIX then enters the run level that corresponds to the *initdefault* entry in the */etc/inittab* file. Generally, this is run level 2 (multiuser mode).

Booting Reliant UNIX in single-user mode

Enter one of the following commands (example of *sdisk000s0* in an RM600-xxx):

Logical boot mode:

```
boot2: boot -f /unix initarg=-s
```

Physical boot mode:

```
boot2: boot -f sd(0,0,0)unix initarg=-s
```

Short form:

```
boot2: boot initarg=-s
```

The *initarg* argument causes Reliant UNIX to enter single-user mode directly.

Switching to the board debug monitor

RM600-xxx: [Ctrl]+[A]

RM600 E: [Ctrl]+[@], [[Shift]]+([B],[D],[M])

3.8 Powering up the system

There are various situations in which the system powers up:

- The computer is switched on. The firmware loads boot1 (or the PROM/Firmware monitor) and boot2 (or the SASH), which in turn loads Reliant UNIX from the system disk. The default start file is `/unix`. Load parameters can be set in the board debug monitor (see the sections "[Setting the boot mode](#)" and "[Setting the bootstring](#)").
- Reliant UNIX is booted from the board debug monitor or from boot2. This is the case, for example, if you have performed administration tasks in one of the interfaces mentioned above (see the sections "[Exiting the board debug monitor](#)" and "[Exiting boot2](#)").
- Reliant UNIX is prompted to execute a restart (see the [Section "Restarting the system"](#)). In this case, the system shuts down, checks in the process whether or not a new start file is to be set up and then does this if necessary. It then mounts all file systems and transfers control to the firmware, which then boots the system again.
- Reliant UNIX aborted itself as a result of a serious error. It is then rebooted automatically – as in the case of a restart.

Various commands are available for starting up and shutting down the system. [Figure "Commands for starting up and shutting down the system \(RM200 - RM400\)"](#) and [Figure "Commands for starting up and shutting down the system \(RM600\)"](#) provides an overview of these. The commands required will be determined by the boot mode set.

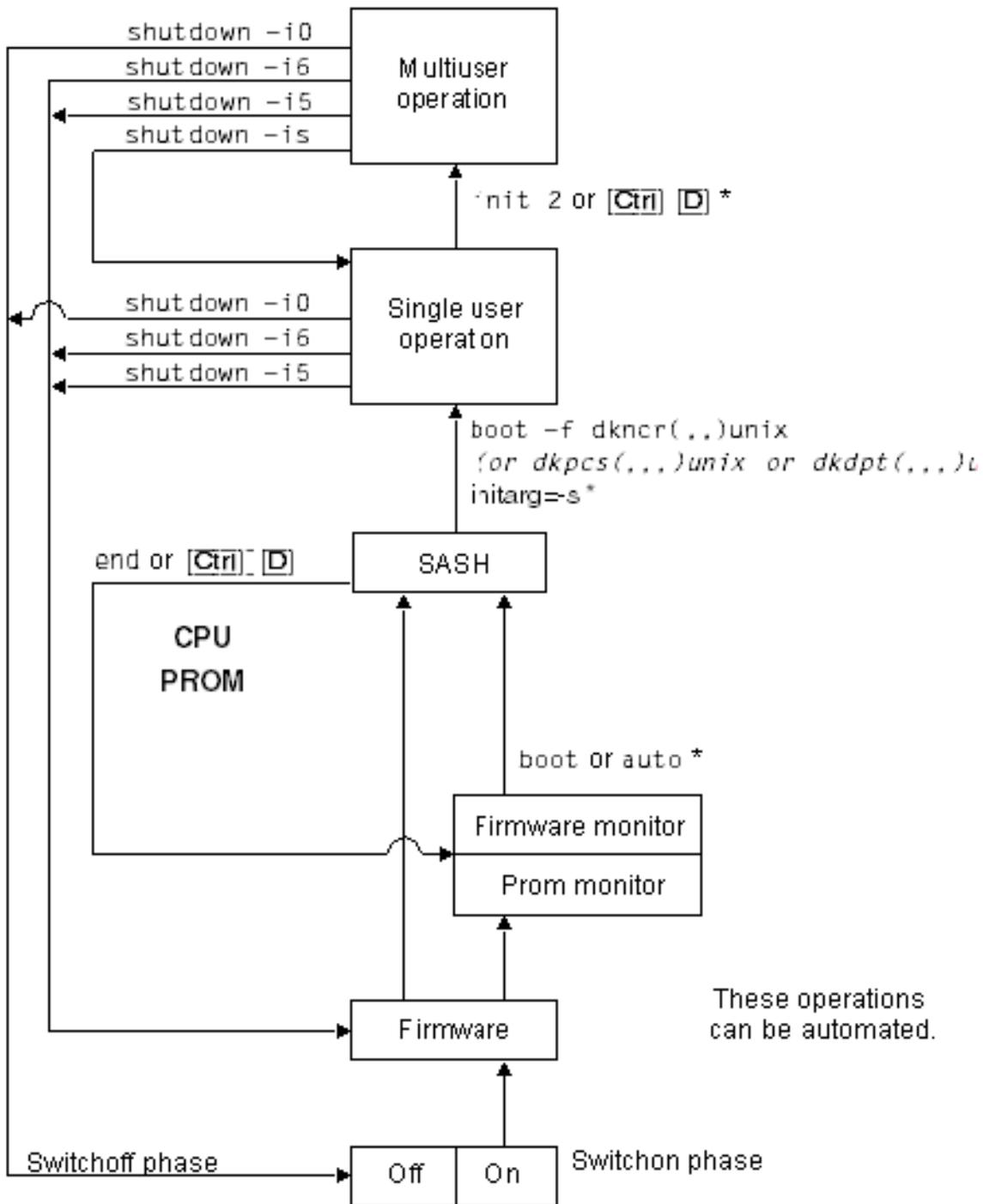


Figure 8: Commands for starting up and shutting down the system (RM200 - RM400)

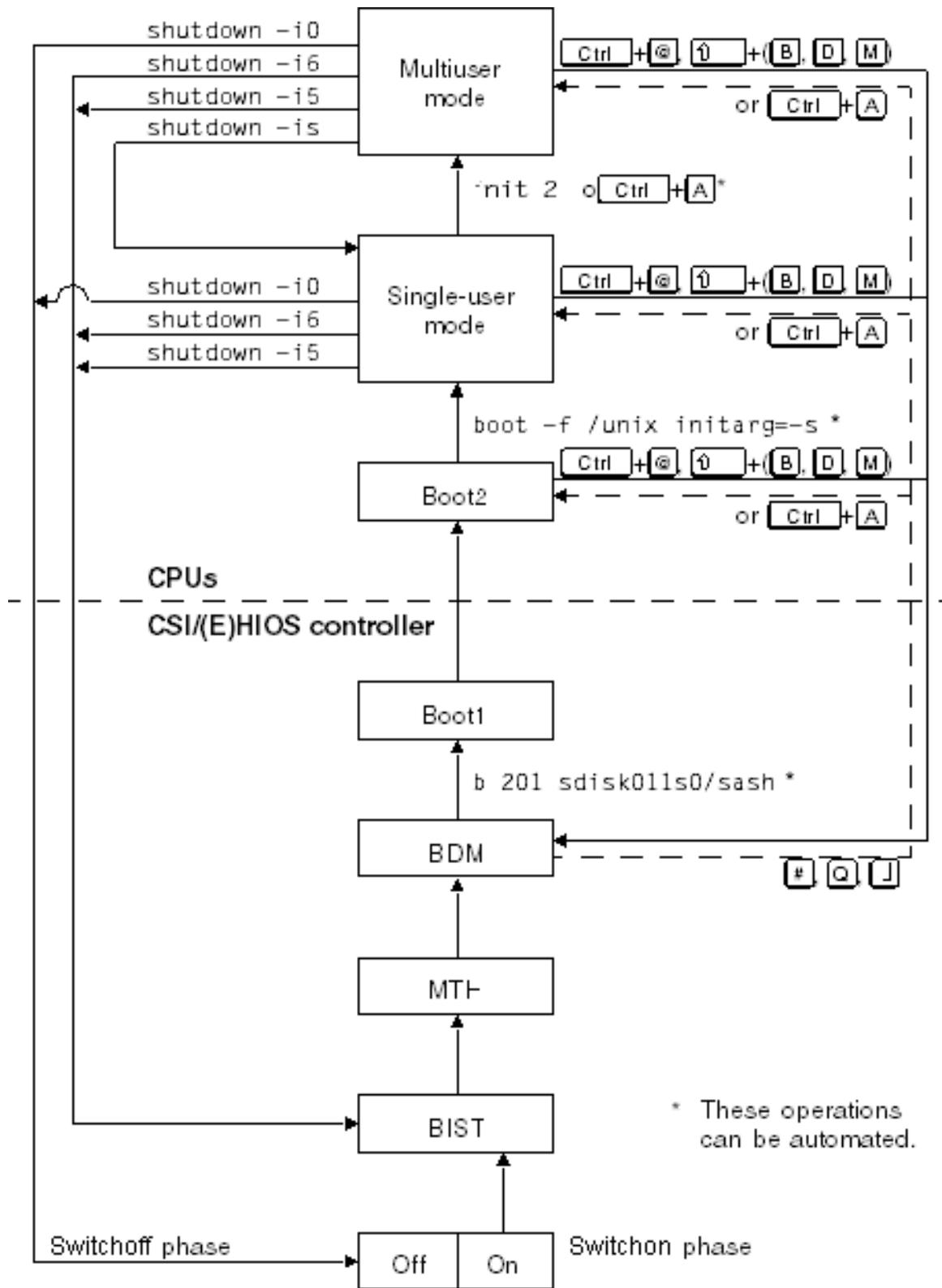


Figure 9: Commands for starting up and shutting down the system (RM600)

3.8.1 Powering up to the default run level (RM200 - RM400)

If you do not change the boot process, Reliant UNIX automatically boots to the run level that corresponds to the *initdefault* entry in the */etc/inittab* file.

Autoboot is switched on

If the startup tests ran without errors, the firmware displays a message indicating that the SASH will be loaded after a short waiting time (approximately 20 seconds), e.g.:

```
Autoboot: Waiting to load dkncr(0,0,10)sash (Ctrl-C to abort, RETURN to expedite)
```

Autoboot is switched off

If the startup tests ran without errors, you access either the PROM monitor or the firmware monitor.

```
Prom monitor      >> auto
```

```
Firmware monitor  SINIX boot (auto)
```

The firmware now displays a message indicating that the SASH will be loaded after a short waiting time (approximately 20 seconds), e.g.:

```
Autoboot: Waiting to load dkncr(0,0,10)sash (Ctrl-C to abort,
RETURN to expedite)
```

Booting SASH and Reliant UNIX

If you do not react to this message or press a key, the firmware waits for 20 seconds before it boots the SASH automatically. You can reduce the waiting time by pressing **Ú**.

The file whose name is stored in the *bootfile* environment variable is booted as the SASH. This environment variable is located in the non-volatile memory of the computer. Please refer to [Switching to the PROM monitor after the computer is switched on](#) or [Section "Setting the boot mode"](#) for information on how to change or display this variable.

The SASH boots Reliant UNIX immediately. Reliant UNIX boots to the run level that corresponds to the *initdefault* entry in the */etc/inittab* file. If this is multiuser mode, you will be requested to log on as a user as soon as the boot process is concluded.

```
The system is ready.
paderborn
Console Login: _
```

Enter your login name and the associated password. Information on the Reliant UNIX version, the copyright and the time of your last login is displayed on the screen. The shell prompt is then displayed.

3.8.2 Booting to the default run level (RM600)

If you do not change the boot process, Reliant UNIX automatically boots to the run level that corresponds to the *initdefault* entry in the */etc/inittab* file.



Please note that your individual configuration may cause your system output to differ slightly from what appears below.

Autoboot activated (last digit of boot flags is 0)

If the startup tests have run successfully, boot1 and boot2 are loaded automatically. Continue reading from the section [Booting Reliant UNIX](#).

Autoboot deactivated (last digit of the boot flags is 1)

If the startup tests ran without errors, you access boot2. Enter the following command (example for *sdisk000s0* in a RM600-xxx):

```
Logical boot mode
```

```
boot2: boot -f /unix
```

Physical boot mode

```
boot2: boot -f sd(0,0,0)unix
```

Autoboot deactivated (last digit of the boot flags is 8 or 9)

If the startup tests ran without errors, you access the board debug monitor. Enter the following command (example of *sdisk000s0* in an RM600-xxx):

Logical boot mode

```
BDM(MBII0t)>b200sdisk000s0/sash
```

Physical boot mode

```
BDM(MBII0t)>b200sd(0,0,0)sash
```

Booting Reliant UNIX

Boot2 then announces that Reliant UNIX will be loaded after a brief waiting period:

```
Autoboot: Waiting to load disk000s0/unix sd(0,0,0)unix root=ios0/sdisk000s0 sw
ap=ios0/sdisk000s1
(Ctrl-C to abort, RETURN to expedite)
```

If you do not react to this message or if you do not press a key, boot2 waits approximately 10 seconds before it boots Reliant UNIX automatically. You can reduce this waiting period by pressing `U`. Reliant UNIX then boots to the run level that corresponds to the *initdefault* entry in the */etc/inittab* file. If this is multiuser mode, you will be requested to log on as a user as soon as the boot process is concluded.

```
The system is ready.
paderborn
Console Login: _
```

Enter your login and the associated password. Information on the Reliant UNIX version, the copyright and the time of your last login are displayed on screen. The shell prompt is then displayed.

3.8.3 Booting to single-user mode

If you do not change the boot process, Reliant UNIX boots automatically to the run level that has been preset in the */etc/inittab* file. You can also have Reliant UNIX boot to another run level. This can be useful, for example, if you want to perform certain administration tasks that require single-user mode directly after booting the system.

Autoboot switched on (last digit of the boot flags is 0)

If the startup tests ran without errors, boot2 announces that Reliant UNIX will be loaded after a brief waiting period (approximately 10 seconds):

```
Autoboot: Waiting to load disk011s0/unix sd(1,8,0)unix root=ios0/sdisk011s0 sw
ap=ios0/sdisk011s1
(Ctrl-C to abort, RETURN to expedite)
```

Use this waiting period to abort the boot process by pressing `[Ctrl]+[C]`. You then enter boot2. In this case, you should now read the section [Booting Reliant UNIX](#).

Autoboot switched off (last digit of the boot flags is 1)

If the startup tests ran without errors, you automatically access boot2. In this case, you should now read the section ["Booting Reliant UNIX"](#).

Autoboot switched off (last digit of the boot flags is 8 or 9)

The procedure for the RM600 is described in [Section "Procedure for the RM600"](#).

3.8.3.1 Procedure for RM200 - RM400

Booting SASH and Reliant UNIX

```
Prom monitor    >> boot
```

Firmware monitor Stand-alone `shell` `boot` (boot)

After a short period of time, SASH responds with the `sash: prompt`. You can now boot Reliant UNIX as follows e.g.:

```
sash: dkncr(,,)unix initarg=-s
```

Enter `dkpcs` or `dkdpt` instead of `dkncr`, depending on the value of `bootfile` (see also ...).

The `initarg` argument is transferred to the system kernel during loading. It cancels the `initdefault` entry in the `/etc/inittab` file, which normally has the value 2 (multiuser mode).

When the boot process is complete, you are given two options:

```
Type Ctrl-d to proceed with normal startup,
(or give root password for system maintenance): _
```

- If you press [Ctrl] + [D] you are asked to specify the run level to which you want to boot UNIX:

```
ENTER RUN LEVEL (0-6, s or S): _
```

- If you enter the root password, the single-user mode prompt is displayed:

```
Entering System Maintenance Mode
# _
```

You can now perform the desired administration tasks. When you have finished, switch to multiuser mode:

```
# init 2
```

UNIX then boots to multiuser mode and responds with the `Login: prompt`. You can now log on to the console, either as the system administrator with the login `root` or as a normal user.

3.8.3.2 Procedure for the RM600

If the startup tests ran without errors, you access the board debug monitor. Enter the following command (example of `sdisk000s0` in an RM600-xxx):

```
Logical boot mode        BDM(MBII 0 t) > b 201 sdisk000s0/sash
```

```
Physical boot mode      BDM(MBII 0 t) > b 201 sd(0,0,0)sash
```

You then enter `boot2`.

Booting Reliant UNIX

Start Reliant UNIX as follows (example of `sdisk000s0` in an RM600-xxx):

```
Logical boot mode:      boot2: boot -f /unix initarg=-s
```

```
Physical boot mode:     boot2: boot -f sd(0,0,0)unix initarg=-s
```

Boot2 then announces that Reliant UNIX will be loaded after a brief waiting period (approximately 10 seconds):

```
Autoboot: Waiting to load sdisk000s0/unix sd(0,0,0)unix root=ios0/sdisk000s0 sw
ap=ios0/sdisk000s1
(Ctrl-C to abort, RETURN to expedite)
```

The `initarg` argument is transferred to the kernel during loading. It cancels the `initdefault` entry in the `/etc/inittab` file, which normally has the value 2 (multiuser mode). When the boot process is concluded, you are given two options:

```
Type Ctrl-d to proceed with normal startup,
(or give root password for system maintenance): _
```

- If you press [Ctrl]+[D], you are asked to specify the run level to which you want to boot Reliant UNIX:

```
ENTER RUN LEVEL (0-6, s or S): _
```

- If you enter the root password, you get the single-user mode prompt:

```
Entering □ System □ Maintenance □ Mode  
# □ _
```

Now perform the desired administration tasks. When you have finished, switch to multiuser mode:

```
# □ init □ 2
```

Reliant UNIX then boots to multiuser mode and responds with the Login: prompt. You can now log on to the console, either as the system administrator with the login *root* or as an ordinary user.

Continue reading from the [Section "Messages from the master test handler \(RM600\)"](#).

3.8.4 Startup tests for the RM200 - RM400

If the boot mode is not set to *d* (*bootmode* environment variable), the diagnostic tests are carried out after the computer has been switched on. Detailed information on the environment variables is contained in the manpage for *prom*(8) and *nvr*am(7).

Skipping the startup tests

If you want to skip the startup tests so that the system boots faster, press the [s] key after the message:

Hit s key to SKIP PON Tests...

The system prompts for confirmation that the startup tests are to be skipped:

Skip Power-On Tests!

The SASH is booted automatically after this message if autoboot mode is enabled or the PROM monitor is loaded (see the [Section "Firmware messages \(RM200 - RM400\)"](#)).

Messages from the startup test

The messages from the tests are sent to the 2681-Duart, which serves the terminal on *tty*(0). If you have not connected a terminal on *tty*(0), you cannot see these messages.

The figure shown below illustrates the tests that are carried out during the startup phase (*bootmode c, m* or *s* for the RM200 and alphanumeric console) which can also be displayed on a graphics terminal.



Please note that your individual configuration may cause your system output to differ slightly.

Each startup test is output in a separate line. The lines contain the following information:

- At the beginning of each test, the name of the test is displayed followed by 3 dots. The result is displayed at the end of the test:

PASSED	The test ran without errors.
FAILED	An error was found.
SKIPPED	The test cannot be performed because an error was found in the previous test.

```
Running Power-On Diagnostics...
Hit s key to SKIP PON Tests... Not Skipped
Cache Test #1...PASSED
Secondary Cache Test...PASSED
ECC Cache Test...PASSED
Data Cache MATS+ Test...PASSED
Memory Test...1 2 3 4 5 6 7 8 9 A B C PASSED
Battery Check Test...PASSED
NVRAM Test...PASSED
Duart 1 26C92 Port Tests...Channel B...PASSED
FP Test #1...PASSED
FP Test #2...PASSED
SCSI CONTROLLER 0:
  Read/Write Test...PASSED
  Inc/Dec Test...PASSED
  SCSI FIFO Test...PASSED
  DMA FIFO Test...PASSED
  SCRIPT Interrupt Test...PASSED
  DMA copy Test...PASSED
IdProm Test...PASSED
  Ethernet address: 00:00:e4:03:00:0e:
  Ethernet Checksum: 0x00
LAN Coprocessor (i82596) Test
  Port Self-test...PASSED
  Lance Internal LoopBack Master Test...PASSED
  Lance External LoopBack Master Test...PASSED
ICU Test...PASSED
```


Continue reading from the [Section "Messages displayed when loading the SASH \(RM200 - RM400\)"](#).

3.8.6 Messages from the master test handler (RM600)

Each time the system boots, the built-in self tests (BIST) are first carried out. The messages of the tests are sent to the cit driver, which serves the terminal on *tty(0)*. If you have not connected a terminal to *tty(0)*, you cannot see these messages.

RM600-xxx

The following figure shows the format of the built-in self test messages.

```
Power On Tests Running
SPbus Configuration and BIST Status
Slot Product Code Vendor Rev Class Test Status
0 MMM-167101 SNI 00H 03H 5 PASS
2 CPU-156800 SNI 01H 01H 51 PASS
7 MBP-269704 SNI 00H 02H 81 PASS
14 CPU-156800 SNI 01H 01H 51 PASS
Multibus Configuration and BIST Status, Bus_ID: 80H
Slot Product Code Vendor Rev Class SSR SLR Test Status
0 CSI-195501 SNI 05H F2H 00H A3H 123 PASS
1 SIH-195700 SNI 03H F2H 00H 23H 103 PASS
2 TCA-279900 SNI 01H F2H 00H 23H 111 PASS
9 DCS_273400 SNI 06H F2H 00H 23H 81 PASS
11 DCS_273400 SNI 06H F2H 00H 23H 81 PASS
16 SIM-195600 SNI 03H F2H 00H 23H 96 PASS
17 MBP-269704 SNI 00H F2H 00H 23H 81 PASS
SYSTEM CONFIGURATION AND BIST STATUS.....PASS
SPBUS INITIALIZATION PHASE SUMMARY Proceed to
Slot Product Code Test Summary Status Boot Phase?
0 MMM-167101 Passed Active Yes
2 CPU-156800 Passed Active Yes
7 MBP-269704 Passed Active Yes
14 CPU-156800 Passed Active Yes
MULTIBUS INITIALIZATION PHASE SUMMARY, Bus_ID: 80H Proceed to
Slot Product Code Test Summary Status Boot Phase?
0 CSI-195501 Passed Active Yes
1 SIH-195700 Passed Active Yes
2 TCA-279900 Passed Active Yes
9 DCS_273400 Passed Active Yes
11 DCS_273400 Passed Active Yes
16 SIM-195600 Passed Active Yes
17 MBP-269704 Passed Active Yes
System now entering Boot Phase...
```

The SP bus modules are tested first and then the MBII module (MBII = Multibus II). Each test is output in a line containing the following information:

- Number of the slot in the system unit
- Product name of the controller
- Test results:

PASSED

The test ran without errors.

FAILED

An error was found.

SKIPPED

The test cannot be performed because an error was found in the previous test.

RM600 E

The following figure shows the format of the built-in self test messages.

```

BISTs running
T5 SYSTEM ARCHITECTURE - Master Test Handler
Copyright 1995, SNI Corporation
Reset Type: COLD RESET
CONFIGURATION
Cab_ID Bus_ID Agent Slot Agentname Test_ID Test_Stat
-----
0x00 0x00 00000000 000002 CP4-V01 38 00000000
0x00 0x00 00000000 000002 CP4-V01 38 00000000
0x00 0x00 00000001 000002 CP4-V01 38 00000000
0x00 0x00 00000002 000004 CP4-V01 38 00000000
0x00 0x00 00000002 000004 CP4-V01 38 00000000
0x00 0x00 00000003 000004 CP4-V01 38 00000000
0x00 0x00 00000004 000007 HIOS 100 00000000
0x00 0x00 00000005 000008 HIOS 100 00000000
0x00 0x00 00000006 000010 MBP-286002 81 00000000
0x00 0x8A 00000000 000000 MBP-286002 81 00000000
0x00 0x8A 00000003 000003 LCE-279400 123 00000000
0x00 0x8A 00000004 000004 SIM-195600 96 00000000
0x00 0x8A 00000005 000005 TCA-279900 111 00000000
SYSTEM CONFIGURATION AND BIST STATUS.....PASSED
System now entering Boot Phase...
    
```

Each test is output in a line containing the following information:

- Number of the cabinet
- Number of the bus
- Number of the slot
- Product name of the controller
- Number of the test program
- Test results:
 - 0 The test ran without errors.
 - 8 Warning: An error was found which impairs the functioning of the controller.
 - 32 No test was performed because the controller is marked as LAR defect.
 - 128 An error was found.

Board debug monitor

If autoboot is switched on (the last digit of the boot flags is 0), boot1, boot2 and Reliant UNIX are now booted automatically. This occurs, for example, when you start up the computer for the first time. In this case, you should now turn to [Section "Messages displayed when loading boot1 and boot2 \(RM600\)"](#). However, you can switch off autoboot at a later stage if you wish (see ...). In this case, the firmware loads the board debug monitor, which returns with the following prompt.

```
□□BDM□Board□Debug□Monitor
□□Copyright□(c)□Siemens□Nixdorf□Informationssysteme□AG□1990/1991
BDM(SPB□□8□t)>_
```

3.8.7 Messages displayed when loading boot1 and boot2 (RM600)

The following messages appear when boot1 is loaded:

```
*** SYSTEM BOOT
System Type: RM600-Midrange
System Name: UNIX RM600-M230
System bootable: yes
Cabinet # 0: 10 boards - FAN MID
*** CPU activated - 1st bootstage complete
boot1 revision: 1.22
```

The following messages appear when boot2 is loaded:

```
Bootstring: ,sdisk011s0"
New physical Bootstring: ,sd(1,8,0)"
Load-Addr.: 0x80030000
Length: 0x3FF2C
Entry-Point: 0x8003001C
CHECK_NVRAM NVRAM-CHECKSUM: 908a CHECKSUM OK
Second Level Boot: Version 5.45
```

If autoboot is switched on (the last digit of the boot flags is 0), Reliant UNIX is now booted automatically. This occurs, for example, when you start up the computer for the first time. However, you can switch off autoboot later on if you wish (see ...). In this case, you then see the boot2: prompt.

Continue reading from the [Section "Messages displayed when loading Reliant UNIX \(RM600\)"](#).

3.8.8 Messages displayed when loading the SASH (RM200 - RM400)

An appropriate message appears when the SASH is being loaded, e.g.

```
loading
300960+130512+629952 entry: 0xa0600000
SNI Standalone Shell Version 3.0005 SNI_R4000 OPT
Wed May 4 11:41:34 WET 1994 livsa542
```

3.8.9 Messages displayed when loading Reliant UNIX (RM200 - RM400)

An appropriate message appears when Reliant UNIX is being loaded, e.g.:

```
Loading dkncr(0,0,0)unix
2185644+1468592+661592 entry: 0x80021000
SNI RM400
UNIX(R) Reliant UNIX Release 5.45 Version B00
Copyright (c) 1984, 1986, 1987, 1988, 1989, 1990, 1991, 1992
Pyramid Technology C
Copyright (c) 1984, 1986, 1987, 1988, 1989, 1990, 1991, 1992 AT&T
Copyright (c) 1991, 1992, 1993, 1994
Siemens Nixdorf Informationssysteme AG
All Rights Reserved
Processor Configuration:
Boot processor: Cpu #0 (p0)
Other processor(s): None.
```

This is followed by a series of other messages. The hardware of the computer and the way in which Reliant UNIX is configured will determine which messages are output. The messages are logged in the `/var/adm/log/osm` file (See also the "Diagnostics and Troubleshooting" manual).



During this time you must not switch off the computer or press the RESET key. Otherwise, data may be lost.

Continue reading from the [Section "Procedure in the event of an error \(RM200 - RM400\)"](#).

3.8.10 Messages displayed when loading Reliant UNIX (RM600)

If autoboot is switched on (the last digit of the bootflags is 0), Reliant UNIX is now booted automatically. This occurs, for example, when you start up the computer for the first time.

```
Autoboot: □Waiting □to □load □sdisk011s0/unix □sd(1,8,0)unix □root=ios0/sdisk011s0 □sw
ap=ios0/sdisk011s1
(Ctrl-C □to □abort, □RETURN □to □expedite)
```

When Reliant UNIX is loaded, the following messages appear first:

```
3884524+2424704+7224536 entry: 0x801c0400
Physical memory size = 640 MB (40960 pages, pagesize=16 KB)
SNI RM600
UNIX(R) SINIX-Y Release 5.45 Version B0000
Copyright (c) Siemens AG 1990-1997
Basis: DC/OSx (R), Copyright (c) Siemens Pyramid Information Systems, Inc. 1984
UNIX (R), Copyright (c) X/Open Company Limited 1983
All rights reserved
SINIX is a registered trademark of Siemens AG
DC/OSx is a registered trademark of Siemens Pyramid Information Systems, Inc.
Reliant is a registered trademark of Siemens Pyramid Information Systems, Inc.
X/Open is a registered trademark, and the X device is a trademark,
of X/Open Company Limited.
UNIX is a registered trademark in the United States and other countries,
licensed exclusively through X/Open Company Limited.
Processor Configuration:
Boot processor: Cpu #0 (p0)
Other processor(s): #1 (p1), #2 (p2), #3 (p3), #4 (p4)
#5 (p5), #6 (p6), #7 (p7)
```

This is followed by a series of other messages. The hardware of the computer and the way in which Reliant UNIX is configured will determine which messages are output. The messages are logged in the */var/adm/log/osm* file and in other files.



During this time you should not switch off the computer. Otherwise, data may be lost. Continue reading from the [Section "Procedure in the event of errors \(RM600\)"](#).

3.9 Procedure in the event of an error (RM200–RM400)

This section describes a basic procedure in the event of an error. You will find detailed information in the Diagnostics manual.

If an error occurs during the startup phase, the corresponding output reads *FAILED* instead of *PASSED*. The boot mode is automatically set to *e* (error) to prevent the PROM monitor from loading the Reliant UNIX operating system. In addition, the tests that rely on components whose test produced an error are also skipped. In this case, *SKIPPED* is displayed instead of *PASSED* (e.g. the *DATA Cache MATS+* test is skipped if the *Cache Test 1* failed).

The example below shows an error in the SCSI 0 read/write test. This error was caused by a connection in the computer being cut.

```
<preceding startup tests>
SCSI CONTROLLER:
  Read/Write Test(fail:0a)...FAILED
<subsequent startup tests>
```

You must first recover the underlying errors before you can boot Reliant UNIX.

Then reset the boot mode to *c*. To do this, use the `setenv bootmode c` command. Finally, you must carry out one of the following actions:

- ▶ Press the RESET key for at least 1 second (execute reset).
- ▶ Switch off the computer and then switch it back on again (execute cold start).
- ▶ Enter the `auto` command and press `U`.

The first two options run the tests of the startup phase again in order to ensure that the error has been corrected successfully.

3.9.1 Booting an alternative system kernel

The new system kernel cannot be correctly booted in a few cases, for example if you have modified a system parameter or combination of parameters such that the new system kernel is too large or does not initialize itself correctly.

In such a case, you must first reboot the old system kernel to make the operating system executable. To do this:

- ▶ Switch to the PROM monitor or firmware monitor. Retrieve the bootstring with the firmware *printenv* command:

```
>> printenv bootfile
```

- ▶ Boot the SASH, e.g

```
>> boot -f dkpcs(0,0,0,10)sash
```

- ▶ Boot the old system kernel

```
sash: boot -f dkpcs(0,0,0,0)stand/unix.old
```

This kernel was renamed to */unix.old* the last time Reliant UNIX was booted. You may also boot an individually saved system kernel instead of *unix.old*.

- ▶ Login as the user *root*.
- ▶ Save the executable system kernel in case the next version also does not function.

```
# mv /unix.old /unix.sav
```

- ▶ Carry out the necessary modifications.
- ▶ Execute the *idbuild* command again.
- ▶ Reboot Reliant UNIX again.

If, for any reason, this system kernel is also non-functional, boot from */unix.sav* instead of */unix.old* and try again.



Each system kernel occupies approximately 9 Mbytes in the */stand* directory!

3.9.2 Booting from a mirror disk

If your system disk is no longer available, you may also boot from a mirror disk if one is present.

If the system disk specified in the NVRAM *bootfile* variable can no longer be booted, use the *dkprint* command to determine the allocation of channels IDs to the numbers assigned by the firmware, e.g.

```
>> dkprint
```

```
PCI-Path Channel-ID Bootfile Comments
```

```
PCI#ONBOARD#0 0 -
```

```
PCI#ONBOARD#1 1 -
```

```
PCI#2.0 3 dkdpt(0,0,0,10)sash
```

```
PCI#2.1 2 -
```

```
PCI#2.2 4 -
```

```
PCI#3 5 -
```

```
PCI#1/0 6 dkpcs(0,0,1,10)sash
```

```
PCI#1/1 7 dkpcs(0,1,1,10)sash
```

```
PCI#1/2□□□□□□□□□□8□□□□□□□□□□-
```

- ▶ If a dash (-) is output in the *Channel-ID* column, no channel ID has been assigned till now. A dash in the *Bootfile* column means that the disk is not bootable.
- ▶ Boot the operating system. If the mirror disk in the above example is the one with channel ID 7, its Reliant UNIX name is */ios1/sdisk071*. The system can then be booted with the following command:

```
>>□ boot -f dkpcs(0,1,1,0)/stand/unix root=ios0/sdisk071s0
swap=ios0/sdisk071s1 bootflags=0x0
```



It is generally recommended to note the most important system data. You should therefore note the logical name of the mirror disk when setting up the mirror root.

If you cannot make a logical assignment, you can also connect the mirror disk in place of the original disk and boot the operating system without making further changes.



If you configured root mirroring with a RAID controller using RAID 1, you can boot without any special intervention. The RAID controller automatically uses the functional half of the mirror pair.

From Reliant UNIX 5.44B00 you can boot automatically from the mirror disk. To do this, boot mode must be set to *auto* or the *auto* command must be issued in the PROM monitor. The PROM firmware then searches for a SASH according to the following sequence:

1. boot from *bootfile*
2. boot from *dkncr(c,t,10)sash (c=0,1; t=0-4)*
3. boot from *dkdpt(c,b,t,10)sash (c=0-N_PCI; b=0-2; t=0-6)*
4. boot from *dkpcs(c,b,t,10)sash (c=0-N_PCI; b=0-2; t=0-6)*

where *N_PCI* indicates the module-dependent number of PCI slots.

The first SASH found is loaded and booted. The SASH compares the contents of *bootfile* with the device name of the device from which the SASH was booted. If the device names, including the parameters passed in brackets, do not correspond, the channel ID (*chid*) and the disk number (*target*) are determined and passed automatically for the Reliant UNIX system to be loaded.

Example: PROM monitor command *auto*

```
>> auto
```

No SASH can be loaded from *bootfile=dkncr(0,0,10)sash*. The PROM firmware finds the first SASH in *dkncr(0,3,10)*. This SASH is then booted. The SASH supplements the Reliant UNIX boot as follows:

```
dkncr(0,3,0)unix chid=0 target=3
```

The assignment of the channel ID corresponds to that of the *dkprint* command. Using the parameters passed, Reliant UNIX determines the corresponding root and swap device.

3.9.3 Booting from an alternative disk

If the system disk is no longer available, you can boot the operating system from another disk.

If you cannot make a logical assignment for the new disk, execute the `dkprint` command as described in the previous [Section "Booting from a mirror disk"](#). Boot the SASH with the `boot <bootstring>` command, e.g.

```
>> boot -f dkpcs(0,0,0,10)sash
```

You can then boot the operating system, e.g.

```
sash: dkpcs(0,0,0,0)unix
```

You can, however, also define the `bootfile` variable with the `printenv` command and start the operating system directly from the PROM monitor. Since this saves the new value of `bootfile`, it is automatically made available for all subsequent system starts, e.g

```
>> setenv bootfile dkpcs(0,0,0,10)sash
>> auto
```

If incompatible system modifications have been made (e.g. channel IDs mixed with `upd_channel_id`), you must reconfigure the system disk. The following section describes how you reconfigure the system disk.

3.9.4 Reconfiguring the system disk

If the solutions suggested above have not solved the problem, it is also possible to reconfigure the system disk.



You must reconfigure the system disk if it has been mounted on a different SCSI ID, SCSI channel or controller.

- ▶ You must initially boot the mini root file system from CD-ROM to reconfigure the system disk. Proceed as described in the manual "Reliant UNIX Installation".
- ▶ Select the *Switch into a shell* menu item.
- ▶ Call the *rr* (root recovery) command in the mini root file system. This command creates a list in the following form

```
bootable system(s) found on:
/dev/ios0/sdisk055s0□□□dkdpt(1,0,5,0)□□□□sash found□□□
or unix but no sash
....
```

- ▶ Now select the desired system disk (e.g. */dev/ios0/sdisk055s0*).

The appropriate nodes are subsequently created or modified, files and NVRAM entries are corrected and the system is then rebooted. This ensures that the system boots to single-user mode.



The system administrator is responsible for all further modifications (e.g. */etc/vfstab*, */etc/dktab*).



If none of the measures described in this section is successful, the operating system must be reinstalled.

3.10 Procedure in the event of errors (RM600)



A detailed description of the errors that can occur and what you can do to resolve these errors can be found in the manual "Diagnostics and Troubleshooting".

If an error occurs during the startup phase, the corresponding output reads *FAILED* instead of *PASSED*. The Reliant UNIX operating system is nevertheless booted, provided the error is not critical. However, the relevant board cannot be accessed subsequently. In addition, the tests that rely on components whose test produced an error are also skipped. In this case, *SKIPPED* is displayed instead of *PASSED* (e.g. the *DATA Cache MATS+* test is skipped if the *Cache Test* failed).

The example below shows an error in the SCSI 0 read/write test. This error was caused by a connection being cut in the computer.

```
<previous power-on-tests>  
SCSI CONTROLLER:  
  Read/Write Test (fail:0a)...FAILED  
<further power-on-tests>
```

In this case, please notify our Service department (see ...).

3.10.1 Starting the system with defective hardware components

The ability to diagnose hardware errors has been improved as of Reliant UNIX 5.43. Defective boards and SCSI devices are automatically left out of operation when starting the system. The LAR function (LAR = Logout Auto Recovery) is responsible for this. This function has the following features:

- Defective hardware components are marked during the system runtime.
 - Marking is either carried out automatically by the error reporting procedure and threshold value analysis or manually by the system administrator (using the *WebSysAdmin* tool). Information on the defective components is stored in the NVRAM of the CSI or (E)HIOS.
- Selected components are not started at system startup.
 - The decision as to which boards should not be started is made either directly in the BIST or in the controller configuration, if Reliant UNIX is loaded.

For further details on *WebSysAdmin* refer to the [Chapter "Node administration with WebSysAdmin"](#) later on in this manual.

Defective boards deactivated in the BIST

The status *FAIL* is returned if defective boards are not started in the BIST:

```
SPbus Configuration and BIST Status
Slot Product Code Vendor Rev Class Test Status
15 MBP-271101 SNI 04H 02H 81 PASS
15 MBP-271101 SNI 04H 02H 81 PASS
14 MBP-271101 SNI 04H 02H 81 PASS
14 MBP-271101 SNI 04H 02H 81 PASS
12 MMM-167201 SNI 00H 03H 238 FAIL <===
7 MMM-167201 SNI 00H 03H 5 PASS
0 MMM-167201 SNI 00H 03H 238 FAIL <===
2 MMM-167201 SNI 00H 03H 238 FAIL <===
13 MMM-167201 SNI 00H 03H 238 FAIL <===
.....
.....
```

After the BIST, the information in the CSI/(E)HIOS-NVRAM is used to establish which boards are defective when loading boot1. Boot1 stops system startup if the last board is defective in any way. The user is asked whether he/she wants to reboot the system despite boards being defective (exception: if the last CPU is defective).

```
CPU is LAR-defect. Do you want to continue (Y/N)? _
```

While it is possible to reboot the system, you should notify our Service department (see ...) so that the defective boards can be replaced.

Defective boards not deactivated in the BIST

Defective boards that were not deactivated in the BIST are not started in the controller configuration when loading Reliant UNIX. The following message is displayed:

```
CMN: WARNING: cmn 56594 MBII 3 mb2_pre_init(): Skipping broken CPU board.
MBII: 55: ctr_name: bus_type: cab_id: slot_id: test_id: test_stat: test_level
MBII: 1: CPU-150500: SP 0: 10: ee: 80: c0
```

Detecting defective boards

You can use the *WebSysAdmin* user interface to detect defective boards in the active system. You can also use the `/sbin/showconf -t` command to display all the boards not started at system start because of a defect. Such boards have the *FAILED* status. The *testid* column also has the value *ee*.

```
# /sbin/showconf -t
ctrl_name: chtyp: bustyp: cab: busid: slot: chan: tstid: tstlev: tst: tst_stat
MBP-286002: SP 0: 0: 14: 0: 51: cc: 0: PASSED
HIOS: SP 0: 0: 7: 0: 64: 0: 0: PASSED
HIOS: 300: SP 0: 0: 7: 0: 64: 0: 0: PASSED
HIOS: 300: SP 0: 0: 7: 1: ee: 0: 0: FAILED <===
```


► Select an NVRAM backup.

All checksums **identical**: You can use the NVRAM backup on any disk.

Checksums **not identical**: Use the NVRAM backup of one of the hard disks that was located in the system cabinet the last time the system was shut down. The checksums of these disks are identical.

► Copy the data from the selected hard disk to the NVRAM:

```
boot2: nv_restore sd(n,n,n)
```

For *n,n,n* enter the name of the desired hard disk as output by *nv_list* (e.g. *nv_restore sd(1,0,0)*).

► Switch to the board debug monitor.

```
RM600-xxx: [Ctrl]+[A]
```

```
RM600□E: [Ctrl]+[@], [[Shift]]+([B],[D],[M])
```

► Activate autoboot and execute a cold start:

```
BDM(SPB□□8□t)>wf□200
BDM(SPB□□8□t)>creset
```

Check 2

The first check described when the boot2 phase starts is repeated when Reliant UNIX is booted. If the NVRAM data was not restored after the first check failed, the error message described in the first check is output again and the boot process is aborted. To restore the NVRAM data, you have to work through the actions described in the first check. You can then boot Reliant UNIX.

Check 3

Check 3 compares the checksum of the NVRAM with the checksum of the NVRAM backup on the boot disk. If an error is detected, the system startup is interrupted. You get the following warning:

```
NVRAM_CHECKSUM and NVRAM_BACKUP_CHECKSUM of Rootdisk are different
boot2:
```

- If the warning is a result of one of the first two checks, proceed as described there.
- If the boot disk has been replaced, restore the NVRAM and continue system startup:

```
boot2: nv_backup
boot2: boot
```

3.10.3 Failure of the active system disk at system startup

Boot2 cannot be loaded if it is ascertained at system startup that the active system disk is defective. The following message is displayed:

```
Scanning□disks□using□Disk-ID□'SEAGATE□ST11200N□□□□□□□□00050906'
.....
BOOT:□no□disk□with□Disk-ID□'SEAGATE□ST11200N□□□□□□□□00050906'
1.□Press□Ctrl-A□to□return□to□BDM
2.□Check□Bootparameters
3.□Enter□'creset'□command□to□restart□the□system
```

Three different cases are possible:

Case 1:

An intact mirrored disk (e.g. *sdisk011*) of the active system disk exists. However this disk is (perhaps) not yet entered in the root partition list.

- Load boot2 from the intact mirrored disk:

```
BDM(SPB□□8□t)>b□1□sd(1,8,0)sash
```

- ▶ Enter the intact mirrored disk in the root partition list.

Use the *root* `boot2` command to do this. Example: The intact mirrored disk should be entered in line 1 of the root partition list.

```
boot2: root -n 1 -l sdisk011s0  
boot2: root -n 1 -w sdisk011s1  
boot2: root -n 1 -p sd(1,8,0)  
boot2: root -n 1 -s 2
```

- ▶ Check that the information in line 1 of the root partition list is correct and complete.

boot2: **root -l**



Incomplete or incorrect information in the root partition list can result in a loss of data.

- ▶ Continue starting the system.

boot2: **boot**

Case 2:

No mirrored disk of the active system disk exists.

However the system cabinet contains another system disk that is entered in the root partition list.

- ▶ First check the root partition list. Use the *root* boot2 command to do this.

boot2: **root -l**

- ▶ Activate the new system disk, for example in line 1.

boot2: **root -n 1 -s 2**

- ▶ Continue starting the system.

boot2: **boot**

Case 3:

The system cabinet contains neither an intact mirrored disk nor another system disk.

- ▶ Perform a new installation or contact our Service department (see ...).

3.10.4 Swapping the active system disk

In certain circumstances it may be a good idea to replace the active system disk with a "foreign" system disk from another RM600 system. The following example shows how to do this. The example is based on the assumption that the new disk has the physical address *1,8,0* and the logical address *sdisk011*.

- ▶ Turn off the machine.

- ▶ Swap the active system disk.

- ▶ Turn on the machine.

The system start ends automatically in the board debug monitor because the boot parameters do not match the new system disk.

- ▶ Load boot2 from the new disk.

```
BDM(SPB□□8□t)>b□1□sd(1,8,0)sash
```

- ▶ Enter the new disk in the root partition list, for example in line 1.

```
boot2: root -n 1 -p sd(1,8,0)
```

This command also enters the unique ID name (*disk_id*) of the new hard disk.

- ▶ Update the NVRAM with the new parameters.

```
boot2: nv_backup
```

- ▶ Switch to the board debug monitor.

```
RM600-xxx: [Ctrl]+[A]
```

```
RM600□E: [Ctrl]+[@], [[Shift]]+([B],[D],[M])
```

- ▶ Update the bootstring, activate autoboot and execute a cold start:

```
BDM(SPB□□8□t)>wn□0□sdisk011s0
```

```
BDM(SPB□□8□t)>wf□200
```

```
BDM(SPB□□8□t)>creset
```

3.10.5 Failure of a mirrored system disk while the system is running

The following section describes in brief what to do if the mirror of the active system disk is to be used instead of the active system disk for the next system restart. Detailed information can be found in the manual "Virtual Disks".

The following **preconditions** must be met:

- The active system disk must be mirrored on another hard disk.
- The Reliant UNIX operating system must be running.
- The active system disk must fail.

The following message is displayed:

```
Mirror vdisk7: piece 0 Read error at block 24.
Mirror vdisk7: piece 0 is now disabled.
Mirror vdisk7: 1 online piece left - this Mirror Disk is now NOT-MIRRORED.
Mirror vdisk7: Retrying the read on another piece.
```

You can also ascertain the error by listing the current status of the mirrored disk using the `dkmirror -l` command.

If the active mirrored disk has failed, its mirrored disk continues to run without interruption, but it is no longer secured by a mirror. Do the following:

- ▶ Activate the mirrored disk as the new system disk:

```
#bootdisk devicename
```

The `bootdisk(8)` command does the following:

- It ascertains the ID of the `devicename` disk.
- It enters the logical device name in the bootstring.
- It enters the disk as the active system disk in the root partition list and overwrites the previous entry.



This command may only be used if the aforementioned conditions are met. If these are not met and the command is used to activate another system disk (in particular one originating from another system), data losses can occur since the configuration files of the other system disk do not match the system hardware.

- ▶ Replace the defective hard disk.

3.11 Restarting the system

You must restart the system each time you reconfigure Reliant UNIX:

- A new kernel was generated and is to become effective.
For more detailed information, please refer to the [Section "Booting the new system kernel"](#).
- New software packages were installed.
Reliant UNIX should be reinitialized for correct operation.

Depending on the current run level, you should proceed as follows:

Multiuser mode

Before shutting down the system, check whether or not other users are logged on. To do this, use the `who(1)` command (see ...). If you are the only user logged on, you can shut down Reliant UNIX as you would in single-user mode (see overleaf).



The `shutdown` command automatically informs all users who are logged on about the imminent change in the run level (see ...). If you would like to give users more detailed information, you can use the

`wall(1M)` command (see the [Section "Informing users who are logged on"](#)).

Now enter the following commands:

```
# cd /
```

```
# shutdown -i6 -g900
```

The arguments are explained below:

-i6 Boots the system to run level 6

-g900

Sets the time for logging off to 900 seconds (i.e. 15 minutes)

Single-user mode

Because you are the only user logged on to the system, you can shut down Reliant UNIX immediately:

```
# cd /  
# shutdown -i6 -g0 -y
```

The arguments are explained below:

- i6** Boots the system to run level 6
- g0** Sets the time for logging off to 0 seconds
- y** Further confirmation is not requested

Reliant UNIX is then shut down. A check is run to determine whether or not a new kernel has been generated. If a new kernel has been generated, a new start file is created. The following actions are then performed:

- Control is transferred to the firmware,
- Boot2 is booted automatically,
- Reliant UNIX is booted automatically.

Reliant UNIX boots again to the default run level with no external intervention required.

3.12 Shutting down the system

You should proceed as follows depending on the current run level:

Multiuser mode

Before shutting down the system, check whether or not other users are still logged on. To do this, use the *who(1)* command (see ...). If you are the only user logged on, you can shut down Reliant UNIX as you would in single-user mode (see this page).



The *shutdown* command automatically informs all users who are logged on of the imminent change in the run level (see ...). If you want to give users more detailed information, you can use the *wall(1M)* command (see the [Section "Informing users who are logged on"](#)).

Now enter the following commands:

```
# cd /
# shutdown -i0 -g900
```

The arguments are explained below:

-i0 Boots the system to run level 0

-g900

Sets the time for logging off to 900 seconds (i.e. 15 minutes)

Single-user mode

Because you are the only user logged on to the system, you can shut down Reliant UNIX immediately:

```
# cd /
# shutdown -i0 -g0 -y
```

The arguments are explained below:

-i0 Boots the system to run level 0

-g0 Sets the time for logging off to 0 seconds

-y Further confirmation is not requested

The shutdown procedure ends differently depending on the computer model:

Procedure for RM200 - RM400

Many computer models turn off the power supply using the software while some models are shut down using the on/off switch or the keyswitch. The following message is issued for these models:

```
POWER OFF: INIT 00000000: PLEASE SWITCH OFF THE POWER SUPPLY!
```

A beep sounds at the same time. The *POWER* LED only extinguishes when you have switched off the computer.



You should check your operating manual to see if your model can be switched off from the software.

Procedure for the RM600

When the shutdown procedure is concluded, the firmware switches off the power supply to the computer. The *POWER* LED extinguishes.

3.13 Run levels

Reliant UNIX can operate in eight different run levels. The individual run levels are distinguished from each other by the processes that each level can run. [Table ""](#) provides an overview of the run levels.

In practise, several terms are used which are synonymous with the term "run level":

- "init status",
- "mode of operation",
- "operating mode" and
- "system status".

The term "run level" is used consistently throughout this manual.

There are several options available for changing the run level:

- using the *shutdown* command
- using the *init* command
- via the *SYSADM* user interface

With each of these three options, the *init* command is called. Information on this command can be found in the [Section "The *init*\(1M\) command"](#) and in the manual entry for *init*(1M).

3.13.1 Overview of the run levels

Leve	Description
1	
<i>s, S</i>	<p>Single-user mode</p> <p>If Reliant UNIX boots to this run level, no file systems are mounted for user-owned files and only important processes are active in the kernel.</p> <p>If Reliant UNIX is shut down to this level, all file systems remain mounted. All processes started by <i>init</i>, which should only run in multiuser mode, are terminated with <i>kill</i>. This also applies for line monitoring programs as well as all services started by them (e.g. login services). The screen from which the change in the run level was triggered is considered to be a console from now on (in contrast to a "normal" console).</p>
<i>0</i>	<p>Shutdown</p> <p>Shuts down Reliant UNIX and switches off the power supply to the computer.</p>
<i>1</i>	<p>System administrator mode</p> <p>All file systems are mounted and only important processes in the kernel are active. All files can be accessed. Only one user is logged on.</p>
<i>2</i>	<p>Multiuser mode</p> <p>All file systems are mounted. All terminal processes and daemons required for multiuser mode are started. This run level is preset when the computer is shipped.</p>
<i>3</i>	<p>Extended multiuser mode</p> <p>The RFS processes (RFS = Remote File Sharing) and daemons are started and remote resources are mounted. This run level is also referred to as the RFS level.</p>
<i>4</i>	<p>Alternative multiuser mode</p> <p>This run level is not used in the Reliant UNIX version supplied.</p>
<i>5</i>	<p>Restart□</p> <p>Shuts down Reliant UNIX and switches to the PROM monitor (RM200 - RM400) or to boot2 (RM600), regardless of the boot mode set.</p>
<i>6</i>	<p>Restart</p> <p>Shuts down Reliant UNIX and boots it again. In the process, Reliant UNIX boots to the run level that corresponds to the <i>initdefault</i> entry in the <i>/etc/inittab</i> file.</p>
<i>a, b, c</i>	<p>Pseudo levels</p> <p>Only processes the entries in the <i>/etc/inittab</i> file for which the run level <i>a, b</i> or <i>c</i> is entered. The pseudo levels are used to enable you to define certain commands without having to change the current run level. For further information in this regard, please refer to the manpage for <i>inittab(4)</i>.</p>
<i>q, Q</i>	<p>Checks the <i>/etc/inittab</i> file again.</p>

Table 19: Possible Reliant UNIX run levels

3.13.2 Directories for the run levels

A directory is allocated to each of the run levels *0*, *1*, *2* and *S*: */etc/rc0.d*, */etc/rc1.d*, etc. The directories contain files that are executed when the respective run level begins or ends. Your task is to start or stop the different system services and background processes. Most of these files are linked to files in the */sbin* directory (*/sbin/rc0*, */sbin/rc1* etc.).

The following syntax applies for the names of the files:

SNN<name> or
KNN<name>

Each name consists of three parts:

- S* Activities should be started at the beginning of the new run level. (Call with the *start* option).
- K* Activities should terminate at the beginning of the new run level. (Call with the *stop* option).
- NN* A number between 00 and 99 determines the ascending sequence in which the processes should be started (*S00*, *S01*, *S02*, etc.) or terminated (*K00*, *K01*, *K02*, etc.). The same number can be used a number of times. For example, all *S12** scripts are executed before *S13** is started.

<name>

The rest of the file name is identical to the name of the file in the */etc/init.d* directory to which the file is linked.

For example, the */etc/rc2.d/S75cron* and */etc/rc0.d/K70cron* files are linked to the */etc/init.d/cron* file, which is a shell script. When this script is called with the *start* option, it executes the */usr/bin/cron* file. However, if it is called with the *stop* option, it terminates the *cron* process using *kill*.

If you switch to run level *2*, the files in */etc/rc2.d* are executed. The *S75cron* file is executed with the *start* option:
sh S75cron start

When you switch to run level *0*, the */etc/rc0.d/K70cron* file is executed with the *stop* option:
sh K70cron stop

Instead of these files, you can also execute the */etc/init.d/cron* script directly with whichever option you require, either *start* or *stop*.

Like all shell scripts, these files are readable and can also be modified as required. However, modified shell scripts should be renamed, as the scripts supplied in future Reliant UNIX versions can change.



Update installation:

With this type of installation, all *rc2* scripts and *cron* tables which are available by default are installed. If you have renamed *rc2* scripts on your system (e.g. for using OBSERVE), to prevent certain system components starting automatically, your system will also contain the original scripts after the update installation has been performed. The same is true of *cron* tables. As a result, you have to reincorporate the necessary changes.

When you generate your own scripts, you should

- store the script in the */etc/init.d* directory and
- link the script to files in appropriate run level directories (see *ln(1)*). Here, you must observe the name syntax described above.



If you incorporate errors into the scripts, it may no longer be possible to start up or shut down the system.

3.13.3 The /etc/inittab file

Each time the run level is changed, the *init* command is invoked. The task of this command is to generate those processes that are required for the new run level. It thus relies on data stored in the */etc/inittab* file. The file contains a table, which consists of several lines with four fields each. The fields are separated from one another by a colon. The second field defines one or more run levels in which the entry is to be processed.

If Reliant UNIX is rebooted, it boots to the run level that is preset by the *initdefault* entry in the */etc/inittab* file. This entry may appear as follows, for example:

```
is:2:initdefault:
```

Generally, a 2 is entered in the second field. This number causes Reliant UNIX to boot to multiuser mode immediately. If the *initdefault* entry is missing, you will be asked to specify the run level to which Reliant UNIX should be booted:

```
ENTER[RUN]LEVEL[0-6,s or S]:_
```

If you would like Reliant UNIX to boot up automatically to a run level other than the level defined in the */etc/inittab* file, there are two options available:

- You can modify the default setting. To do this, enter the desired run level in the second field in the *initdefault* line of the */etc/inittab* file.
- You can switch to boot2 or the SASH and boot Reliant UNIX manually from there. In this case, use the *initarg* argument.

The following entry must be made for the RM200 - RM400:

```
sash:[boot-f]dkncr(,,)unix[initarg=-s
```

The following entry must be made for the RM600:

```
boot2:[boot-f]/unix[initarg=-s
```

3.13.4 Processes required for multiuser mode

Several levels are run through when Reliant UNIX boots:

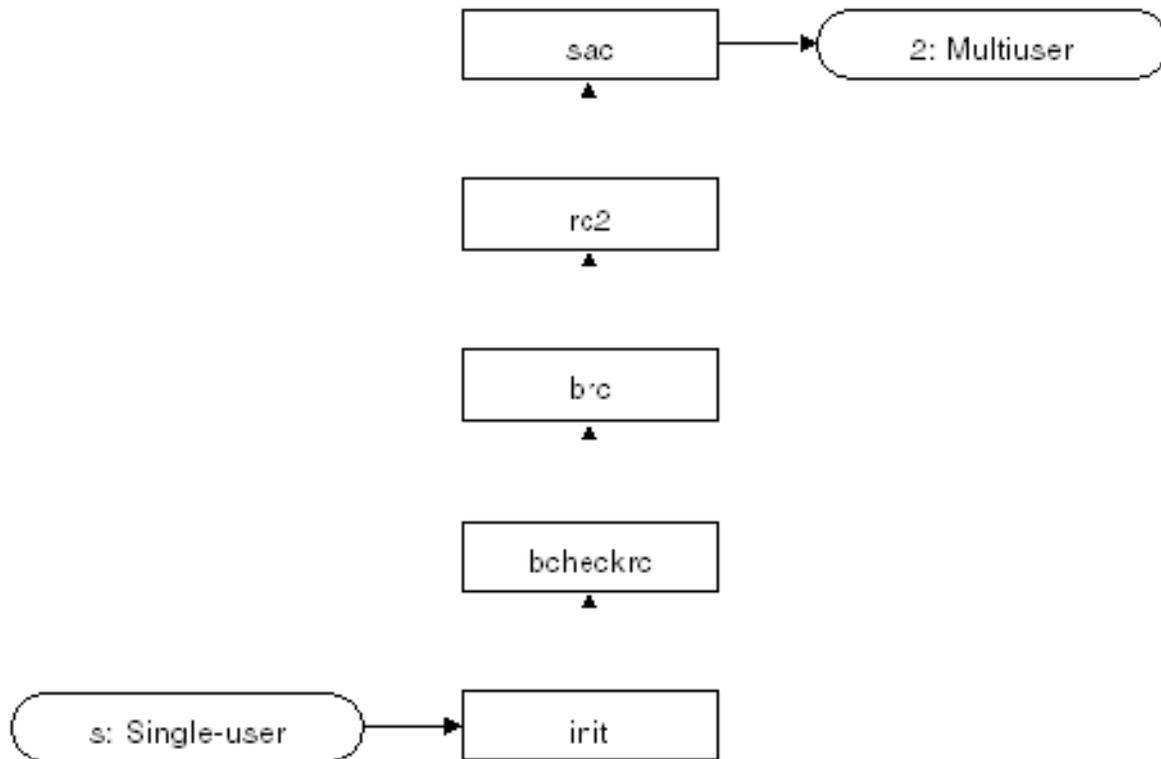


Figure 10: Overview of system initialization

Starting system initialization

As soon as Reliant UNIX has been loaded into main memory, the *init* process is generated. It searches the */etc/inittab* file for entries of the type *sysinit*.

Example:

```

ap::sysinit:/sbin/autopush -f /etc/ap/cons.ap
bchk::sysinit:/sbin/bcheckrc </dev/console >/dev/console &2 &1
lar::sysinit:/sbin/scsilar_rc </dev/console >/dev/console &2 &1
sofs::sysinit:/sbin/soconf </dev/console >/dev/console &2 &1
brc:1234:bootwait:/sbin/brc >/dev/console &2 &1 </dev/console

```

The entries are processed in sequence:

- The first entry sets up the connection between the system and the console.
- The other entries define */dev/console* as the default device for input/output.
- *bcheckrc* checks the file systems and mounts them.
- *scsilar_rc* deactivates defective SCSI devices (LAR = Logout Auto Recovery).
- *brc* performs other initializations.

Ascertaining the required run level

The system must now be booted to a specific run level. To do this, *init* searches the */etc/inittab* file for the *initdefault* entry. Example:

```
is:2:initdefault:
```

If an entry of this type is found, this means that the run level defined in the second field of the line is preset. If no *initdefault* entry is found, you are prompted to select a run level.



The default setting can be canceled by booting Reliant UNIX manually from boot2 and specifying the *initarg* argument (see ...).

Preparing to change the run level

The *init* process now searches the */etc/inittab* file for the processes that are assigned to the required run level. The figure below shows typical entries for processes in run level 2.

```
r2:23:wait:/sbin/rc2>/dev/console<&1</dev/console
li:23:wait:/usr/bin/lm/dev/systty/dev/syscon>/dev/null<&1
sc:234:respawn:/usr/lib/saf/sac-t<300
co:1234:respawn:/usr/lib/saf/ttymon-g-p"ConsoleLogin:"<-d/dev/ttylc-l-con
000:1234:respawn:/usr/lib/saf/ttymon-g-p"SystemnameLogin:"<-d/dev/term/000
```

The entries are processed in sequence:

- The first entry calls the */sbin/rc2* script, which in turn calls the */etc/rc2.d/S** scripts (see the [Section "Directories for the run levels"](#)). The latter scripts perform the following tasks:
 - Compiles and mounts the file system
 - Starts the background process *cron*
 - Displays the current hardware configuration of the system
 - Establishes the Ethernet connection (start *inetd*)
 - Activates the LP print service (if installed)

- Starting SAF components:
 - Start the service access controller (*sac*) for the interfaces
 - Start screen monitoring (*ttymon*) for the console

The full multiuser environment is then configured. The system is in run level 2 and waits for the user to log on (see ...).

3.13.5 Informing users who are logged on

Many administration tasks directly affect other users who are logged on to the system. If, for example, you switch from multiuser mode to single-user mode, other users are denied access to the system. Before you perform this kind of administration work, you should therefore carry out two tasks in general:

- Give all users who are logged on prior notice of what you are going to do.
- Allow these users sufficient time to log off.



If you use the *shutdown(1M)* command to change the run level (see ...), you can transfer both tasks to this command. It is therefore not necessary to send two separate messages to the users who are logged on.

Proceed as follows to inform the users who are logged on to the system:

1. Use the *who(1)* command to check whether or not other users are active in the system.

```
# who -uH
NAME LINE TIME IDLE PID COMMENTS
hjk tty010 Jul 7 07:51 12582
hjk tty011 Jul 7 08:10 4:49 12583
hofu tty020 Jul 7 08:36 8298
jib tty140 Jul 7 09:17 1:08 18059
krenz tty150 Jul 7 08:12 14034
hoffma tty190 Jul 7 09:26 28684
hgf tty220 Jul 7 09:04 0:14 20995
berni tty210 Jul 7 09:06 0:12 18477
```



who(1) does not indicate whether users are accessing the system from other computers via *nfs* or *pcnfs*.

Inform the users who are logged on that the system will be shut down shortly. To do this, use the *wall(1M)* command.

Enter an appropriate message. Conclude the message by pressing [Ctrl]+[D] at the beginning of a new line:

```
# wall
The system will be shut down in 5 minutes.
Please log off now.Ú
[Ctrl]+[D]
```

The message displayed below then appears momentarily on the screens of all users who are logged on (including your own):

```
Broadcast Message from root (console) on orion Mo 12.Sep 08:30:27...
The system will be shut down in 5 minutes.
Please log off now.
```



The first line of the message only appears in German if as user *root* you have set the *\$LANG* environment variable to German (see the [Section "Important environment variables"](#)).

3.13.6 The `shutdown(1M)` command

We recommend that you use the `shutdown(1M)` command to change the run level. This procedure has two advantages:

- The users who are logged on to the system are informed automatically
- You can use a special option to define the time period after which the system should change to the new run level

The `shutdown` command can always be used if you want to change to run levels `0`, `1`, `5`, `6` or `s`. However, if you want to change to run levels `2`, `3` or `4`, you must use another command, for example `init(1M)`.

The `shutdown(1M)` command has the following syntax:

```
shutdown [ -i<init_state> ] [ -g<grace_period> ] [ -y ]
```

The various options are explained below:

- i** Boots the system to the specified run level
(default: run level `s`)
- g** Waits the specified number of seconds
(default: 60 seconds)
- y** Further confirmation is not requested
(default: request confirmation)

Procedure

1. Log on under the login name `root`.
2. Switch to the `/` directory.
3. Start the `shutdown` command with the required options.

The following message is displayed on your screen:

```
Shutdown started.  Mo 12.Sep. 08:57:40 MDT
```

If other users are logged on to the system, they are requested to log off. In the following example, the `-g900` option is used. A waiting period of 900 seconds, i.e. 15 minutes, should be adequate to allow all users to log off:

```
Broadcast Message from root (console) on orion Mo 12.Sep 08:57:41...
The system will be shut down in 900 seconds.
Please log off now.
```

When the waiting period has expired, a second warning is output:

```
Broadcast Message from root (console) on orion Mo 12.Sep 09:12:41...
THE SYSTEM IS BEING SHUT DOWN NOW!!
Log off now or risk your files being damaged
```

The system then waits for another 15 seconds before executing the *shutdown* command again. What happens next depends on whether or not you have specified the *-y* option.

-y option not specified: confirmation requested

You must confirm again whether or not the system should actually be shut down:

```
Do you want to continue? (y or n): _
```

If you answer the question with *n*, the *shutdown* command is aborted. The users who are logged on then receive the message:

```
Broadcast Message from root (console) on orion Mo 12.Sep 09:12:41...
False Alarm: The system will not be brought down.
```

However, if you answer the question with *y*, execution of the *shutdown* command is continued. The next steps are described below.

-y option specified: confirmation not requested

The following message is output (example for *-is*):

```
Changing to init state s - please wait
```

The system executes the following commands when shutting down, depending on the run level to be achieved:

init state	Actions
<i>s, S</i>	The shell script <i>/sbin/rc0</i> is executed. The command <i>/sbin/init</i> is executed.
<i>1</i>	The command <i>/sbin/init</i> is executed.
<i>0</i>	The command <i>/sbin/init</i> is executed. The system shuts down and switches off the computer.
<i>5, 6</i>	The command <i>/sbin/init</i> is executed. The system shuts down and transfers control to the firmware. The system then boots to the required run level.

Table 20: Actions carried out during shutdown

Messages

Appropriate messages are displayed on the console, which are determined by both the old and the new run level. Examples see

As long as you stay within run levels *1* to *4*, you do not need to enter a new password. However, you must press the **U** key as soon as the new run level has been reached.

3.13.7 The *init(1M)* command

If you want to change the run level, you can use the *init(1M)* command to do so. If Reliant UNIX is currently in multiuser mode (run level *2*, *3* or *4*), you should perform the tasks listed in [Section "Informing users who are logged on"](#) before changing the run level:

1. Give all users who are logged on prior notice of what you are going to do (see the [Section "Informing users who are logged on"](#)).
2. Allow these users sufficient time to log off.



If you want to change to run levels *0*, *1*, *5*, *6* or *s*, you should use the *shutdown(1M)* command (see ...). You can then transfer both tasks to this command. It is therefore not necessary to send two separate messages to the users who are logged on.

The *init(1M)* command has the following syntax:

```
init <run_level>
```

Procedure

1. Log on under the login name *root*.
2. Switch to the */* directory.
3. Start the *init* command.

Messages

Messages are displayed on your screen which are dependent on both the old and the new run level.

As long as you stay within run levels *1* to *4*, you do not need to enter a new password.



When switching between run levels *1* to *4* please note the following:

When the new run level has been reached, you do **not** get a new prompt. The prompt was already output **before** the messages of the *init* command (*# INIT ...*). Therefore, you can start another Reliant UNIX command immediately. However, you can also press the **U** key to get a prompt again.



When switching to single-user mode please note the following:

Input / output is only possible via the console. If you changed the run level from a different screen, it can no longer respond.

Example: Switching from run level 2 to run level 1

```
# INIT: New run level: 1
Changing to state 1.
 
done.
Change to state 1 has been completed.
U
# _
```

Example: Switching from run level 1 to run level 2

```
# INIT: New run level: 2
The system is coming up. Please wait.
 
done.
The system is ready.
U
# _
```

Example: Switching to run level 6

```
# INIT: New run level: 6
The system is coming down. Please wait.
 
killall: done.
setting up new kernel as next default kernel: done.
 
The system is about to go down.
The system is being rebooted.
```

3.13.8 Run level s (single-user mode)

Some administration tasks, such as backing up file systems, can only be executed in single-user mode.

Enter the following commands:

```
# cd /
# shutdown -is -g900
```

When you shut down to single-user mode, the */etc/rcS.d/K** scripts are executed. These scripts perform the following tasks:

- All open files are closed
- All user processes are stopped
- All background processes and services are stopped
- All system buffers are written to the disk

If you have created your own applications, which should not run in single-user mode either, you can terminate them by incorporating an additional script into the */etc/rcS.d* directory. Further information can be found under *rcS(1M)*.

As soon as you enter single-user mode, you are given two options:

```
Type Ctrl-d to proceed with normal startup,
(or give root password for system maintenance): _
```

1. If you press [Ctrl]+[D], you are asked to specify the run level to which you want to boot Reliant UNIX:

```
ENTER RUN LEVEL (0-6, s or S): _
```

2. If you enter the root password, you get the single-user mode prompt:

```
Entering System Maintenance Mode
# _
```

3.13.9 Run level 2 (multiuser mode)

Reliant UNIX normally operates at this run level. In this case, several users can be logged on and use the resources at the same time (according to the license purchased). For this reason, run level 2 is also referred to as multiuser mode.

Enter the following command:

```
# init 2
```

This command calls the */sbin/rc2* script, which itself executes the scripts *S** in the */etc/rc2.d* directory. The service access facility (SAF), which controls access to the system via various interfaces and communication units, is also called. Detailed information in this regard is contained in the chapter "Service Access" in the

"System Administrator's Guide".

3.13.10 Run level 6 (restart)

Reliant UNIX is restarted at this run level. In the process, all active processes are terminated with *kill*. The computer reaches a level which closely resembles that at which the system is switched on. You must restart the system, for example, if you have installed a software package and want to reinitialize the system to ensure correct operation.

Enter the following commands:

```
# cd /
# shutdown -i6 -g900
```

The *-i6* option applies for the entries in the */etc/inittab* file.

Example:

```
r6:6:wait:/sbin/rc6 reboot /dev/console 2>&1</dev/console
rb:6:wait:/sbin/uadmin 2 1 /dev/console 2>&1</dev/console
```

The command calls the */sbin/rc6* script. This script ensures that Reliant UNIX is shut down and then rebooted.

3.13.11 Run level 0 (shut down and switch off system)

If you want to switch off the system, use the following commands:

```
# cd /
# shutdown -i0 -g900
```

The *-i0* option applies for the entries in the */etc/inittab* file.

Example:

```
r0:0:wait:/sbin/rc0 off /dev/console 2>&1</dev/console
sd:0:wait:/sbin/uadmin 2 0 /dev/console 2>&1</dev/console
```

The command calls the */sbin/rc0* script. This script unmounts the file systems and terminates all user processes, background processes and other services. The */tmp* directory is also emptied.

3.13.12 Shutting down the system in emergencies

Reliant UNIX can become inoperable for other users as a result of software or hardware errors. An imminent crash can often be detected by the fact that the system is suddenly very slow to respond or that it reacts abnormally to input. Proceed as follows in such a case:

Actions in Reliant UNIX

- ▶ Try to back up the data in the main memory, including the current information in the superblock and the inode entries, on to disk:

```
# sync
```

The *sync(1M)* command can be called by any user on any terminal.

- ▶ You should attempt to shut down the system properly.
 - Log on as user *root*.
 - Switch the system to single-user mode. To do this, follow the instructions given on [Section "Run level s \(single-user mode\)"](#). If no time is available to send warning messages, enter the following command:

```
# init s
```

Actions in the kernel debugger



For RM200 - RM400:

As of V5.43C, the kernel debugger is no longer activated by default. It can be activated temporarily with *setikdb on*. If the debugger is to be activated permanently, the *SETIKDB=on* parameter must be set in the */etc/default/dumpsave* file. When the debugger is active, the system branches to the kernel debugger

in the event of a serious error. If you set the parameter *SETIKDB=off* in the */etc/default/dumpsave* file, the debugger is deactivated at a set time after booting. This time is specified by the *SETIKDBTIME=number_of_minutes* parameter. If the number of minutes is not specified, the debugger remains active. If the debugger is not active, the system generates a memory dump and reboots automatically in the event of a serious error.

For all systems:

If the system is still blocked after a few minutes, call up the kernel debugger *ikdb(8)* (IKDB = Internal Kernel Debugger) from the console:

- ▶ On graphics screens, first switch to ANSI mode. Then press [Ctrl] + [SysRq] + [H].
- ▶ Start the IKDB by entering [Ctrl]+[@], [[Shift]]+([I],[K],[D],[B]).

A detailed description of all commands available in the debugger is contained in the manual entry for *ikdb(8)*. You can exit the debugger using the *ex* or *q* command.

- ▶ Attempt to "kill" all running processes:

```
(lcpu0,pcpu0)ikdb>ki-150
(lcpu0,pcpu0)ikdb>ex
```

- ▶ If the debugger is not terminated or if you do not get into single-user mode within approximately 5 minutes, call the debugger again:

```
[Ctrl]+[@], [Shift]+([I],[K],[D],[B])
```

- ▶ Force an error situation:

```
(lcpu0,pcpu0)ikdb>pa
(lcpu0,pcpu0)ikdb>ex
(lcpu0,pcpu0)ikdb>ex
```



Turn to [Section "Next steps on the RM600"](#) for information on the RM600.

3.13.12.1 Next steps on the RM200 - RM400

The following warning message is issued when the debugger terminates:

Do not switch off, wait for boot procedure

Do not switch off your system, not even if it takes a long time before the subsequent boot process starts up:

Pressing the RESET key

- ▶ If this measure also fails, press the RESET key for more than 2 seconds. The system is then reset to the initial status.

3.13.12.2 Next steps on the RM600

- ▶ Enter *ex* again to exit the IKDB:

```
(lcpu0,pcpu0)ikdb>ex
```

The system attempts to restart on the basis of the bootflags and bootstrings set.

Actions in the board debug monitor

- ▶ If this measure also fails, switch to the board debug monitor:

```
RM600-xxx: [Ctrl]+[A]
```

```
RM600E: [Ctrl]+[@], [[Shift]]+([B],[D],[M])
```

- ▶ Then initiate a warm start:

```
BDM(SPB8t)>wreset
```

- ▶ Save a selective dump to a magnetic tape cassette:

```
boot2:dump-fst(0,32,0)
```

- ▶ Initiate a restart:

```
BDM(SPB8t)>creset
```

3.13.13 System failure

Reliant UNIX constantly checks its internal data structures and registers at specific positions in the kernel in order to identify logical inconsistencies in the hardware or software. All values are checked that

- should be located within a specific domain,
- should be identical to an expected value and
- must not be equal to an error value

If Reliant UNIX detects an inconsistency, it outputs a message on the console. The message always contains the word *PANIC*.

The system automatically starts the debugger. A dump of the current core image is written to the swap space where it can later be retrieved using *dumpsave(1M)*. The system then executes a restart.

During the restart procedure, the */etc/brc* script executes the *dumpsave(1M)* command. This command reads the swap area and stores the core image in */var/crash/dump.X*, where *X* is a digit which is incremented by 1 after each *dumpsave* command. *dumpsave* also copies the *stand/unix* file (that contains the executable kernel) to */var/crash/unix.X*. In this case, *X* is also incremented by 1 after each *dumpsave* command.



If a kernel other than *stand/unix* is executed at the time of the error, *dumpsave* cannot recognize this and stores the wrong executable kernel. If this occurs, you must copy the kernel which is active at the time of the error to */var/crash/unix.X* in order to enable you to analyze the dump with */etc/crash* later on. The *unix.X* file must be identical to the kernel that was executed when the error occurred.



If the */var/crash* area is too small, it can be redefined in the */etc/default/dumpsave* file.

If there is insufficient space in the `/var/crash` directory for the `dump.X` and `unix.X` files, `dumpsave` fails. In this case, an incomplete (and thus unusable) dump is generated. This is not fatal provided that you react quickly, before the system begins with the swap actions:

- Switch the system to single-user mode immediately using `init 1` to keep the number of processes running in the system as low as possible.
- Remove the old `dump` and `unix` files as well as other files from the `/var/crash` directory if necessary (or from other directories located on the same area of the disk) in order to create more space.
- Then repeat the `dumpsave` command from the console.

3.14 Generating a new system kernel

When Reliant UNIX is booted, an executable kernel is loaded into main memory. The kernel is an executable file located on one of the available memory units. Its default file name is `/stand/unix` or `/unix` (hardlink).

You must generate a new system kernel in the following cases:

- If you have installed or uninstalled individual software packages and have received the following message for example:

```
Do you wish to rebuild the kernel after <packagename> installation?
(default: n) y,n,?
```

In this case, installation or deinstallation does not take effect until you generate a new system kernel and restart Reliant UNIX. A new system kernel is generated automatically during a new installation or update installation.

- You changed the kernel parameters using the `idtune(1M)` command. Once again, your modifications do not come into effect until you generate a new system kernel and restart Reliant UNIX.

The `/etc/conf/bin/idbuild` command is provided for generating a new system kernel.

- The `-M` option generates a multiprocessor kernel.



Do not use this option on RM400-xxx.

- The `-S` option generates a mono-processor kernel.
- If **no** option is specified, Reliant UNIX itself checks how many processors are available.

The generation process will take some time, during which you will receive the following messages, for example:

```
■■■■■■■■The UNIX Operating System will now be rebuilt.
■■■■■■■■This will take some time. Please wait.
idaddsyms: 12682 symbols (202912 bytes), 159006 bytes of strings
■■■■■■■■The UNIX Kernel has been rebuilt.
```

The `idbuild` command generates a file called `MP_unix` in the `/etc/conf/cf.d` directory. The new file is linked by reference to the `unix` file in the same directory. Finally, a lock file called `.new_unix` is created in the `/etc` directory.

3.14.1 Booting the new system kernel

You must restart Reliant UNIX so that the kernel takes effect:

- ▶ Enter the following commands:

```
# cd /
# shutdown -i6 -g900
```

Detailed information on restarting the system can be found in [Section "Restarting the system"](#).

The next time Reliant UNIX switches to run level `0`, `5` or `6`, it checks whether or not a `/etc.new_unix` lock file exists. If so, it carries out the following steps:

- The `/stand/unix.old` file is deleted.

- The `/stand/unix` file is renamed `/stand/unix.old`.
- The `/stand/unix.default` file is deleted.
- The `/etc/conf/cf.d/unix` file is copied to `/stand/unix.default`.
- The `/stand/unix.default` file is linked to `/stand/unix`.
- The `/stand/unix` file is linked to `/unix`.
- The `/stand/unix.old` file is linked to `/unix.old`.
- The `/etc/.new_unix` file is deleted.

This procedure ensures that the old kernel is reactivated if problems arise in the newly installed version. If one of the steps listed above cannot be executed, you can boot the operating system with the old system kernel.

The connections between `/unix` and `/stand/unix` or between `/unix.old` and `/stand/unix.old` are established for the following reason:

- Boot2 boots the `/unix` file by default.
- Reliant UNIX applications need the system kernel in the `/stand` directory.

The following message is displayed when the new system kernel has been activated:

```
setting up new kernel as next default kernel: done.
```

3.14.2 Booting the old system kernel

There are few situations where the new kernel cannot be booted correctly. Such a situation may occur, for example, if you modified a tunable parameter or a combination of parameters in such a way that the new kernel is too big or does not initialize correctly. In this type of situation, you must first reboot the old kernel to set up an executable operating system. To do this, proceed as follows:

- ▶ Switch to the board debug monitor.
- ▶ Boot boot2.

```
BDM(SPB 8t)>b 201
```

- ▶ Boot the old system kernel.

This kernel was renamed `/unix.old` when Reliant UNIX was last booted.

```
boot2: boot -f /unix.old
```

- ▶ Log on as user `root`.
- ▶ Save the executable kernel just in case the next version does not function either.

```
# mv /unix.old /unix.sav
```

- ▶ Cancel the last configuration.

For example, uninstall the packages that you installed beforehand or change the tunable parameters in the `/etc/conf/cf.d/stune` file back to the original values (using the `idtune` command).

- ▶ Execute the `idbuild` command again.
- ▶ Then reboot Reliant UNIX.

If this kernel does not function for any reason, boot from `/unix.sav` instead of from `/unix.old` and try again.



Each system kernel occupies approximately 9 Mbytes in the `/stand` directory.

3.15 Displaying system information

3.15.1 Hardware components

Information about the hardware components is displayed as follows:

Shell command:

```
autoconf -l
```

Menu tree for `SYSADM`:

Menu items: `machine – configuration – summary`

You will find some sample output on the next pages:

- Several HD02, HD04 and HD14 hard disks
- There is no RAM disk

For further information, please refer to the manpage for *autoconf(8)*.

3.15.2 System disk partitioning

You can use the *dkpart* command to display how the hard disks are partitioned. This command requires root authorization.

```
# dkpart -l /dev/ios0/rsdisk011s0
```

A HD07 hard disk can have the following partitioning, for example:

```
HD07 „New Part“ Geometry 34712-3:16:64 (cyls:heads:sectors)
Sector size = 512 bytes 17356.0 Total Mbytes (1MB = 1024 * 1024 bytes)
Partition Start Rule End Rule First Cyl Last Cyl Cyl Count Size (MB)
0 >p10 256mb 8 519 512 256.0
1 >p0 2560mb 520 5639 5120 2560.0
2 >p1 1024mb 5640 7687 2048 1024.0
3 >p2 512mb 7688 8711 1024 512.0
4 >p3 512mb 8712 9735 1024 512.0
5 >p4 <p6 9736 26515 16780 8390.0
6 26516 <p15 26516 34707 8192 4096.0
7 0 0 *** Unused ***
8 0 0 *** Unused ***
9 0 0 *** Unused ***
10 0 4mb 0 7 8 4.0
11 0 0 *** Unused ***
12 0 0 *** Unused ***
13 0 0 *** Unused ***
14 0 0 *** Unused ***
15 34708 1 34708 34708 1 0.5
# Partition layout (not to scale):
|10|-0|---1---|2---|3-|4-|-----5-----|-----6-----|15|
```

Figure 12: Partitioning the system disk using the HD07 as an example



There are some limitations concerning the partitioning:

- Partition 5 is partly available for user data. During a new installation, this partition is automatically mounted at the */home* mount point. Its size is the remaining rest after all other partitions have been created.
- Partition 6 will always be placed directly before partition 15. It mostly will be used as dump device.
- Partition 10 cannot be used for user data. It contains boot2 as well as the header information for the disk.
- The last cylinders in each disk are not available for user data either. They contain the trailer information of the disk.

Further information can be found in the manpage for *dkpart(8)*.

3.15.3 Establishing which users are logged on

Before you carry out any administration tasks that could affect the work of users who are logged on, you should first check who is currently logged on.

Command:

who -uH

SYSADM:

Menu items: machine – who's on

The following screen dump is an example of this:

```
# who -uH
NAME LINE TIME IDLE PID COMMENTS
hjk tty010 Jul 7 07:51 . 12582
hjk tty011 Jul 7 08:10 4:49 12583
hofu tty020 Jul 7 08:36 . 8298
jjb tty140 Jul 7 09:17 1:08 18059
krenz tty150 Jul 7 08:12 . 14034
hoffma tty190 Jul 7 09:26 . 28684
hgf tty220 Jul 7 09:04 0:14 20995
berni tty210 Jul 7 09:06 0:12 18477
```

Further information can be found in the manpage for *who(1)*.

3.15.4 Name, version, revision version of Reliant UNIX

This information is displayed as follows:

Command:

```
uname -rsv
```

SYSADM:

```
Menu items: machine – configuration – system
```

Example:

```
# uname -rsv
ReliantUNIX-Y 5.45 B0000
```

The version number of the operating system and the revision version, here 5.45 B0000, are important. You should always use the current revision version of the software. Further information is contained in the manpage for *uname(1)*.

3.15.5 Current run level

Information on the current run level is displayed using the *who -rH* command. Example:

```
# who -rH
NAME LINE TIME IDLE PID COMMENTS
run-level 2 Aug 15 15:23 2 1 S
```

In this case, run level 2 is set (multiuser mode). This run level has been operating since August 15th at 15:23. The PID (Process Identifier) specifies that the system already operated at this run level once (since the last generation of the kernel). The comment refers to the run level at which the system was operating before (S = single-user mode). Further information can be found in the manpage for *who(1)* and in the diagnostics and troubleshooting manual.

3.15.6 Size of the main memory

A distinction must be made between various specifications: the physical memory, the real memory and the available memory.

Physical memory

This information is displayed using the *uname -M* command. Example:

```
# uname -M
8/1024
```

In this case, the computer has 8 processors and a physical memory of 1024 Mbytes.

Real memory

The size of the real main memory is calculated as follows:

1024 Mbytes	Size of the physical main memory
– 0 Mbytes	Size of the RAM disk
<hr/>	
1024 Mbytes	Size of the real main memory

You can also find out the size of the real main memory from the messages that are displayed on the console while the system is booting. These messages are logged to the */var/adm/log/osm* file. Analyze the lines in */var/adm/log/osm* as follows:

```
# egrep mem /var/adm/log/osm
```

Sample output:

```
Total real memory = 1073741824 B, (1024 MB)
Available memory = 996556800 B, (950 MB)
```

The two lines are explained below:

Total real memory

Available memory, i. e. physical memory minus RAM disk

Available memory

Free memory, i. e. available memory not occupied by the system kernel when the system starts up.

In the above example, the real memory accounts for 1,073,741,824 bytes = 1024 Mbytes and the available memory accounts for 996,556,800 bytes = 950 Mbytes.

3.16 Overview of important commands

Restarting Reliant UNIX:

Command:	Multiuser mode: <input type="checkbox"/>	shutdown -i6 -g900 <input type="checkbox"/>
	Single-user mode:	shutdown -i6 -g0 -y
SYSADM:	Menu items: <input type="checkbox"/>	<i>Restart machine</i> <input type="checkbox"/>
	Function keys:	<i>CONTINUE</i>

Reliant UNIX shuts down, is restarted and boots to the default run level.

Shutting down Reliant UNIX:

Command:	Multiuser mode: <input type="checkbox"/>	shutdown -i6 -g900 <input type="checkbox"/>
	Single-user mode:	shutdown -i6 -g0 -y
SYSADM:	Menu items: <input type="checkbox"/>	<i>Restart machine</i> <input type="checkbox"/>
	Function keys:	<i>CONTINUE</i>

The *-y* and *-g0* options are not critical in single-user mode as you are the only user who is logged on.

Ascertaining the hardware configuration:

Command:		autoconf -l
SYSADM:	Menu items:	<i>machine – configuration – summary</i>

Ascertaining the name and version number of the operating system:

Command:		uname -rsv
SYSADM:	Menu items:	<i>machine – configuration – system</i>

Ascertaining users who are logged on:

Command:		who -uH
SYSADM:	Menu items:	<i>machine – whos on</i>

4 Processing software with SYSADM

This chapter describes how to install additional software which does not belong to the Reliant UNIX operating system. It also describes how to display an overview of the installed software and how to deinstall it. You can use the *SYSADM* user interface for this. If you have never worked with *SYSADM*, you will find a brief introduction in [Section "The SYSADM user interface"](#) of this manual.

4.1 Selecting the right menu item

SYSADM offers two menu items for installing, listing and deinstalling software: *software* and *software_prod*:

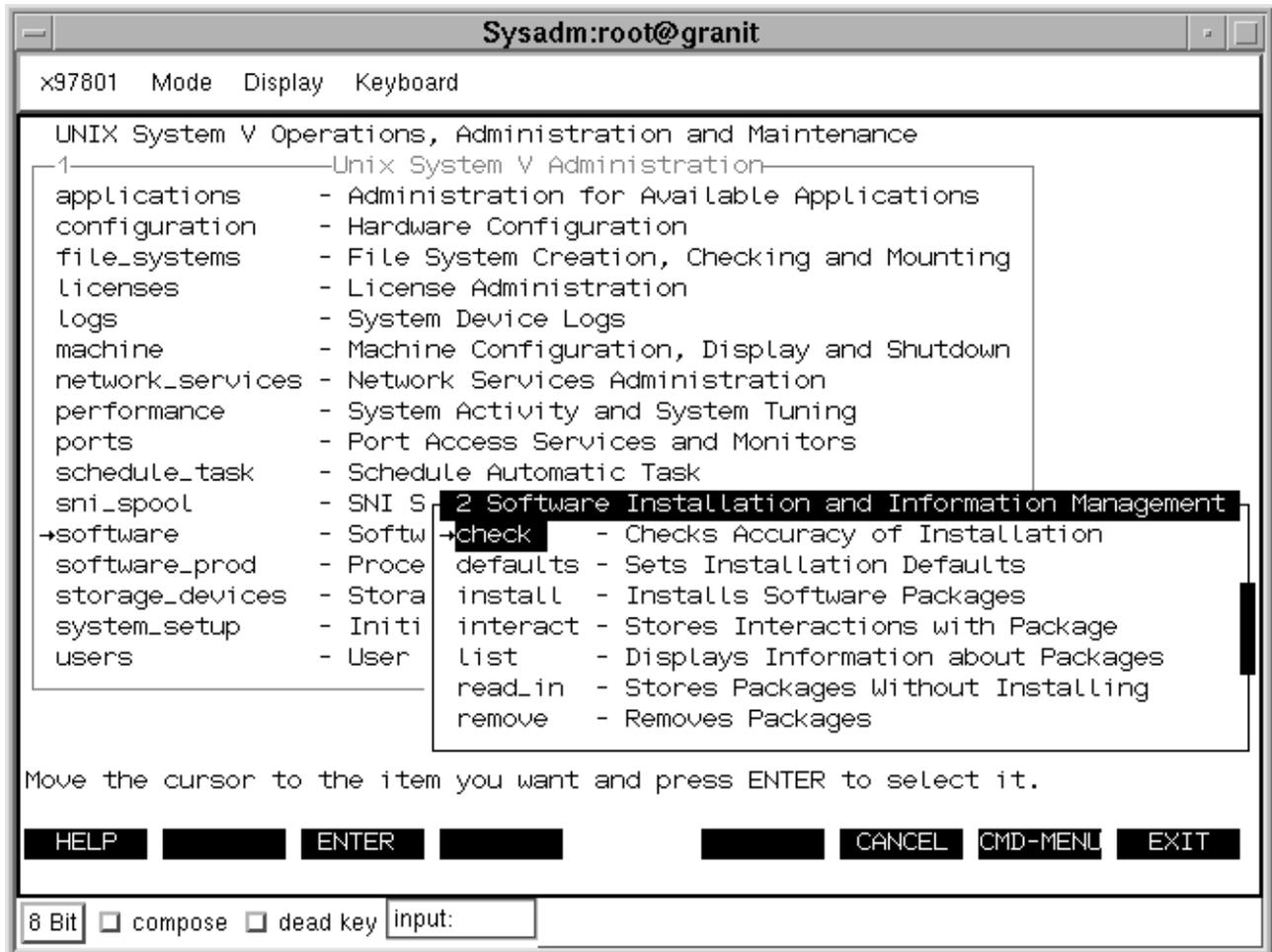


Figure 13: SYSADM menu – software menu item

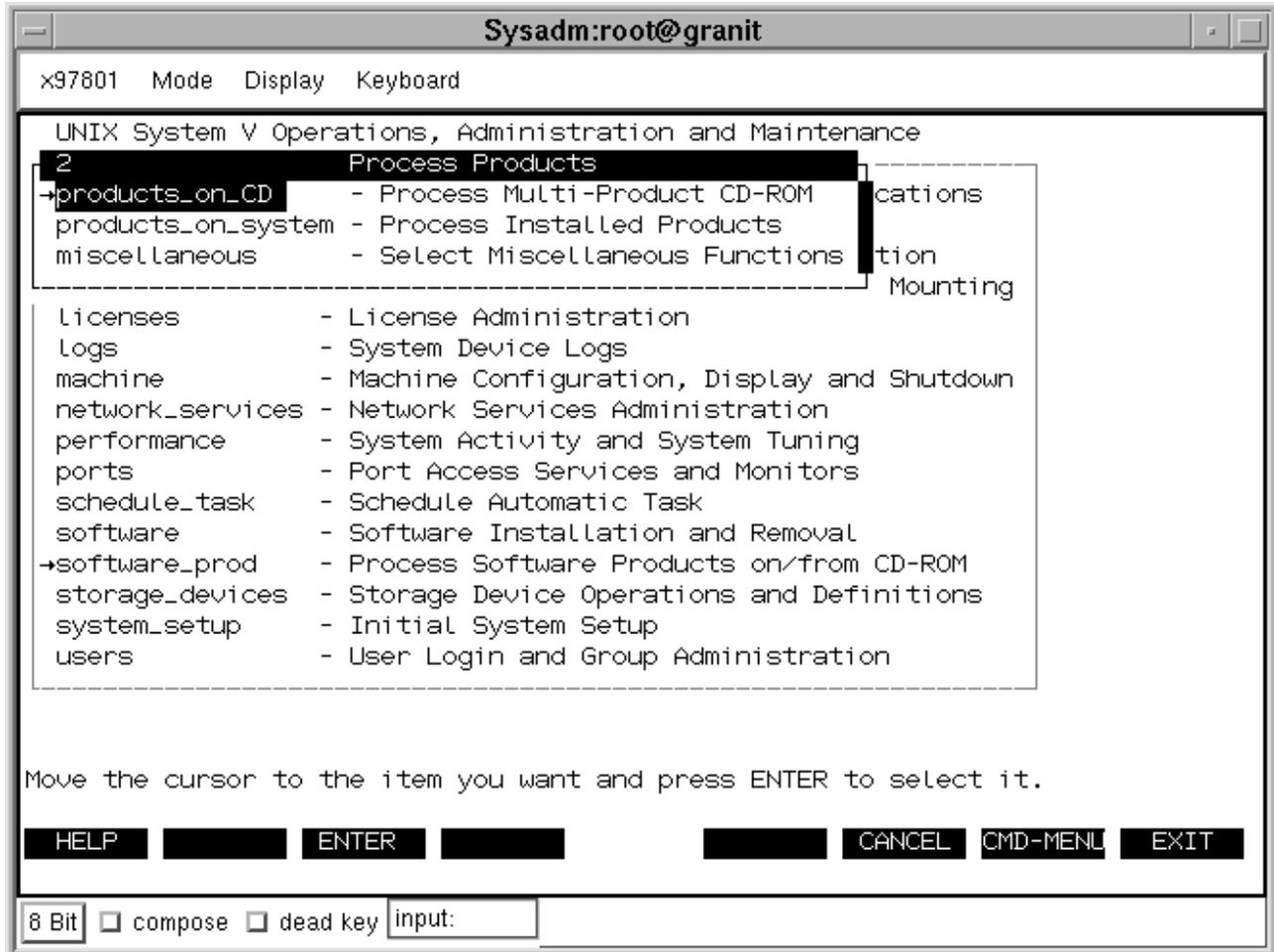


Figure 14: SYSADM menu – software_prod menu item

The menu item you choose depends on the type of software to be processed.

Two types of software can be distinguished:

1. System software

The software belonging to the Reliant UNIX operating system.

System software is system-specific, i.e. it will only run on specific RM systems. System software is traditionally structured in packages. There are currently around 100 packages.

Installation:

You need a system CD-ROM from Fujitsu Siemens, which indicates the model of your system. You can then perform either a new or update installation (see the corresponding section in this manual). You can install single system software packages with *pkgadd(1M)*.

Deinstallation:

You can deinstall single software packages with either the *SYSADM software* or *software_prod* menu item (pseudo product "OTHERS": all packages not contained in a product).

2. Application software

The software belonging to applications. It is irrelevant whether the applications come from Fujitsu Siemens or other companies. This software is also known as **add-on products**.

Add-on products are only installed when the Reliant UNIX operating system is up and running. They will generally run on all our RM systems. You will find a few add-on products on the system CD-ROM (e.g. drivers for various controllers). Additional add-on products can, for example, be found on the "CD-SYS-MI"

CD-ROM.

Add-on products are also structured as packages. The utility which is started via the *software_prod* menu item, uses a product structure to improve legibility. This groups all packages belonging to a common add-on product together under a product name. Packages not belonging to an add-on product are assigned to the "OTHERS" pseudo product (e.g. the system software packages). The utility stores the information on the installed products and their packages in a separate product database.

Installation:

You can install add-on products you receive on CD-ROM using the *SYSADM software_prod* menu item and other add-on products via the *software* menu item.

Deinstallation:

You should use the *SYSADM software_prod* menu item to deinstall add-on packages. *SYSADM* then automatically deinstalls all software packages belonging to the add-on product and also updates the product database.

It is, however, also possible to use the *software* menu item instead. In this case, you must ensure that all software packages belonging to the add-on product are deinstalled. The product database is, however, not updated and the next time you use the *software_prod* menu item you will receive the message: *n* package(s) inconsistent in data base, where *n* stands for the actual number of inconsistent packages.



If you use a graphics terminal as a console, you can also install and deinstall via the SINIX/windows *Software Manager*.

The following table provides you with an overview of the tasks that can be performed with each of the two *SYSADM* menu items:

Task	Menu item <input type="checkbox"/> <i>software_prod</i>	Menu item <input type="checkbox"/> <i>software</i>
Installation
Install add-on products from the system CD-ROM (e.g. drivers for controllers or a UPS)	yes	
Install add-on products from a product topic area CD-ROM	yes	
Install software from tape cartridge or diskette		yes
Install key diskette (...)	yes	yes
Re-install separate system software packages	use the <i>pkgadd(1M)</i> command for this	
Deinstallation
Deinstall add-on products	yes	
Deinstall separate software packages	yes	yes
Information
Display information on installed add-on products	yes	
Display overview of installed packages		yes
Display detailed information on installed packages	yes	yes
Print information on add-on products or packages	yes	

Table 21: Using SYSADM software and SYSADM software_prod

4.2 Installing software products

Many Reliant UNIX add-on products are supplied on CD-ROM. This section explains how to install such add-on products on your system.



Some add-on products and software packages require a key diskette containing a special authorization file. Please install the key diskette **before** you install the add-on product or packages belonging to it. Please refer to the [Section "Installing the key diskette"](#).



If you wish to install software which is not on a CD-ROM, please read the [Section "Installing single software packages"](#).



If you wish to install system software packages subsequently, please use the `pkgadd(1M)` command.

4.2.1 Preparing for installation

Add-on products can be installed locally or remotely and each requires different preparatory steps.

Local installation

- ▶ Insert the CD-ROM into the CD-ROM (DVD) drive on the **local** computer.

Please refer to the [Section "CD-ROM form"](#).

Remote installation

Before installing add-on products remotely, you must select a server with a CD-ROM (DVD) drive. You must then prepare the selected server for the remote installation:

- ▶ Insert the CD-ROM into the CD-ROM (DVD) drive on the **server**.
- ▶ Start `SYSADM`.
The main menu appears.
- ▶ Select the `software_prod` menu item.
The `Process_Products` menu appears.
- ▶ Select the `miscellaneous` menu item.
The `Miscellaneous Functions` menu appears.
- ▶ Select the `prepare_server` menu item.

The `Prepare as Server` form appears.

Sysadm:root@granit

x97801 Mode Display Keyboard

UNIX System V Operations, Administration and Maintenance

2 -----Process Products-----

4 Prepare as Server

CD-ROM device name: base

Local mountpoint: g

Clients:

machine - Machine Configuration, Display and Shutdown

network_services - Network Services Administration

performance - System Activity and System Tuning

ports - Port Access Services and Monitors

schedule_task - Schedule Automatic Task

software - Software Installation and Removal

→software_prod - Process Software Products on/from CD-ROM

storage_devices - Storage Device Operations and Definitions

system_setup - Initial System Setup

users - User Login and Group Administration

Fill in the form and then press SAVE.

HELP CHOICES SAVE CANCEL CMD-MENU EXIT

8 Bit compose dead key input:

CD-ROM device name

Device name of the CD-ROM (DVD) drive.

The name of the device file or the alias name as entered in the */etc/device.tab* file is permitted. Press the **CHOICES** function key to determine the valid device names.

Local mountpoint

Name of the directory under which the CD-ROM (DVD) drive is mounted.



The name of this mount point is also used as the "local" mount point on the client. This means that no other file system can be mounted on the client under this mount point.

Clients

Names of all client systems

The name of a client system corresponds to the output of *uname -n*. The names must be separated from each other by a blank.

- ▶ Fill in the form.
- ▶ Press the **SAVE** function key.
- ▶ Press the **Ú** key.
- ▶ Exit **SYSADM**.

To do this press the **[EXIT]** function key.

You can now switch to the console of the first client system. The remaining part of the installation procedure is

analogous to that on a local system.

4.2.2 CD-ROM form

- ▶ Start *SYSADM*.

The main menu appears. [Figure "Overview of SYSADM software_prod products_on_CD"](#) on the next page provides you with an overview of the remaining program execution.

- ▶ Select the *software_prod* menu item.

The *Process_Products* menu appears.

- ▶ Select the *products_on_CD* menu item.

The *Process Multi-Product CD-ROM* form appears (see ...). CD-ROMs containing one or more add-on products are called a "Multi-Product CD-ROM".

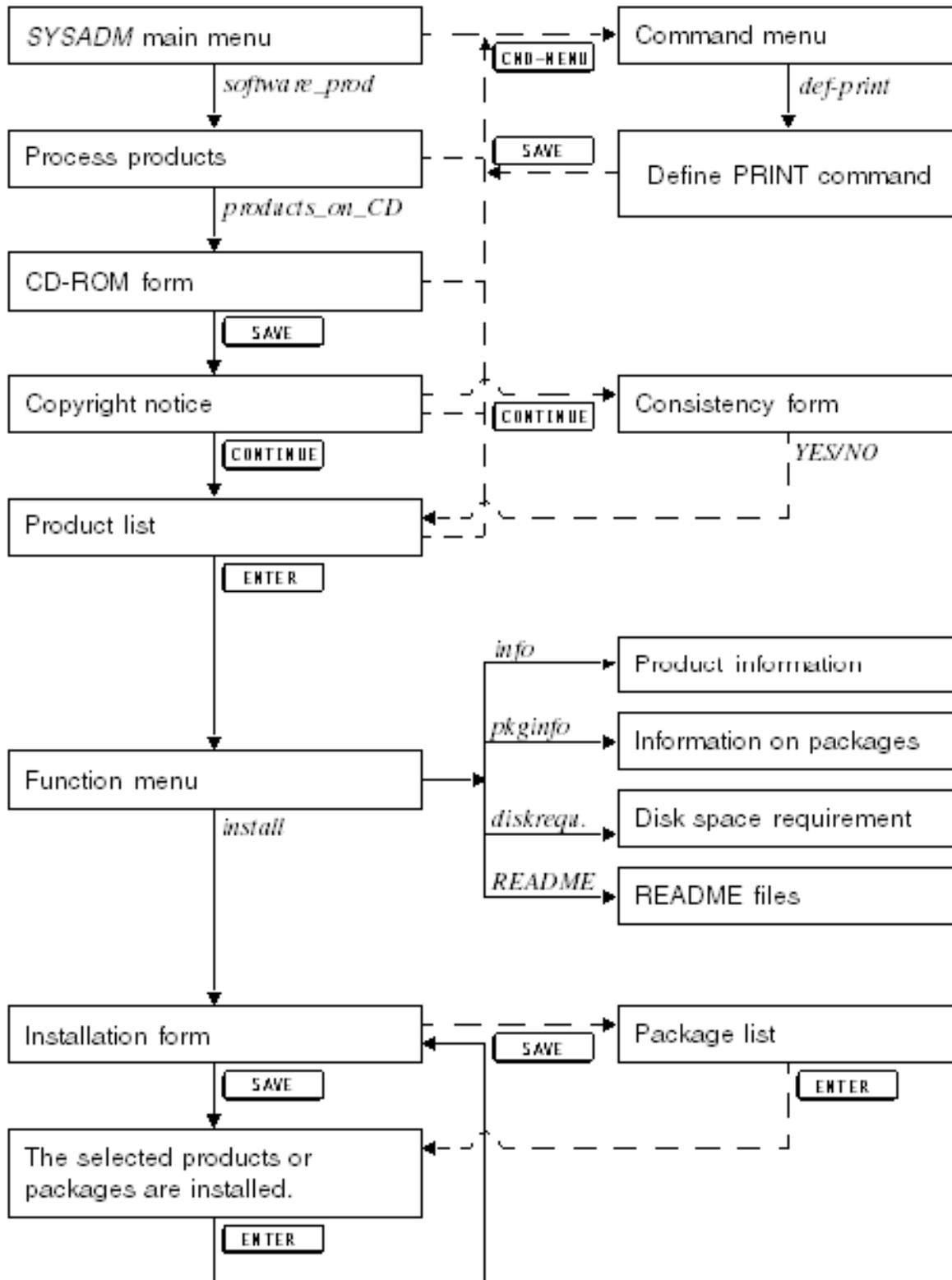


Figure 15: Overview of SYSADM – software_prod — products_on_CD

Sysadm:root@granit

x97801 Mode Display Keyboard

2 -----Process Products-----

3 Process Multi-Product CD-ROM

Server: local

CD-ROM device name: cdrom0

Remote pathname /

Local mountpoint: /cdrom0

network_services - Network Services Administration

performance - System Activity and System Tuning

ports - Port Access Services and Monitors

schedule_task - Schedule Automatic Task

software - Software Installation and Removal

→software_prod - Process Software Products on/from CD-ROM

storage_devices - Storage Device Operations and Definitions

system_setup - Initial System Setup

users - User Login and Group Administration

Fill in the form and then press SAVE.

HELP CHOICES SAVE CANCEL CMD-MENU EXIT

8 Bit compose dead key input:

Server

Name of the computer to which the CD-ROM (DVD) drive is connected.

Local installation: *local* (default)

Remote installation: name of the server

CD-ROM device name

Device name of CD-ROM drive.

The field is only significant if the CD-ROM (DVD) drive is connected to the local computer. The name of the device file or the alias name as entered in the `/etc/device.tab` file is permitted.

Remote pathname / Local mountpoint

- Local installation (parameter *Server* **equal** to *local*):
Name of the directory under which the CD-ROM (DVD) drive is mounted
- Remote installation (parameter *Server* **not equal** to *local*):
Name of the *nfs* file system under which the CD-ROM (DVD) drive is accessible

▶ **Fill in the form.**

Selection menus are available for the *CD-ROM device name* and *Remote pathname / Local mountpoint* fields and can be reached via the [CHOICES] function key.

▶ **Press the [SAVE] function key.**

A text window containing the copyright notice appears.

- ▶ Acknowledge the copyright notice with the [CONTINUE] function key.

The CD-ROM data format is checked

CD-ROMs can have different data formats. You do not normally need to worry about this. *SYSADM* automatically checks the format of the inserted CD-ROM and if an error occurs during this check, an appropriate message is output. Example:

```
It is not possible to determine type of CD-ROM format.
Please select from the following menu.
1  pure ISO9660 format (mount with dos option)
2  rock ridge format (mount without dos option)
1,2 (default: 1) -->
```

In such a case, please note:

- The *dos* option refers to the *mount(1M)* command executed automatically by *SYSADM* which must either use the *-odos* option or not, depending on the CD-ROM data format (see also the two following points).
- The system CD-ROM, for example, has a Rock Ridge format. CD-ROMs with this format should normally be mounted without the *dos* option.
- Add-on product CD-ROMs, for example, have an ISO9660 format. CD-ROMs with this format must be mounted with the *dos* option.

Proceed as follows:

- ▶ Select the appropriate menu item.
If the *mount(1M)* command succeeds, the text window containing the copyright notice then appears.
- ▶ Acknowledge the copyright notice with the [CONTINUE] function key.

If the `mount(1M)` command does not succeed, the **CD-ROM form** reappears. In this case, you should check

- that the CD-ROM is correctly inserted in the drive,
- whether you have accidentally selected the wrong menu item.

Once you have eliminated the error, press the [SAVE] function key again.

4.2.3 Consistency form

The system now checks to see whether all the installed products are also contained in the product database. If this is not the case, you are asked whether the consistency of the database is to be established automatically.

Sysadm:root@granit

x97801 Mode Display Keyboard

2-----Process Products-----

3-----Process Multi-Product CD-ROM-----

5 Establish Consistency of Product Database

Execute?: YES

Local mountpoint:

network_services - Network Services Administration
 performance - System Activity and System Tuning
 ports - Port Access Services and Monitors
 schedule_task - Schedule Automatic Task
 software - Software Installation and Removal
 →software_prod - Process Software Products on/from CD-ROM
 storage_devices - Storage Device Operations and Definitions
 system_setup - Initial System Setup
 users - User Login and Group Administration

Fill in the form and then press SAVE.

HELP CHOICES SAVE CANCEL CMD-MENU EXIT

8 Bit compose dead key input:

- ▶ Reply with *YES*. If you answer *NO*, you will not receive some of the information described below about installed products.



Inconsistencies always occur when packages are installed or deinstalled without using the *software_prod* menu item in the *SYSADM* user interface. This is, for example, the case with an update installation.

If you reply with **NO**, some of the subsequent information on previously installed products will not be displayed.



If the product database is so badly damaged that the consistency cannot be re-established automatically, you can select the *miscellaneous — consistency* menu item under *software_prod* and answer *y* to the question *Extensive check ?*

4.2.4 Product list

A list of all products on the CD-ROM is displayed.

```

Sysadm:root@granit
x97801  Mode  Display  Keyboard

-----Process Products-----
-----Process Multi-Product CD-ROM-----
5      SINIX_Y 5.45A0040 19/11/98
MAN    5.45A00 - -  online manual pages
MLX-J  5.45A00 - -  DC/OSx Multi-Lingual Extensions, Japan only
MLX    5.45A00 - -  DC/OSx Multi-Lingual Extensions, except Japan
ROCOMPAT 5.45A00 - -  RISC/OS Compatibility
SAXLSM 2.3A10  - -  Local Software Maintenance
SENDMAIL 5.45A00 - -  Sendmail_8.8.7
SIACC  5.45A00 - -  SCSI Autochanger Device Driver
SIDPTG2 5.45A00 - -  Sinix DPTG2-Protocol System Components
SIMPD  5.45A00 - -  MultiPathDriver for Galaxy Box
SINIX  5.45A00 - -  ReadMe for SINIX Y
SINUS  5.45A00 - -  SINIX new update set
SISAGAPI 5.45A00 - -  SIsagapi for SINIX Y
SISPLHP 5.45A00 - -  Sinix High Performance Printer SCSI Device Driver
SITHREADS 5.45A00 - -  DCE Threads and Reentrant Libraries
SITRACEX 5.45A00 - -  TRACEX - SINIX kernel tracer
SIXMSB 2.1A00 - -  XMS-base/menu

HELP  MARK  ENTER  CANCEL  CMD-MENU  PRINT

8 Bit  compose  dead key  input:

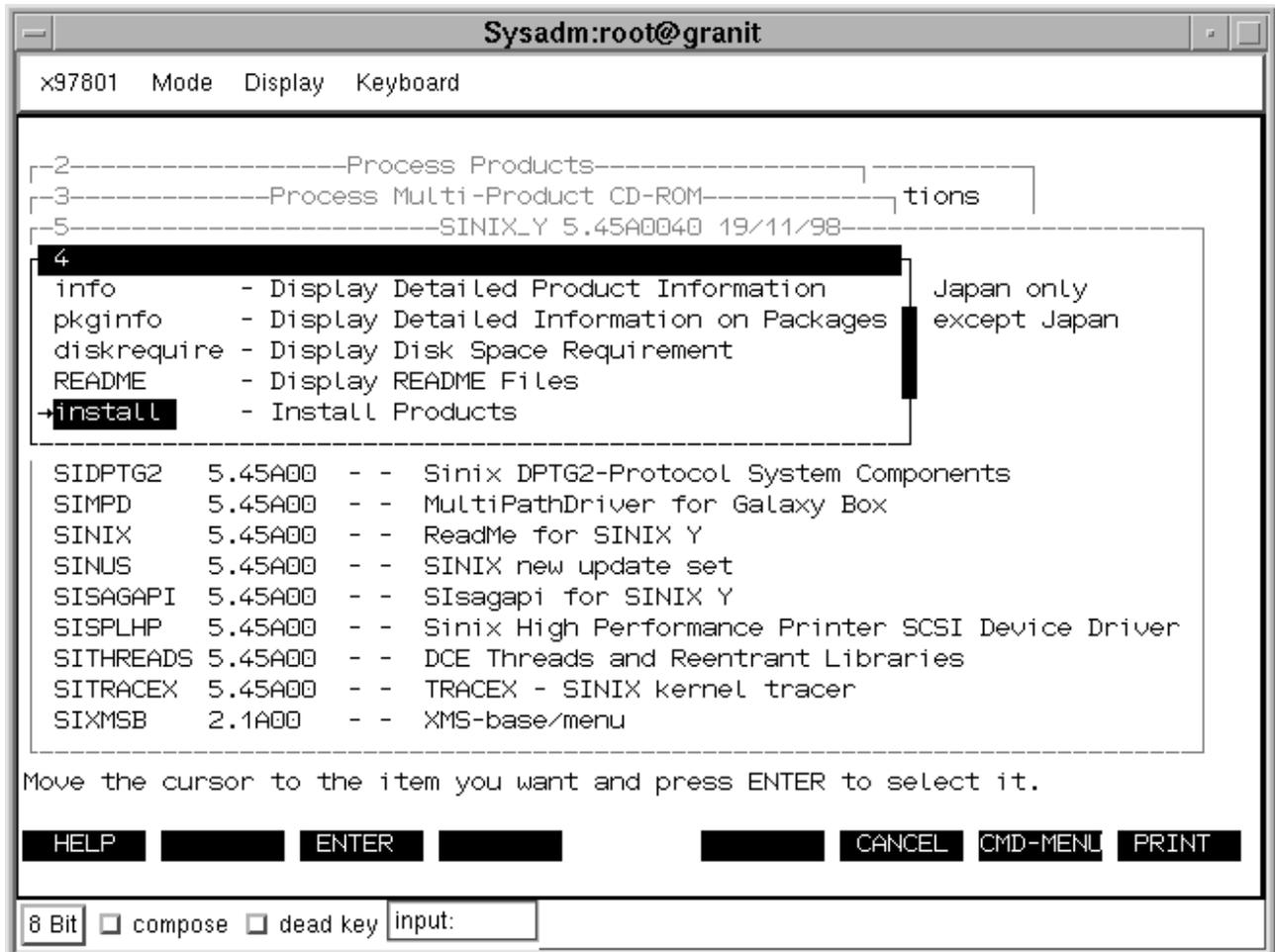
```

The following information on each product is shown:

- Name
- Version
- Version ID
 - S* The same version is already installed.
 - D* Another version is already installed.
 - The product has not been installed yet.
 - U* Undefined, cannot be determined
(because of an inconsistent product database)
 - E* Error
- Installation status
 - I* Completely installed
 - P* Partially installed
 - Not installed
 - U* Undefined, cannot be determined
(because of an inconsistent product database)
 - E* Error
- Brief description
- ▶ Mark all products you wish to install.
Use the [MARK] function key for this. Pressing [MARK] a second time will deselect a (mistakenly) marked product.
-  License check. In the case of license products, you will be asked during the installation whether the correct licenses have been purchased. The product will only be installed if you can answer *YES*.
- ▶ Press the [ENTER] function key.
-  You can also print the information displayed in this window and some of the subsequent ones. This option is available if the [PRINT] function key is offered. A description of this can be found in the [Section "Printing information"](#).

4.2.5 Function menu

If you selected one or more products from the **Product list** in **Section "Product list"**, the *Function menu* then appears. Here you can select the actions to be executed with the selected products.



info, *pkginfo*, *diskrequire* and *README*

Information on the products you previously selected is displayed. Examples can be found in the **Section "Displaying information"**.

install

The **Installation form** appears (see next page).

4.2.6 Installation form

If you selected the **install** menu item from the **Function menu** (see ...), the following form appears.

```

Sysadm:root@granit
x97801  Mode  Display  Keyboard

-----Process Products-----
-----Process Multi-Product CD-ROM-----
-----SINIX_Y 5.45A0040 19/11/98-----
-----
6  Install
-----
Installation extent:      Complete Product
Installation mode:       automatic
Installation defaults filename: default
Path to response files:  on CD-ROM
-----
SIMPD   5.45A00  - -  MultiPathDriver for Galaxy Box
SINIX   5.45A00  - -  README for SINIX Y
SINUS   5.45A00  - -  SINIX new update set
SISAGAPI 5.45A00  - -  SISagapi for SINIX Y
SISPLHP 5.45A00  - -  Sinix High Performance Printer SCSI Device Driver
SITHREADS 5.45A00 - -  DCE Threads and Reentrant Libraries
SITRACEX 5.45A00 - -  TRACEX - SINIX kernel tracer
SIXMSB  2.1A00   - -  XMS-base/menu

Fill in the form and then press SAVE.

HELP  CHOICES  SAVE  CANCEL  CMD-MENU  EXIT

8 Bit  compose  dead key  input:

```

Installation extent

Complete Product

All packages of the selected products

Package

Single packages of the selected products

SreadmeM

Only packages containing the *README* files (*SreadmeM* packages)

Installation mode

Type of dialog

automatic

The installation parameters are taken from existing reply files.

dialog

The installation parameters for all selected products are requested before starting the installation.

Installation defaults filename

Name of the file containing general installation parameters

If no absolute path name is specified, the file is searched for under */var/sadm/install/admin*. (The

contents of this file are described in the "System Administrators Reference Manual" under *admin(4)*).

Path to response files

Path name under which the response files are stored

Either **on CD-ROM** (default) must be specified or an absolute path name under which the response files for all selected packages are stored.

▶ Fill in the form.

Selection menus are available for the *Installation extent* and *Installation mode* fields and they can be reached via the [CHOICES] function key.

If you specify the *dialog* parameter as the *Installation mode*, the path name for temporary files into which the replies are written is automatically entered in the *Path to response files* field. These response files are deleted after installation is completed. You will have to change the default path name if you want to retain the files for future installations.

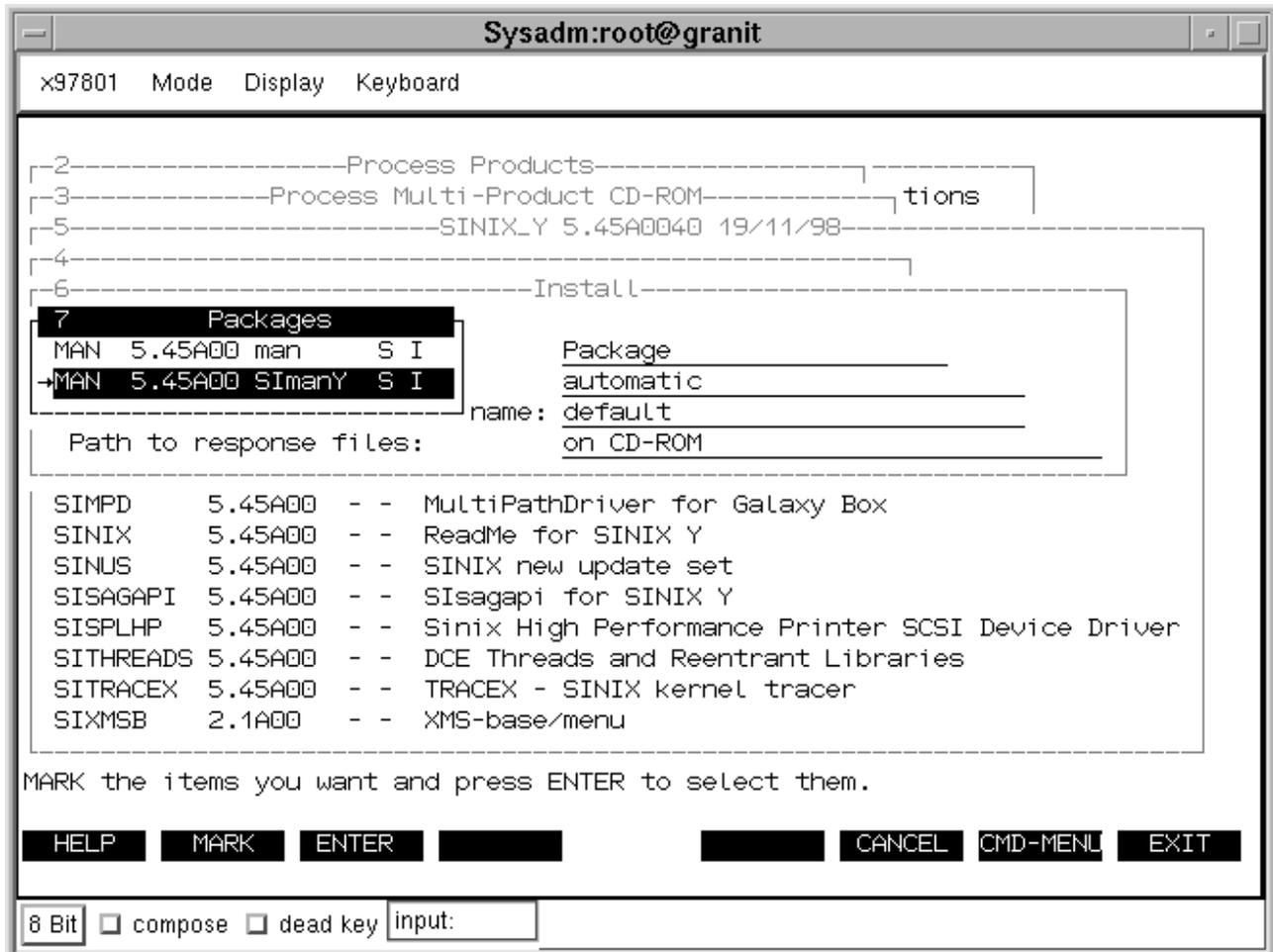
▶ Press the [SAVE] function key.

The subsequent sequence depends on the *Installation extent* you selected:

- If you selected *Complete Product* or *SReadmeM*, the installation process now starts (see the [Section "Installation process"](#)).
- If you selected *Package*, you must first select the packages you wish to install.

4.2.7 Package list

If you selected *Package* in the **Installation form**, a selection list now appears containing all packages you can install.



The following information is displayed for each package:

- Product name
- Product version
- Package name
- Version ID
 - S* The same version is already installed.
 - D* Another version is already installed.
 - The product has not been installed yet.
 - U* Undefined, cannot be determined (because of an inconsistent product database)
 - E* Error
- Installation status
 - I* Completely installed
 - P* Partially installed

– Not installed

U Undefined, cannot be determined
(because of an inconsistent product database)

E Error

- ▶ Mark all packages in the list that you wish to install.
Use the [MARK] function key for this.
- ▶ Then press the [ENTER] function key.

4.2.8 Installation process

The installation process is accompanied by relevant messages. You may also be asked additional questions about the sequence and scope of the installation.



If the software product is a licensed product (key diskette, LVS or written license), you will be prompted to specify whether you have acquired the appropriate licenses. The product can only be installed if you have the proper licenses.

Example of manpages in Reliant UNIX:

```
The following output is logged in: /var/sadm/products/log/15
Installation from CD-ROM: SINIX_Y 5.45B0000 26/06/00
Creation time of CD-ROM: Mon Jun 29 15:22:14 2000
Check for multiple product versions selected and previously
installed product versions.
-- MAN 5.45B00 --
Check for necessary licenses.
Calculate space required.
-- MAN 5.45B00 --
Sufficient space available.
Check whether user input is needed.
-- MAN 5.45B00 --
Response file being created. Do not remove the data medium.
□□□□□□□□□□:
□□□□□□□□□□:
Installation in progress. Do not remove the data medium.
□□□□□□□□□□:
□□□□□□□□□□:
□□□□□□□□□□:
□□□□□□□□□□:
No error occurred.
Log file is /var/sadm/products/log/15.
Continue with ENTER
```

► Press **Ú** when the installation is complete.

This returns you to the **Installation form** if package mode was selected. In product mode, on the other hand, you are returned to the product menu if the installation is **successful** (see ...).

After installing add-on products and terminating *SYSADM*, you are informed that email has arrived for you:

```
you□have□mail
```

► Call the *mail(1)* command:

```
#□mail
```

You are then informed whether the installation was successful:

```
Installation□of□machine dependent manual pages (SImanY)□was
successful.
-- 2 of 2 packages (100%) processed --
No errors.
```

You can also read how the installation progressed in the specified log file (in this case: */var/sadm/products/log/25*).

► Remove the CD-ROM from the CD-ROM (DVD) drive.

4.3 Deinstalling software products

This section describes how to deinstall add-on products installed on your system.



If you wish to delete single system software packets, you can also use the *SYSADM software* menu item. In this case, refer to the section of the same name in **Section "Deinstalling single software packages"**.

► Start *SYSADM*.

The main menu appears. [Figure "Overview on SYSADM software prod products on system"](#) on the next page provides you with an overview of the remaining program execution.

- ▶ Select the *software_prod* menu item.

The *Process_Products* menu appears.

- ▶ Select the *products_on_system* menu item.

A check is now made as to whether all installed products are in the product database. If not, the database is updated automatically (i.e. there is no **Consistency form** as with installation).



Inconsistencies always occur when packages are installed or deinstalled without using the *software_prod* menu item in the *SYSADM* user interface. This is the case with an update installation, for example.

If all system software are inconsistent (e.g. after an update installation), automatic database updating can take a few minutes.

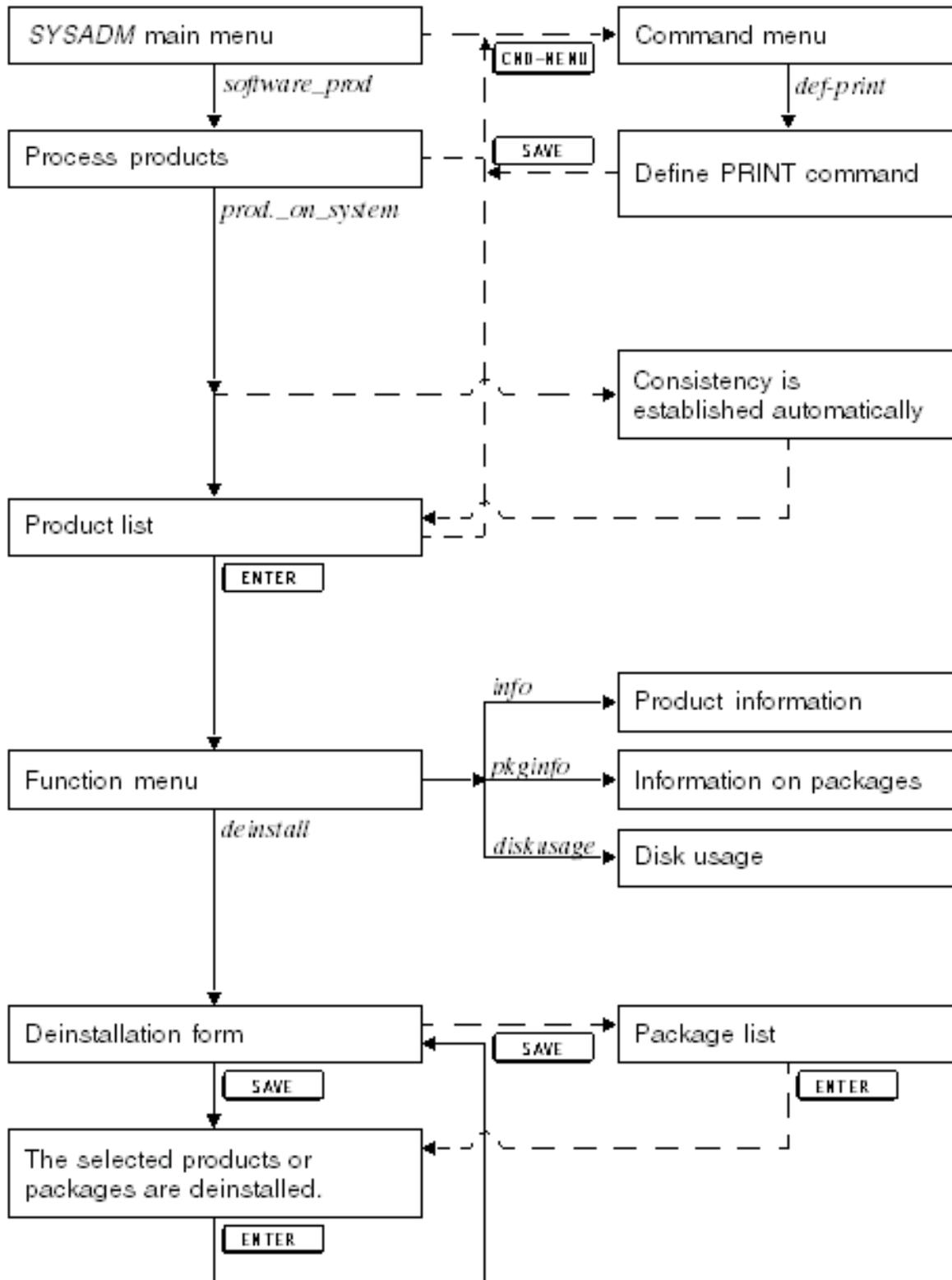
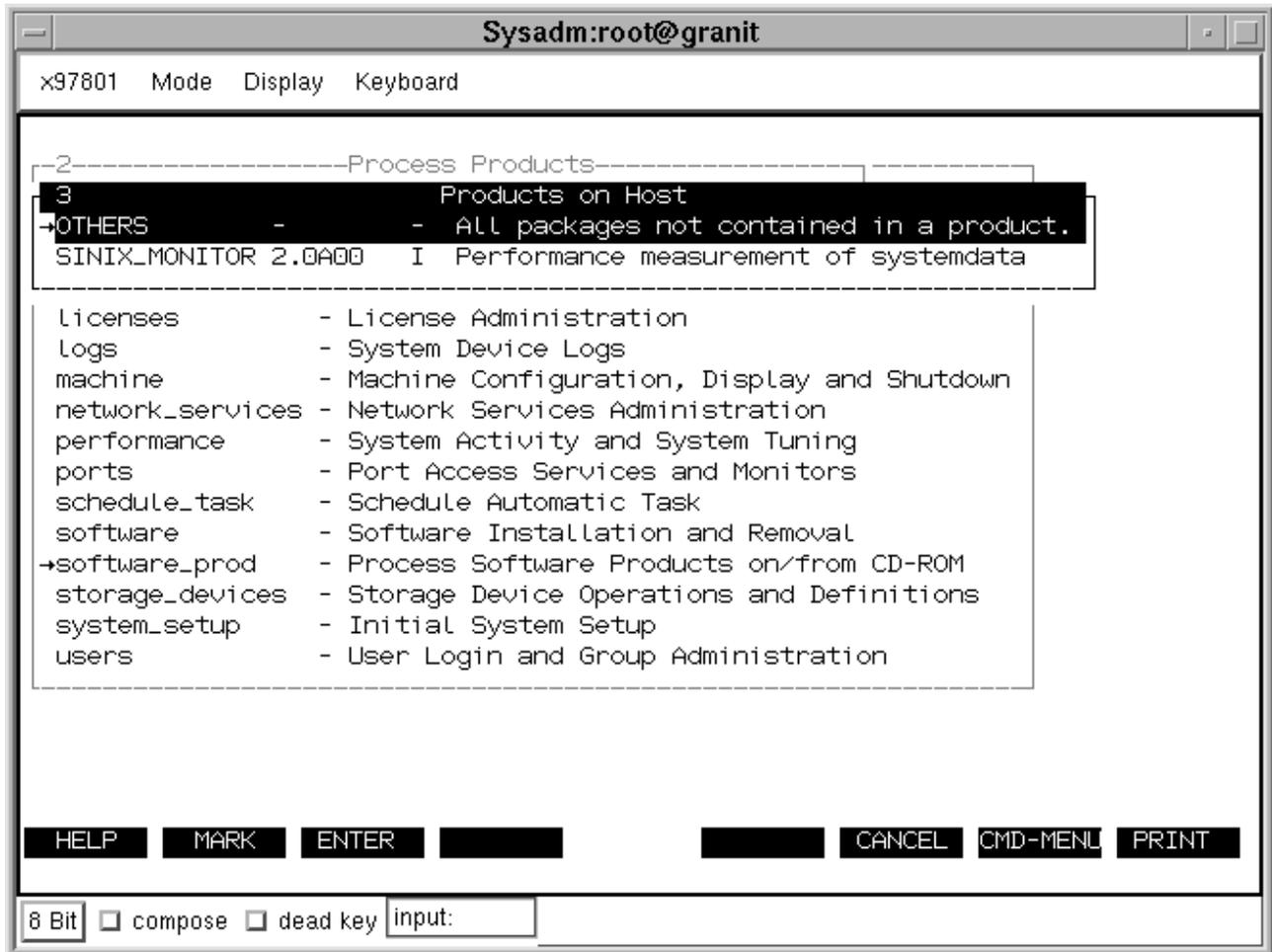


Figure 16: Overview on SYSADM – software_prod — products_on_system

4.3.1 Product list

A list of all products on the system is displayed.



The following information is displayed for each product:

- Name
- Version
- Installation status
 - I* Completely installed
 - P* Partially installed
 - U* Undefined, cannot be determined
(because of an inconsistent product database)
 - E* Error
- Brief description

All packages not belonging to an add-on product are grouped together under the "OTHERS" pseudo product. This includes, for example, all system software packages.

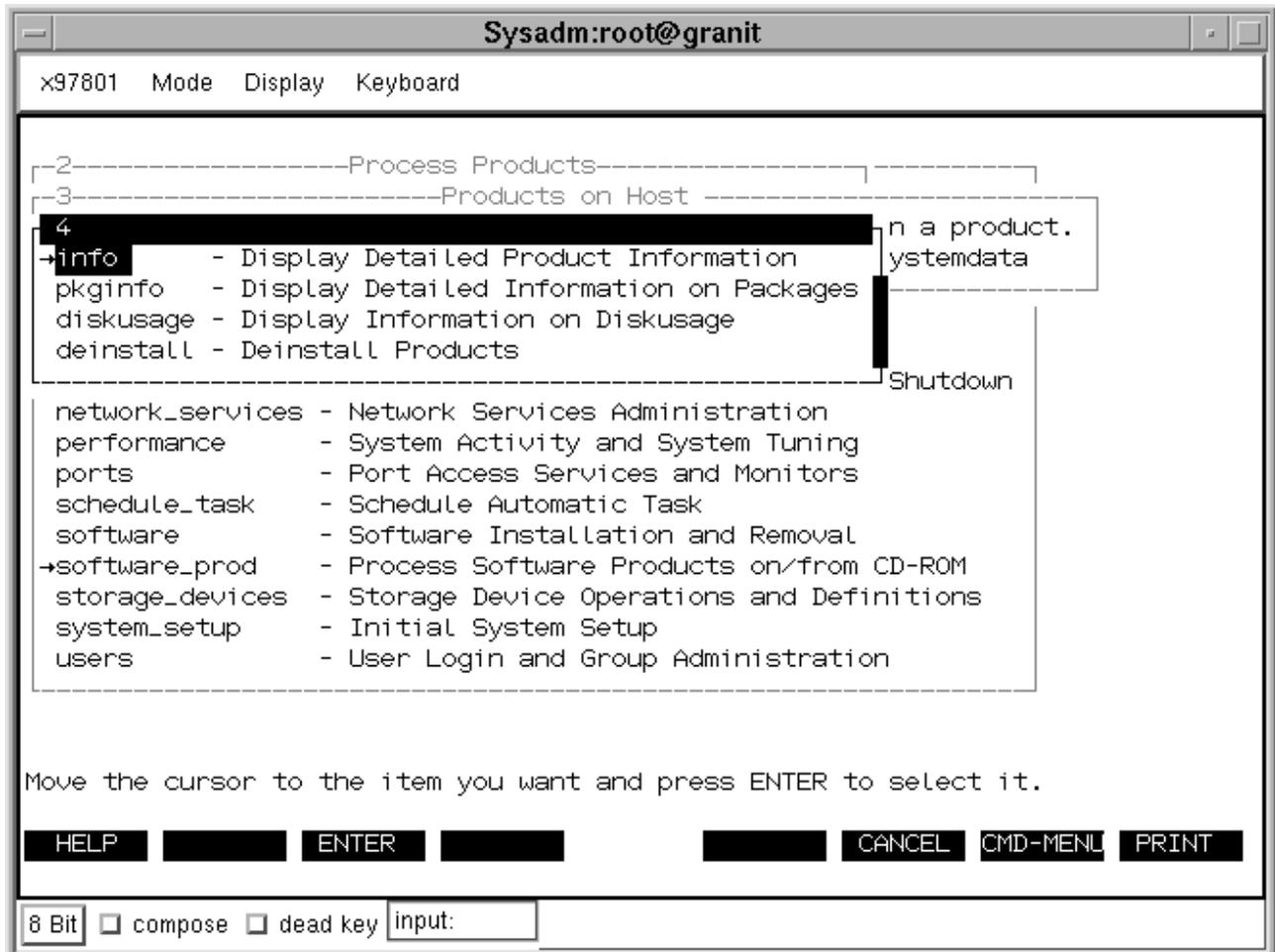
- ▶ Mark all products you wish to deinstall.
Use the [MARK] function key for this. Pressing [MARK] a second time will deselect a (mistakenly) marked product.
- ▶ Press the [ENTER] function key.



You can also print the information displayed in this window and some of the subsequent ones. This option is available if the [PRINT] function key is offered. A description of this can be found in the [Section "Printing information"](#).

4.3.2 Function menu

If you selected one or more add-on products from the [Product list](#), the [Function menu](#) now appears. Here you can select the actions to be executed with the selected products.



info, pkginfo, diskusage

Information on the products you selected previously is displayed. Examples can be found in the [Section "Displaying information"](#).

deinstall

The [Deinstallation form](#) appears.

4.3.3 Deinstallation form

If you selected the *deinstall* menu item from the **Function menu** (see previous page), the following form now appears:

```

Sysadm:root@granit
x97801  Mode  Display  Keyboard

-----Process Products-----
-----Products on Host -----
-----n a product.-----
5  Deinstall

Deinstallation extent:      Complete Product
Deinstallation defaults filename: default

network_services - Network Services Administration
performance      - System Activity and System Tuning
ports             - Port Access Services and Monitors
schedule_task    - Schedule Automatic Task
software          - Software Installation and Removal
->software_prod   - Process Software Products on/from CD-ROM
storage_devices  - Storage Device Operations and Definitions
system_setup     - Initial System Setup
users            - User Login and Group Administration

Fill in the form and then press SAVE.

HELP  CHOICES  SAVE  CANCEL  CMD-MENU  EXIT

8 Bit  compose  dead key  input:

```

Deinstallation extent

Complete Product

All packages of the selected products

Package

Single packages of the selected products

Deinstallation defaults filename

Name of the file containing general deinstallation parameters. If no absolute path name is specified, the file is searched for under `/var/sadm/install/admin`. (The contents of this file is described in the "System Administrators Reference Manual" under *admin(4)*).

► Fill in the form.

A selection menu is available for the *Deinstallation extent* field and can be reached via the [CHOICES] function key.

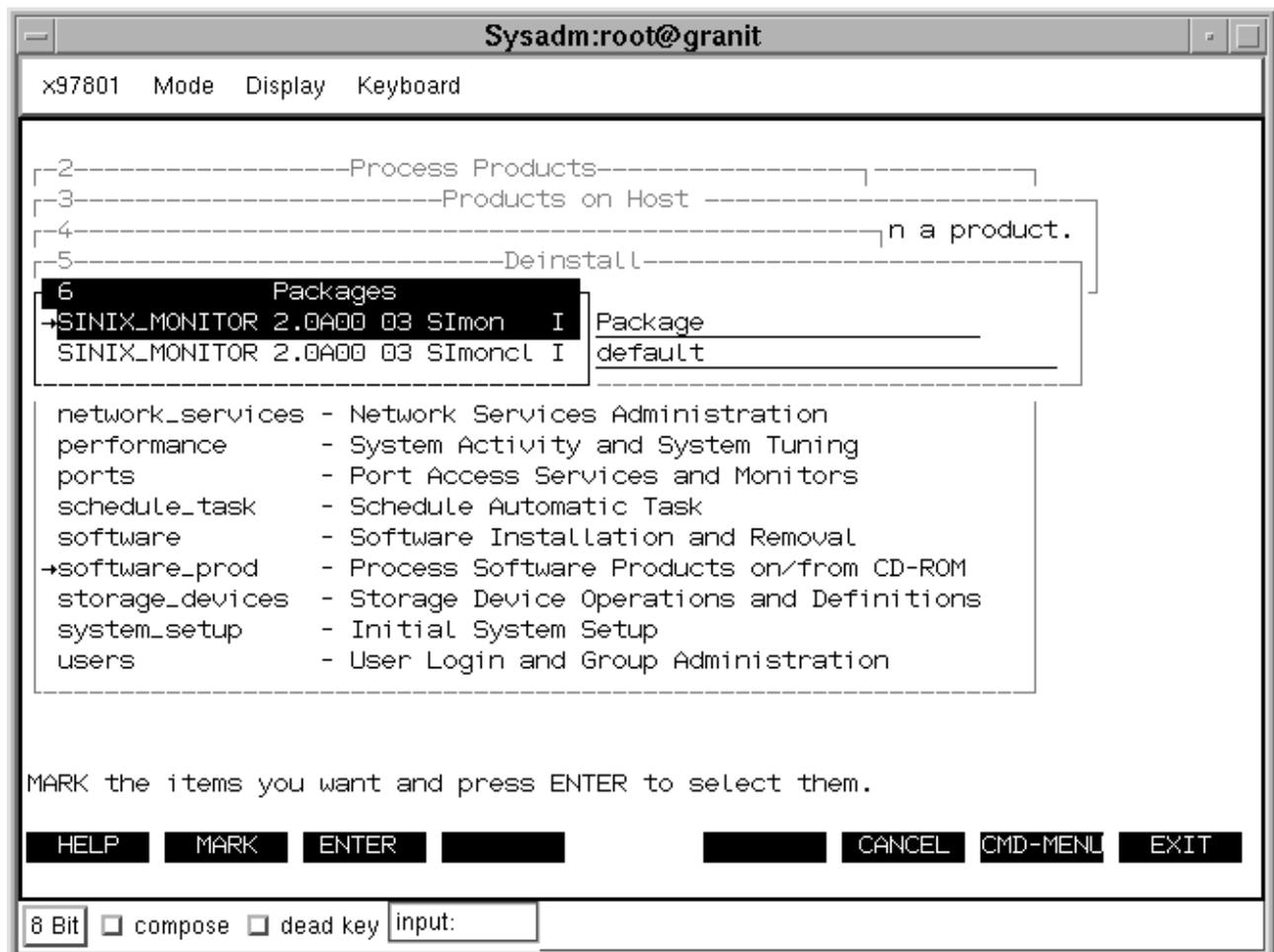
► Press the [SAVE] function key.

The remaining process depends on the **Deinstallation extent** you selected:

- If you selected *Complete Product*, the deinstallation process now starts (see [Section "Deinstallation process"](#)).
- If you selected *Package* you must first select the packages you wish to deinstall.

4.3.4 Package list

If you selected *Package* in the **Deinstallation form**, a selection list containing all installed packages belonging to the selected products now appears.



The following information is displayed for each package:

- Product name
- Product version
- Production state
- Package name
- Installation status

I Completely installed
P Partially installed
U Undefined, cannot be determined
(because of an inconsistent product database)
E Error

- ▶ Mark all packages in the list that you wish to deinstall.
Use the [MARK] function key for this.



If you deinstall system software packages, you cannot reinstall them later with *SYSADM*. Use the *pkgadd(1M)* command in such a case (or perform a new installation).

- ▶ Then press the [ENTER] function key.
- ▶ Deinstallation now starts.

4.3.5 Deinstallation process

The deinstallation process is accompanied by relevant messages. You may also be asked additional questions on the deinstallation sequence and extent. Example of the product *SINIX/monitor*:

```

The following output is logged in: /var/sadm/products/log/26
Determining packages contained in selected products.
--SINIX_MONITOR2.0A0003--
Checking whether packages are subject to multiple selection.
Checking whether package is being used by some other product.
--SINIX_MONITOR2.0A0003--
*** Starting deinstallation.***
**SINIX_MONITOR2.0A0003**
--SImoncl--
The following package is currently installed:
SImoncl SINIX/monitor collect client
(R3000)2.0A00
Do you want to remove this package [y,n,?,q]y
## Removing installed package instance <SImoncl>
## Verifying package dependencies.
## Processing package information.
## Removing pathnames in <none> class
/opt/monitor<shared pathname not removed>
## Updating system information.
.
.
.
.
No error occurred.
Protocol file is /var/sadm/products/log/26.
Press ENTER to continue

```

► Press **U** after deinstallation is completed.

You are then returned to the **Deinstallation form**.

After installing add-on products and terminating *SYSADM* you are informed that email has arrived for you:

```
you have mail
```

► Call the *mail(1)* command:

```
# mail
```

You are then informed whether the deinstallation was successful:

```
Deinstallation of SINIX/monitor on unbidden as Version 1.0A00 Load 02
of package instance <SImon> was successful.
```

You can also read how the installation progressed in the specified log file (in this case: */var/sadm/products/log/26*).

4.4 Installing single software packages

This section describes how to install single software packages supplied on tape cartridge or diskette (i.e. not on CD-ROM).



If you wish to install software products from CD-ROM, please read the [Section "Installing software products"](#).



If you wish to install system software packages subsequently, please use the `pkgadd(1M)` command (or perform a new installation).

- ▶ Start `SYSADM`.

The main menu appears. The following figure provides an overview of the remaining program execution.

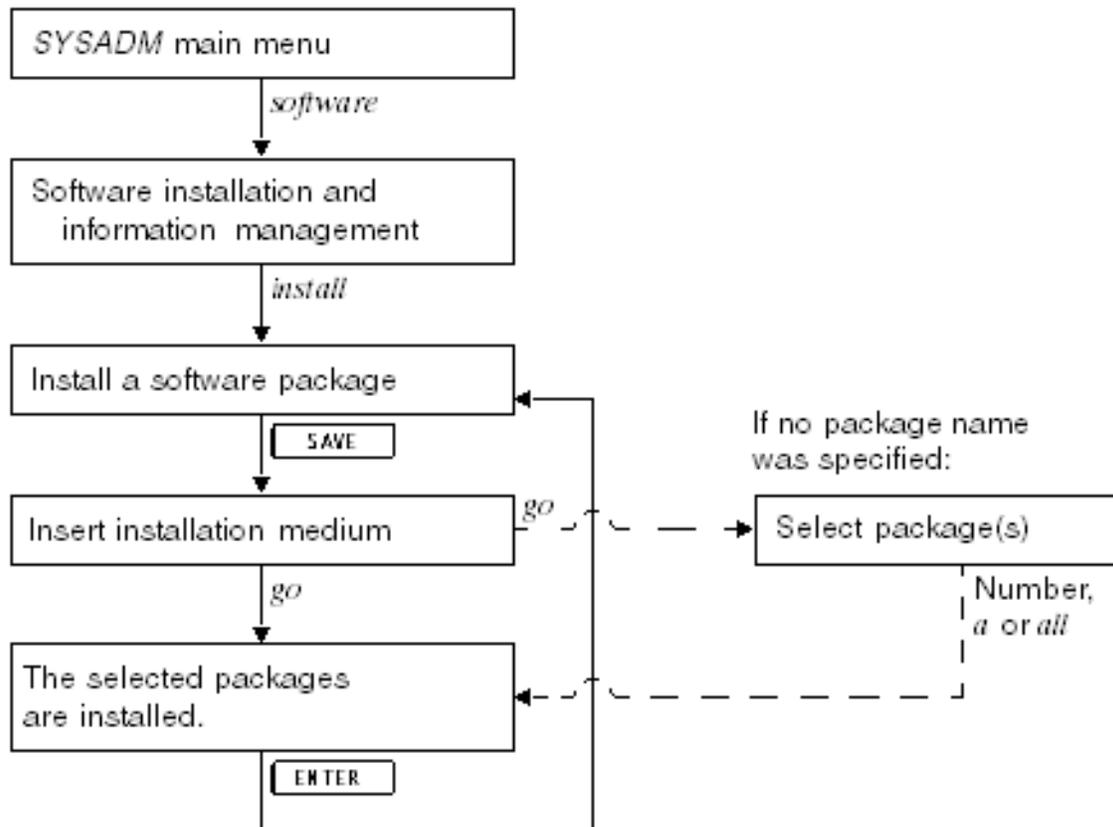
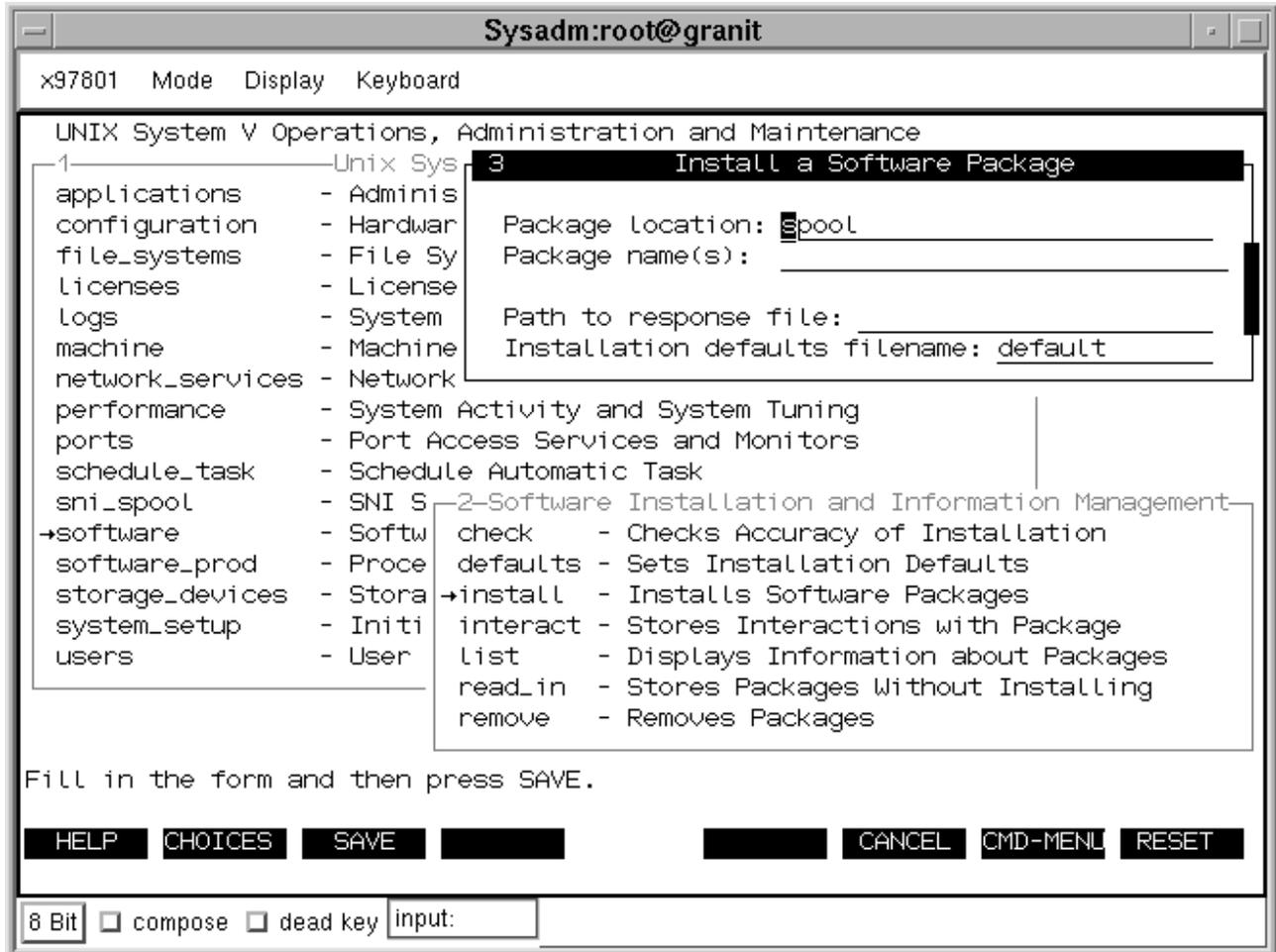


Figure 17: Overview of SYSADM – software — install

- ▶ Select the `software` menu item.
The *Software Installation and Information Management* menu appears (see ...).
- ▶ Select the `install` menu item.
The following form is displayed:



Package location

Storage device containing the software to be installed.

Use [CHOICES] to obtain a list of all devices.

spool stands for the */var/spool/pkg* directory.

Package name(s)

Names of the packages to be installed.

The names must be separated with a comma. Enter **all** if all packages on the installation medium are to be installed.

If you leave the field empty, a list of all packages on the installation medium is displayed. Select the packages you wish to install.

Path to response file

If the dialog process required by the installation routine was stored in a file, you can specify its path name here. Otherwise, leave this field empty.

Installation defaults filename

If the defaults required by the installation routine were stored in a file, you can specify its path name here. Otherwise, specify default.

- ▶ Fill in the form.
- ▶ Press the [SAVE] function key.
You are then requested to insert the installation medium into a specific drive.
- ▶ Insert the installation medium into the drive.

- ▶ Confirm with *go*.

The installation process now starts. As soon as it is finished, you receive the message:

Press ENTERto continue

- ▶ Press *Ú*.

You are returned to the *Install a Software Package* form (see previous page).

4.5 Deinstalling single software packages

This section describes how to deinstall single software packages that are installed on your system.



If you wish to delete complete software products, you should use the *SYSADM software_prod* menu item. In this case, read the section of the same name in [Section "Deinstalling software products"](#).

- ▶ Start *SYSADM*.

The main menu appears. The following figure provides an overview of the remaining program execution.

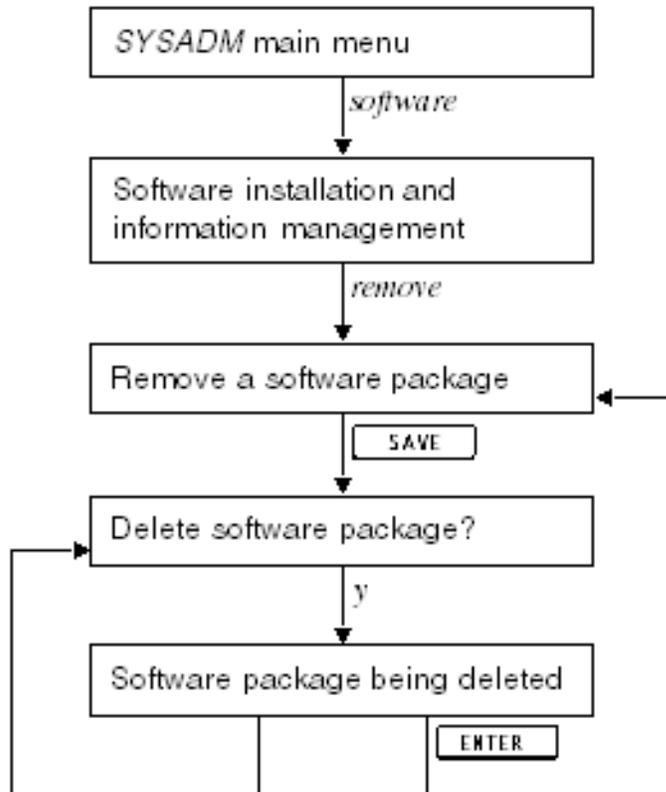


Figure 18: Overview of SYSADM – software — remove

- ▶ Select the *software* menu item.
The *Software Installation and Information Management* menu appears (see ...).
- ▶ Select the *remove* menu item.

The following form is displayed:

Sysadm:root@granit

x97801 Mode Display Keyboard

UNIX System V Operations, Administration and Maintenance

1 — 3 Remove a Software Package

appl
conf Location of the package to be removed: installed
file Package name(s) to be removed: _____
lice

Logs - System Device Logs
machine - Machine Configuration, Display and Shutdown
network_services - Network Services Administration
performance - System Activity and System Tuning
ports - Port Access Services and Monitors
schedule_task - Schedule Automatic Task
sni_spool - SNI S
→software - Software Installation and Information Management
software_prod - Proce
storage_devices - Stora
system_setup - Initi
users - User

check - Checks Accuracy of Installation
defaults - Sets Installation Defaults
install - Installs Software Packages
interact - Stores Interactions with Package
list - Displays Information about Packages
read_in - Stores Packages Without Installing
→remove - Removes Packages

Fill in the form and then press SAVE.

HELP CHOICES SAVE CANCEL CMD-MENU RESET

8 Bit compose dead key input:

Location of the package to be removed

Specify *installed* if only packages installed on the system disk are to be deinstalled.

Package name(s) to be removed

Names of the packages to be deinstalled.

The names must be separated by a comma. You can use [CHOICES] to display a list of all installed packages.

- ▶ Fill in the form.
- ▶ Press the [SAVE] function key.



You may also enter *all* in the *Package name(s) to be removed* field instead of specific package names. You must **never** do this on the system disk! Otherwise, it will no longer function as a system disk.



If you deinstall system software packages, you cannot reinstall them later with *SYSADM*. Use the *pkgadd(1M)* command in such a case (or perform a new installation).

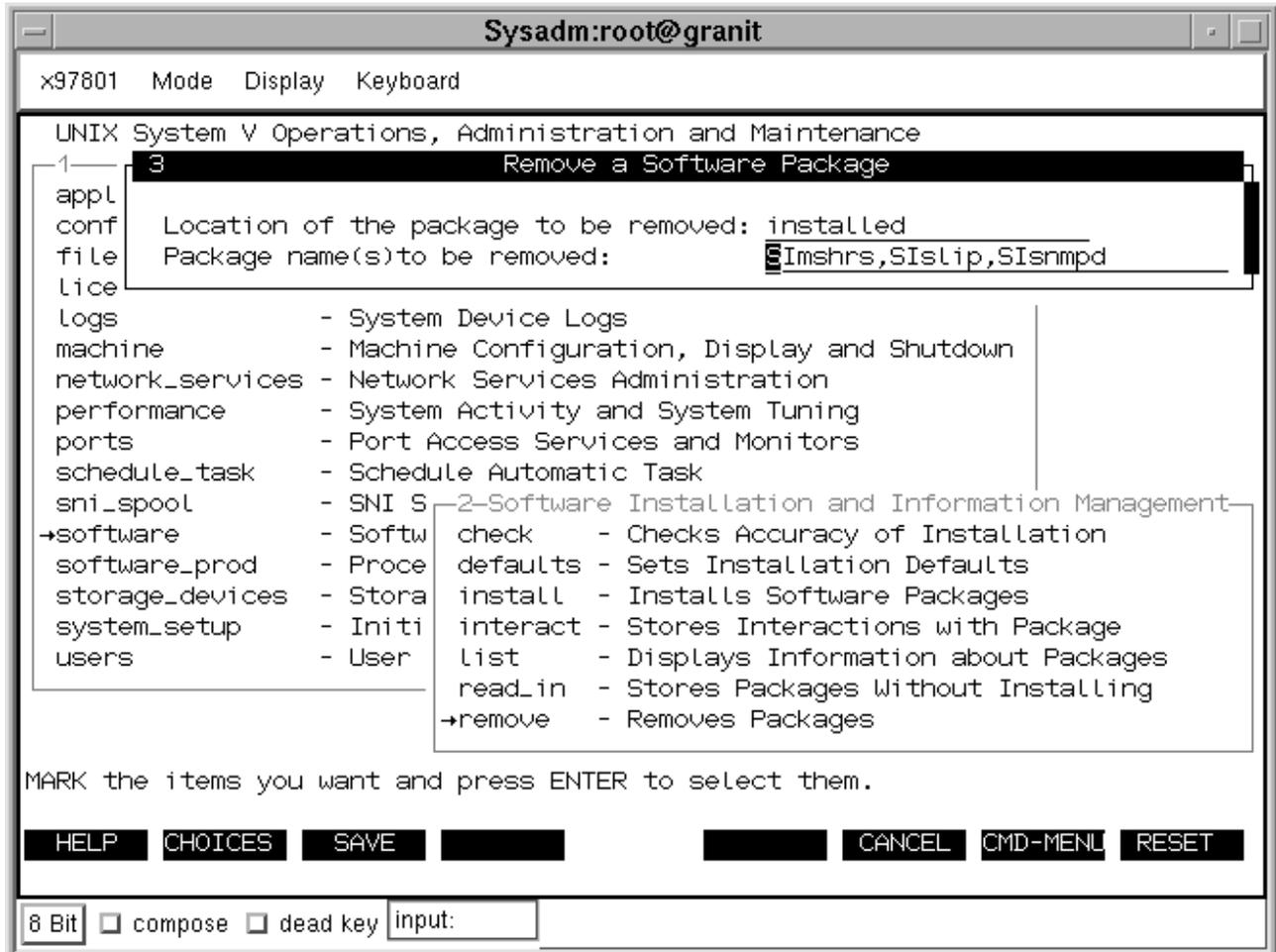


To deinstall some packages, you need to generate a new system kernel and then reboot your computer with the new operating system.



Some add-on products and software packages require a special authorization file. If you accidentally deinstall the relevant package, all add-on products that require an authorization file can no longer be started. If this happens, please read the [Section "Installing the key diskette"](#) to find out what you have to do.

The following figure shows a typical sequence for deinstalling a package. The *Smshrs*, *SIslip* and *SIsnmpd* packages are to be deinstalled as an example.



► Press the [SAVE] function key.

```

The following package is currently installed:
  SImshrs Sinix msh runtime system (English+German)
  (R3000) 1.0A00
Do you want to remove this package [y,n,?,q] y
## Removing installed package instance <SImshrs>
## Verifying package dependencies.
## Processing package information.
...
## Updating system information.
Removal of <SImshrs> was successful.
The following package is currently installed:
  SIslip OpenNet slip
  (R3000) 5.45B00
Do you want to remove this package [y,n,?,q] y
## Removing installed package instance <SIslip>
## Verifying package dependencies.
## Processing package information.
...
Do you wish to rebuild the kernel after <SIslip> removal?
(default: n) [y,n,?] n
## Updating system information.
Removal of <SIslip> was successful.
The following package is currently installed:
  SIsnmpd Sinix SNMP Agent
  (R3000) 3.0A40
Do you want to remove this package [y,n,?,q] y
## Removing installed package instance <SIsnmpd>
## Verifying package dependencies.
## Processing package information.
...
## Updating system information.
Removal of <SIsnmpd> was successful.
Press ENTER to continue

```

► Press Ú.

You are returned to the *Remove a Software Package* form (see ...).

Package dependencies

If you selected a package for deinstallation on which other packages are dependent, a corresponding warning is output. You must then decide whether you wish to continue the deinstallation or abort it.

Example: package *tcp*

The following package is currently installed:

```

tcp OpenNet TCP/IP
(R3000) 5.45B00

```

Do you want to remove this package [y,n,?,q]y

Removing installed package instance <tcp>

Verifying package dependencies.

WARNING:

```

The <nfs> package depends on the package currently
being removed.

```

WARNING:

```

The <Stulc> package depends on the package currently
being removed.

```

WARNING:

```

The <Sifmlan> package depends on the package
currently being removed.

```

WARNING:

```

The <Silvs> package depends on the package currently
being removed.

```

Dependency checking failed.

Do you want to continue with the removal of this package [y,n,?,q]n

Removal of <tcp> was terminated due to a user request.

No changes were made to the system.

Press ENTER to continue

This check mechanism allows you to intervene in time to prevent deinstallation of a package on which other packages depend.

4.6 Displaying information

You can use the *SYSADM* user interface to display information you require in relation to software. Two menu items are provided for this purpose: *software* and *software_prod*. [Table ""](#) shows the information you can call up via the two *SYSADM* menu items.

4.6.1 SYSADM – software_prod menu item

If you require information on add-on products, you can use the *SYSADM software_prod* menu item. Then switch to the [Function menu](#). A description of how to proceed for this can be found in [Section "Installing software products"](#) or in [Section "Deinstalling software products"](#).

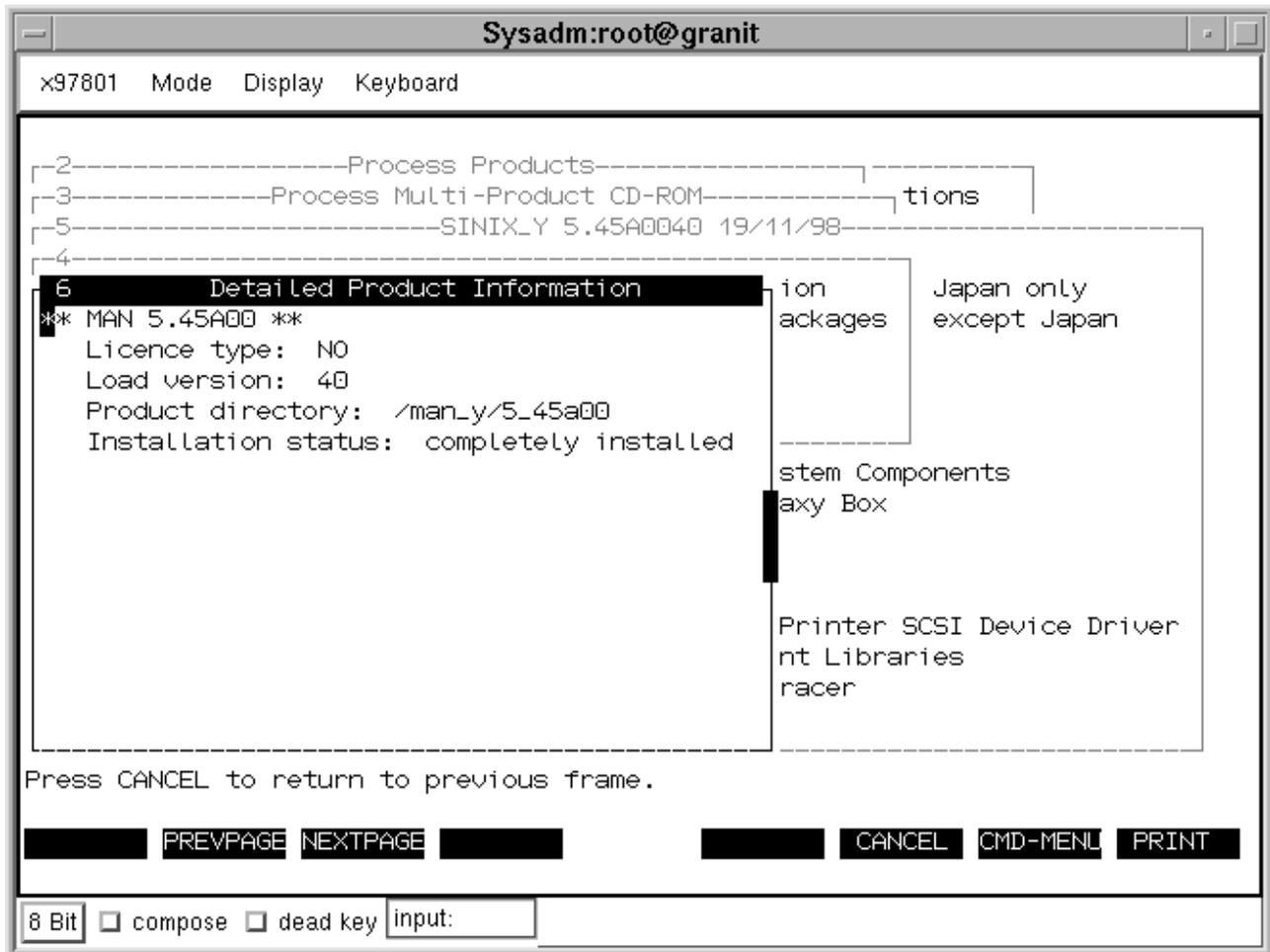


You can use the [NEXTPAGE] and [PREVPAGE] function keys to scroll in the subsequent *SYSADM* windows. Press the [CANCEL] function key to terminate the display.

You can also print the displayed information out. To do this:

- ▶ Define a suitable print command (see the [Section "Printing information"](#)).
- ▶ Press the [PRINT] function key.

Example of the *Display Detailed Product Information* menu item:

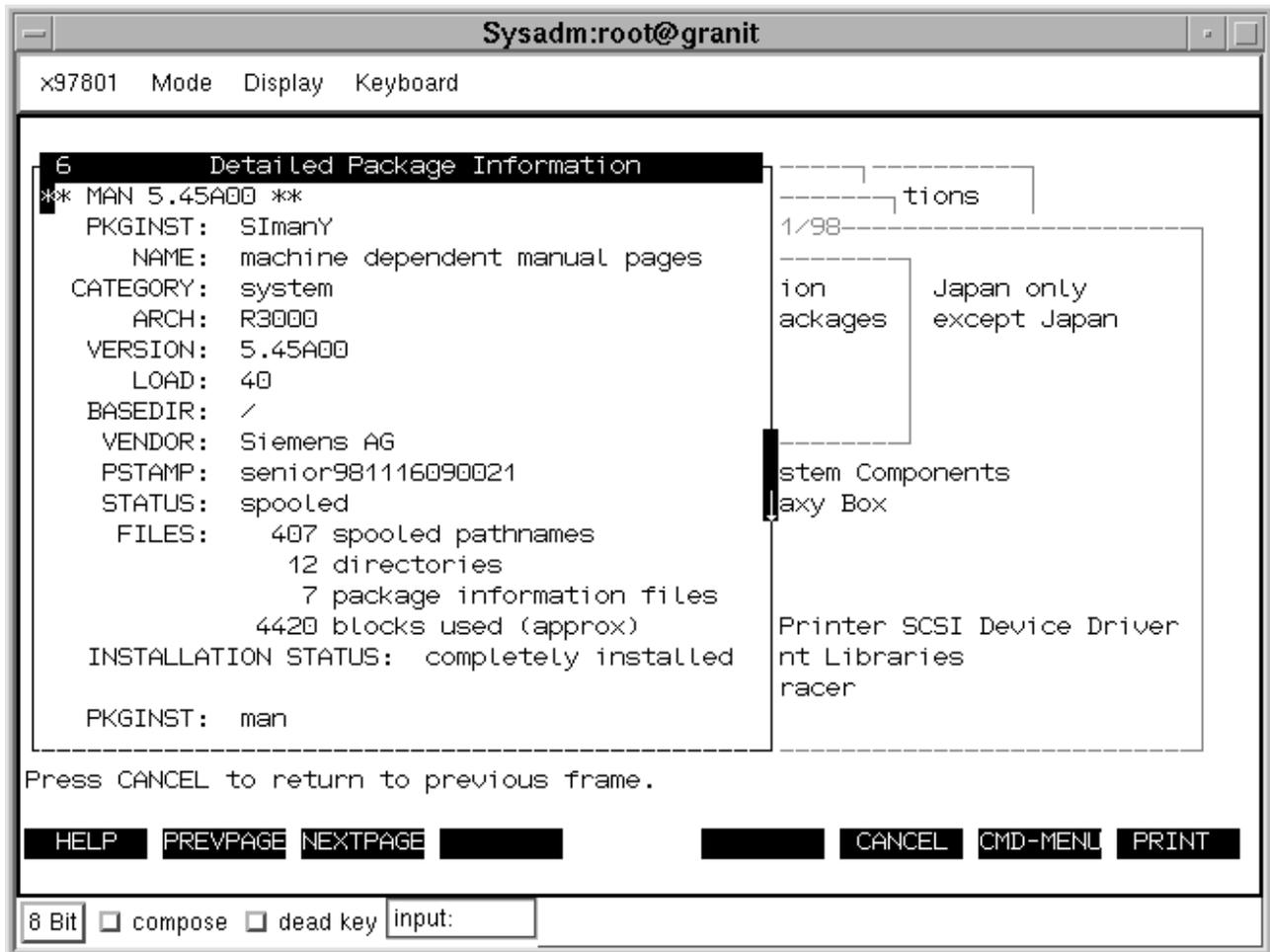


The following information is displayed for all products you marked in the **Product list** (see ... or ...): product name and version, type of licensing, production state, product directory and installation status.



The "OTHERS" pseudo product contains all packages not belonging to an add-on product. This includes, for example all system software packages. No detailed product information is available for the pseudo product.

Example of the *Display Detailed Information on Packages* menu item:



The following information is displayed for all products you marked in the **Product list** (see ... or ...):

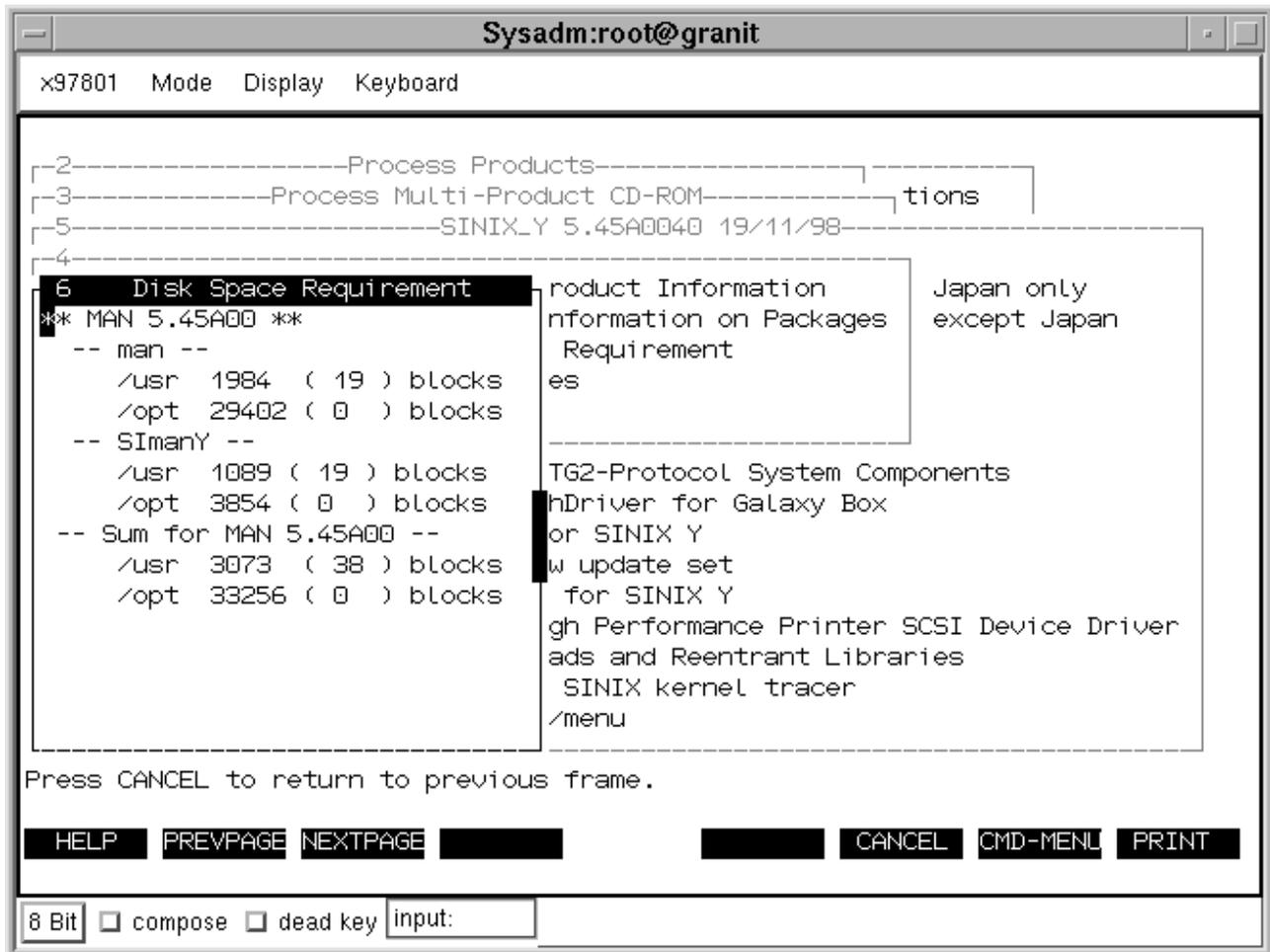
- Detailed information on the installed packages
- A list of all non-installed packages of the product
- The status of the package containing the *README* file (*S!readmeM* package)



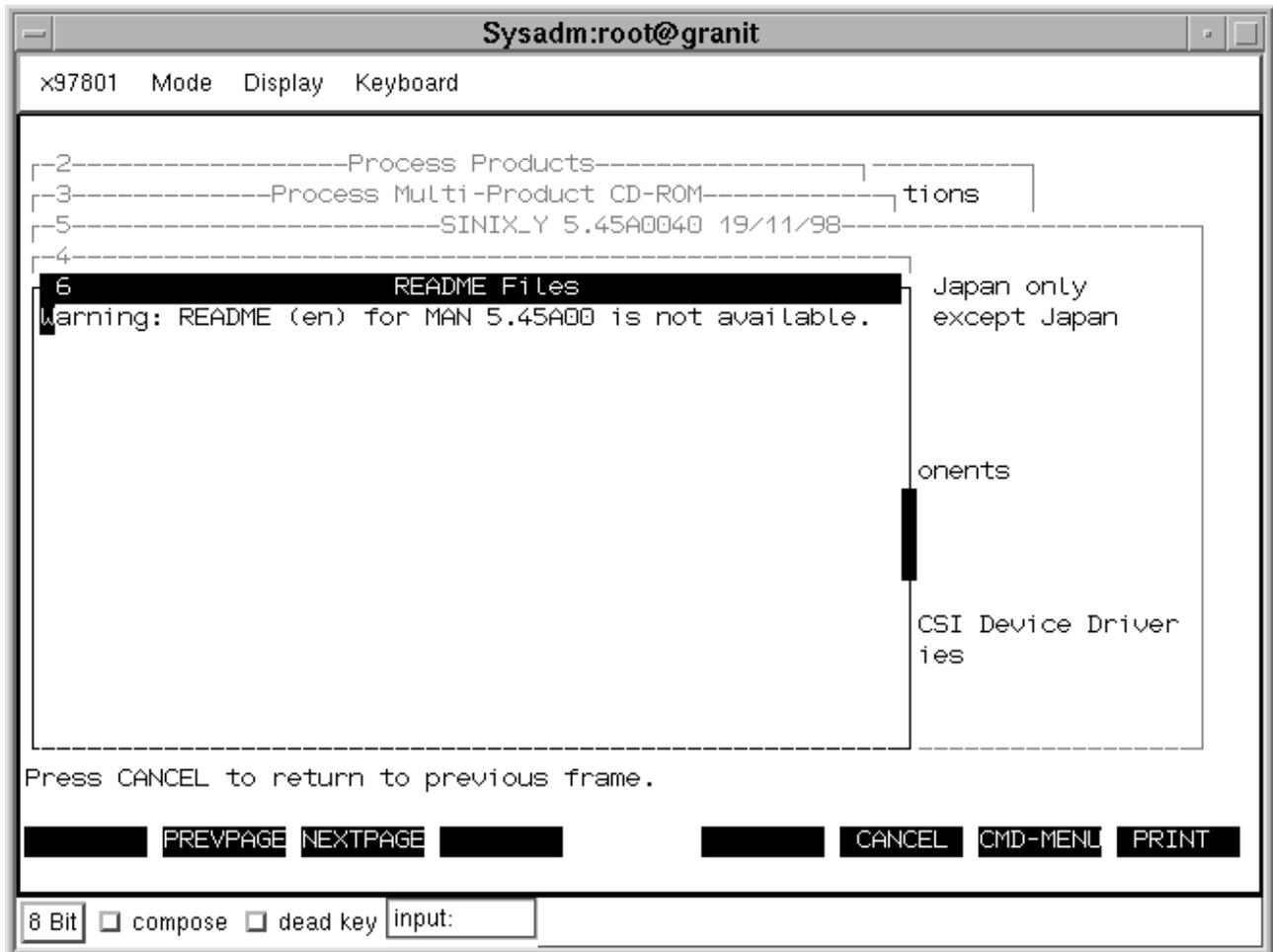
The displayed information corresponds to that displayed via the *SYSADM* software — *list* — *Amount of information to display* — *long* menu item (see ...).

A summary of all packages of the products you marked in the **Product list** (see ... or ...) can be found in the **Package list** (see ... or ...).

Example of the *Display Disk Space Requirement* menu item:



Example of the *Display README Files* menu item:



4.6.2 SYSADM – software menu item

Proceed as follows if you require information on single software packages (e.g. on system software packages):

- ▶ Start *SYSADM*.

The main menu appears. The following figure provides an overview of the remaining program execution.

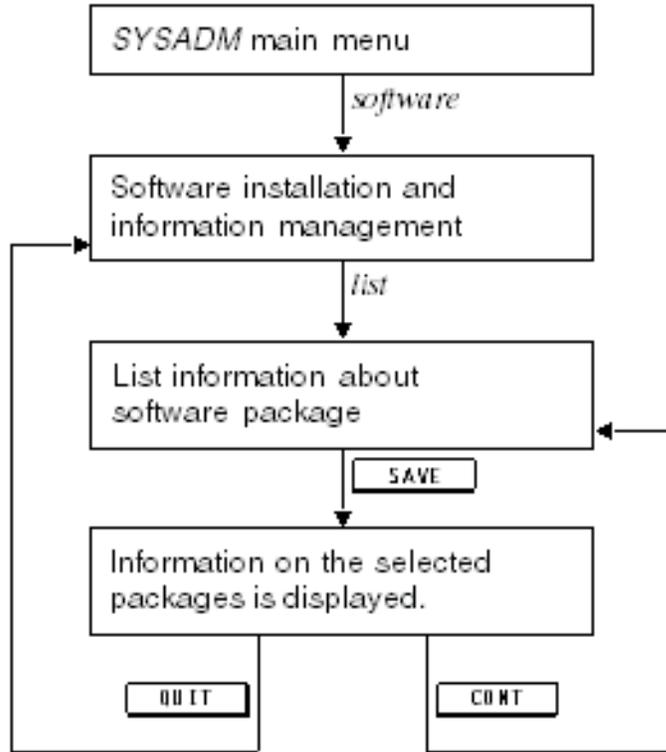
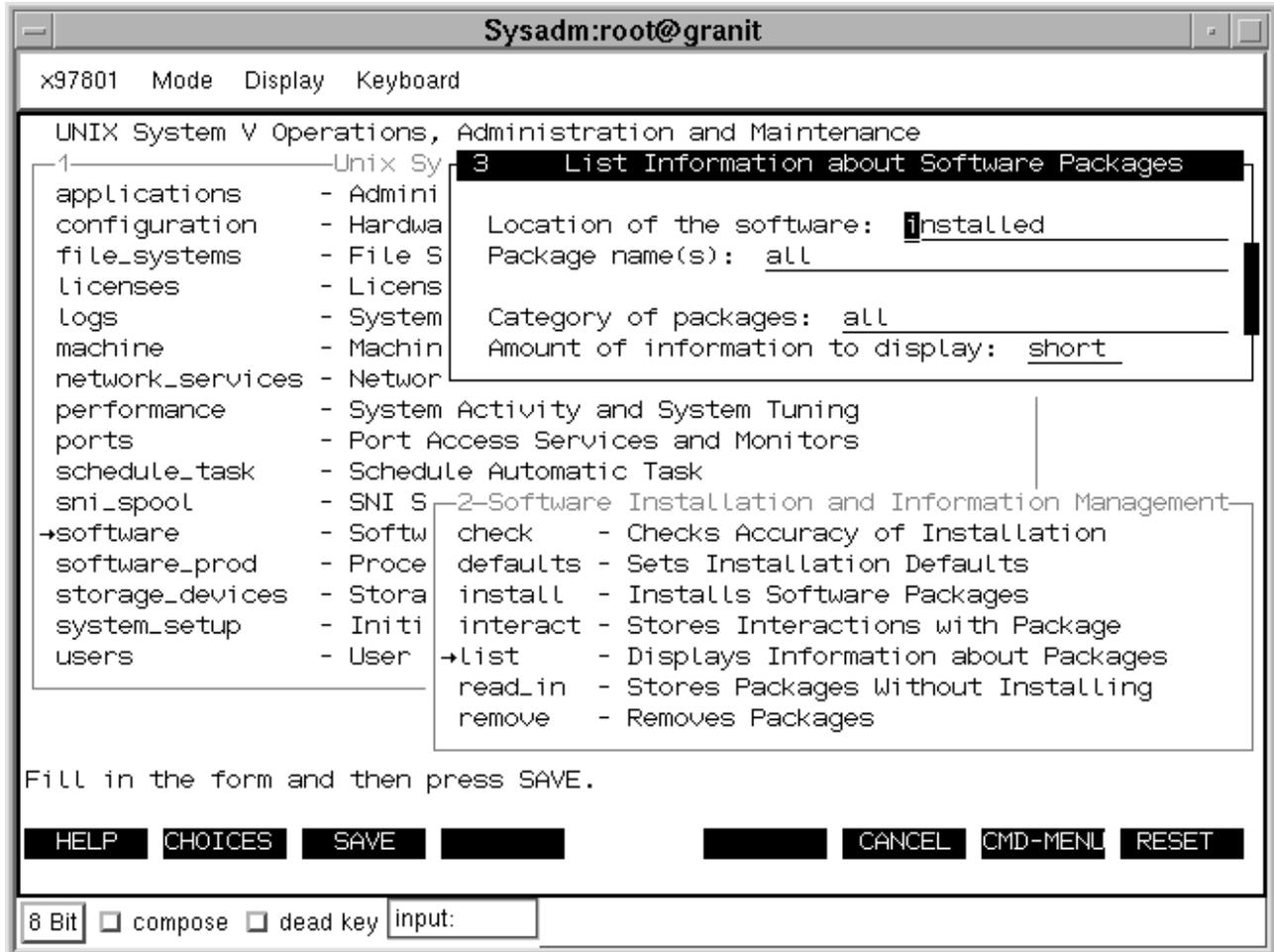


Figure 19: Overview of SYSADM – Software — List

- ▶ Select the *software* menu item.
The *Software Installation and Information Management* menu appears (see ...).
- ▶ Select the *list* menu item.
The *List Information about Software Packages* form appears (see ...).

**Location of the software**

Specify *installed* if you only wish to display software installed on the system disk.

Package name(s)

Specify *all* if you require information on all packages. If you only require information on single packages, input their names separated by a comma.

Category of packages

Specify one or more categories: *system, application, optional*. Specify *all* for all categories.

Amount of information to display

The *short, medium, long* options are available here.

- ▶ Fill in the form.
- ▶ Press the [SAVE] function key.

Example of *Amount of information to display installed, all, all, short:*

```

Sysadm:root@granit
x97801  Mode  Display  Keyboard

UNIX System V Operations, Administration and Maintenance
-----1-----Unix Sy  --3-----List Information about Software Packages-----
4 Packages Installed on the System
system      SIX11fml    SINIX/windows - X11R6 servers -
Administration menus
system      SIX11srv    SINIX/windows - X11R6 servers
system      SIacc       Sinix SCSI Autochanger Device Driver
system      SIackey     Ansi console keytables and fonts
system      SIadmcon    Console keyboard configuration
system      SIboed      Sinix Object Editor Basis
system      SIboedM     Sinix Object Editor Basis Messages
system      SICdin      Sinix Multi-product CD-ROM Installation
system      SICdinM     Sinix Multi-product CD-ROM Installation / Germ.
Suppl.
system      SICdrom     Sinix High Sierra and IS09660 File System
system      SICmxstub   CMXstub Driver
system      SICOed     Sinix Object Editor Character Interface
system      SICOedM    Sinix Object Editor Character Interface
Messages
system      SICompat    Sinix compatibility package

-----
Press CONT to repeat this task or CANCEL to return to the menu.

QUIT  CONT  PREVPAGE  NEXTPAGE  CANCEL  CMD-MENU

8 Bit   compose  dead key  input:

```

The following information is displayed for all packages: category, package name and brief description.

You can use the [NEXTPAGE] and [PREVPAGE] function keys to scroll in the text windows. Press either the [CONT], [QUIT] or [CANCEL] function key to terminate the display.

Example of *Amount of information to display installed, all, all, medium*:

```

Sysadm:root@granit
x97801  Mode  Display  Keyboard
UNIX System V Operations, Administration and Maintenance
1-----Unix Sy--3-----List Information about Software Packages-----
4 Packages Installed on the System
SIX11fml      SINIX/windows - X11R6 servers - Administration menus
              (R4000) 1.2A00
SIX11srv      SINIX/windows - X11R6 servers
              (R4000) 1.2A00
SIacc         Sinix SCSI Autochanger Device Driver
              (R3000) 5.43B00
SIackey       Ansi console keytables and fonts
              (R3000) 2.0A10
SIadmcon      Console keyboard configuration
              (R3000) 5.43B0050
SIboed        Sinix Object Editor Basis
              (R3000) V543B00
SIboedM       Sinix Object Editor Basis Messages
              (R3000) V543B00
SIcdin        Sinix Multi-product CD-ROM Installation
              (R4000) 2.1A00
SIcdinM       Sinix Multi-product CD-ROM Installation / Germ. Suppl.
-----
Press CONT to repeat this task or CANCEL to return to the menu.
QUIT  CONT  PREVPAGE  NEXTPAGE  CANCEL  CMD-MENU
8 Bit   compose  dead key  input:

```

The following information is displayed for all packages: category, package name, brief description, structure and version.

Example of *Amount of information to display installed, all, all, long*:

```

Sysadm:root@granit
x97801  Mode  Display  Keyboard
-----
UNIX System V Operations, Administration and Maintenance
1-----Unix Sy--3-----List Information about Software Packages-----
4-----Packages Installed on the System-----
  PKGINST: SIX11fml
    NAME: SINIX/windows - X11R6 servers - Administration menus
  CATEGORY: system
    ARCH: R4000
  VERSION: 1.2A00
    LOAD: 04
  VENDOR: Siemens Nixdorf Informationssysteme AG
  PSTAMP: coburg960913163741
  INSTDATE: Oct 01 1996 10:32 AM
  STATUS: completely installed
  FILES:   45 installed pathnames
          16 shared pathnames
          22 directories
          21 executables
          96 blocks used (approx)

  PKGINST: SIX11srv
-----
Press CONT to repeat this task or CANCEL to return to the menu.
  QUIT  CONT  PREVPAGE  NEXTPAGE  CANCEL  CMD-MENU
8 Bit   compose  dead key  input:

```

Detailed information is displayed for all packages.



This information corresponds to that displayed via the *SYSADM software_prod*—*products_on_system*—*pkginfo* menu item (see ...).

4.7 Installing the key diskette

Some add-on products and software packages require a key diskette containing a special authorization file. You are informed if this applies when you order the product concerned. You can call up this information in case of doubt via

- the *software_prod*—*info* menu item (see ...)
- the *software_prod*—*README* menu item (see ...)

You should install the key diskette **before** installing the corresponding add-on product or package. Two options are provided within *SYSADM* for this:

- The *software*—*install*—*Package location: floppy0 or mdens0HIGH*—*Package name(s): all*—*go* menu item (see also the [Section "Installing single software packages"](#))
- The *software_prod*—*miscellaneous*—*keydiskette* menu item

If you accidentally deinstall the package containing the authorization file, you will no longer be able to start any add-on products that require an authorization file. If this happens, simply reinstall the key diskette.

4.8 Managing user profiles (RM200 - RM400)

As already mentioned in the [Section "User profile \(RM200 - RM400\)"](#), the scope of the system software can be customized according to individual requirements. Three user profiles are available for this purpose:

- Full System
- Core System (can be extended to support German)
- Application Client System (can be extended for language and developer support)



If the application client system was extended to provide developer support, it is to be addressed as "Developer Client System" in the user profile management.

A concrete user profile was defined during the new installation. If, while working on the computer, you find that you need additional software from the Reliant UNIX basic system, you can subsequently install the necessary groups or packages using *SYSADM*. The same procedure applies when you have installed too much software. The less system software you have installed on the system disk, the less space is required for it in the individual partitions. This saves space, which can then be used for other purposes.

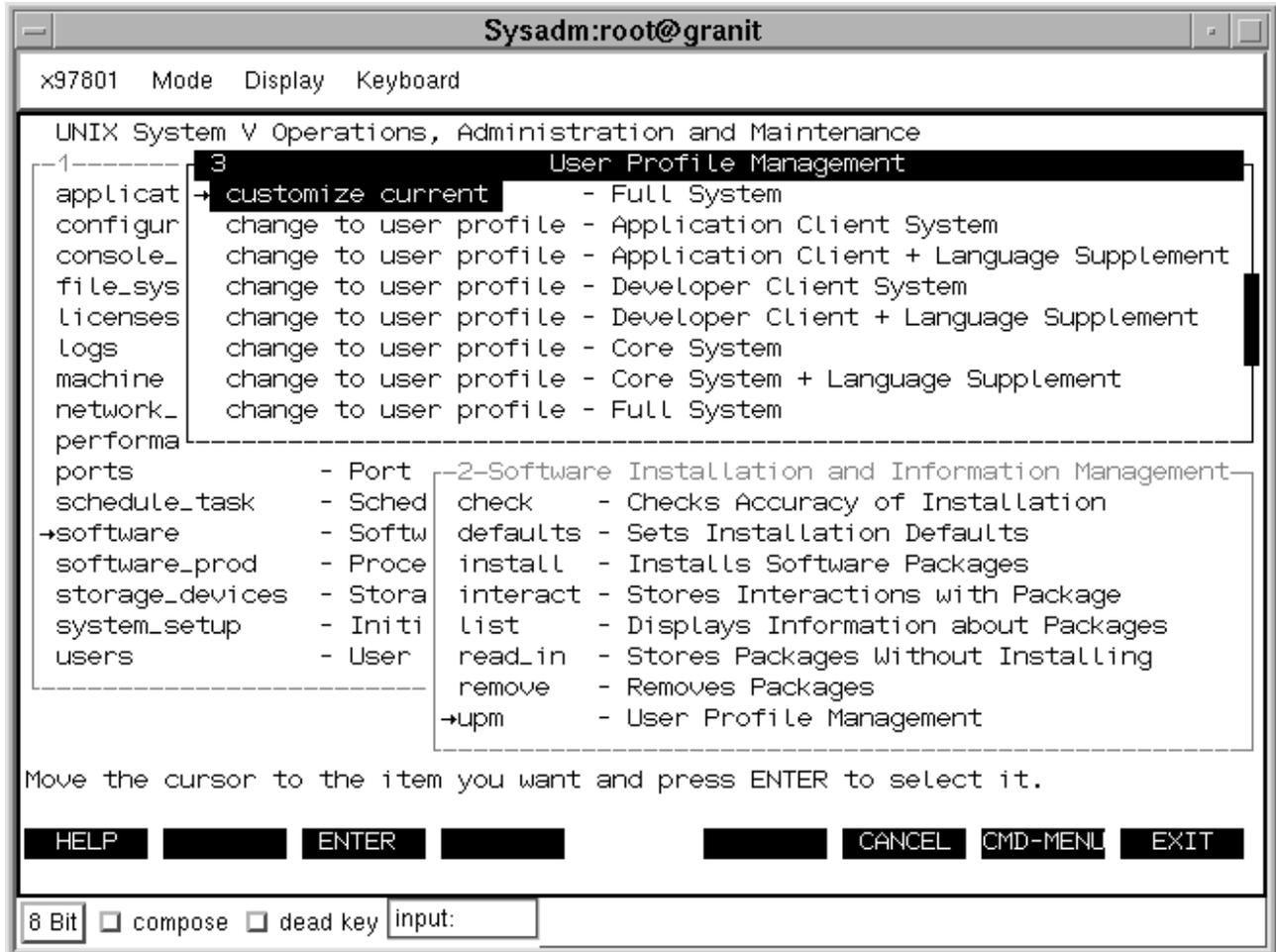


It is only possible to install additional system software if the system disk partitions are larger than the minimum values specified in the manual "Reliant UNIX Installation".

If you want to modify the current user profile, you have to:

- ▶ Select the *Software* menu item.
The *Software Installation and Information Management* window is displayed.
- ▶ Choose the *Management* menu item.

A menu appears showing the possible user profiles:



customize current Currently installed user profile.

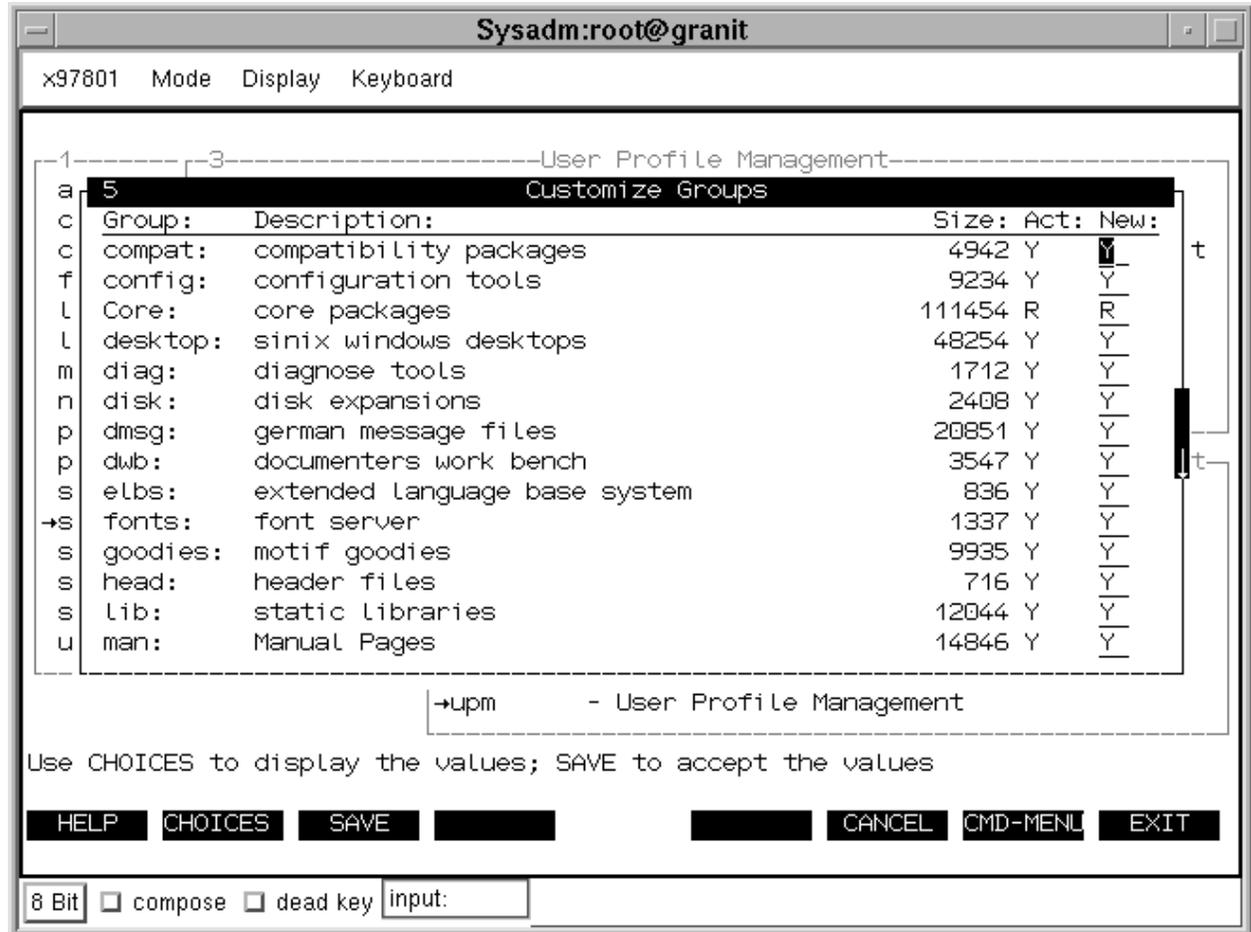
change to New user profile.

- ▶ Select the desired user profile.
- ▶ Press the [ENTER] function key.

You can then decide in a further menu whether the selected user profile is to be installed in its default form (*install* menu item) or whether you want to modify it (*customize* menu item).

- ▶ Select the desired menu item.
- ▶ Press the [ENTER] function key.

If you choose the *customize* menu item, another window is displayed:



This window contains a list of the package groups that are currently installed, as well as the package groups that also belong to the selected user profile. The *Act* column contains one of the following entries:

- N* not installed
- P* particular installed
- R* installed and protected; protected package groups cannot be deleted
- Y* installed

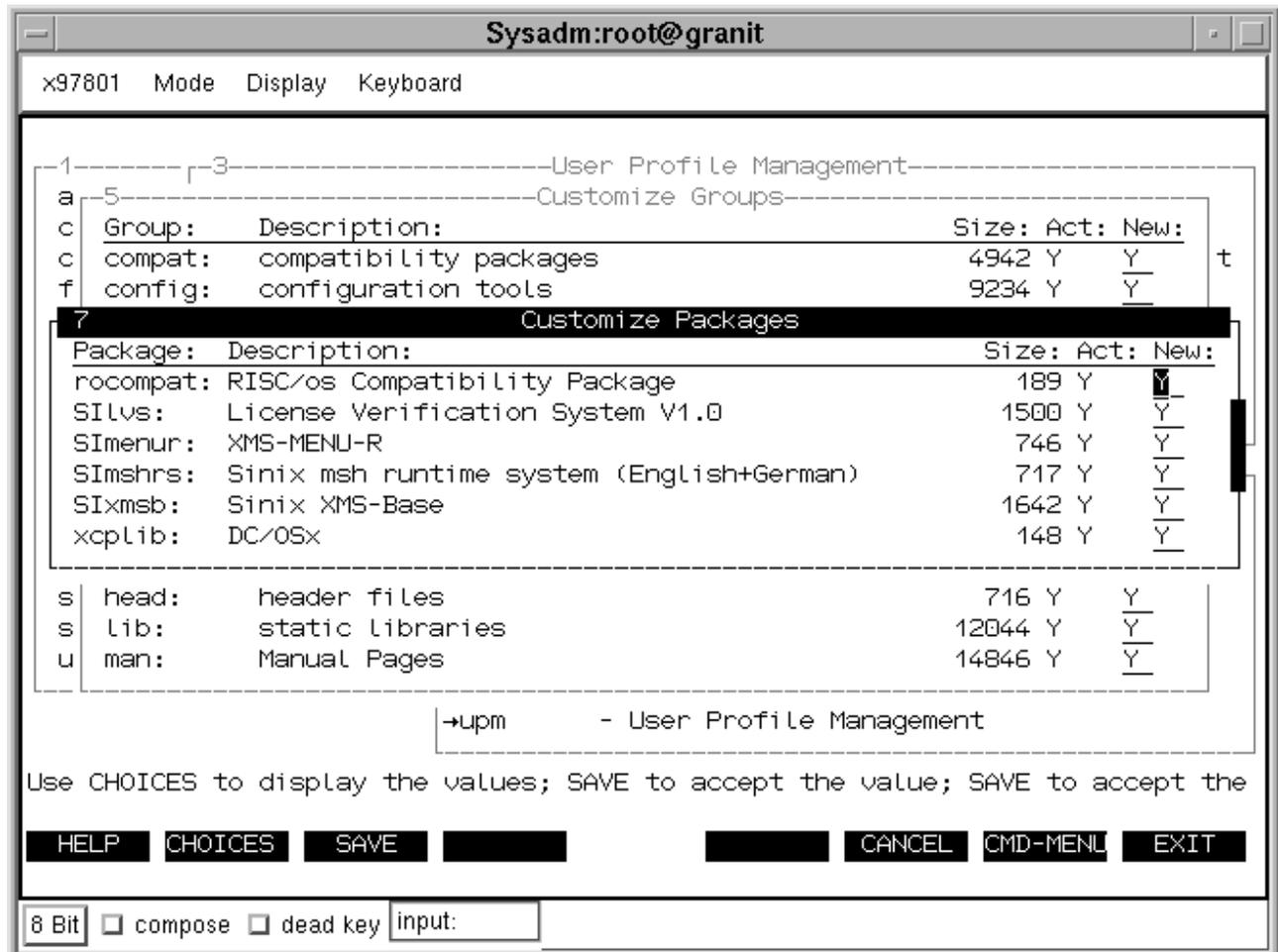
In the *New* column, you can specify which non-protected package groups are to remain or be installed (*Y*) and which are to be deleted or are not to be installed (*N*).

- ▶ If you want to delete non-protected package groups or exclude new package groups from the installation, move the cursor to the relevant row and enter either *Y* or *N*.

- ▶ If you do not want to execute the selected action on individual packages, press the [CHOICES] function key instead.

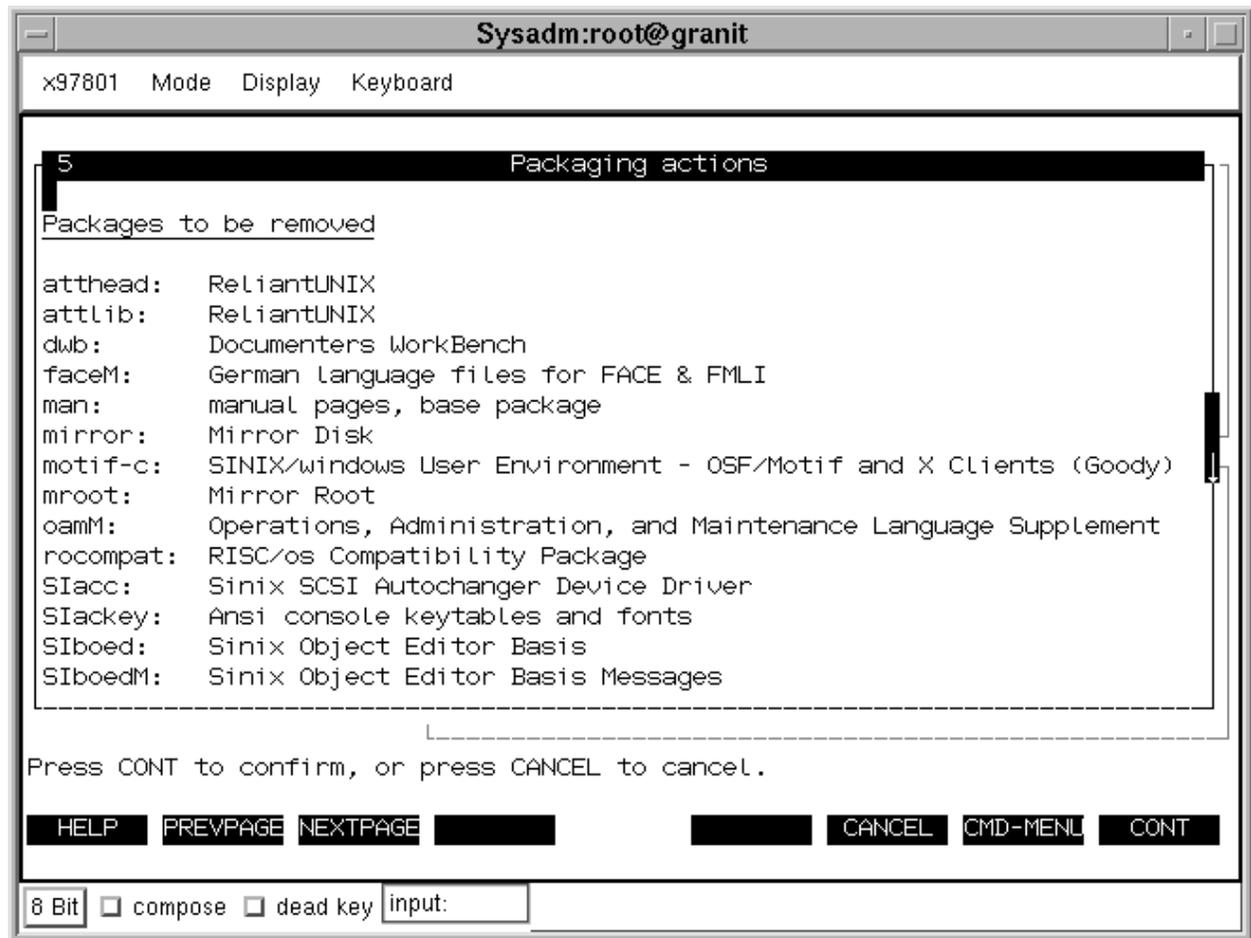
Another menu is displayed.

- ▶ Select the *customize* menu item.
- ▶ Press the [ENTER] function key. □
The following window now appears:



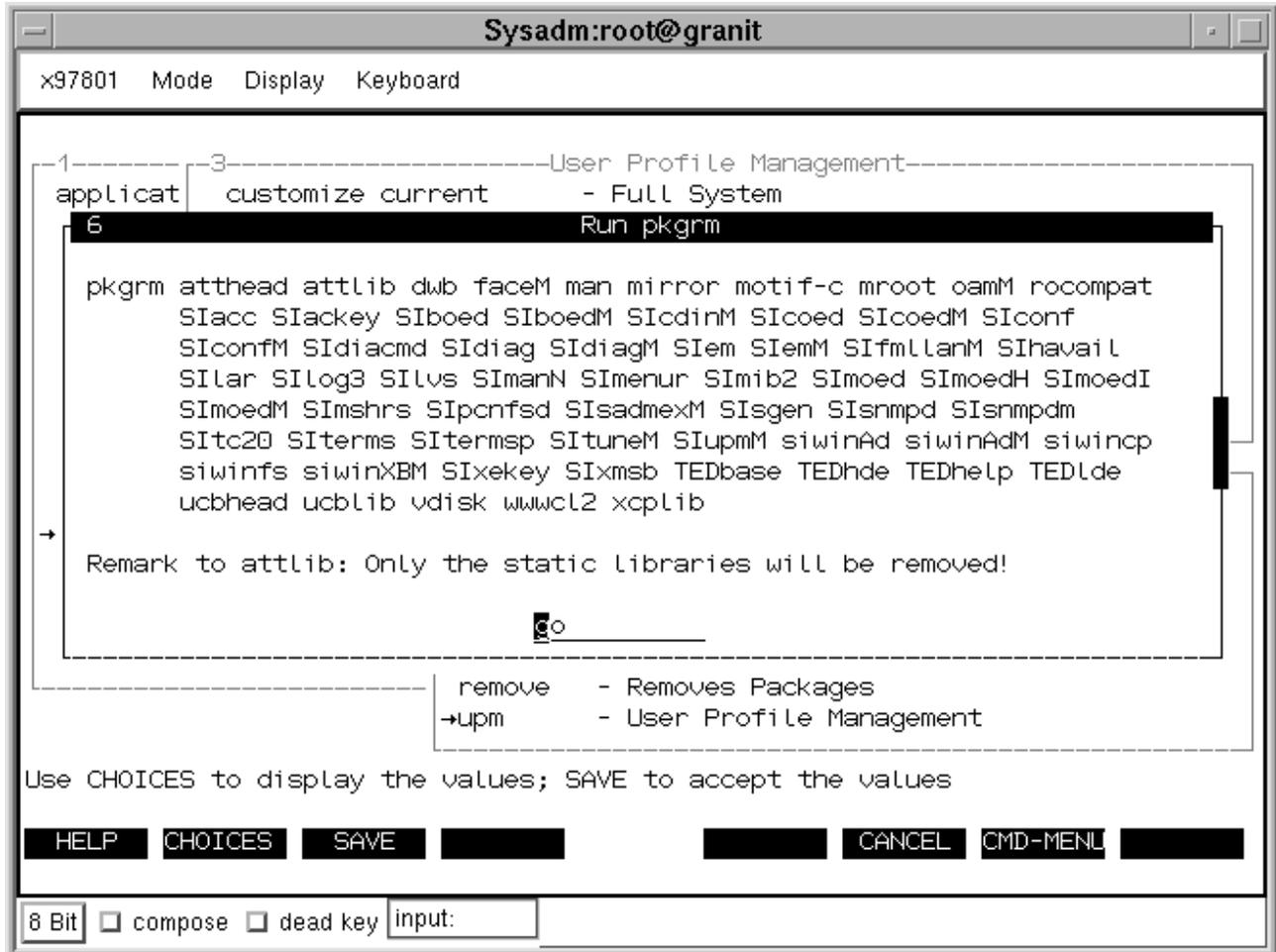
- ▶ If you want to delete individual packages or exclude them from the installation, move the cursor to the relevant row and enter either *Y* or *N*. You can also use the [CHOICES] function key for this purpose.
- ▶ Then press [SAVE]. □
The *Customize Groups* window appears again.
- ▶ If the settings are now correct, press [SAVE] again.

The following window is then displayed:



This window once again contains a list of all the packages that are to be deleted or installed.

- ▶ Check carefully to ensure that the correct package groups and packages are displayed and that the correct actions (delete and install) are specified.
- ▶ If the settings are correct, press the [CONTINUE] function key.
- ▶ Before the packages are deleted or installed, you must confirm this action again:



- ▶ If the settings are correct, enter *do*. Otherwise, enter *cancel*. You can also use the [CHOICES] function key for this purpose.



Data will be deleted on your system disk in the following cases:

- If you have selected package groups or packages to be deleted in the windows displayed earlier.
 - If you choose *do* in this window and then press [SAVE].
- ▶ If the settings are correct, press [SAVE].

5 Node administration with WebSysAdmin

WebSysAdmin (for Web-based System Administration) is the new user interface for administrators working on RM systems. It offers greater ease of use and more efficient administration even in large system networks. Used in conjunction with the add-on product *DomainAdmin* it is even possible to manage several systems using *WebSysAdmin*. More detailed information can be found in the "System Administration within a Domain (Reliant UNIX)" manual.

WebSysAdmin comprises the following components:

- *Accounts*

All of the functions are available here for managing users and groups, for example creating, modifying and deleting user and group definitions or managing user passwords.

- *VConfig*

This component includes functions for configuring and administering the hardware, as well as individual computer nodes and administration domains.

- *Diagnostic*

The *Diagnostic* component includes functions for evaluating log files and managing events across the entire domain. You will find detailed information in the chapter "Evaluating and managing log books" in the manual "Reliant UNIX – Troubleshooting and Diagnostics".

- *Tasks*

The *Tasks* component makes it possible for the *root* user to manage task definitions for both local computers and for all the nodes in a domain. Task definitions correspond to *cron* jobs and thus to entries in the file *crontab*.

- *Software*

This component allows management of the installed software packages on all nodes of a domain or on the local computer.

More detailed information about this point can be found in the "System Administration within a Domain (Reliant UNIX)" manual.

The components can be used intuitively thanks to the self-explanatory user interface. Nevertheless, should questions arise, the integrated help text should provide the answers. We have therefore kept detailed explanations to a minimum here and have limited ourselves to the key elements.

5.1 Configuration requirements

Hardware and software configuration when using a PC

- PC or VGA console as the output device
- Microsoft® Windows 9x/NT™ operating system
- JAVA-enabled browser (recommended: Netscape Communicator 4.5 or above)
- JAVA Plug-In (automatically offered for downloading)

Hardware and software configuration when using a LAN console

- LAN console as the output device
- Solaris™ operating system
- JAVA-enabled browser (recommended: Netscape Communicator 4.5 or above)
- JAVA Plug-In (automatically offered for downloading)

5.2 Starting WebSysAdmin

WebSysAdmin can be called in both write mode and read mode. You must remember that the interface can only be called a write mode by a single user at any one time. If any additional users try to start *WebSysAdmin* on the

same server, then it is only possible to call the system in read mode.

5.2.1 Starting WebSysAdmin under Windows 95/98/NT (PC console)

- ▶ Start your browser.
- ▶ To call the user interface in **Write mode** enter the following in the URL line:

`http://computer-name:8881`

For example, :

`http://cosmos.pdb.siemens.de:8881`

The user interface is loaded from the server and the start screen will appear on your monitor (see ...).

If you wish to call the user interface in **Read mode** enter the following:

`http://computer-name:8882`

- ▶ Select the dialog language (default is English) and then click the icon of the component you require (see below).
- ▶ A dialog box appears in which you are asked to identify yourself. Enter your user ID and your password.



Figure 20: The WebSysAdmin start screen

- ▶ Now enter the root password.
You can now begin your administration work.



WebSysAdmin requires a minimum screen resolution of 1024x768, in order to display information in a way that can read easily.

5.2.2 Starting WebSysAdmin under Solaris 7 (LAN console)

To start the user interface in **Write mode**, enter the following at the command level:

```
# /opt/SMAW/bin/wsa computer-name
```

The start screen will then appear and you can continue as described in [Section "Starting WebSysAdmin under Windows 95/98/NT \(PC console\)"](#) (setting the language, selecting the component etc.).

To start the user interface in **Read mode** enter the following:

```
# /opt/SMAW/bin/wsa -readonly computer-name
```

Carry on as described (setting the language, selecting the component etc.).

5.2.3 Starting WebSysAdmin on a SPARC system (VGA console)

The procedure for starting the user interface as described in [chapter 2](#), also applies when using a VGA console on a SPARC system.

5.3 Introduction to VConfig

VConfig is a component of *WebSysAdmin* and is used to configure and manage the hardware of individual RM systems as well as an entire administration domain.

VConfig offers you the following advantages:

- Menu-guided modification of configuration data.
- Consistency checking of changes, which helps avoid errors.
- Modification of all affected system files on the node with *VConfig* and instigation of all processes required to operate the hardware.

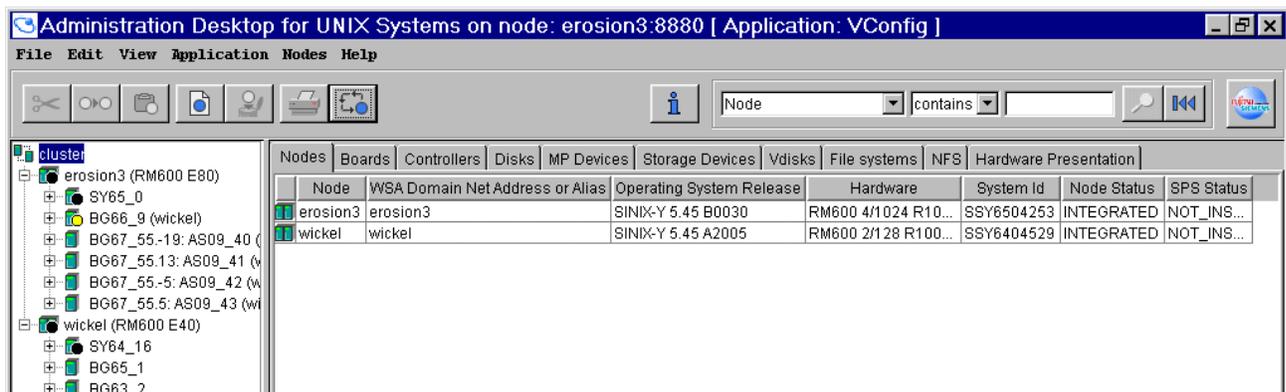


Figure 21: Example: VConfig component window (diagram similar)

5.3.1 Functionality of VConfig

VConfig allows you to perform the following actions, for example:

- Management of the installed cabinets, boards and storage devices
- Display and, if possible, modification of device attributes
- Analysis of installed hardware
- Recognition of defective power supplies, fans, boards and hard disks
- Manual deactivation and subsequent reactivation of boards and hard disks
- Online replacement of defective hard disks

- Online installation of new hard disks
- Setting up and modification of virtual disks
- Creation of SPS configurations
- Management of multipath capable devices

5.3.2 User interface elements

The user interface essentially comprises the following elements:

Menu bars and toolbars

Here you are able to carry out a range of different actions. Some of these actions are context-sensitive and cannot be carried out at all locations. The context-sensitive actions are deactivated on screen when the option is not available.

Menu bar	Toolbar	Action
<i>File</i>		Output object list
	–	Export object
	–	Terminate session
<i>Edit</i>		Cut object
		Copy object to clipboard
		Insert object
<i>View</i>		Refresh display
<i>Application</i>	–	Change to a different application
Application-specific menus (the selected tab appears, for example, <i>Node</i>)		Create object
	–	Modify object
		Delete object
	–	Copy object to another computer within the administration domain
	–	Other application-specific actions
<i>Help</i>	–	Overview (application-specific)
	–	Contents (application-specific)
	–	About ... (version information)
–		Display system information

		
-		Selection list: Select object
-		Selection list: Undo object selection
-		Interrupt loading process

Selection configurator

The selection configurator is part of the toolbar, this can be used to find objects using the search algorithms of the regular expressions. This function allows you to restrict the number of hits.



Figure 22: Example of a selection configurator

Hierarchy browser

The hierarchy browser assists navigation and allows you to carry out the following actions

- within an administration domain, to select either the entire domain or the individual computers within a domain,
- on a local computer, to select individual hardware components in the VConfig component.

In accordance with the selections you make in the hierarchy browser, you will be shown a list of objects (see the section "**Object table**" below) which takes into account the entire administration domain or the selected computer.

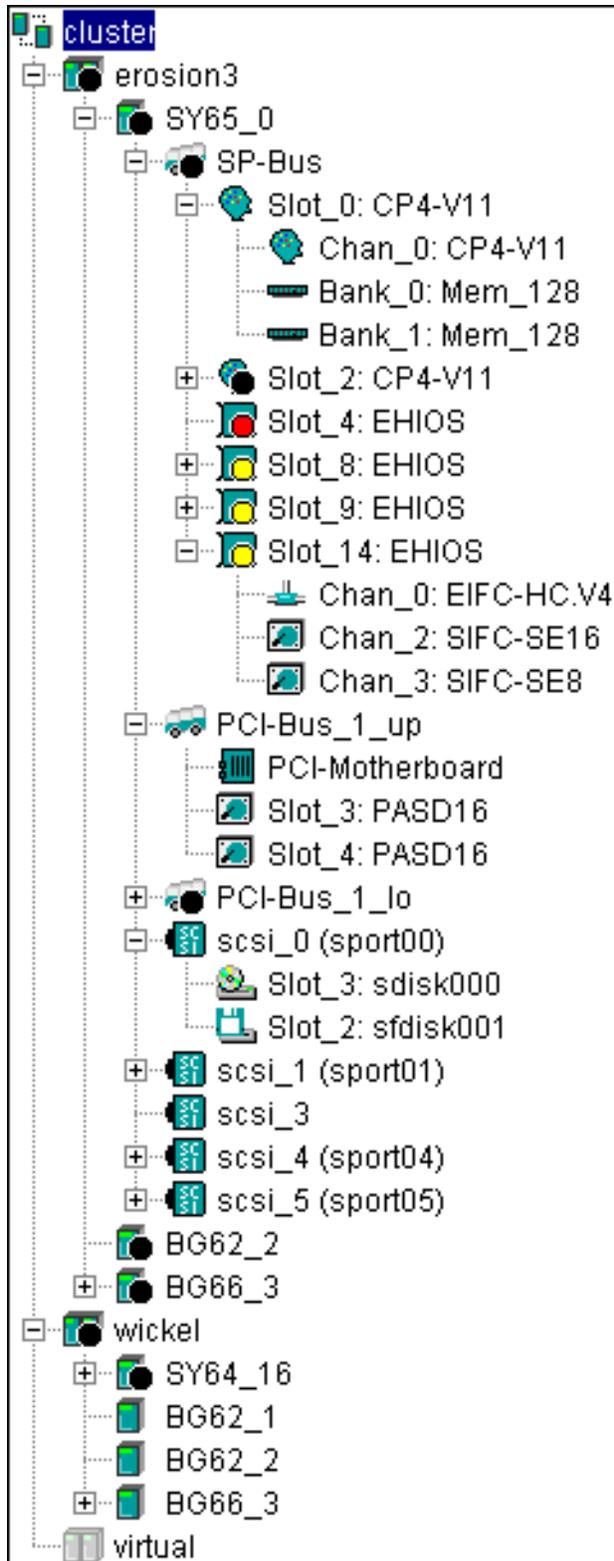


Figure 23: Example of the hierarchy browser of an administration domain

Object table

The object table covers the following areas:

- The upper part consists of the **tab bar**. The type and number of tabs available will depend on the application you have selected.

Nodes	Boards	Controllers	Disks	MP Devices	Storage Devices	Vdisks	File systems	NFS	Hardware Presentation
-------	--------	-------------	-------	------------	-----------------	--------	--------------	-----	-----------------------

Figure 24: Example of a tab bar

- The **object list** can be found below the tabs. The type and number of attributes shown will depend on the tab/sub-tab that you have selected (see below) and on your selection in the hierarchy browser and the selection configurator within the toolbar.

	Node	WSA Domain Net Address or Alias	Operating System Release	Hardware	System Id	Node Status	SPS Status
	erosion3	erosion3	SINIX-Y 5.45 B0030	RM600 4/1024 R10...	SSY6504253	INTEGRATED	NOT_INS...
	wickel	wickel	SINIX-Y 5.45 A2005	RM600 2/128 R100...	SSY6404529	INTEGRATED	NOT_INS...

Figure 25: Example of an object list

The first column of the object list contains the same information about the status of the individual objects as is shown in the hierarchy browser (see above).

- The lower part of the display may contain a **sub-tab**. The existence of this tab and the type and number of sub-tabs available depends on the tab you have selected.

Selection options

In order to make sure that you don't lose your way in the object list when you have large configurations with several hundred entries, it is possible to reduce the volume of lines displayed. This can be done:

- via the hierarchy browser (see ...)
- via the selection configurator (see ...)

The administrator is thus able to filter out the information that is important to him or her.

5.4 Special features of administration domains

This chapter deals with the relationship between *WebSysAdmin* and administration domains. Both underlying factors and the functionality of the *DomainAdmin* software product will be addressed. More detailed information can be found in the "System Administration within a Domain (Reliant UNIX)" manual.



DomainAdmin is a licensed product and may only be installed if you hold a valid license!

General observations

When you connect several computer systems using a local network as the communications medium, this is termed a **computer network**. The systems and media have different architectures and sometimes are even from different manufacturers. The common element in this computer network is the communications protocol which is used to control the exchange of data.

In contrast to a computer network, an **administration domain** is also a network of computers. But in this case the computers are more closely linked to one another. The operating systems and configurations of the individual computers are also interlinked by means of a global administration system (*DomainAdmin*).

Advantages of forming administration domains

Considerable improvements can be achieved in many areas by forming administration domains, for example:

- "Single Point of Administration"

All the nodes of a domain are managed from a single administration workstation and must no longer be operated individually from the various different system consoles.
- Extensibility

A computer node and its resources can be incorporated in and removed from the administration domain at any time. A special command is issued to the administration system (*DomainAdmin*) after the operating system has booted. Each node can then survive independently as a Reliant UNIX system and can be incorporated in or removed from the network by means of a command from the global administration system.
- Transparency

Since an administration domain appears like a single system ("Single System View") to the user, a network of nodes can be administered in the same way as a single operating system. Regardless of the node on which an administrator starts the user interface, he or she can access disk media and other peripherals in the entire administration domain at any time.
- Ease of maintenance

A node can be removed temporarily (deactivated) from the administration domain for maintenance and diagnostics purposes. This allows diagnostics and maintenance to be carried out easily.
- Diagnostics

It is possible to evaluate the log books of all nodes in the administration domain from a single central workstation.

From this central workstation, it is also possible to configure the diagnostics system specifically for each node.

The domain-wide administration system

In order to make it possible to manage several systems simultaneously within a domain, Reliant UNIX as of release 5.45 has been extended by the add-on product *DomainAdmin*. This must be installed on all those nodes that are to make up the administration domain. The LAN connections available between the systems enable system-wide administration.

The product *DomainAdmin* has all the functions and services that make it possible to administer and control an administration domain. It is a global administration system which coordinates all nodes in an administration

domain.

DomainAdmin resides with equal authorization on all participating nodes. It accesses services available on each node and which are implemented as operating system commands. DomainAdmin therefore accesses the actual operating system in the individual nodes.

5.4.1 Functionality of the administration system

DomainAdmin provides functionality used by *WebSysAdmin* to manage nodes and their resources. This allows *WebSysAdmin* to carry out the following activities within the node:

- Configuration management (the "VConfig" component)
This includes services required for configuring the administration domain itself, such as incorporating a new node in the network or removing a node from a network.
- Maintenance and diagnostics (the "Diagnostics" component)
This includes the domain-wide evaluation of log files.
- User management (the "User" component)
This includes services for setting up, logging on and off users.
- Software management (the "Software" component)
This includes installation and version checking.
- Task management (the "Tasks" component)
This includes functions for domain-wide setup and modification of tasks.

DomainAdmin and the WebSysAdmin user interface

DomainAdmin has been implemented for the first time in Reliant UNIX in Release 5.45. All administration functions are combined under the new common *WebSysAdmin* interface.

This provides many advantages in relation to administration domains:

- Faster overview of the entire administration domain
- Easy access to operations that could otherwise only be executed using complex commands
- "Live" feedback for actions

WebSysAdmin currently includes the applications *VConfig*, *Diagnostics*, *Accounts*, *Software* and *Tasks*. These applications provide important functions for the administrator. Detailed information about *VConfig* can be found in [Section "Introduction to VConfig"](#). The other components are described in the "System Administration within a Domain (Reliant UNIX)" manual.

6 UPSs in RM systems

Reliant UNIX supports the connection of uninterruptible power supplies (UPS). If the power supply fails, the UPS supplies the power for a certain time period. The maximum period for which it can fulfill this function depends on the battery capacity of the UPS model and on the number of devices connected. For further information, please refer to the operating manual for the UPS.

6.1 UPS on the RM200 - RM400

Under V5.42 and V5.43, the 9071 UPSs are only supported on the basis of **signal control**. Monitoring of the UPS signals is carried out by the UPS daemon. As of V5.44, **protocol control** is also supported for the 9071-Pxx6 UPS. In this case, the PUPS daemon is used.

You must install the *Slups* package if you want to work with signal control. The drivers from the *Spups* package are required for protocol control. Use the *SYSADM* user interface for installation. If you have not yet worked with *SYSADM*, you should refer to the brief introduction to the interface in [Section "The SYSADM user interface"](#).

Installing device drivers

- ▶ Insert the "Reliant UNIX 5.45" CD-ROM.
- ▶ Start *SYSADM*.
- ▶ Select the *software_prod* menu item (see also ...). The *Process_Products* menu appears.
- ▶ Select the *products_on_CD* menu item. The *Process Multi-Product CD-ROM* form appears.
- ▶ Fill in the mask. Save the mask by pressing the function key [SAVE] or [F3].
- ▶ Confirm the copyright notice. To do this, press the function key [CONTINUE] or [F8].
- ▶ Mark the *UPS* or *PUPS* package in the selection list. To do this, press the function key [MARK] or [F2]. Press the function key [ENTER] or [F3].
- ▶ Select the *Install* menu item. Press the function key [ENTER] or [F3].
- ▶ Start the installation. Press the function key [SAVE] or [F3].
- ▶ In reply to the question as to whether a new kernel is to be generated, enter n.
- ▶ Remove the CD-ROM from the drive.

6.1.1 UPS using signal control

Before you can start using a UPS, you will need to carry out some preparatory steps as described below.

Connecting the UPS

- ▶ Connect the cable from the UPS (COM 1) to the UPS interface you have set up on your RM system.
- ▶ Shut down the system unit and connect the UPS to your system cabinet (see the operating instructions for your computer) and connect the power cable of your system unit to the UPS output.
- ▶ Connect the UPS to the main power supply.
- ▶ Reboot the system unit.
- ▶ Check whether the default Port_1 (COM2) is free on the motherboard and activate it if necessary.



If you want to use Teleservice, the UPS must be configured for Port_0 (COM1). In this case, you must use the SVGA graphics console instead of any console that may be connected to Port0.

If you have connected the UPS to port 1, you can skip the next step and continue reading from "Activating the UPS function" (see ...).

UPS connected to port 0: Configure installation parameters

If you have connected the UPS to port 0, you need to configure a number of installation parameters and then generate a new system kernel.

- ▶ Edit the `/etc/conf/pack.d/ups/space.c` file and change the number of the physical V.24 port from 1 (default) to 0:

```
long upsport = PORT0;
```



The default value must only be changed from 1 to 0 if the UPS is actually connected to the physical V.24 port 0 (see above).



If you do not set the port correctly, problems arise when booting the system.

- ▶ Save the modified file and exit the editor.
- ▶ Enter the following command:

```
# termdel -T serial ups 2>/dev/null
# termadd -T serial -b motherboard0,5 -n ups -d 2>/dev/null
```

In this way you can ensure that the assigned port is not used inadvertently for terminal generation.



In the case of the RM200-C, configuration for PORT0 may be restricted. Please refer to the readme file.

Activating the UPS function

- ▶ Check whether the UPS function has already been activated. To do this, look at the `/etc/conf/sdevice.d/ups` file:

```
# cd /etc/conf/sdevice.d
# pg ups
```

Two output lines are displayed. The second field specifies whether the UPS function has been activated (*Y*) or deactivated (*N*). Both lines must have the same content. If the UPS function has not yet been activated, enter a *Y* instead of an *N* using an editor (e.g. *vi*).

The UPS parameters are explained below:

[1] *Time from line fail until start of shutdown procedure (in seconds)*

Default value: 20 Delay time in seconds, after which the system is shut down. Requirement: the *normal shutdown* is set (see menu item [3]).



The delay time set here only starts when the *powerfail* signal is received. The UPS sends this signal approximately 10 seconds after the power to the system has failed. This means that power failures of less than 10 seconds are **not** signalled. The restoration of the power supply is also signalled after approximately 10 seconds. If this occurs within the initiated delay time, the system is not shut down.

[2] *Time to wait until battery is loaded (in minutes)*

Default value: 120 Time in minutes that the system waits after restoration of power supply before it powers up again. The loading time should ensure that the system only powers up if the UPS battery has been fully loaded again. Requirement: the *safe boot* is set (see menu item [4]).

[3] *Shutdown behaviour (0 = normal shutdown, 1 = immediate shutdown)*

Default value: 1

Power failure behavior:

- | | | |
|---|---------------------------|---|
| 0 | <i>normal shutdown</i> | The system is only shut down if the delay time that was set under [1] has expired. |
| 1 | <i>immediate shutdown</i> | The system is powered down immediately without waiting for the delay time that was set under [1]. |



Change the value to 0.

[4] *Boot behaviour (0 = safe boot, 1 = temporary unsafe boot)*

Default value: 1 Behavior after restoration of power supply

- | | | |
|---|------------------------------|--|
| 0 | <i>safe boot</i> | The system is only powered up when the load time that was set under [2] has expired. |
| 1 | <i>temporary unsafe boot</i> | The system is powered up immediately, regardless of the loading time set under [2]. |

[5] *Quit*

Exit the menu. The displayed values are stored in NVRAM.

Testing the UPS monitoring function

There is one final test to perform. This test is perfectly safe. Even in the case of defective configurations, system operation is not interrupted. Carry out the following actions:

- ▶ Switch the switch on the front panel of the UPS from position 1 (inverter mode, the green INVERTER-LED lights up) to  (bypass mode, the yellow BYPASS-LED lights up). The following message appears on the system console after 10 seconds:

WARNING UPS-Bypass active (overload or defect in UPS)!

- ▶ Then switch the switch back again to the inverter position. After a period of 10 seconds, the same message as above appears.

MESSAGE UPS_Bypass not active!

If this message is **not** displayed, you must check carefully the steps "**Connecting the UPS**" and "**UPS connected to port 0: Configure installation parameters**".

- Is the UPS connected to the correct port? If not, plug the cable into another port and repeat the test. You can check the entry for the UPS with the command.

`termshow | grep ups`

- Was the kernel generated and booted correctly? If not, repeat the steps "**Installing device drivers**" and "**Connecting the UPS**".



When the UPS has been connected successfully, you must ensure that the switch on the UPS front panel is at **position 1** (INVERTER mode), otherwise the system will **not switch** to battery mode during a power failure.



If UPS monitoring has been activated, the message below may appear on the console:

```
WARNING: ..... serial line is busy (Port 0), (IRQ 4)
```

In this case, start the *SYSADM* user interface. Call up the menu items *configuration*, *load* and *alpha terminals* one after the other. Delete the alpha terminal *t0* (port 0) or *t1* (port 1). For further information in this regard, please refer to the manpage for *ups(1M)*.

6.1.2 UPS using protocol control

Protocol control is supported by the 9071-Pxx2 and Pxx6 UPS models.

Setting up the UPS interface

You must connect the protocol-controlled UPS to a free V.24 serial interface, e.g. to the motherboard (Port_0/COM1 or Port_1/COM2) or to the TC4P controller (TC40). You set up the UPS with *Config*.

- ▶ In *Config*, call up the mask for serial devices:

```
sysadm > configuration > serial devices
```

- ▶ Add a standard device with [Actions]:

```
[F7] > add > standard device
```

- ▶ Select the device name UPS.

- ▶ In the *UPS* mask, define the interface you have chosen. Make sure that your specifications for the UPS port match the following values:

```
Interface type: □□□□□□□□V.24
```

```
TTY label: □□□□□□□□□□serial
```

```
Activate service: □□□□□□no
```

```
Node name: □□□□□□□□□□/dev/term/UPS
```

- ▶ Activate the new configuration when you exit *Config*.

Connecting the UPS

- ▶ Connect the cable from the UPS (COM A or COM2) to the UPS interface of your RM system that you have set up.
- ▶ Shut down the system unit and connect the power cable of the system unit to the output from the UPS.
- ▶ Connect the UPS to the power supply.
- ▶ Reboot the system unit.

Setting the UPS parameters

The relevant entries are made with an editor (e.g. vi) in the */var/adm/ubfi/_upstab* file. The following entries are required:

- *ups_type*:
Enter 9071_P for the 9071-Pxx6 UPS using protocol control (the default value *unused* is required for a UPS using signal control).
- *ups_polltime*:
Specify the interval in seconds at which the UPS interface is polled. Do not choose too short an interval, otherwise there is a danger of overloading the system.
- *ups_downdelay*: UPSs using a protocol interface must switch themselves off shortly after the operating system has been shut down. To allow this to be monitored, Reliant UNIX terminates all activities when shutting down, but the kernel continues to run. After a set time (*ups_downdelay*), the kernel checks whether it is still "alive". If the kernel is still active, the UPS is not switched off and a reboot is initiated. You specify the time for *ups_downdelay* in minutes. The values 1 through 10 are valid.

Sample entries:

```
...
ups_type:9071_P
ups_polltime:16
ups_downdelay:3
...
```

Setting the reaction parameters for error states

These entries must be present in the *_upstab* file.

- *ups_active*:
Specifications in the event that the main power supply has failed and the UPS is supplying power. By default, a slow shutdown (*shutdown -y -g0 -i0*) is performed.
- *ups_active+batlow*:
Actions to be carried out if the primary voltage has failed and the UPS battery capacity is low. By default, a slow shutdown (*shutdown -y -g0 -i0*) is performed.
- *ups_no_comm*:
If this entry does not exist or if the entry has the reaction "no_shut_down", an error message is generated as before after five unsuccessful attempts (with a ten-second interval between each attempt). If an entry exists with the reaction "no_shut_down", the user reaction is executed in addition.
In both of the previously mentioned cases, a further attempt is made after two minutes to make contact with the UPS.
If an entry exists with the reaction "slow_shutdown" or "fast_shutdown", an attempt is made fifteen times (with a ten-second interval between each attempt) to make contact with the UPS before the user reaction and the required shutdown sequence are initiated.

Default entries:

```
...
ups_active:any:20:30:0:slow_shutdown:echo
ups_active+batlow:any:0:0:0:slow_shutdown:echo
ups_no_comm:any:0:0:0:no_shutdown:echo...
```

If the default entries do not suit your configuration, you can change them. The entries for the two reaction parameters have the same structure:

```
error:cabno:waittime:scripttime:warntime:reaction:script [parameter]
```

Field	Value	Meaning
error	<i>ups_active</i>	UPS active (main power failure).
	<i>ups_active</i> + <i>batlow</i>	UPS active and the UPS battery capacity <input type="checkbox"/> is low.
	Other values	Not relevant for UPS.
cabno	<i>any</i>	Cabinets which are not explicitly defined by an entry and a cabinet number.
	<i>0 ... 15</i>	Logical cabinet number.
waittime	<i>0</i>	Not evaluated for RM200/300/400.
scripttime	Time in seconds	Maximum permitted duration for the procedure specified under <i>script</i> . After this time has elapsed, the <i>reaction</i> is deleted.
wartime	<i>0</i>	Not evaluated for RM200/300/400.
reaction	<i>slow_shutdown</i>	The system is shut down with the command: <input type="checkbox"/> shutdown -y -g0 -i0
	<i>fast_shutdown</i>	The system is shut down with the command: <input type="checkbox"/> uadmin 2 0
	<i>no_shutdown</i>	Only the procedure specified with <i>script</i> is executed.
script <input type="checkbox"/> [parameter]	Name and parameters	Shell procedure or program called with the specified parameters.
	<i>echo</i>	Preset to the command <i>echo</i> .

Activating the configuration data

If you edit the *_upstab* file while the system is running, the changes you make only apply the next time the system is booted. You can also force activation of the new configuration data by stopping and then starting the EECD and PUPS daemons:

```
# /etc/rc2.d/S72eeed stop
# /etc/rc2.d/S72eeed start
# /etc/rc2.d/S98pups stop
# /etc/rc2.d/S98pups start
```

Setting the system behavior with *pwr*

You can set the following system characteristics with the *pwr* command:

- Auto power-on
Startup behavior after power returns
- Wait time for the UPS to shut down
To allow a shutdown of the UPS to be monitored, Reliant UNIX terminates all activities when shutting down, but the kernel continues to run. After a set time, the kernel checks whether it is still "alive". If the kernel is still active, the UPS is not switched off and the kernel initiates a reboot. This value should match the entry for *ups_downdelay* in the *_upstab* file.
- Wait time for secure startup
The kernel waits for the specified time at an early stage in the boot phase to allow the UPS to recharge. A power failure during this time will not endanger the system.
In this way, you avoid the risk of data loss if the capacity of the UPS is not sufficient to fully shut down your system if the power fails again.

- Activation

The system behavior on power failure defined by this command can be activated or deactivated. The settings are stored in NVRAM.

Syntax:

/sbin/pwr [-a [a_flag]] [-w [w_value]] [-s [s_value]][-p [p_flag]]

Option	Parameter	Value	Meaning
-a	<i>a_flag</i>	<i>n</i>	No automatic restart when power is restored.
		<i>y</i>	The system restarts when power is restored.
-w	<i>w_value</i>	Time in seconds	Wait time for the UPS to shut down (compare with <i>ups_download</i>)
-s	<i>s_value</i>	Time in minutes	Wait time for secure startup.
-p	<i>p_flag</i>	<i>n</i>	Deactivation of the system behavior defined by <i>pwr</i> .
		<i>y</i>	Activation of the system behavior.

Table 22: Options and parameters of the `pwr` command

The following example shows a recommended setting:

```
# pwr -a y -w 180 -s 10 -p y
```

Testing the UPS monitoring function

You should finally carry out the following test (the test is not dangerous, since system operation is not interrupted, even if the configuration is incorrect):

- ▶ Set the switch on the front panel of the UPS from *I* (inverter mode: the green INVERTER LED lights up) to *O* (bypass mode: the yellow BYPASS LED lights up).

After 10 seconds, the following message appears on the system console, preceded by the date and time:

```
prags unix: pupsd[10527]: pupsd:10005: BYPASS ACTIVE
```

- ▶ Now return the switch to the inverter position. After 10 seconds, the following message appears:

```
prags unix: pupsd[10527]: pupsd:10005: BYPASS ACTIVE is fixed
```

If the message is not displayed, check the preceding steps carefully. You can check the entry for the UPS with the `#termshow | grep UPS` command.



Once the test has been completed successfully, you must ensure that the switch on the front panel of the UPS is in position *I* (INVERTER mode), since only then does it switch to battery mode if the power fails.

6.2 Power failure or hazardous temperatures when using RM600

The computers in the RM600 model series are fitted with sensors that can detect a power failure or abnormal temperatures in the housing. If such a situation arises, a special piece of hardware generates an interrupt and sends it to the CPU. The *ubft* driver receives the interrupt and sends a signal to the *ubft* daemon, called *ups*. The daemon controls the subsequent response in accordance with the definitions in the */var/adm/ubft/_upstab* file and the settings made using the *Config* tool. If there are temperature problems in the system, the system is shut down automatically with the *shutdown(1M)* command.



A power failure can have disastrous consequences as there is no time to shut down the system in an orderly fashion if the hardware does not protect against power failure. The consequences are:

- All data contained in the main memory is lost.
- The file systems on the hard disks may become inconsistent.

Two peripheral devices are available to protect against power failure:

BBU Battery Backup Unit

A BBU protects the system. □

RM600-xxx: External devices (e.g. the console) are not protected.

UPS Uninterruptable Power Supply

A UPS protects both the system and external devices such as the console, ECON or jukeboxes.

A power failure is bridged for a specified length of time, which is limited by the performance of the BBU or the UPS. This means that the system can be shut down in an orderly fashion if the power failure lasts longer than a specified length of time.

Reliant UNIX automatically reboots once power is restored, depending on the boot mode set. The file system is checked for consistency if autoboot is turned on (the last number of the boot option is 0). If the check returns a positive result, the system switches back to multiuser mode.

6.2.1 Hardware monitoring and error detection

Both the computer environment and changes in status of the hardware (e.g. error events) are monitored using a special processor: the Environmental Interface Processor, or EIP for short. This type of processor is found in both the system cabinet and the peripherals cabinets (if any). All EIP status and error messages can be output on the system cabinet control panel. You will find information relating to this in your computer's operating manual.

Reliant UNIX starts a special *ubft* daemon when the system is started. The *ubft* daemon analyzes the information coming to it from the EIP via the *ubft* driver. It reacts to the following error events as follows:

acf2 Failure of the main power supply or end of UPS

acf1 Failure of the backup power supply

fan_unit_fail

Failure of a fan-out unit

temp_fail

Increase in temperature

The daemon name is derived from these error events:

UBFT = UPS, BBU, Fan, Temperature.

The *ubft* daemon takes the necessary steps if an error event occurs:

- It issues appropriate messages that can be directed to TRANSVIEW or Tele-X via Logging V3.0.
- It executes a special shell script if the system administrator has defined one.
- It shuts down the system, if necessary.

The mode of operation of the *ubft* daemon can be adapted to suit the relevant hardware:

- With the specifications in the `/var/adm/ubft/_upstab` file
- With settings made using the `Config` tool

The `/var/adm/ubft/_upstab` file contains default values that do not normally need to be changed.

If you are using an external BBU or a UPS, you must enter the required settings using the `WebSysAdmin` tool. In the case of an external BBU, cabinets must be allocated to the BBU driver that will supply power to them. In the case of a UPS, you must specify the cabinets that are to be powered by the UPS. With an RM600-xxx, the type of UPS must also be defined.

More detailed information on `WebSysAdmin` can be found in the [Chapter "Node administration with WebSysAdmin"](#).

You need only read the following section if you want to learn more about the operating principles and parameters of the `ubft` daemon, perhaps with a view to modifying them also. The manual entries `ups(1M)`, `ubft(7)`, `_upstab(4)` and `user_reaction(7)` also contain further information.

6.2.2 How the `ubft` daemon works

The `ubft` daemon processes error events according to a reaction model. [Figure "How the `ubft` daemon works"](#) on the next page provides an overview of this. When the `ubft` daemon detects an error, it executes the following actions:

- It sends an appropriate error message to the console.
- After a certain **delay time**, it checks whether the error still exists. If the error has not been corrected, it does the following:
 - It creates an identification file called `shutdown_<event type>.PBAY<n>` in the `/etc/opt/havail` directory. `<event type>` can be `FANFAIL`, `TEMPFAIL` or `POWERFAIL`, `<n>` is the cabinet number.
 - It starts a script called `/etc/opt/havail/user_reaction`, if it exists. The `OCCURRED` parameter is transferred to this script. The script is also called a user response program. More detailed information can be found under the manual entry `user_reaction(7)`.
- After a certain **user time**, the `ubft` daemon checks for a second time whether the error still exists. It then does the following:
 - If both the error and the identification file still exist, it starts the script using the `EXISTING` parameter.
 - Otherwise, it starts the script using the `DISAPPEARED` parameter.
- After a certain **warning time**, the `ubft` daemon checks for a third time whether the error still exists. It then does the following:
 - If both the error and the identification file still exist, it shuts down the system. To do this, it issues the `shutdown -y -g0 -i0` command (also called `slow_shutdown`).
 - Otherwise, it starts the script using the `DISAPPEARED` parameter.

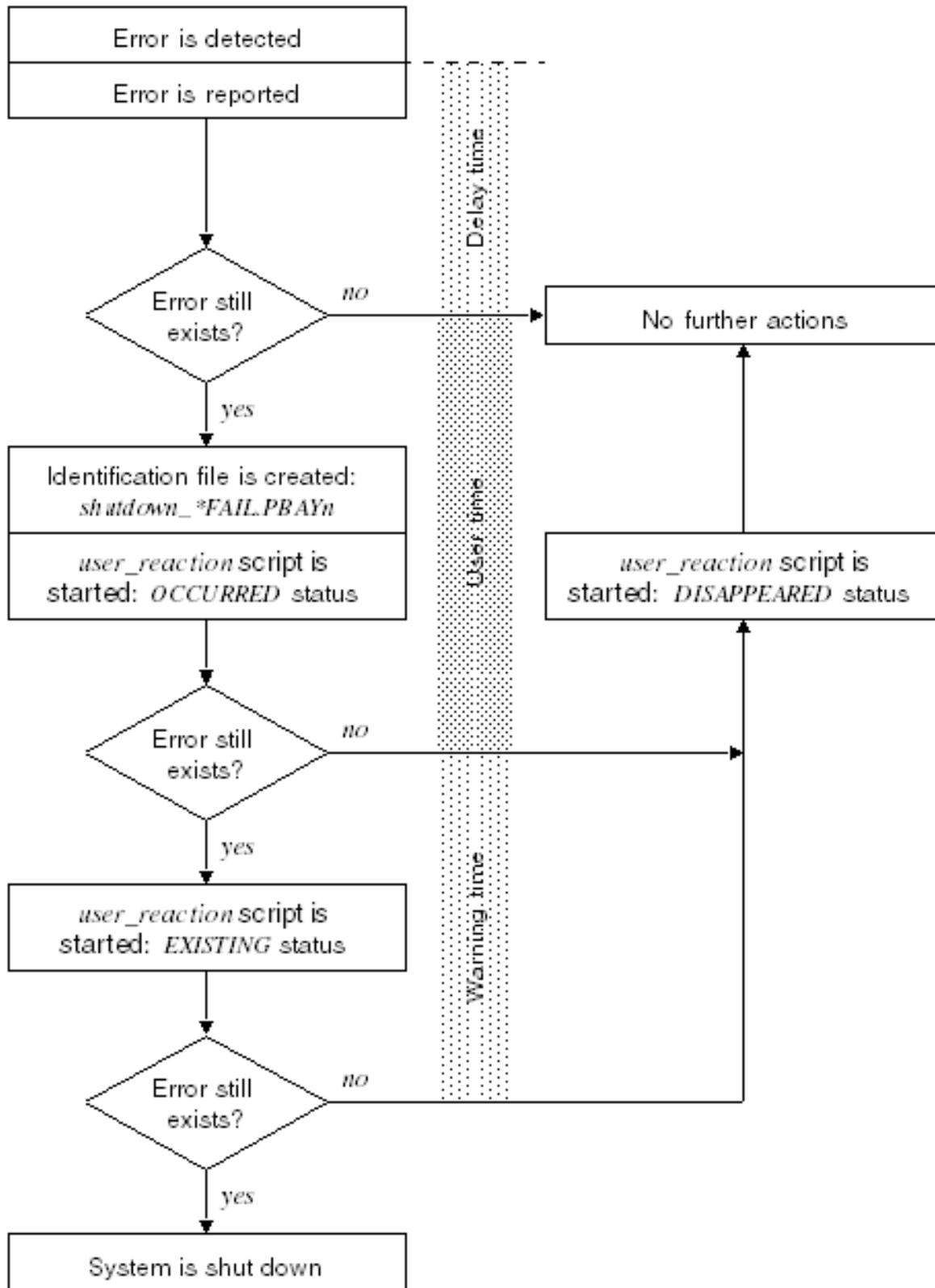


Figure 26: How the ubft daemon works

The user response program can be used to initiate further actions. For example, you can prevent the system from shutting down by deleting the identification file. The user response file has at least four parameters:

Parameter name	Parameter value
Event type	<i>FANFAIL, TEMPFAIL, POWERFAIL</i>
Event status	<i>OCCURRED, EXISTING, DISAPPEARED</i>
Number of the affected cabinet	<i>0 ... 63</i>
Time remaining before shutdown	Number of seconds
Optional string	Freely definable

Table 23: User response program parameters

Since the *ubft* daemon only categorizes the error events reported by the EIP into three error types (*FANFAIL*, *TEMPFAIL* and *POWERFAIL*), the user response program may be started a number of times with the same parameters. This program must therefore deal with such a situation.

Example:

A machine has an external BBU (*acf2+extbbu_active*). During a power cut, a fan breaks down in the power unit of a cabinet (*power_fan_fail*). In this case, the user response program would be started twice with the error type *POWERFAIL*, provided the *_upstab* file is configured accordingly.

The operation of the *ubft* daemon, in particular the length of Delay time, User time and Warning time, is controlled using the */var/adm/ubft/_upstab* file. The contents of this file are described in the manual entry *_upstab(4)*. The file is in the form of a table in which the columns are separated by colons. Example:

```
fan_unit_fail:any:20:10:0:slow_shutdown
```

For example, the following table entries are evaluated by the *ubft* daemon:

Error	Cabinet	Time period in seconds			Reaction
		Delay time	User time	Warning time	
fan_unit_fail	any	20	10	0	slow_shutdown
temp_fail	any	20	10	0	slow_shutdown

You can edit the */var/adm/ubft/_upstab* file if you want to enter other parameters. In this context, please note the following:

- Depending on the error event, the time available before the system **must** be shut down is very limited. When selecting the Delay time, User time and Warning time, you must ensure that sufficient time remains for shutting down the system.
- When you define the Delay time, User time and Warning time, you must also take into consideration the *AUTO POWER OFF* delay times of the EIP for *TEMPFAIL* and *FANFAIL* for the system cabinet.
- The EIP Delay time must be greater than the sum of the response times of the corresponding error event, plus the time required for shutdown. Otherwise, the EIP switches off the system cabinet automatically before the error response is finished. The response times may have to be shortened in the *_upstab* file. The EIP delay times are upper limits which are set by the hardware and should not normally be changed.
- If you have edited the file, the *ubft* daemon must be notified of the new entries (see the manual entry *ups(1M)*). To do this, establish the PID of the *ups* process (the *ubft* daemon process) using *ps -ef*. Then send a signal to the *ups*:

```
# kill -2 <ups-PID>
```


7 Telephone and Fax Numbers of the Support Center

Germany

Call Management Center (CMC)

Nationwide telephone number: **+49 1 80 5 40 40**

Nationwide fax number: **+49 1 80 5 33 67 79**

USA and Canada

Corporate Customer Support Center (CCSC)

Telephone number USA and Canada: **(800) 695-22 72**

Telephone number international: **(408) 428-90 00**

Fax number (first): **(408) 428-74 50**

Fax number (second): **(408) 428-70 20**

International (except Germany, USA and Canada)

Ask your local Fujitsu Siemens provider or branch for the telephone and fax numbers of the Support Center near you.

Glossary

address

An address refers to the position of data in the physical or virtual **memory** of the computer.

admin domain

Close coupling of computers. An example of this is that the disk storage of the individual computer nodes can be **shared** by all nodes in the domain.

ascii file

See **text file**

basic operating system

The basic operating system behaves like a subset of the Reliant UNIX **operating system**. It contains the most important UNIX **commands**, which are used to install the add-on packages and to generate the Reliant UNIX **system kernel**.

board debug monitor

The board debug monitor (BDM) is part of the **firmware** of a special **controller**:

- RM600-xxx: CSI controller□
 (CSI = Central Services & Interfaces)
- RM600 E: (E)HIOS basic controller□
 (referred to as (E)HIOS for short in the manual;□
 (E)HIOS = (Enhanced) High Performance Input Output
 System)

When the **operating system** is **booted**, the BDM is started automatically once the **master test handler** has completed its tests correctly. Depending on the **boot mode** set, it either automatically initiates **boot1**, or awaits user input with his prompt:

- RM600-xxx: BDM(MBII□0□t)>
- RM600 E: BDM(SPB□□8□t)>

boot1

Boot1 (first-level boot) loads **boot2** from the load device (**system disk** or **system CD-ROM**) into the main memory and starts the boot2 process.

boot2

Boot2 (second-level boot) is a system interface that works independently of the **operating system**. It is located in **partition 10** of the **system disk**. It is loaded into the main memory by **boot1** and executed there as a bootstrap process. Depending on the **boot mode** set, it either loads the Reliant UNIX operating system automatically, or waits for user input with the boot2: prompt. The term SASH (stand alone shell) is occasionally used as a synonym for boot2.

boot

Starting the **operating system** is referred to as booting. The **system kernel** must first be loaded from a data medium (e.g. the **system disk**) into main memory. Booting involves a number of **boot phases**, namely the **built-in self test**, **master test handler**, **board debug monitor**, **boot1** and **boot2**. All **logins** and **processes** are killed when the operating system is booted. The load parameters required for booting (e.g. the device name of the system disk) are saved in the **NVRAM**.

boot mode

There are a number of modes in which the system can be booted:

- Autoboot and manual boot
 In autoboot, all **boot phases** are run through automatically until the **login** prompt of the Reliant UNIX

operating system appears. With manual booting, the **boot** procedure is stopped in the **board debug monitor** and **boot2** phases. A prompt appears and the user can enter **commands**.

— Logical and physical boot

The name is derived from whether the boot medium is accessed using the logical or the physical **address**. The address is stored in the boot string. The physical boot mode is required if the operating system is not to be booted from the active **system disk**, but instead from another storage medium (e.g. **system CD-ROM**). The logical boot mode offers the advantage that a number of system disks that can be connected to any **controller** in the system cabinet can be administered. The logical addresses are assigned to the physical addresses here via a root partition list.

Prerequisites for the RM600-xxx:

The physical boot mode is only possible if the system disk is connected to the CSI. The logical boot mode is only possible with **Reliant UNIX V5.43** or higher and **firmware V2.42** for the CSI.

boot phases

Several boot phases are run through when the **operating system** is **booted**, namely **built-in self test**, **master test handler**, **board debug monitor**, **boot1**, and **boot2**. Depending on the **boot mode** set, the next boot phase is started automatically in the board debug monitor and in boot2 (autoboot), or a prompt is displayed (manual booting).

buffer

A buffer is an area for temporarily storing input/output data of any length. The data is divided into smaller units that are easier for system functions to handle. Buffers are used both by the **file system** [3] and by the standard functions for input/output (*stdio*).

buffering

This is the term used to describe temporary data storage in **buffers**.

built-in self test

The built-in self tests (BIST) are part of the **firmware** that is stored in the **controller**. The BISTs are automatically started when the computer is switched on. They check all of the controllers and store the results in a special controller:

RM600-xxx: in the CSI controller (CSI = Central Services & Interfaces)

RM600 E: in the (E)HIOS basic controller (referred to in this manual as (E)HIOS for short; (E)HIOS = (Enhanced) High Performance Input Output System

The results of the check are then queried by the **master test handler**.

child process

See **fork**

client

If a system is integrated in a network, certain activities can be executed **remotely** on this system (i.e. via another remote computer). In this case, the **local** system is called the **client** and the remote system is the **server**. For example, the software, which is required for each different activity on the **client**, can be read in via the **server**.

command

A command is

1. an instruction to the **shell**, which normally starts a **program** [1] as a **child process** (see **fork**).
2. in a wider sense, any executable **file**.

command file

See **shell script**

configuration

The configuration of a system means

1. the characteristics of the software running on that system. The characteristics of the **operating system** are influenced by certain parameters (e.g. *\$SHELL*, *\$TZ*, *\$PS1*), certain **files** (e.g. */etc/profile*, */etc/passwd*) and the **tunable parameters** (defined in the */etc/conf/cf.d/mtune* and */etc/conf/cf.d/stune* files).
2. the hardware components belonging to the system. This includes the integrated **controllers**, **hard disks** and **drives**, as well as the available peripherals.

console

The console is the **terminal** connected directly to the system. This terminal is used by the **system administrator** to communicate with the system.

controller

A controller is an intelligent control unit that controls more or less independent data input between the central processor and a peripheral device. The controller generally consists of a slot card that is inserted in a slot on a system board and is connected with the peripheral devices via a cable.

core image

The core image (dump) is a copy of all segments of a **process**. It is created if the process is not terminated "normally" (e.g. by means of a signal). This copy can be located in the main memory, the **swap area**, or in a core file named *core*. The *core* file is used for debugging. It is always created in the current **directory** of the process. Special switches and **commands** can also be used to dump the main memory. These files are stored under */var/crash*.

cylinder

This refers to the set of all tracks on a **hard disk** that are equidistant from the rotating center of the disk, but can be on different surfaces. All tracks on the cylinder can be read without having to reposition the read/write unit.

daemon

A daemon is a **process** that runs in the background (usually permanently) and executes a public task throughout the system, e.g. *calendar(1)* and *cron(8)*.

device

A device is

1. a **file** that is neither a normal file nor a **directory**, e.g. a tape device, the null special file or a device unit,
2. a physical input/output unit.

directory

A directory is a **file** that contains a catalog of **file names** [2]. Based on the organizational principles of **file systems** [2], a directory consists of entries that identify files (and also other directories). This means that a directory forms a node in the **directory tree**.

directory entry

A directory entry is

1. the assignment of a name to an **inode number**, **i-number**. Each entry in a **directory** has this format.
2. the name part of such an assignment.

directory hierarchy

See **directory tree**

directory tree

This term refers to the entire tree containing all *directories*. In this tree, each directory can be accessed from the **root** via a series of subdirectories.

drive

Drives are hardware devices that can contain or accommodate data media (**hard disks**, CDs, magnetic tapes or diskettes).

dump

See **core image**

environment

An environment is

1. a set of character sequences that is made available to a **process** in addition to its arguments when it is executing a **file** [2]. Normally, the environment is inherited with **exec(2)** operations,
2. a particular environment [1] that is administered by the **shell**.

error

An error is a hardware or software-related situation that prevents a system or user process from being executed successfully. In such cases, the **operating system** outputs an error message on the **console** or on the user's terminal.

error message

See **error**

exec

exec is a **system call** that enables users to execute another **program**.

file

A file is

1. in general, a possible source for input or a target for output,
2. in particular, an **inode** and/or associated file contents, i.e. a normal file, a **special file** or a **directory**,
3. a **directory entry**, where different directory entries can refer to the same file [2],
4. in its simplest form, a normal file.

If the execute permissions have been set, the file is referred to as executable (see **permission**). If the file is used as a source for input or a target for output, it is referred to as open. A file can be open several times simultaneously.

file name

A file name is

1. a *pathname*
2. the last component of a pathname

file system

A file system is

1. a collection of **files** that can be mounted in a block-oriented **special file**. The **inode** for each file appears precisely once in the **i-list** of the file system. Each file can be accessed via a *path* from the **root** directory of the file system.
2. the complete set of all files [1] on the computer.
3. that part of the **system kernel** that deals with file systems [1].

firmware

Firmware refers to **programs** that are stored in the EEPROMs of the **controllers** (EEPROM = Electrically Erasable Programmable Read Only Memory). Firmware consists of: **built-in self test**, **master test handler**, **board debug monitor** and **boot1**. The programs are activated in sequence if the **operating system** is loaded in main memory during the **boot** process. The firmware can be updated as required.

fork

This **system call** is used to duplicate a **process** so that a parent process and a child process then exist

with separate (but initially identical) text segments, data segments and batch segments.

formatting

This refers to a process whereby a **hard disk** is assigned an addressing schema. The disk is divided into tracks and **sectors**.

function key

The term **function key** is used to refer to certain keys on the keyboard that are numbered consecutively (e.g. F1 through F12). Some application programs use these keys to control how the program operates (e.g. **SYSADM** and the dialog program of the mini operating system). The various functions that are assigned to the function keys depend on the program context. The current function key assignments are displayed at the bottom of the screen.

group

A group is

1. a set of **permissions** which, in addition to the **owner's** permissions, regulate access to a **file**,
2. a set of **user IDs** that can assume the priority rights of a group [1],
3. the **group ID** of a file.

group ID

The group ID is an integer that is usually assigned to one or more **login** names. If **files** are created by a **process**, they are assigned the **user ID** of the process as the **owner** and the group ID as the **group** [3].

hard disk

Hard disks are data media that are usually installed directly in the computer. This distinguishes them from removable data media, such as diskettes, cassettes and CDs. The **system disk** is particularly important, since it contains the **operating system**. Each hard disk is divided into several areas referred to as **partitions**. The existing partitioning is stored in a **partition table**.

i-list

The i-list contains all **inodes** belonging to a **file system** [1]. This list is referenced by means of the **inode number, i-number**.

inode

An inode (information node) is an individual data structure in the **i-list**. Each **file** [2] in a **file system** [1] has precisely one inode. This inode contains information for managing the file. The link to the i-list is established by means of the **inode number, i-number**. An inode contains the following information: the file type, the **permissions**, the number of **links**, the **owner**, the **user ID**, the **group**, the **group ID**, the size in bytes, the date on which the inode was last modified, the date on which the file was last modified, and last accessed, and references to the file contents.

inode number, i-number

Each **inode** is assigned a number. The inode number combined with the device number refer to the unique relevant entry in the **i-list** of a **file system** [1].

installation

This is the process of loading software packages from a data medium (installation medium) and copying them to a **hard disk**. There are two different *types of installation*: **new installation** and **update installation**. During the new installation of Reliant UNIX, several interim levels of the **operating system** are installed. These include the **mini operating system** and the **basic operating system**.

Internet address

The Internet address is the unique **address** of a computer in the network. It is four bytes long and contains information as to which network and which computer in the network is to be addressed. If the computer is to be connected to the official Internet, the Internet address has to be applied for from the Network Information Center (NIC).

link

A link is

1. an additional **directory entry** for an existing **file**. This type of link can be created using the *link(1M)* command and deleted using the *unlink(1M)* command,
2. in a wider sense, a normal **directory entry**,
3. in a general sense, any **directory entry**, apart from the entry taken to be the main entry for a predefined **inode** referenced using either a normal **link** [1] or a **symbolic link**.

local

The computer to which the user's terminal is connected is called the *local* computer. Any other computer that can be accessed over the network, on the other hand, is referred to as the **remote** computer. If the local computer is used exclusively for an activity, the activity is also referred to as local.

log files

Log files contain records of actions executed on the system by the software. During **installation**, all actions are logged under */var/sadm/pkg/.prot**.

login

A login is:

1. the **program** that executes the login on the **operating system**
2. the procedure for logging on to the operating system
3. in a wider sense, the session on the system that begins after login [2].

master test handler

The master test handler (MTH) is part of the **firmware** that is stored in the **controller**. The MTH is started automatically once the **built-in self tests** have checked the controllers. It queries the result of these tests, checks other hardware modules and logs the results on the **console**. This log contains a test status for each module checked. If an error is detected in a module, the MTH prevents the **operating system** from being booted.

memory

This term can have the following meanings:

1. a synonym for **core image**,
2. a physical memory that reflects the available space in the main memory.
3. an extension of the main memory. Programs can use memory on the **hard disk** as an extension of the main memory by means of virtual memory management.

memory image

This is a synonym for **core image**.

mini operating system

The mini operating system behaves like a subset of the Reliant UNIX **operating system**. It only supports **commands** that are needed for installing Reliant UNIX or for working with the *DSSI* **program**. It is located on the **system CD-ROM** and can be booted from **boot2**.

miniroot file system

The miniroot file system is a small root **file system**. It is located on the **system CD-ROM** and can be copied from **boot2** to the RAM disk.

mount

Mounting a **file system** extends the directory hierarchy. This is achieved by assigning the **root** of the new file system [1] to a **directory entry** of a file system already mounted. The *mount(1M)* command is used for this purpose. The *umount(1M)* command has the opposite effect.

multiuser mode

Multiuser mode is a **run level** that allows several users to access the **operating system**.

networking

Networks are used to transfer data between various systems. Communication lines such as coaxial cables, telephone lines, etc., are used for this purpose. Networking services include **file** transfer, remote **login** and remote execution of **commands**.

new installation

If you want to install Reliant UNIX on a **hard disk** on which this **operating system** was **not** previously installed, you must perform a new installation. Normally, this is only required if a new **system disk** has been integrated in the system. All data that was previously stored on the hard disk, including the add-on products, is overwritten in this **type of installation**. Exception: the existing **/home file system** can be preserved.

node name

Each system has its own name, known as the node name. The system can be addressed by means of this name if it is integrated in a network. The name must be unique and can be displayed using **uname -n**.

NVRAM

The load parameters required for the **boot** process (e.g. the device name of the **system disk**) are stored in a non volatile storage medium, the NVRAM (Non Volatile Random Access Memory). The NVRAM is located in a special **controller**:

RM600-xxx: in the CSI controller (CSI = Central Services & Interfaces)

RM600 E: in the (E)HIOS basic controller (referred to in this manual as (E)HIOS for short; (E)HIOS = (Enhanced)High Performance Input Output System)

operating system

The operating system is a **program** for managing the computer resources. For example, it handles input/output procedures and **process** planning. This relieves user programs of these tasks. Well-known operating systems include DOS, UNIX and **Reliant UNIX**. During **installation** of Reliant UNIX, several interim levels of the operating system are installed. These include the **mini operating system** and the **basic operating system**.

owner

The owner of a **file** is the **user ID** of the **process** that created the file. Owners have special **permissions** for their files.

parent process

See **fork**

partition

Each **hard disk** is divided into a number of areas (partitions) which can overlap. The existing partitioning is stored in a **partition table** on the hard disk. The partitions of the **system disk** must have a minimum size. **boot2**, which is required in order to **boot** the system, is stored in partition 10.

partition table

The partitioning of each **hard disk** is defined in a partition table. This table is stored on the hard disk. Partition tables are created and administered using the **dkpart(8) command**.

path, pathname

Both terms are synonyms, used to refer to a name string which identifies a **file**. A relative pathname always starts at the current **directory**. For example, if the current directory contains the subdirectory **A**, which in turn contains the subdirectory **B**, which contains the file **xyz**, this file is referred to as **A/B/xyz**. A

complete pathname always begins at the **rootfile system** (`/`), e.g. `/home/krenz/A/B/xyz`.

permission

This controls certain types of access to a **file**, namely read access, write access, and execute permission (in the case of a **directory**, the last permission is the search permission). Permissions are assigned separately to **owners**, **groups** and others.

pipe

A pipe is a direct data stream connection between **processes**. The output data of one process becomes the input data of the other process.

pipeline

A pipeline is a concatenation of several **programs** by means of **pipes**.

process

A process is a sequence of related statements. When a process is started, three **files** are usually opened: **standard input** (`stdin`), **standard output** (`stdout`) and **standard error** (`stderr`) from the `stdio` standard library. Initially, all three files refer to the **terminal**. The attributes of a process are specified by its **core image**. The core image contains the command counter and information about the current **directory**, the set of opened **files**, the controller terminal, the **user ID** and the **group ID**.

process ID, process number

The process ID is an integer for identifying a **process**.

profile

A profile is

1. an optional **shell script** called `.profile`. Normally, this script is run by the **shell** when a user logs on in order to set the **environment** [2] and other working conditions that the user expects.
2. a file for capturing the time profile of a **process**. This contains the regularly scanned states of the command counter.

program

A program is

1. an executable **file**
2. a **process**

prompt

A prompt is a character that indicates to the user that the computer is ready to receive input.

queue

A queue is a list of actions waiting to be processed.

Reliant UNIX

Reliant[®] UNIX[®] is the UNIX[®] derivative of Fujitsu Siemens. It is a standardized open **operating system** which satisfies high requirements with regard to reliability and availability – both in commercial and technical usage. "Reliant UNIX" is the new name for Version 5.43 and later versions of the **SINIX[®]** operating system. All of the proven SINIX features are still available in Reliant UNIX.

remote

The computer to which the user's terminal is connected is called the **local** computer. Every other system that can be accessed over a network, on the other hand, is referred to as the *remote* computer. If a remote computer is used to perform an activity, this activity is also described as remote. In a remote operation, the **local** computer is also called the **client** and the remote computer is called the **server**. For example, the software required for any activity on the **client** is loaded via the **server**.

root

root is

1. the **login** name of the user *root*. As **system administrator** or **superuser**, this user has special rights (*root* authorization).
2. a special **directory** (/) that forms the **root** of the directory hierarchy in a **file system** [1]. This is also the **login** directory of the user *root*.
3. the source of the entire **file system** [2] with the usual *pathname* /,
4. the source of a **directory** in a **file system** [1].

run level

The run level is a software **configuration** of the **operating system**, which permits the existence of a particular group of **processes**. A distinction is made between **single-user mode** and **multiuser mode**.

SASH

SASH stands for the stand alone shell, and is used as a synonym for **boot2**.

search path

The search path of the **shell** consists of a list of directory names. If a **command** is invoked, it is prefixed with the directory names in sequence until an executable **file** is found. The search path is stored in the *\$PATH* shell variable.

sector

Each **hard disk** is logically divided into concentric circles called tracks. These tracks are in turn divided like a cake into parts called sectors.

server

If a computer is integrated in a network, certain activities on this computer can be carried out **remotely** (i.e. using another, remote computer). In this case, the **local** computer is called the **client**, while the remote computer is called the **server**. For example, the software required for any activity on the **client** can be loaded via the **server**.

shell

The shell is

1. the *sh*(1) **program** which activates other programs when **commands** are entered. Normally, this is the program set by the user, which is started when the user logs on to the **operating system** (see also **login**).
2. the term commonly used for every program started after login.

shell procedure

See **shell script**

shell script

A shell script is an executable **file** whose contents are used by the **shell** as command input.

single-user mode

A **run level** in which only one user has access to the **operating system**.

SINIX

SINIX[®] (Siemens UNIX[®]) is an **operating system** based on UNIX for systems developed by Siemens AG. It is a standardized open operating system with commercial applications. The new name Reliant[®] UNIX[®] is used for Version 5.43 and later versions of the operating system. All the proven SINIX features are still available in Reliant UNIX.

special file

A special file is formed by an **inode** that refers to a **device**.

standard error

This term refers to one of the three **files** that are normally already open when a **process** is started. The **file name** of standard error output is *stderr*. Initially, the file refers to the **terminal**.

standard input

This term refers to one of the three **files** that are normally already open when a **process** is started. The **file name** of standard input is *stdin*. Initially, the file refers to the **terminal**.

standard output

This term refers to one of the three **files** that are normally already open when a **process** is started. The **file name** of standard output is *stdout*. Initially, the file refers to the **terminal**.

startup

See **boot**

superuser

See **system administrator**

swap area

The swap area is the part of the **hard disk** which stores the "anonymous" memory contents when swapping out. It is a separate area and is not part of the **file system**.

symbolic link

A symbolic link is a **file** containing the *pathname* of another file. When a symbolic link is accessed, the file with the specified pathname is accessed.

SYSADM

SYSADM is an interactive program package for system administration. It is started by the *sysadm(1M)* **command** and supports most system administration tasks via its own user interface.

system administrator

The system administrator or **superuser** is the user who is logged on with the **login** name **root** and the **user ID** 0. Only the system administrator is authorized to execute certain administration tasks. The rights required for this purpose are also called *root* authorization. Regardless of the **permissions** set, he can access all **files** and activate certain **system calls**.

system call

System calls are

1. a set of **operating system** functions used to assign, start, monitor, handle and terminate all operating system operations.
2. calls issued by user processes for executing system-dependent functions. These include input/output and process creation.

system CD-ROM

System CD-ROM refers to a CD-ROM that contains the software of the **operating system**. A system CD-ROM from Fujitsu Siemens bearing your system model is required to install the **Reliant UNIX** operating system. Example: "Reliant UNIX 5.45 A00 – RM200, RM300, RM400". This CD-ROM also contains the **firmware** that is necessary for the respective Reliant UNIX version.

system crash

Hardware or software errors can lead to conditions that the **operating system** cannot handle. If this occurs, the system is terminated automatically and "crashes". Such conditions arise, for example:

- if the operating system cannot assign any further resources,
- if it cannot manage any further **processes**,
- if it cannot support any further **system calls**,
- if the power supply is unstable.

system disk

Each system can contain several **hard disks**. Any hard disk containing the root file system (/) is known as the system disk. One or more system disks, of which one is active, can be administered depending on the **boot mode**. The **partitions** of the system disk must have a minimum size. Partition 10 of the system disk contains **boot2**, which is required to **boot** the system. Partition 15 stores the status

information if the system disk is mirrored.

system kernel

The system kernel is the actual **operating system**. It is located on the **system disk** as the */unix* or */stand/unix* file. This file is approximately 9 Mbytes in size. It is loaded into the main memory during the **boot** process and remains resident there. The */etc/conf/bin/idbuild* command can be used to generate a new system kernel. This is then loaded automatically the next time the system is booted.

terminal

A terminal is an input/output device consisting of a keyboard and a monitor (and possibly a mouse). The terminal that is connected directly to the system is referred to as a **console**.

text file

A text file or ASCII file only contains ASCII characters.

tunable parameter

Tunable parameters are used to define sizes and limits for various control structures of the Reliant UNIX **operating system**. This enables the operating system to be adapted to a certain extent to the respective operating conditions. The tunable parameters are defined in the */etc/conf/cf.d/mtune* and */etc/conf/cf.d/stune* files. They can be changed using the *idtune(1M)* command.

type of installation

There are two different types of installation: **new installation** and **update installation**.

update installation

An **installation** is referred to as an update installation if two conditions apply: a particular version of Reliant UNIX has already been installed on the system and a new version is now to be installed. With this **type of installation** all system data used to configure the system is retained, together with all user data and add-on products.

user ID

The user ID is a positive integer that is normally assigned to a **login** name. The user ID of a **process** is inherited by all processes created by that process (using **fork**). It is also entered as the **owner** of all files created by the process.

virtual memory

A memory management procedure used by most modern **operating systems**. See also **memory**.

Abbreviations

ASCII	American Standard Code for Information Interchange
ANSI	American National Standards Institute
BBU	Battery Backup Unit
BDM	Board Debug Monitor
BIST	Built-In Self Test
CCSC	Corporate Customer Support Center
CD-ROM	Compact Disk - Read Only Memory
CMC	Call Management Center
CPU	Central Processing Unit
CSI	Central Services & Interfaces
DAT	Digital Audio Tape
DCS	Disk Controller SCSI
DHCP	Dynamic Host Configuration Protocol

DNS	Domain Name Service
DSM	Distributed Systems Management
DSSI	Data backup and system installation
DVD	Digital Versatile Disk
ECON	Environment Controller
EEPROM	Electrically Erasable Programmable Read Only Memory
EHIOS	Enhanced High Performance Input Output System
EIP	Environmental Interface Processor
EISA	Extended Industry Standard Architecture
FMLI	Form and Menu Language Interpreter
HIOS	High Performance Input Output System
ID	Identifier
IKDB	Internal Kernel Debugger
IOS	Input Output Subsystem (SCSI)
IP	Internet Protocol
LAN	Local Area Network
LAR	Logout Auto Recovery
LSM	Local Software Maintenance
MAC	Media Access Control
manpage	Online command description
MBII	

	Multibus II
MET	Middle European Time
MTC	Magnetic tape cassette
MTH	Master Test Handler
NFS	Network File System
NIC	Network Information Centre
NVRAM	Non Volatile Random Access Memory
OLR	Online Replacement
PC	Personal Computer
PID	Process ID
PIF	Product Interchange File
PROM	Programmable Read-Only Memory
RAM	Random Access Memory
RFS	Remote File Sharing
ROD	Rewritable Optical Disk
SAC	Service Access Controller
SAF	Service Access Facility
SASH	Stand Alone Shell
SAX	Software Administration for Open Systems
SCSI	Small Computer System Interface
SINIX	Siemens UNIX

SInus	SINIX New Update Set
SMC	Streaming Mode Cartridge
SNMP	Simple Network Management Protocol
SP bus	Synchronous Pipelined Bus
SPI	Siemens Pyramid Information Systems, Inc.
SVGA	Super Video Graphics Array
TCP	Transmission Control Protocol
TZ	Time Zone
UBFT	UPS, BBU, Fan, Temperature
ufs	UNIX File System
UPS	Uninterruptable Power Supply
VxFS	VERITAS File System
WORM	Write Once Read Many

Related publications

Ordering manuals

The manuals listed here can be ordered from your local office.



Please refer also to the documentation overview in the readme file. This file can be found under the name *readme*

- in the */opt/readme/sinix.GB* directory on the system disk.
- in the */cdrom/sinix_y/5_45b00/en* directory resp. in the */cdrom/sinix_n/5_45b00/en* on the CD-ROM "Reliant UNIX".

Some manuals can also be downloaded from the Internet. □

Address: http://www.siemens.de/servers/man/man_us.htm

Reliant UNIX 5.45

Diagnostics and Troubleshooting

RM200, RM300, RM400, RM600

Target Group

System administrators, service engineers

Contents

The first part of this manual describes the errors that can occur when the system starts up and when it is operating, how the system responds and what actions you can take. The second part describes the user interfaces for system diagnostics, for example Logging □ V3.0, Config, Diagnostic menu. The third part includes descriptions of various routines that you may require for troubleshooting and diagnostics, an explanation of the LAR concept and a list of boot1 error messages on RM600 systems.

Reliant UNIX 5.45

Commands. User's Reference Manual

Reference Manual

Target Group

Users, system administrators

Contents

Description of the user commands of the Reliant UNIX 5.45 operating system.

Reliant UNIX 5.44

User's Guide

User Guide

Target Group

Users

Contents

Description of the fundamental elements of the Reliant UNIX operating system, for example the file system, process execution, the shell.

Reliant UNIX 5.44

System Administrator's Guide

RM600-xxx, RM200, RM300, RM400

Target Group

System administrators

Contents

Overview of UNIX system administration on the specified machines and on PCs.

Reliant UNIX 5.45

Network Administration

System Administrator Guide

Target Group

System administrators

Contents

Description of the network administration tasks that arise when using the TCP/IP software on Reliant UNIX 5.44 and the network basic functions (BNU).

Reliant UNIX 5.45

Networking – Reference Manual

System Administrator Guide

Target Group

System administrators

Contents

Description of commands, C functions, utilities and files that are important for computer networks.

Reliant UNIX

NetWorker V4.2C – Installation and Maintenance

System Administrator Guide

Target Group

System administrators who want to install NetWorker

Contents

Description of how the NetWorker software is installed on server and client systems.

Reliant UNIX

NetWorker V4.2C – User's Guide

User Guide

Target Group

Users who want to implement NetWorker

Contents

Description of how data is backed up and restored using NetWorker. The manual explains all NetWorker functions available to the user.

Reliant UNIX

NetWorker V4.2C – System Administrator's Guide

System Administrator Guide

Target Group

System administrators and users

Contents

Description of how NetWorker can be configured and managed.

Reliant UNIX 5.44

RAIDmaster

RM200, RM300, RM400, RM600

Target Group

System administrators

Contents

Description of how RAIDmaster controllers are used in Reliant UNIX systems.

Reliant UNIX 5.45

System Administration within a Domain

System Administrator Guide

Target Group

System administrators

Contents

Description of how domains are created and managed using *DomainAdmin*.

Reliant UNIX 5.45

System Administrator's Reference Manual

RM200, RM300, RM400, RM600

Target Group

System administrators

Contents

Description of the commands and application programs for system maintenance as well as file formats and special system administration files.

Reliant UNIX 5.45

Reliant UNIX Installation

System Administrator Guide

Target Group

System administrators

Contents

Description of how to install the Reliant UNIX operating system.

Reliant UNIX 5.44

System Administration and Hardware Configuration Using the SYSADM User Interface

System Administrator Guide

Target Group

System administrators

Contents

Description of hardware configuration and system administration with SYSADM: managing file systems and network services, installing software, configuring systems, user administration.

Reliant UNIX 5.44

Tuning Guide

RM200, RM300, RM400, RM600

Target Group

System administrators

Contents

Description of how system performance can be analyzed and tuned and when it make sense to upgrade the hardware. The tunable parameters are also described.

Reliant UNIX 5.44

Virtual Disks

RM200, RM300, RM400, RM600

Target Group

Network administrators

Contents

Description of the different types of virtual disks in Reliant UNIX and how they are configured. The manual describes in detail what to do should a mirror disk fail.

Reliant UNIX, Solaris

LAN Console

Product manual

Target Group

System administrators, service engineers

Contents

Description of the installation, configuration and operation of LAN consoles and the diagnostics of these devices.

Distributed Systems Management V1.3

DSM V1.3 – New Functions and Modifications

System Administrator Guide

Target Group

System administrators

Contents

Description of the new functions of DSM V1.0 (DSM = Distributed Systems Management), the DSM V1.3 packaging process, as well as the operation of SAX-LSM (SAX = Software Administration for Open Systems, LSM =Local Software Maintenance).