

Reliant UNIX 5.45

System Administration within a Domain

Comments... Suggestions... Corrections...

The User Documentation Department would like to know your opinion on this manual. Your feedback helps us to optimize our documentation to suit your individual needs.

Fax forms for sending us your comments are included at the back of the manual.

There you will also find the addresses of the relevant User Documentation Department

Copyright and Trademarks

Copyright © 2000 Fujitsu Siemens Computers GmbH.

All rights reserved.

Delivery subject to availability; right of technical modifications reserved.

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

This manual is printed on
paper treated with
chlorine-free bleach.

Preface

What is an administration domain?

Domain administration with WebSysAdmin

Operating an administration domain

Solving problems

Abbreviations

Related publications

Index

Contents

1	Preface	1
1.1	Target groups	1
1.2	Notational conventions	1
2	What is an administration domain?	3
2.1	Advantages of administration domains	3
2.2	The domain-wide administration system	5
2.2.1	Installing DomainAdmin	5
2.2.2	The CLOS domain operating system	5
2.2.3	The CLODB administration domain database	6
2.3	The administration desktop	6
2.3.1	Administration desktop requirements	7
2.3.1.1	WebSysAdmin clients under Windows 95/98/NT (PC console)	7
2.3.1.2	WebSysAdmin clients under Solaris 7 (LAN console)	7
2.3.1.3	WebSysAdmin clients via VGA console with X11 interface (on RM200, RM300 and RM400)	8
3	Domain administration with WebSysAdmin	9
3.1	Features of WebSysAdmin	9
3.2	Starting WebSysAdmin	10
3.2.1	Starting WebSysAdmin under Windows 95/98/NT (PC console)	10
3.2.2	Starting WebSysAdmin under Solaris 7 (LAN console)	11
3.2.3	Starting WebSysAdmin via the VGA console (on RM200, RM300 and RM400)	12
3.3	Elements of WebSysAdmin	12
3.3.1	Menubar and toolbar	12
3.3.2	Selection configurator	14
3.3.3	Hierarchical browser	14
3.3.4	Object table	16
4	Operating an administration domain	17
4.1	Creating a domain	18
4.2	Deleting a domain	21
4.3	Adding a node	21
4.4	Removing a node	23
4.5	Adding a user	24
4.6	Cloning a user	27
4.7	Removing a user	28
4.8	Assigning a global password	29
4.9	Creating virtual disks (vdisks)	31
4.9.1	Creating local vdisks on a node	31

Contents

- 4.9.2 Configuring vdisks on shared resources 37
- 4.10 System diagnosis 41
- 4.10.1 General troubleshooting 42
- 4.10.2 Object-specific troubleshooting 45
- 4.11 Event and reaction definitions 49
- 4.12 Domain-wide file system access 53
- 4.13 Checking software versions 56
- 4.14 Creating tasks 58

- 5 Solving problems 63**
- 5.1 Problems with Netscape or WSA 63
- 5.2 Performance problems 65
- 5.3 Problems with node administration 66

- Abbreviations 71**

- Related publications 73**

- Index 77**

1 Preface

This manual describes how several nodes can be bundled in an administration domain and thus managed jointly. This is possible using a domain administration system, in this case the *DomainAdmin* add-on product. Users are given access to the essential functions of this administration system by means of the *WebSysAdmin* (WSA = Web based System Administration) user interface. This manual deals less with *DomainAdmin* itself, than with using *DomainAdmin* via the WSA user interface for managing the systems in an administration domain.

1.1 Target groups

This manual is written primarily for system administrators and field engineering staff. Sound operating system and hardware knowledge is required as well as familiarity with the basics of the *WebSysAdmin* graphical user interface.

1.2 Notational conventions

The following notational conventions are used in this manual.

italics

Denote names of commands, system calls, functions, files, procedures, programs etc. as well as menu options and input/output fields from figures in the main body of text.

Name (extension)

Commands, system calls, functions, files, interfaces etc., are given an extension in brackets if manual pages are available for them. The extension indicates the chapter in which the manual page is described in one of the following books:

- User's Reference Manual, see also *ref_urm*(1)
- System's Administrator's Reference Manual, see also *ref_sarm*(1)
- Programmer's Reference Manual, see also *ref_prm*(1)
- Networking Reference Manual, see also *ref_net*(1)

Not all manual pages are printed in the reference manuals. Some manual pages, for add-on products for example, are only available online.

You can show the manual pages on the screen with the *man* [see *man(1)*] command. See the manual command for *man* itself for information about using this command. To do this, enter *man man*. You will find information about printing a manual page in the *manprint(1)* manual page. Changes to the manual pages because of new releases are logged in the *man-news(1)* manual page.

fixed-pitch font

Indicates system output, such as error messages, messages, notes, file excerpts and program examples.

fixed-pitch semi-bold font

Denotes user input in examples.

► Indicates activities to be carried out by the user.



Highlights additional information that should be noted to assist understanding of the surrounding text passages.



Indicates the possible risk of data loss.

2 What is an administration domain?

An administration domain is a group of nodes that are linked via a network and managed jointly. This chapter provides more detailed information about:

- The advantages of an administration domain
- The administration system on which this administration domain is based (operating system and administration domain database)
- The advantage of a single administration desktop from which the entire administration domain can be managed
- Specific requirements for operating an administration desktop within a domain

2.1 Advantages of administration domains

The setting up of an administration domain offers improvements in many respects:

- Single point of administration

All nodes in a domain are supported jointly from a single administration desktop and need no longer be operated individually from the respective system console.
- Extensibility

An active node including all its resources can be inserted in or removed from the administration domain at any time.
- Transparency

Because an administration domain appears to a user like a single system (single system view), a cluster of nodes can be administered in the same way as a single operating system. Regardless of the node on which the system administrator has started the user interface, he or she can access the disk storage and other peripherals anywhere in the administration domain at any time for the purpose of administration and configuration.

- Maintenance/troubleshooting

The logbooks for all nodes in the administration domain can be evaluated from a central desktop.

Furthermore, the *Logging* diagnostics system can be configured specifically for each node from a central desktop.

The diagram below attempts to clarify the advantages of an administration domain.

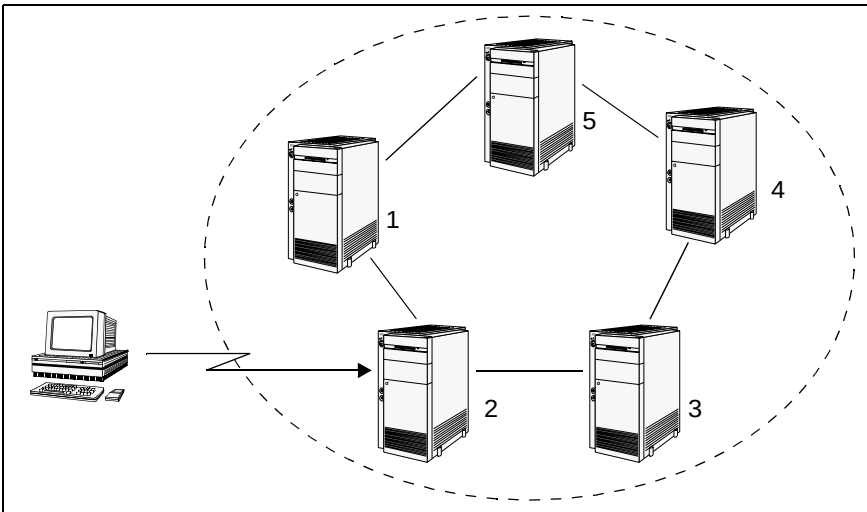


Figure 1: Example of an administration domain with an administration desktop (WSA client)

This administration domain comprises five nodes. The five nodes are connected to one another via TCP/IP (Ethernet). Communication is enabled on this basis for administration purposes. The entire computer network is managed from a single administration desktop (single point of administration) via the *WebSysAdmin* (WSA) user interface. This is why the administration desktop is also referred to as a *WebSysAdmin* client. WSA can be started on any node in the network (e.g. node 2) from which all nodes can be administered. This node is then referred to as the *WebSysAdmin* server. The administration information is updated consistently for all nodes. The five nodes appear to the system administrator like a single system (single system view).

2.2 The domain-wide administration system

The *WebSysAdmin* user interface has been available for some time now for managing single systems. In order to be able to administer several systems in a domain simultaneously, the *DomainAdmin* add-on product has been included in the Reliant UNIX operating system from Release 5.45. The *DomainAdmin* add-on product must be installed on all nodes in the administration domain being set up. Cross-system administration can then be started if LAN connections exist to other systems.

2.2.1 Installing DomainAdmin

The licensed *DomainAdmin* product is contained on the current system CD and is called *CD-SYS-MI_1*; it is a component of the Reliant UNIX 5.45 CD set. The product is installed with *SYSADM*.

The “Reliant UNIX Operation” manual describes how to install an add-on product with *SYSADM*.

When you have installed the *DomainAdmin* add-on package with the *SIclos* and *SItelx* packages, you can start using the domain-wide administration system via the *WebSysAdmin* user interface.

 A domain with the internal name CADMIN is set up automatically when you install the *SIclos* package. This domain is still “empty”, i.e. it does not contain any nodes as yet. You can find out how to fill this domain with nodes in the section “Creating a domain” on page 18.

2.2.2 The CLOS domain operating system

When you install the *DomainAdmin* product, a separate operating system is supplied for domain administration. This operating system is called *CLOS*.

CLOS stands for Cluster Operating System. The name can be explained by the fact that a domain was previously also termed a cluster. Clusters in the current sense (RMS, SPS, ...), however, are simply special cases of an administration domain. You will find a technical description of the *CLOS* administration system in the *clos(7)* manual page.

2.2.3 The CLODB administration domain database

The administration system needs data on all nodes in a domain from which it takes its administration information. This data is stored in several files, with all local information only being stored on the respective node. Files containing global data, on the other hand, (e.g. VDisk Lite configuration data) are stored on all nodes. The volume of such data and files required by the administration system represents the database for the entire domain. The distribution and verification of the consistency of these files is the most important task of the administration system.

The domain database is called *CLODB* (Cluster Object Database) in accordance with the *CLOS* administration system. You will find a technical description of the *CLODB* database in the *clodb(7)* manual page.

2.3 The administration desktop

Each Reliant UNIX system has a system console. Apart from system configuration, this console is used to report events that affect the node administration. In general, these are system messages that require intervention by a system administrator.

It is not necessary for the system administrator in an administration domain to monitor several nodes at the same time. There is just one single administration desktop in an administration domain for monitoring and configuration purposes. From the user's perspective, an administration domain is like a single uniform system.

The administration desktop is not logically assigned to any particular node, rather always physically to one node. When we speak of central administration, this does not simply mean displaying and reacting to messages, rather the entire administration of this domain is carried out on one desktop.

The WebSysAdmin (WSA) user interface

The *WebsysAdmin* graphical user interface is the user interface on the administration desktop. This user interface is supplied as standard with the Reliant UNIX operating system from Release 5.45. Following installation of the *Domain-Admin* add-on product, not only is it possible to have a local view of the system, rather a domain-wide view of all systems.

The *WebSysAdmin* management desktop offers several advantages for operating domains:

- Quick overview of the entire administration domain
- Possibility to perform the same activities on several nodes with one single action
- Secure access to shared resources
- Easy access to operations that would otherwise only be feasible with complex commands

2.3.1 Administration desktop requirements

Various requirements must be fulfilled to operate the administration desktop or *WebSysAdmin* client.

2.3.1.1 WebSysAdmin clients under Windows 95/98/NT (PC console)

The administration desktop should be a PC system with at least a 200-MHz processor and 128-Mbyte memory, which is connected to the LAN.

- The Netscape browser from Version 4.5 is supported.
- A virtual Java machine (JRE 1.2 or JVM 2 as it is also known) is required on the client. If such a machine is not available, an installable version can be downloaded into a selected directory from the server when *WebSysAdmin* is started by following the instructions given. The virtual Java machine is active once the client has been installed and restarted.

2.3.1.2 WebSysAdmin clients under Solaris 7 (LAN console)

The following steps must be followed to start up a *WebSysAdmin* client:

- Install the *WebSysAdmin* components from the *WSA V545B00 for Solaris LAN-Console* Reliant UNIX CD-ROM. To do this:
 - Mount the CD-ROM in the drive of the LAN console.

The File Manager window opens automatically showing the contents of the CD-ROM.

The *installer* script is started from this window:

- ▶ Double-click *installer*. From the list displayed, choose the *WebSysAdm GUI LAN Con* product.
- ▶ Remove the CD-ROM by selecting *File* → *Eject* from the menubar of the File Manager window.
- Install the Java runtime system (SMAWjvrt) from the *CD-SYS-SOL* CD-ROM.
 - ▶ Mount the CD-ROM in the drive of the LAN console. The File Manager window opens automatically showing the contents of the CD-ROM.
 - ▶ Now click *SMAWjvrt*.
 - ▶ Read the *Readme.txt* file.
 - ▶ Now click *Intel*.
 - ▶ Read the *README.i386* files and follow the instructions given.

2.3.1.3 WebSysAdmin clients via VGA console with X11 interface (on RM200, RM300 and RM400)

The following steps must be followed to start up a *WebSysAdmin* client.

- Install the CDS++ runtime system from the CD-ROM *CD-DEV-MI* (with *SYSADM*)
- Extend the *\$PATH* variable by */opt/wsa/bin*
- Install JENV (Java Runtime Environment) from the *CD-SYS-MI_1* CD-ROM (with *SYSADM*)



The color resolution of the VGA screen must be set to a value other than 16 bits.

3 Domain administration with WebSysAdmin

Domains are administered using the *WebSysAdmin* user interface. This chapter provides more detailed information about:

- The user interface in general
- Starting the user interface
- The individual elements of the user interface

3.1 Features of WebSysAdmin

WebSysAdmin is a Web-based user interface for performing system administration of RM systems. Compared to system administration at command level, this interface is easy to use and allows administration to be performed efficiently even in the case of large system networks.

WebSysAdmin currently contains the following standard components:

- *User Management*

This component offers functions for managing users and groups, for example creating, modifying and deleting user and group definitions or managing user passwords.

- *VConfig*

This component contains functions for configuring and managing the hardware both for a single node as well as for a number of nodes in an administration domain. Furthermore, support is also provided for managing storage devices, multipath device files, file systems and virtual disks.

- *Diagnostic*

The *Diagnostic* component offers functions for domain-wide evaluation of log data, for monitoring as well as for administering the log system itself.

- *Software*

This component enables management of the installed software packages (Unix packages) on all nodes in a domain or on a local node.

- *Tasks*

The *Tasks* components allows the user *root* to manage task definitions both for a local node as well as for all nodes in a domain. Task definitions correspond to *cron* jobs and thus entries in the *crontab* file.

Because the user interface is self-explanatory, the components can be used intuitively to a large extent. Should questions arise nevertheless, the integrated help texts provide possible answers. For this reason, the explanations below are purposely kept short and concise and are restricted to the essential elements.

3.2 Starting WebSysAdmin

WebSysAdmin can be invoked in both write and read mode. However, the user interface may only be invoked in write mode on a particular server by one user at any one time. If other users want to start *WebSysAdmin* on the same server, this is only possible in read mode.

3.2.1 Starting WebSysAdmin under Windows 95/98/NT (PC console)

- ▶ Start your browser.
- ▶ If you want to invoke the user interface in **write mode**, make the following entry in the URL line:

`http://hostname:8881`

e.g.: `http://spinlock.pdb.fsc.net:8881`

If you want to invoke the user interface in **read mode** only, enter the following:

`http://hostname:8882`

The user interface is loaded from the specified server and the welcome screen appears on your monitor.

- ▶ Choose the required dialog language (default = English) under *Language* on the menubar and then click one of the application components to the bottom left. A dialog box appears and the welcome screen now looks as follows:

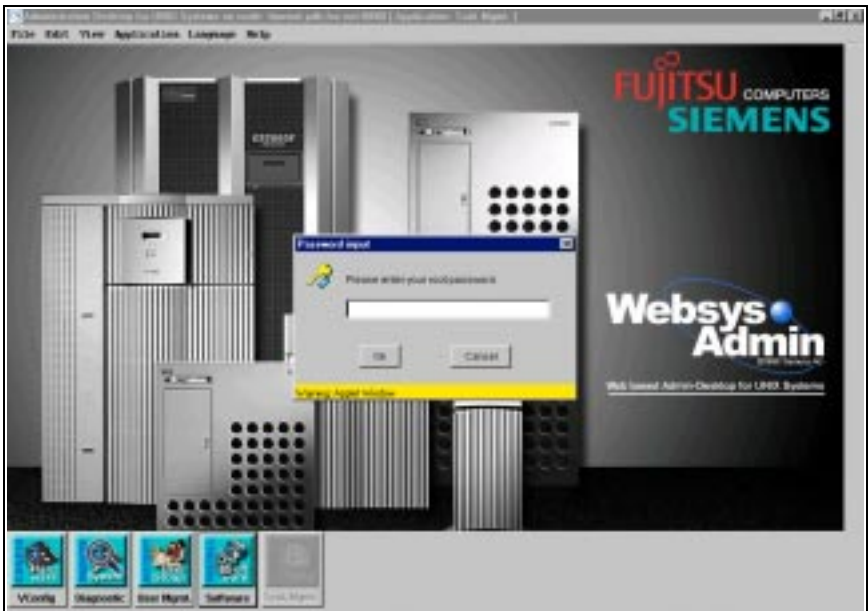


Figure 2: Welcome screen for WebSysAdmin in write mode: Tasks component selected

- Enter the user name and the root password.

You can now start domain administration.

i *WebSysAdmin* requires a minimum screen resolution of 1024x768, so that all displayed information is legible.

3.2.2 Starting WebSysAdmin under Solaris 7 (LAN console)

To start the user interface in **write mode**, enter the following at command level:

```
# /opt/wsa/bin/wsa hostname
```

To invoke the user interface in **read mode** only, enter the following:

```
# /opt/wsa/bin/wsa -r hostname
```

A dialog box then appears and you proceed as described (language selection, application component selection, user name, password).

3.2.3 Starting WebSysAdmin via the VGA console (on RM200, RM300 and RM400)

To start the user interface in **write mode**, enter the following at command level:

```
# wsa hostname
```

To start the user interface in **read mode** only, enter the following:

```
# wsa -r hostname
```

A dialog box then appears and you proceed as described (language selection, application component selection, user name, password).

3.3 Elements of WebSysAdmin

The following diagram shows an example of a window for the *VConfig* component, after you have entered the user name and the root password:



Figure 3: Example: Window for VConfig component (excerpt)

i You will know that the *DomainAdmin* add-on product is installed, because system information is displayed for the nodes “spinlock” and “wait” below *cluster* on the left side of the hierarchical browser (see the section “Hierarchical browser” on page 14). If *DomainAdmin* were not installed, only a single system would be displayed.

The window comprises the following elements:

3.3.1 Menubar and toolbar

You can execute various functions from the menubar and toolbar. Some of the actions are dependent on the context and cannot always be executed. Actions that cannot be executed in a particular context are grayed out on the screen.

Menubar	Toolbar	Action
<i>File</i>		Print object list
	—	Export object
	—	End session
<i>Edit</i>		Cut object
		Copy object to clipboard
		Insert object
<i>View</i>		Update display
<i>Application</i>	—	Start applications
Application-specific menu (the selected tab appears each time, e.g. <i>Nodes</i>)		Create object
	—	Change object
		Delete object
	—	Copy object to another node within an administration domain
	—	Other possible application-specific actions
<i>Help</i>	—	Overview (application-specific)
	—	Contents (application-specific)
	—	About ... (version information)
—		Display system information
—		Choice list: Select objects
—		Choice list: Cancel object selection
—		Abort loading

3.3.2 Selection configurator

The toolbar also offers a selection configurator, which you can use to find objects using the search algorithm for regular expressions and operators. The search output can be restricted as a result.



Figure 4: Example of a selection configurator

3.3.3 Hierarchical browser

The hierarchical browser serves as a navigation aid, which you can use

- in an administration domain to select both the entire administration domain as well as individual nodes or individual hardware components (the latter only in *VConfig*).
- on a local node in the *VConfig* application to select individual hardware components.

Depending on what you select, the object list (see section “Object table” on page 16) will show the objects for the entire administration domain, the selected node or the selected hardware component.

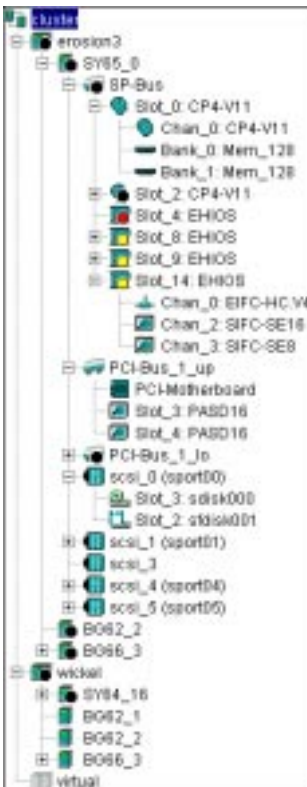


Figure 5: Example of a hierarchical browser in an administration domain

The hierarchical browser shows the following information about the status of the individual objects:

Object icon is highlighted in green

The object can be selected.

Object icon is grayed out

The object cannot be selected at present, because it has been powered down, for example.

Yellow dot

The object is faulty. It can remain in service for the moment but should be replaced at an appropriate time.

Red dot

The object is defective and must be replaced.

- Red-black dot
- The object is defective and must be replaced. It will be deactivated the next time the system starts up.
- Black dot
- The object is deactivated.
- Black-yellow dot
- The object is still deactivated. It will be activated again the next time the system is started, but is faulty.
- Black-green dot
- The object is still deactivated. It will be activated again the next time the system is started and is fault-free.

3.3.4 Object table

The object table comprises the following areas:

- The upper area comprises the **tab bar**. The type and number of tabs depends on the application selected.



Figure 6: Example of a tab bar

- The **object list** appears below the tab bar. The type and number of objects displayed depends on the tab or sub-tab selected (see below) as well as on the selections made in the hierarchical browser and selection configurator.

Node	WSA Domain Net Address or Alias	Operating System	Hardware	System ID	Node Status	OS Status
 spinlock	spinlock	SINBY 5.45 D6332	RM800 40840 R10...	9270580081	INTEGRATED	NOT_INSTALLED
 wall	wall	SINBY 5.45 D6332	RM800 50424 R10...	0611	INTEGRATED	NOT_INSTALLED

Figure 7: Example of an object list with tab bar

The first column in the object list shows the same information about the status of the individual objects as in the hierarchical browser (see above).



The lower area may contain a **sub-tab bar**. Its existence as well as the type and number of sub-tabs offered depends on the tab selected.

4 Operating an administration domain

Various different actions are involved in operating an administration domain. This chapter introduces some standard actions, illustrated with working examples. The standard actions are as follows:

- Setting up a domain and creating and removing a node.
- Creating and removing a user
- Creating a virtual disk (vdisk)
- Analyzing and resolving errors
- Configuring cross-domain events
- Making network file systems available
- Comparing and checking software versions across the domain
- Transferring tasks to other nodes

You will also be told how

- to fill an “empty” domain by creating an initial node
- to add two additional nodes to the domain
- to remove nodes from a domain
- to create a new user on a node in a domain
- to copy (clone) this new user to two additional nodes in the domain
- to assign a global password for a user on several nodes
- to create local vdisks on several nodes
- to set up vdisks on shared resources
- to create a module-independent as well as module-specific filter for trouble-shooting
- to create an event and transfer (clone) it to other nodes
- to import a network file system and mount it on other nodes
- to compare the versions of selected software packages and check for consistency
- to create a task and transfer (clone) it to other nodes

4.1 Creating a domain

The section “Installing DomainAdmin” on page 5 explained that a domain with the internal name CADMIN is set up automatically following installation of the *SIclos* package. This domain is still “empty”, i.e. it does not contain any nodes as yet and must still be “filled”. Filling a domain means giving the empty CADMIN domain its first node. The following section shows you the WSA window following installation of the *SIclos* package on the “spinlock” system and after you have selected the *VConfig* component and the *Nodes* entry on the menubar:

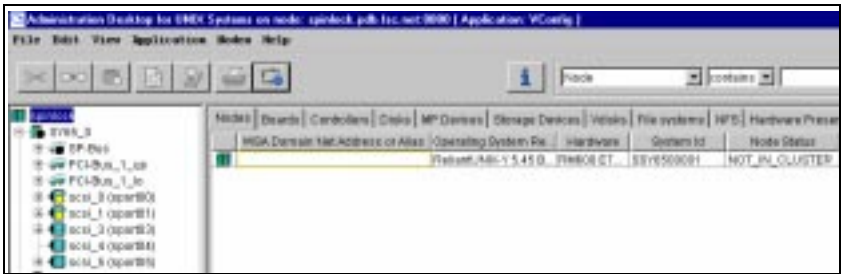


Figure 8: Example of a window for an “empty” domain (excerpt)

The *WSA Domain Net Address or Alias* and *Node status* columns in the object list indicate that the system has not yet been included in a domain. There is no entry in the *WSA Domain ...* column and *NOT_IN_CLUSTER* is entered in the *Node status* column. This means that the “spinlock” node on which you installed *SIclos* and started WSA is itself not a member of the administration domain (the word cluster is often used as a synonym for domain). In this case, the system administrator will possibly have the task of creating this system as the first node in the domain and thus filling the domain.

There are two equivalent approaches to creating the first node in a domain. First of all, select the object in the object list by left-clicking it so that the line is highlighted in blue. Then, you can either

- left-click the *Nodes* entry on the menubar or
- right-click the object.

The following sub-menu is then displayed:

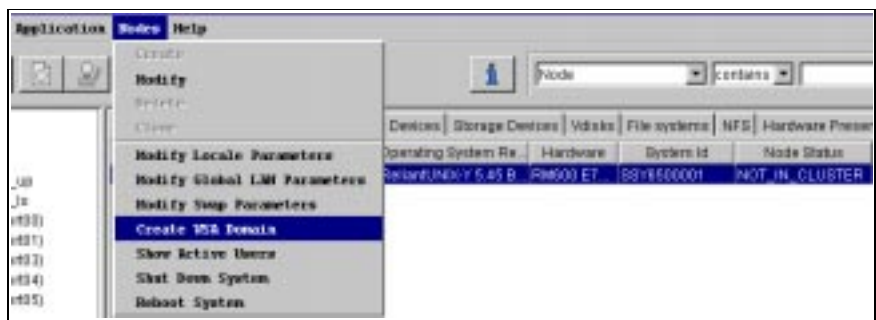


Figure 9: Example of a window for creating a domain and the first node (excerpt)

Now click the *Create WSA Domain* function in the sub-menu.

 To create the first node, you first have to create a domain, since no domain yet exists from the user's perspective.

When you select the *Create WSA Domain* function, the following dialog box opens.



Figure 10: Example of a dialog box for creating a domain

The dialog box for creating a domain contains two fields. In the *WSA Domain Net Address or Alias* field on the right, enter the alias name of the network interface (or the IP address) that is to be used for communication in the administration domain. The node name “spinlock” is proposed automatically in this case for the system; it is taken from the current node name of the UNIX system (*uname -n*). You can accept this name. However, if you want to use a different network interface for domain communication, enter its IP address here in dot notation or alternatively its alias name.

i The configured controllers or the permitted alias names can be established from the *LAN* sub-tab on the *Controllers* tab.

In the *password* field on the right, enter the root password for the node.

When you confirm both entries with *Ok*, an event box pops up, which tells you whether the action was executed successfully:



Figure 11: Example of an event box after creating the first node

You are told that the “spinlock” node now has the status *INTEGRATED*, i.e. the node is integrated in the domain. When you confirm this event box with *Ok*, the new status is also displayed in the object list in the *Node status* column:

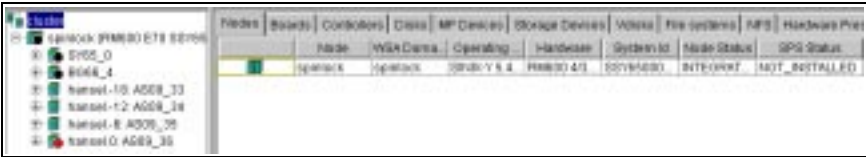


Figure 12: Example of the window after creating the domain and the first node (excerpt)

A new *Nodes* column has now been added to the left in the object list with the entry “spinlock”, and the “spinlock” node is also integrated in the domain with the same domain alias name (*WSA Domain Net Address or Alias* column); as a result, the domain is no longer “empty”.

You can also see from the hierarchical browser that an administration domain now exists, which contains the “spinlock” node: The *cluster* icon has been entered at the top position.

4.2 Deleting a domain

To delete a domain, all integrated nodes with the exception of the server node (server node = node on which WSA was started) must first be deleted successfully (see the section “Removing a node” on page 23).

If the administration domain ultimately only contains a single node, select this node in the object list, choose the *Nodes* menu option on the menubar and then the *Delete WSA Domain* function. A message box is then displayed indicating that the domain will be deleted. If you confirm this box with *Yes*, an event box is displayed indicating that the domain has been deleted. You can now create a domain again and “fill” it as described in the section “Creating a domain” on page 18.

4.3 Adding a node

If you have created the domain as well as the first node and then want to add further nodes to the domain, you can do so as follows. If you have not already done so, switch to the *VConfig* application. You now have two options:

- Click the  button (create object) on the toolbar or
- Choose the *Nodes* menu option on the menubar.

In the latter case, a sub-menu opens where you choose the *Create* menu option:

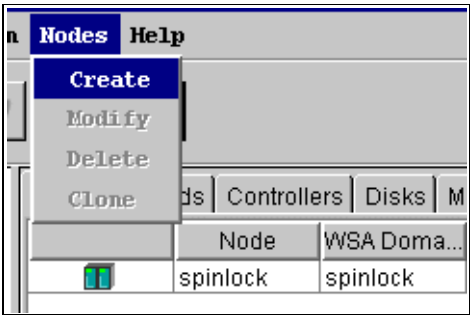


Figure 13: Example of a window for creating additional nodes (excerpt)

When you select *Create*, the *Create node* dialog box opens. The figure below gives you an example of how to fill this dialog box:



Figure 14: Example of the dialog box for creating an additional node

The name of the UNIX system (*uname -n* command) must be chosen for the node name (in this case: *wait*). Once the node name has been entered, it is proposed as the domain net address or the alias. Finally, enter the root password and confirm with *Ok*. The newly added node is then also displayed in the main window.



A node can only be administered fully within the domain if it can access all other nodes and if the integrated nodes can also access it using the domain net address (it must be possible to successfully execute the *ping* command from all nodes).

New nodes can be added successively to the domain by following the procedure described. Once you have successfully created the nodes, you can use the object list in the main window to establish which nodes are integrated in the domain. In the figure below, the domain comprises three nodes, i.e. “post”, “spinlock” and “wait”.

Nodes	Boards	Controllers	Disks	MP Devices	Storage Devices	Vdisks	File systems	NFS	Hardwa
Node	WSA Domain ...	Operating System Release	Hardware		System Id	Node Status			
post	post	SINIX-Y 5.45 B0030	RM600 4/768 R100...		SSY6404357	INTEGRATED			
spinlock	spinlock	SINIX-Y 5.45 B0032	RM600 4/3840 R10...		SSY6500001	INTEGRATED			
wait	wait	SINIX-Y 5.45 B0032	RM600 8/3424 R10...		0011	INTEGRATED			

Figure 15: Example of a domain window with three nodes (excerpt)



The RM200 is not supported as a node in a domain.



The *CLODB* database in the administration domain is transferred to new nodes as they are created.

All local information on the new nodes, which is stored in the database, is then lost (e.g. VDisk Lite configurations).

4.4 Removing a node

In order to remove a node from an administration domain, select the node you want to delete in the object list and then choose *Delete* in the sub-menu. An important warning is then issued:



Figure 16: Warning issued when removing a node

i The warning indicates that shared configuration data in the *CLODB* (vdisks) database will be removed on a deleted node.

If you answer *Yes*, the following message box is displayed:

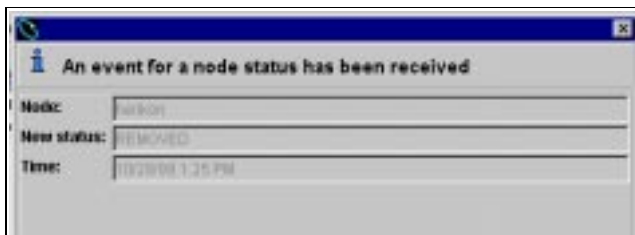


Figure 17: Example of a message box when removing a node (excerpt)

Following confirmation with *Ok*, the deleted node is then no longer displayed in the main window.

i If the administration domain only contains a single node and you want to remove it, you can only do so by deleting the domain fully (see the section “Deleting a domain” on page 21).

4.5 Adding a user

In order to add a user to an administration domain, click the *User Mgmt.* icon in the welcome screen (cf. figure on page 11) after WSA has been started; if you had already started WSA and you are currently accessing a different component, click *Application* on the menubar and choose the *User Mgmt.* application in the sub-menu. You are then shown an object list with all users, who have been created on the node or in the domain.

i When you click the *Name* column header, you can influence the order in which user names are displayed (in ascending or descending alphabetical order).

If you now want to add a new user, click *Users* on the menubar and then choose the *Create* function in the sub-menu; or click the  (create object) button on the toolbar. You are then shown a dialog box and can fill in the fields as you require:



Figure 18: Example of a dialog box for creating a new user

The domain in our example comprises three nodes, i.e. “post”, “spinlock” and “wait”, as the *Node* entry to the top right in the figure indicates. The “spinlock” node is selected and a new user “abraham” is to be created on this node. The remaining input fields are blank or the defaults are copied over, such as the user id (134). A short entry has only been made in the comment field.

When you have filled in the dialog box and confirmed with *Ok*, the new user's home directory is prompted:



Figure 19: Dialog box querying the home directory when creating a new user

You can generally answer *Yes* to this query. This completes the entire action and an *Ok* message is issued:

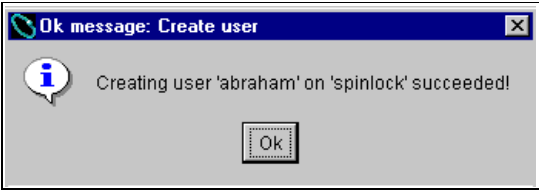


Figure 20: Example of an Ok message for creating a new user

The new “abraham” user is created on the system and entered in the object list of users with its attributes:

Users									
	Name	User id	Node	MB+ domain	Primary group	Home directory	Logon shell	Comment	
	abraham	134	spinlock		other	/home/abraham	/bin/sh	Demonstration	
	adm	4	spinlock		adm	/var/adm	/bin/sh	0000-Admin(0000)	
	adm	4	psd		adm	/var/adm	/bin/sh	0000-Admin(0000)	
	adm	4	wat		adm	/var/adm	/bin/sh	0000-Admin(0000)	
	adm	5	wat		bin	/home/adm	/bin/sh	FACS System Administrator	
	adm	5	spinlock		bin	/home/adm	/bin/sh	FACS System Administrator	
	adm	5	psd		bin	/home/adm	/bin/sh	FACS System Administrator	
	adm	100	wat		other	/home/adm	/bin/sh	Andreas Schaefer, Teil 15062	
	adm	121	spinlock		vm	/home_4400000	/bin/sh	Andreas Schaefer, Teil 15062	

Figure 21: Object list with new user (excerpt)

 Only the first 1000 users are ever listed in the object list. If there are more than 1000 users, a relevant warning is issued.

A newly created user is also not displayed in an object list with more than 1000 entries, if (as in our example) it would have to appear at the very beginning alphabetically. This new user can only be found in this case by performing a search using the selection configurator (see the figure on page 14) on the toolbar.

4.6 Cloning a user

Cloning a user means copying a previously created user including his or her attributes to other nodes in a domain. To do this, select the user to be cloned and click the right mouse button. Then in the sub-menu, click the *Clone* function as shown in the next figure:

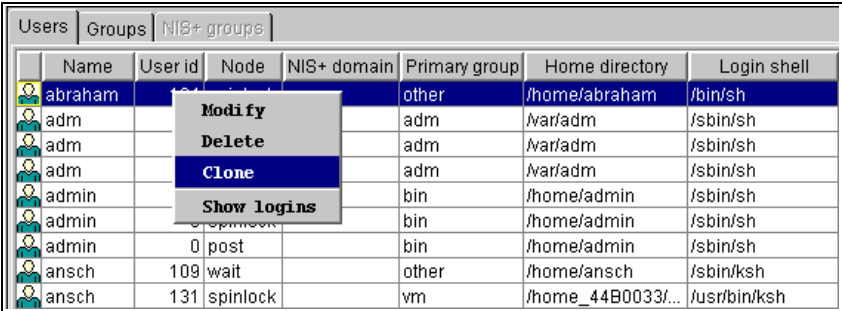


Figure 22: Sub-menu for cloning a user

When you have selected the user to be cloned and the *Clone* function, you then have to select the other nodes in the domain to which the user is to be copied in the next dialog box:

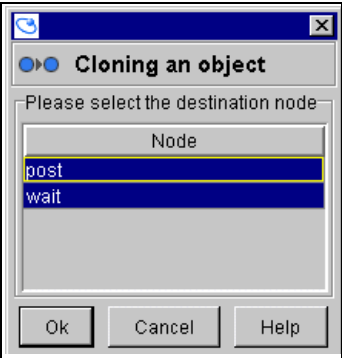


Figure 23: Example of a dialog box for cloning a user

When you confirm the dialog box with *Ok*, the user “abraham” is copied to the two nodes “post” and “wait”. There is no *Ok* message. You can check if the action was successful in the object list of users in the main window. It is only in the event of an error that a relevant message is issued.

i If the user id of a user being cloned is already assigned on a different node, cloning is not possible for this node. An error message is issued to this effect.

4.7 Removing a user

In principle, you remove a user in the same way as you create a user. A detailed description is therefore not necessary. We will simply point out that a user cannot be removed globally from all nodes in a domain rather only individually on the respective node. There is therefore no undo function for cloning. It is possible, however, to remove several users on a node in one action, assuming the appropriate selection is made. The confirmation query prior to deletion then refers to the respective overall selection.

4.8 Assigning a global password

In addition to being able to copy an existing user retrospectively to other nodes in a domain (i.e. cloning), when you add a new user you can create this user immediately on other nodes with the same attributes. To do this, you simply have to select the required node to the top right in the dialog box for creating a new user (see the figure on page 25) in the *Node* column. The new user is then created immediately on all these nodes.

If you want to create a user simultaneously on several nodes in a domain, this user will have the same login name, the same user id, the same group membership etc. on all selected nodes. In addition, you can also assign the same password to this user on all nodes, in other words assign a global or domain-wide password.

In order to assign a global password to a user, choose the *Password* tab to the top left in the mentioned dialog box (see again the figure on page 25) once you have entered the basic data. The following dialog box then opens:



Figure 24: Example of a dialog box for assigning a password

In this dialog box, first select the *password* entry in the *Password status* column (as in the dialog box), then simply enter the global password in the field to the right and reenter it. The global password is then assigned.

You can assign a password globally but you cannot change it in the same way. In order to change a global password, you have to modify it individually on each node.

4.9 Creating virtual disks (vdisks)

If you have installed the *DomainAdmin* add-on product, you are guaranteed that the configuration of virtual disks is consistent throughout the domain. This means:

- Each node in a domain knows the vdisks of the other nodes since this information is stored in the *CLODB* (Cluster Object Database, cf. page 6) database.
- Each time a system is rebooted, the *CLODB* is reconciled.

As with user management, it is possible to create vdisks simultaneously on several nodes in a domain. This is particularly beneficial when managing identical hardware platforms as well as when setting up a vdisk on shared resources for later use of a high-availability application. Two working examples are provided for these two scenarios.



WSA only supports disks created with VDisk Lite and stores these in the *CLODB* administration domain database.

4.9.1 Creating local vdisks on a node

In order to add a vdisk to an administration domain, click the VConfig icon on the welcome screen after you have started WSA (cf. figure on page 11); if you had already started WSA and you are currently accessing a different component, click Application on the menubar and choose the VConfig application in the sub-menu. Then click the VDisks tab. You are now shown an object list with all vdisks, which have been created on the node or in the domain.

In our example, we assume the domain comprises three nodes, i.e. “spinlock”, “timeout” and “wait”. The first step is to create a new group on the “wait” node, since a vdisk must always be assigned to a group.

To create the new group, choose the *VDisks* entry on the menubar and navigate to the *Create a vdisk group* action as shown in the next figure:

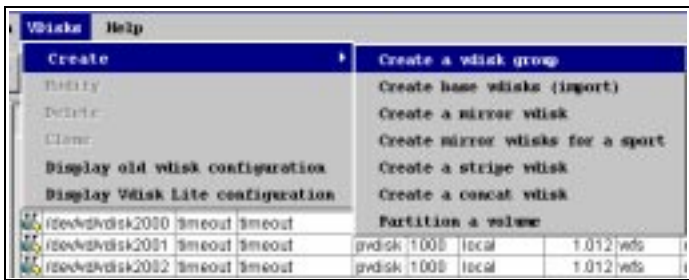


Figure 25: Menu sequence for creating a vdisk group

When you create a vdisk group, a dialog box of the same name opens, which you can fill in as follows:



Figure 26: Example of a dialog box for creating a vdisk group

The individual entries in the dialog box have been filled in accordance with our example and have the following meaning:

Nodes

The “wait” node, on which the new group is to be created, is selected.

If the local vdisk group 3000 is to be created on all nodes, this can be performed in one step by additionally selecting the nodes “timeout” and “spinlock”.

New group

The name of the new group is entered in this field and must comprise a number in the range 1 – 9999. The name or number must not already be assigned.

Statesave

This field contains the name of the statesave vdisk, i.e. the virtual disk for administration information. The name comprises the fixed component `vdisk` and the number of the group, hence in our example `vdisk3000` as well as a valid unassigned vdisk number.

Type

local is selected in the *Type* field, since the new group is only to be used locally on the “wait” node.

Configure level

boot is entered in this field, which indicates that the group is to be configured (activated) automatically when the system boots.

Comment

You can enter any comment up to a maximum of 64 characters in this field.

When you confirm the dialog box with *Ok*, the new group is created on the “wait” node. Now click the *Vdisk groups* sub-tab in the main window. If you created the new group simultaneously for all three nodes, you are shown the following object list as a result of the action:

Nodes Boards Controllers Disks MP devices Storage devices vDisks File systems Hardware location									
Group	Node	Shared with	Group type	Statesave pieces	State	Level	VDisk name	Comment	
1890	timeout	timeout	local	0	mirrored	boot	NodeVdisk1890		
1891	timeout	timeout	local	0	mirrored	boot	NodeVdisk1890		
3890	spinlock	spinlock	local	0	ed-ca...	boot	NodeVdisk3890	Demonstration	
3890	timeout	timeout	local	0	ed-ca...	boot	NodeVdisk3890	Demonstration	
3890	wait	wait	local	0	ed-ca...	boot	NodeVdisk3890	Demonstration	

Tap vdisks | Configuration vdisks | vdisk groups

Figure 27: Example of an object list after creating a vdisk group

The new 3000 group was created on the three nodes “spinlock”, “timeout” and “wait” (*Group* and *Node* column). The group is of type *local* (*Group type* column) and is thus not shared with other nodes (*Shared with* column). Since no disks have been imported into this group as yet (the group is still “empty” so to speak), it also does not contain any statesave pieces (0 in the *Statesave pieces* column) and is also not yet configured (*not-configured* in the *State* column). The *VDisk name* column contains the pathname of the statesave vdisk. The remaining two columns (*Level* and *Comment*) show the respective input in the dialog box.

Once the new vdisk group has been created, you have to fill it in the next step with the disks that are to be used. This step is termed importing a disk. Both steps, creating the group and importing the disks, are mandatory.

The statesave pieces are created on the first imported disks. A maximum of six statesave pieces are used, whereby a maximum of two statesave devices are used for each SCSI port to improve the failsafe performance.

Each virtual disk must be of a particular type. There are several vdisk types depending on the usage [for details of the various vdisk types, see the manual page for *vdisklite(7)*]. To create the simplest *base* vdisk type, choose the *Create base vdisks (import)* action in the sub-menu on page 32. The following dialog box then opens, which may look as follows for example:

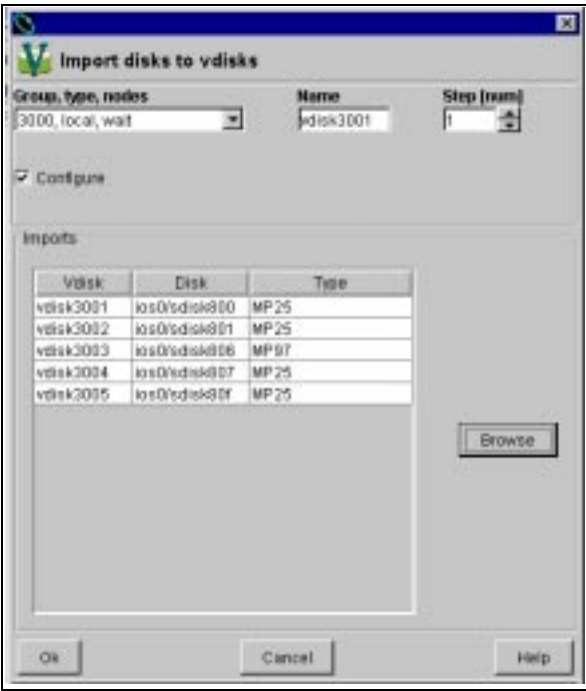


Figure 28: Example of a dialog box for creating base vdisks

The individual entries in the dialog box have the following meaning:

Group, type, nodes

The vdisk group into which the new vdisks being created are to be imported must be selected in this field. In our example, the 3000 group just created of type *local* in the “wait” domain was selected.

Name

This field contains the name of the first vdisk to be created. The field is preassigned the next valid vdisk name.

Step

This field shows the increment for name generation where several disks are imported simultaneously.

Configure

This field is selected so that the imported disk is also configured (activated) after it is created.

Browse

You can use this button to search for the disks that are to be imported. If you click *Browse*, you switch to the disk browser where you can select a range of suitable disks. The disk browser only proposes disks as suitable if they are recognized as a free local resource on the “wait” node. Your selection is displayed in the table to the left with the new vdisk name.

When you confirm the dialog box with *Ok*, the selected disks are integrated in the new 3000 vdisk group on the “wait” node. Now click the *Top vdisks* sub-tab in the main window. The following object list shows you the result of the action:

Nodes Boards Controllers Disks HP devices Storage devices VDisks File systems Hardware location											
VDisk name	Node	Shared with	Type	Group	Group type	Size (GB)	FS Type	ActualMountPoint	State		Comment
vdevvdisk3001	wait	wait	base	3000	local	3.929			configured		
vdevvdisk3002	wait	wait	base	3000	local	3.929			configured		
vdevvdisk3003	wait	wait	base	3000	local	8.472			configured		
vdevvdisk3004	wait	wait	base	3000	local	3.929			configured		
vdevvdisk3005	wait	wait	base	3000	local	3.929			configured		

Top vdisks | Constructive vdisks | Vdisk group

Figure 29: Example of an object list for vdisks

It is clear from the object list that a 3000 vdisk group exists on the “wait” node and contains five vdisks (“vdisk3001” to “vdisk3005”). These vdisks are of type *base* and since the vdisk group is of type *local*, the resources are not shared with any other member of the domain. The vdisks have the status *configured* and can be used immediately on the “wait” node.

4.9.2 Configuring vdisks on shared resources

We described in the previous section how to create vdisks locally. However, since these vdisks all belong to a group of type *local*, they can only ever be used on one node.

We use an example in this section to show how you can create mirror vdisks that are configured on two nodes in a domain. These vdisks are used as a basis for high-availability configurations.

The first step again is to create a vdisk group. The dialog box described in the previous section on page 32 looks as follows for the new example:



Figure 30: Example of a dialog box for creating a vdisk group of type “rms”

The new group is created simultaneously on the two nodes “spinlock” and “wait”. The type *local* may not be selected. The selected *rms* type indicates that this is a vdisk group whose disks can be accessed from several nodes (a so-called multihosted environment).

The only permitted entry in the *Configure level* input field is *manual*. *manual* means that the vdisk group must be configured by the system administrator or from a high-availability monitor.

When you confirm the dialog box with *Ok*, the new group is created on both nodes. If you now click the *Vdisk groups* sub-tab in the main window, you will see the result of the action in the following object list:

Nodes	Boards	Containers	Disks	MP entries	Storage devices	VDisks	File systems	Hardware location	
Group	Node	Shared with	Group type	Grades/no pieces	State	Level	VDisk name	Comment	
1300	lineat	lineat	local	6	mirrored	best	ldvldvdisk1000		
1301	lineat	lineat	local	6	mirrored	best	ldvldvdisk1500		
3000	wait	wait	local	2	mirrored	best	ldvldvdisk3000	Demonstration	
3000	spinlock	spinlock	local	0	not configured	best	ldvldvdisk3000	Demonstration	
3000	lineat	lineat	local	0	not configured	best	ldvldvdisk3000	Demonstration	
4000	spinlock	spinlock,wait	rms	0	not configured	manual	ldvldvdisk4000	Demonstration	
4000	wait	spinlock,wait	rms	0	not configured	manual	ldvldvdisk4000	Demonstration	

Figure 31: Example of an object list after creating a vdisk group of type “rms”

The new 4000 group has been created on the two nodes “spinlock” and “wait”. The main difference compared to the object list in the first example (cf. figure on page 34) is that the *Shared with* column now shows the node names “spinlock” and “wait”. This indicates that the vdisk definitions for this group refer to resources shared by the two nodes. The systems must share these resources, which means that access must be coordinated.

In our previous example of creating local vdisks, the first step after the vdisk group was created was to import the disks. This example now discusses the creation of mirror vdisks. Because all disks are to be mirrored in two “multi-

hosted” connected SCSI lines, the *Create mirror vdisks for a sport* (cf. figure on page 32) is selected instead of the action *Create base vdisks (import)*. The following dialog box then opens:

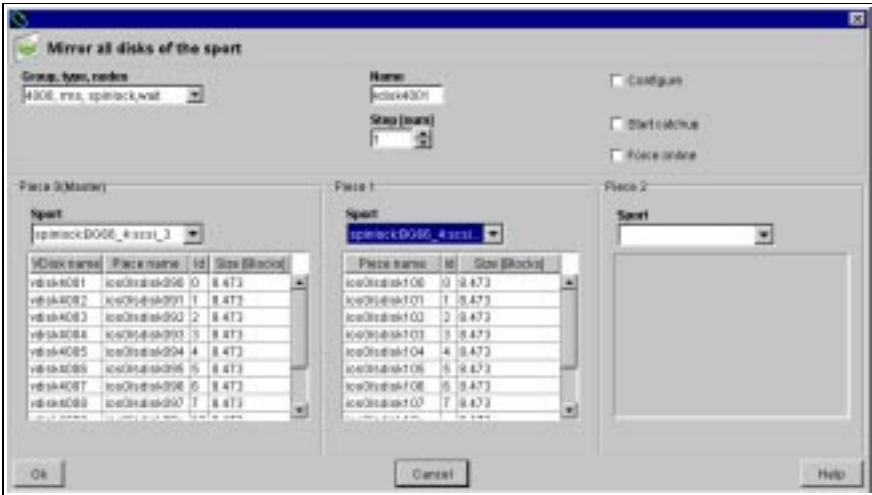


Figure 32: Example of a dialog box for mirroring a SCSI line

The dialog box, which has been filled in accordance with our example, confirms that on node “spinlock” all disks for SCSI line 3 in cabinet BG66_4 (spinlock:BG66_4:scsi_3) are to be mirrored on SCSI line 4 in the same cabinet (spinlock:BG66_4:scsi_4). When Piece 0 is selected in the left selection box, all available disks including their parameters are listed.

Nothing is selected in the status fields to the top right of the dialog box, (i.e. *Configure*, *Start catchup* and *Force online*), because shared resources should only be configured on high-availability applications (RMS, Observe, OPS) and not automatically via WSA. Warnings are also issued to this effect if you use WSA for configuration or deconfiguration.

When you confirm the dialog box with *Ok*, the line is mirrored and can be used on both nodes, i.e. “spinlock” and “wait”. If you now click the *Top vdisks* sub-tab in the main window, you will see the result of the action in the following object list:

Nodes Boards Controllers Disks MF devices Storage devices VDisks File systems Hardware locations											
VDisk name	Node	Shared with	Type	Group	Group type	Size (GB)	FS Type	Actual Mount Point	State		Comment
ldenvdisk4001	wait	wait	base	3000	local	3.929			configured		
ldenvdisk4002	wait	wait	base	3000	local	3.929			configured		
ldenvdisk4003	wait	wait	base	3000	local	3.973			configured		
ldenvdisk4004	wait	wait	base	3000	local	3.929			configured		
ldenvdisk4005	wait	wait	base	3000	local	3.929			configured		
ldenvdisk4006	wait	spinlockwait	mirror	4000	ema	8.473			not configured		
ldenvdisk4007	wait	spinlockwait	mirror	4000	ema	8.473			not configured		
ldenvdisk4008	wait	spinlockwait	mirror	4000	ema	8.473			not configured		
ldenvdisk4009	wait	spinlockwait	mirror	4000	ema	8.473			not configured		
ldenvdisk4010	wait	spinlockwait	mirror	4000	ema	8.473			not configured		
ldenvdisk4011	wait	spinlockwait	mirror	4000	ema	8.473			not configured		
ldenvdisk4012	wait	spinlockwait	mirror	4000	ema	8.473			not configured		

Top vdisks | Configuration vdisks | Vdisk groups

Figure 33: Example of an object list with RMS vdisks

The object table shows the result of the action from the view of the “wait” node. This view is identical with the view of “spinlock”. The individual vdisks in this group are of type *mirror*, since the entire SCSI line (vdisk4001 to vdisk4012) was mirrored. The newly created mirror disks are not yet configured (*not-configured*). To configure them, you have to use a high-availability monitor, as already mentioned.

A warning is issued in the event of a manual configuration via WSA. Note that the RMS vdisk group was only configured on one system, as otherwise inconsistencies could arise in relation to the mirror status as well as the data itself. Where the vdisk is used in parallel, consistent mirror states can only be ensured for OPS/OPSRES vdisks.

You will find additional information on managing vdisks in the manual “Reliant UNIX – Configuring Virtual Disks with VDisk Lite”.

4.10 System diagnosis

The diagnostic system can be configured from the central administration desktop for all nodes in a domain and is used for troubleshooting and diagnostics. Various functions are available for configuration and evaluation purposes.

Domain-wide configuration

- Log administration

Among other things, log administration allows the size and number of logbooks to be defined as well as the logbook components to be assigned.

- Event management

Event management allows events and reactions to be defined. All system messages are observed from a monitor. When the defined events occur, the assigned reactions are performed, for example defining a component as defective or information messages for the administration desktop.

- Monitoring

The monitoring function allows error thresholds to be defined for storage and SCSI components, with the result that the components are marked as faulty if these values are exceeded and possibly deconfigured. It is a component of the LAR concept (LAR = Logout Auto Recovery).

Domain-wide evaluation

Domain-wide evaluation allows all logbooks to be displayed for nodes or their entries. The performance is dependent in this case on the number of nodes and the size and number of logbooks.

When you switch from *VConfig* (via the *Application* action on the menubar) the node selection performed in the hierarchical browser is also effective in the *Diagnostic* application, with the result that only the logbook for the node selected in *VConfig* is evaluated.

The system administrator can basically follow two approaches when troubleshooting:

1. General troubleshooting across all nodes in a domain
2. Object-specific troubleshooting for a specific error message

Both approaches are demonstrated below using two examples.

4.10.1 General troubleshooting

System administrators must be able to analyze and resolve errors quickly when they occur. One method of reducing the particularly large volume of information in the various logbooks in the administration domain is to use search criteria.

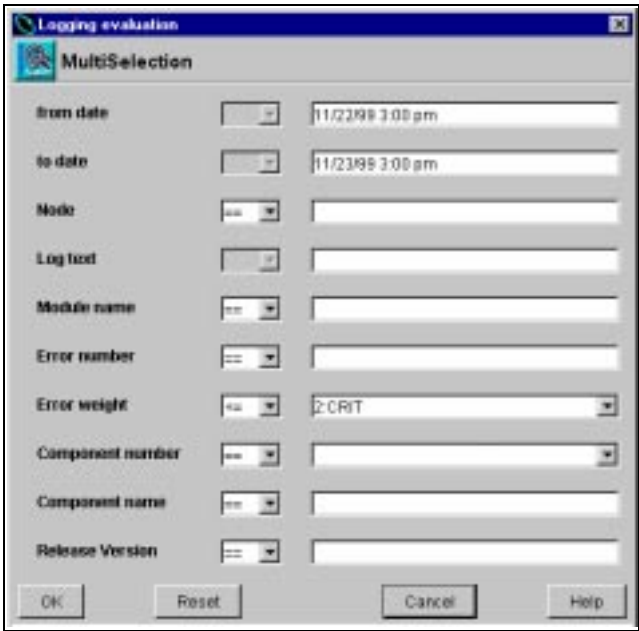
Two options are available to you in this case: the selection configurator and the filter action.

In the **selection configurator** (see the description on page 14 earlier) – as is also the case with the other applications – it is only possible to specify a search criterion + operator, in order to search, for example, throughout a domain for all log entries with a specific weighting or for messages from a particular driver.

However, the following example uses *two* criteria (time period and weighting), in order to reduce the result set further. A filter is needed for this purpose.

Regardless of the object, the **filter** action allows serious errors to be sought specifically for a defined period. For example, a system administrator may want to view all errors less than or equal to ($\leq$) 2, which have occurred in a domain within the last 24 hours.

The example assumes a single domain, which comprises the two nodes “feile” and “unbidden”. Since you may want to record all serious errors that occur in the domain, select the top option in the hierarchical browser, i.e. *cluster*. You can then use the *Logevaluation* menu option and the *Filter* function to open the dialog box where you can specify the filter:



The 'Logging evaluation' dialog box features a 'MultiSelection' tab. It contains several input fields with dropdown arrows: 'from date' (11/23/99 3:00 pm), 'to date' (11/23/99 3:00 pm), 'Mode', 'Log level', 'Module name', 'Error number', 'Error weight' (2:CRIT), 'Component number', 'Component name', and 'Release Version'. At the bottom are 'OK', 'Reset', 'Cancel', and 'Help' buttons.

Figure 34: Example of an object-independent filter

Only the key data for the time period and the error weighting is shown in this object-independent filter. The period covers 24 hours, and the weighting is set to less than or equal to ($\leq$) 2, i.e. all errors that require a reaction soon or immediately (2:CRIT).

 You are shown the precise format for the date fields by dragging the cursor to the *from date* entry.

When you exit the dialog box with *Ok*, all hits are shown in the object list:

Logging evaluation							
Log administration		Event Management		Monitoring			
	from date	mode	Log level	Module name	Error no.	Error weight	
	11/23/99 3:15 PM	file	System Component	ias3/stape003: Limit exceeded on...	0 EE_TAPE	2:ALERT	
	11/23/99 10:58 AM	unhidden	System Component	ias3/stak879 (SEAGATE ST112...	0 EE_DISK	2:CRIT	

Figure 35: Example of an object list with faulty devices based on a filter

In our example, one faulty object was found on each domain node: a tape device on the node “feile” and a defective disk on the node “unbidden”. You can now display the detailed log entry by double-clicking one of the objects, for example the second object (“unbidden” node):

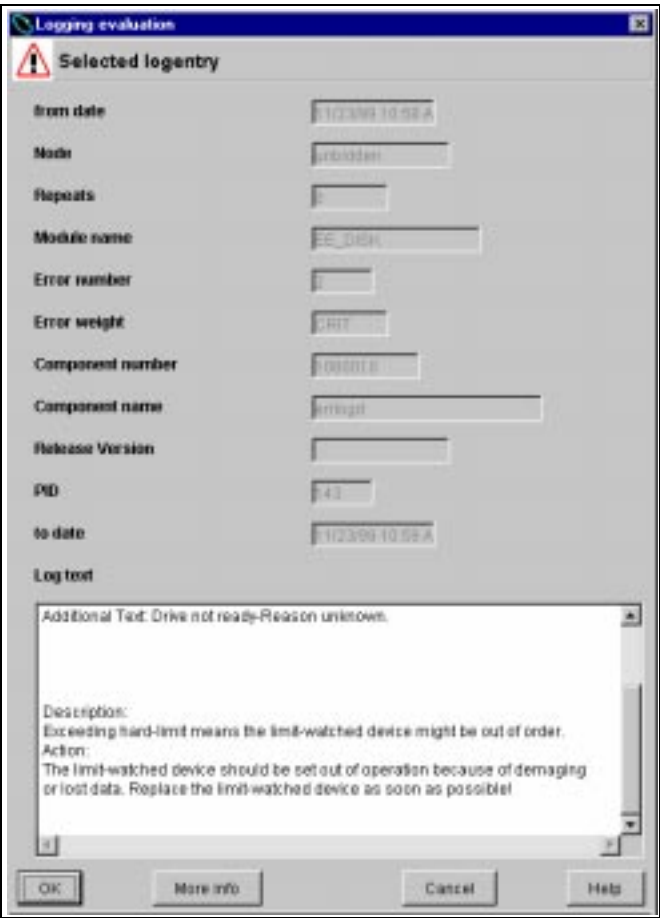


Figure 36: Example of a log evaluation for a faulty disk

This is an example of a log entry comprising a log data header, which identifies the log entry for evaluation purposes (top gray area) as well as several log data records (lower white area) containing a description of the error in the *Log text* field. The administrator is also finally informed here of whatever action he or she should take. In our example, the affected disk is to be replaced.

You then repeat the same log evaluation action for the other object displayed in the object list.

4.10.2 Object-specific troubleshooting

The following example assumes a concrete error situation in order to illustrate the various options for analyzing the error.

When the data backup starts, a particular tape device cannot be accessed. If you start WSA in such a case, you will be informed of the relevant problem directly after you log on in the form of a message, which could have the following contents:

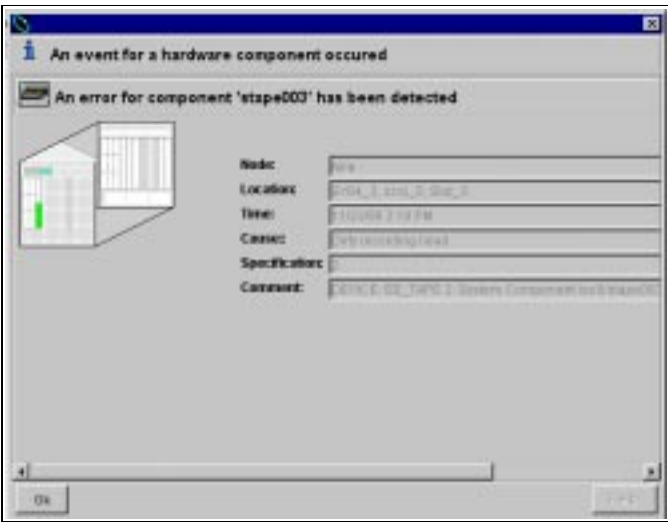


Figure 37: Example of a message regarding a hardware error

The event message already contains the essential information: A tape device (stape003) with a contaminated read/write head was discovered on node “feile” on 23.11.99 at 2:18 PM in the system cabinet SY64_3. The error is serious (error weighting 2) and should be resolved immediately.

Based on the description, the system administrator can now proceed to further analyze the error. The normal way is to switch first to the *VConfig* application and have all devices displayed on the *Storage devices* tab.



Figure 38: Example of an object list with faulty tape device and action selected for analyzing the error

The yellow dot in the object list shows that the “stape003” device is faulty (cf. explanations on the hierarchical browser on page 14). Furthermore, the device is also set to *error* in the *Status (LAR)* column. If you now select the “stape003” device and choose the *Display LAR entries* entry, the following error entry is displayed:

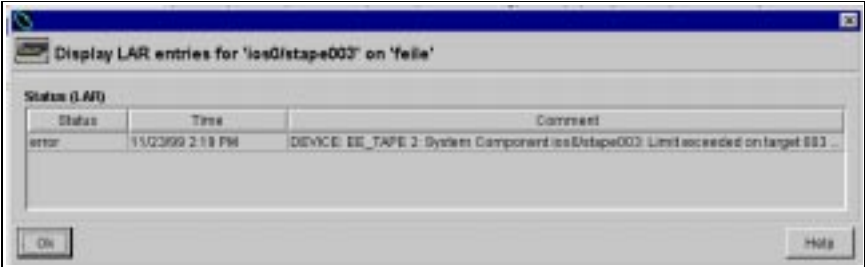
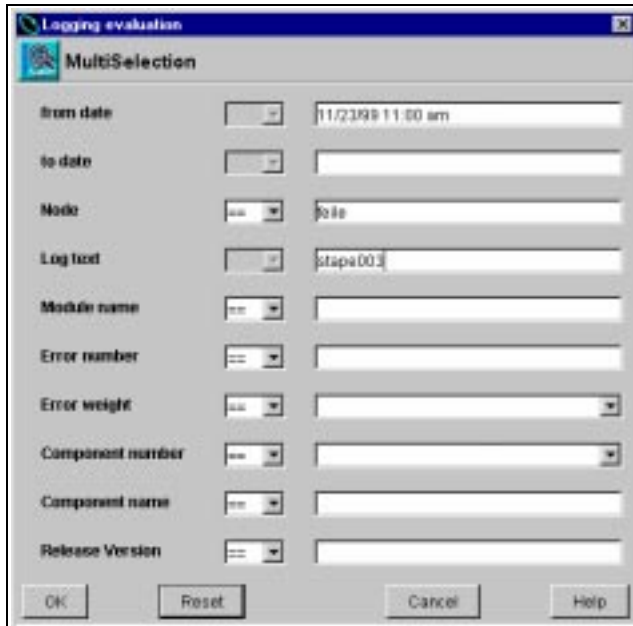


Figure 39: Example of a display box with specification of an error in a SCSI device

In our example, the display box contains an entry about the faulty tape device “stape003”. The error itself is not characterized very precisely with the result that a more detailed analysis is recommended. This is possible using a filter.

To create a filter, switch to the *Diagnostic* application and click the *Logevaluation* menu option on the menubar. Finally, in the next sub-menu choose the *Filter* action, as a result of which a dialog box opens:



Field	Value
from date	11/23/99 11:00 am
to date	
Node	Node
Log text	stape003
Module name	
Error number	
Error weight	
Component number	
Component name	
Release Version	

Figure 40: Example of a filter for a faulty tape device

The dialog box contains various input fields for data that are of interest with regard to a filter. In our example, three fields are filled in. First of all, the period of time during which diagnostics is to be performed has been specified (the last four hours). Then, the name of the node on which the error was reported is entered in the *Node* field. Finally, the specific tape device for which the error was reported is entered in the *Log text* field. The scope of the reported error is thus marked out sufficiently and you can close the dialog box with *Ok*.

When the dialog box is closed with the key data for the filter, an object list is displayed in the main window with the faulty tape devices that have been found. If you select the relevant device and have it displayed with the *Logevaluation* menu option, you will be shown a detailed evaluation of the error based on the entries in the logbook:

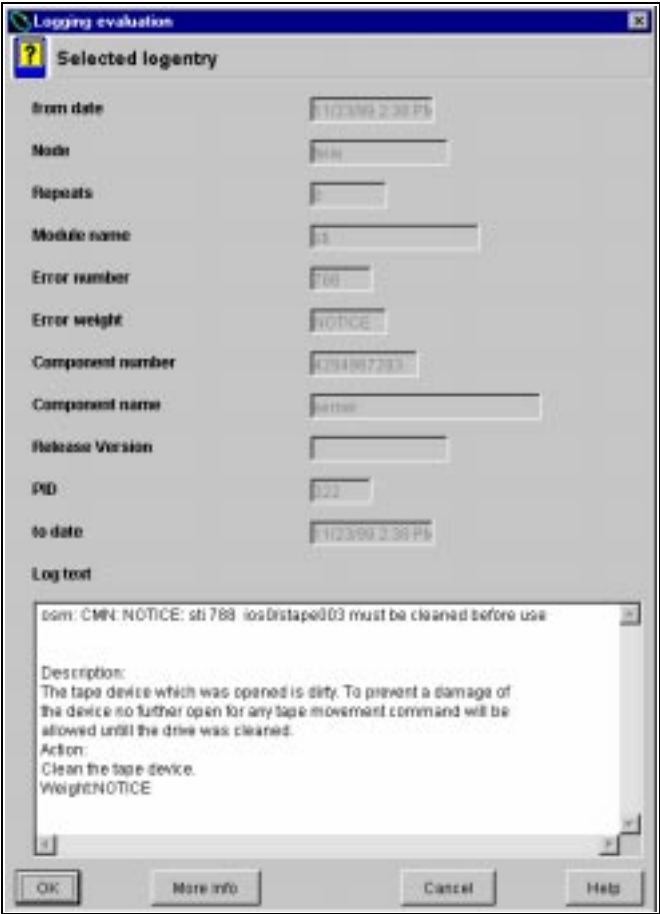


Figure 41: Example of a filter evaluation for a faulty tape device

This is an example of a log entry comprising a log data header, which identifies the log entry for evaluation purposes (top gray area) as well as several log data records (lower white area) containing a description of the error in the *Log text* field. The administrator is also told finally what action to take in the specific error situation. In our example, the necessary measure is clear: The tape device is dirty and hence must be cleaned.

4.11 Event and reaction definitions

Event Management uses messages and the frequency of their occurrence to define events and trigger reactions in accordance with these events.

For example, if you want to take up the option of receiving all system disruptions or warnings throughout the domain by e-mail, you have to proceed as follows:

- 1. Create an event definition.
- 2. Clone this event definition.
- 3. Create a reaction definition.
- 4. Clone this reaction definition.
- 5. Activate the modified event and reaction definitions.

These steps are now illustrated individually using the following example.

Step 1: Create event definition

- Select a suitable entry in the *Unit: Event Name* column in the object list (in the example: *cmn: Memory Multi Bit Error Reached*) and right-click it.

The following sub-menu appears:

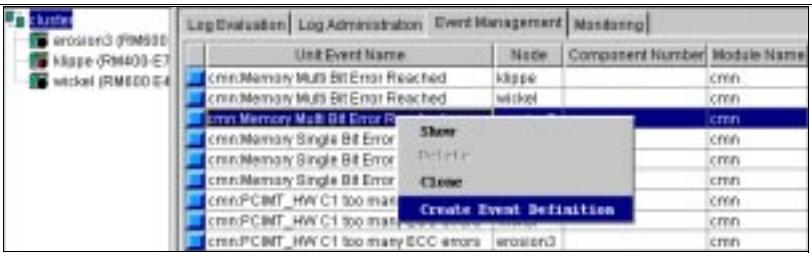


Figure 42: Example of an object list for creating an event definition (excerpt)

- ▶ Click *Create Event Definition*. The following dialog box is displayed:

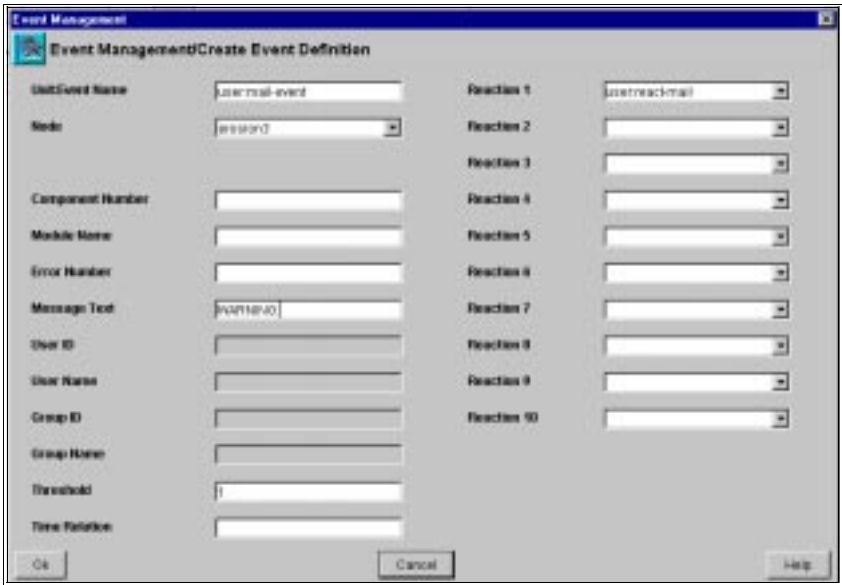


Figure 43: Example of a dialog box for creating an event definition

A combination of unit and event name that does not already exist must now be assigned in this dialog box for the *Unit:Event Name* entry, in our example *user:mail-event*. All other fields can be accepted unchanged, modified or deleted. For better clarity, our example shows the entry *user:react-mail* in the *Reaction 1* field as the unit and reaction name and the flag *WARNING:* in the *Message Text* field.

- ▶ Click *Ok* to exit the dialog box.

You will now find the new event definition *user:mail-event* in the object list.

Step 2: Clone event definition

You have to clone the event definition to transfer it to other systems in the domain. To do this, proceed as follows:

- ▶ Select the entry you just created in the *Unit:Event Name* column in the object list (in the example: *user:mail-event*).
- ▶ Now click the *Clone* action. A dialog box opens where you have to select the destination nodes:

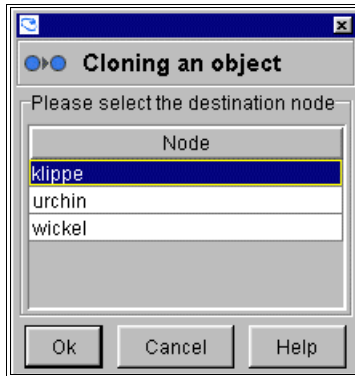


Figure 44: Example of a dialog box for cloning an object

- After you have selected the required node (in the example: *klippe*), click *Ok* to exit the dialog box.

You will then find the new event definition *user:mail-event* for all required destination nodes in the object list.

Step 3: Create reaction definition

- Change to the tab-bar *Reactions*.
- Choose a suitable entry in the *Unit:Reaction Name* column in the object list (in the example: *ubft: UBFT Reactions*) and right-click it. The following sub-menu appears:

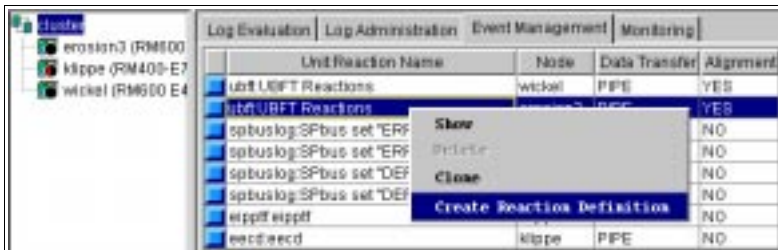


Figure 45: Example of an object list for creating a reaction definition (excerpt)

- Click *Create Reaction Definition*. The following dialog box is then displayed:

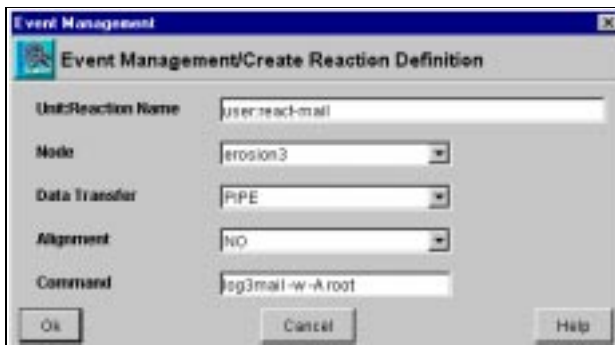


Figure 46: Example of a dialog box for creating a reaction definition

The only fields to edit in this case are *Unit:Reaction Name* and *Command*.

Enter *user:react-mail* in the *Unit:Reaction Name* field as the unit and reaction name. Enter the *log3mail* reaction program in the *Command* field with the *-w* and *-A* options. The *-w* option causes a mail to be sent to a recipient in the network without waiting for the transfer program to terminate. The *-A* option requires the user id of the mail recipient as an argument, in our example this is the user *root*.

- Click *Ok* to exit the dialog box.

Step 4: Clone reaction definition

The reaction definition must also be cloned if it is to be transferred to other systems in the domain. To do this, proceed as described in “Clone event definition” on page 50.

Step 5: Activate modified event and reaction definitions

The following steps are required to activate the newly created event and reaction definitions:

- Change to the tab-bar *Events*.

- ▶ Click the *Activate* action. The following dialog box is displayed:

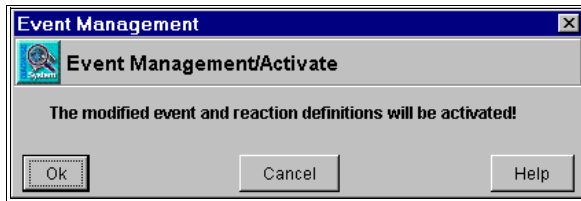


Bild 47: Example of a dialog box for activating event/reaction definitions

- ▶ Click *Ok* to activate the newly created event and reaction definitions.

This completes all necessary steps. The new events will now also be monitored and will trigger the required reactions where necessary.

4.12 Domain-wide file system access

WebSysAdmin offers all nodes in the domain the option to access a file system. This section uses an NFS file system to show how this can be managed in a few simple steps. The following tasks are to be performed in this example:

- An entry should be made on all systems for the file system in the */etc/vfstab* file.
- The file system should be mounted immediately.
- The system should also be mounted automatically when this system reboots.

 In order to enable domain-wide access to an NFS file system, the relevant resource must first be exported on the *NFS* tab.

Importing an NFS file system

To import an NFS file system, you have to:

- ▶ Change to the *File systems* tab (*VConfig* application).
- ▶ Click the *File systems* option on the menubar.
- ▶ Now choose the *Import network file system* option, sub-option *Nfs*.

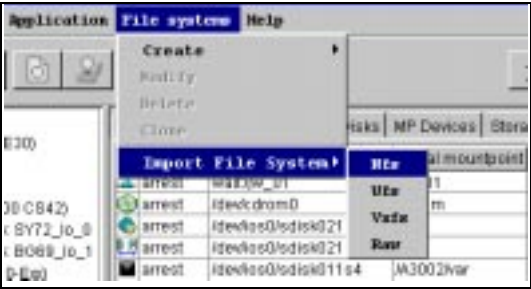


Figure 48: Example of a selection menu for importing file systems

The following dialog box is displayed:

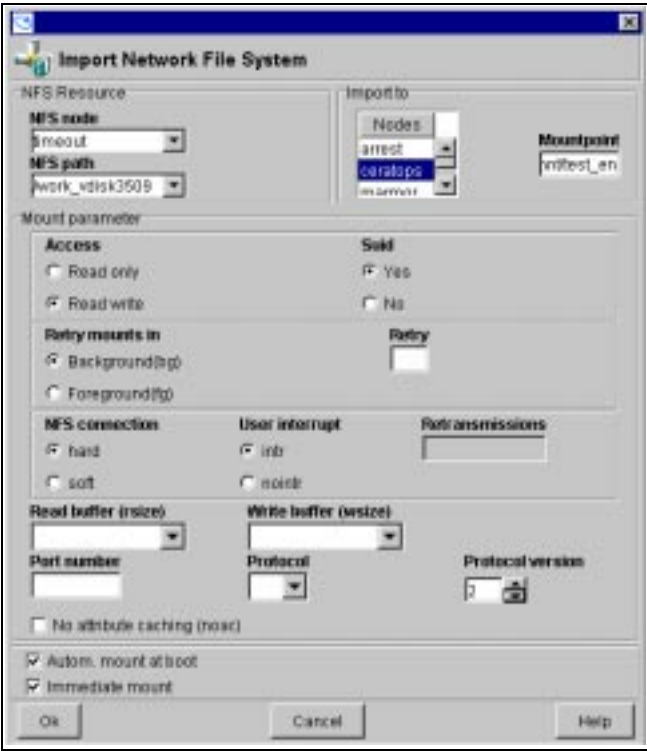


Figure 49: Example of a dialog box for importing an NFS file system

Fill in the dialog box as displayed in the diagram above:

- ▶ In the *NFS node* field, enter the node on which the file system being imported is located (in the example: “timeout”).
- ▶ In the *NFS path* field, enter the path of the file system being imported on the specified node (in the example: `/work_vdisk3509`).



When importing an NFS file system, you can specify both a node name and NFS node path, which do not belong to the domain.

- ▶ In the *Nodes* list, select all systems that are to receive access to the NFS file system (in the example: “ceratops”). The options *Autom. mount at boot* (controlled by the entry in the `/etc/vfstab` file) and *Immediate mount* are executed by default and thus lead to the actions required in the example.
- ▶ In the *Mountpoint* field, you can enter a pathname, which is to be used as the mount point for the NFS file system. The same pathname is used for all selected nodes (in the example: `/work_vdisk3509`). This can subsequently be modified separately for each individual node.
- ▶ All other fields are already assigned default values, which can be copied over here; version 2 has simply been entered in the *Protocol version* field.
- ▶ Click *Ok* to exit the dialog box.

As a result of the previously described actions, the imported NFS file system on the selected node is displayed in the object list on the *Usage parameters* sub-tab:



Figure 50: Example of an object list with imported NFS file system



The search criterion “Actual mount point contains test_en” was specified with the selection configurator to reduce the number of objects displayed.

The NFS file system (in the example: `/work_vdisk3509`) is thus available to all selected NFS clients in the domain with the access rights defined in the dialog box (see figure 49 on page 54).

4.13 Checking software versions

One of the system administrator's main tasks is to maintain the software used. The system administrator therefore has to ensure, for example, that all packages required on the respective systems are available in their current versions. This includes verifying that the same versions are being used on all nodes in a domain to avoid the individual nodes behaving differently.

For this purpose, *WebSysAdmin* offers various options in the *Software* application to ensure simple administration of the software used.

Therefore, if you want to know, for example, whether the same version of a particular package is used on all nodes in the domain, you can establish this as follows:

- 1. Select the entire domain in the hierarchical browser (click *cluster*).
- 2. Specify the abbreviated form of the required package (see the *Pkginst* attribute) in the search field of the selection configurator (in the example: *SIlog3*) and click the  icon:



Figure 51: Example of searching for installed packages

As a result, the example shows an object list with eight packages, which are available with identical versions on all systems:



Node	Pkginst	Name	Category	Arch	Version	Load	Backup
si-log3	SIlog3	SIlog3	application	all	2.0.1.0	20	SIlog3
si-log3	SIlog3	SIlog3	application	all	2.0.1.0	20	SIlog3
si-log3	SIlog3	SIlog3	application	all	2.0.1.0	20	SIlog3
si-log3	SIlog3	SIlog3	application	all	2.0.1.0	20	SIlog3
si-log3	SIlog3	SIlog3	application	all	2.0.1.0	20	SIlog3
si-log3	SIlog3	SIlog3	application	all	2.0.1.0	20	SIlog3
si-log3	SIlog3	SIlog3	application	all	2.0.1.0	20	SIlog3
si-log3	SIlog3	SIlog3	application	all	2.0.1.0	20	SIlog3

Figure 52: Example of targeted selection of installed packages

Checking the packages for consistency

The second step involves selecting six packages and checking them for consistency. This is how you do this:

- Select the packages to be checked in the object list.

- In the menubar, click the *Software* option and choose the *Check ...* action.



Figure 53: Example of a menu for checking packages

The following dialog box is displayed:



Figure 54: Example of a dialog box for checking software packages

Choose the options you require from those listed.



The two options *Check file attributes* and *Check file contents* are already selected by default.

- Start the checking process by clicking *Ok*. A progress indicator in the form of an increasing bar as well as a percent figure show how advanced the checking process is.

As soon as the checking process is complete, a dialog box is opened containing the result of the checks:

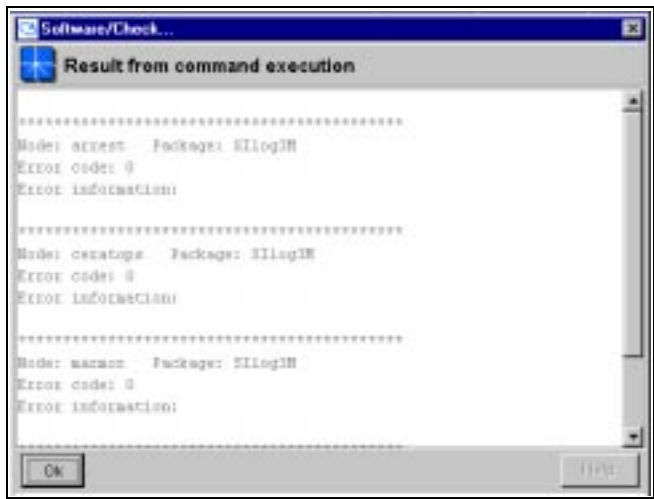


Figure 55: Example of a dialog box with the result of a package check

i If errors are listed here (e. g. incorrect access rights), you can resolve these if appropriate in a further check run by choosing the *Correct known inconsistencies* option in the dialog box.

- Click *Ok* to exit the dialog box.

The checking of packages is now complete.

4.14 Creating tasks

Every system administrator will transfer repetitive tasks to the cron system from Reliant UNIX. These system tasks (or simply tasks) can be defined easily on one or more nodes under *WebSysAdmin* using the *Task Management* application.

If you want to create a task, for example, which is to execute daily backups on all nodes in a domain at 10 pm every evening, you have to:

- Click the *Tasks* option in the menubar and choose the *Create* action.



Figure 56: Menu for creating a task

The following dialog box is displayed:



Figure 57: Example of a dialog box for creating a task

To complete the proposed task, proceed as follows:

- Enter the name of the shell script to be executed in the *Command* input field (in this example: */usr/sbin/cron-backup*).
- Now select the node that is to execute this task.

i It is recommended to first try out more complex tasks on one node and only then distribute them through cloning. This approach is also favored here.

- Now fill in the time period fields as shown in figure 57.
- Click *Ok* to activate the new task.

The successful execution of the action is confirmed in a message box:

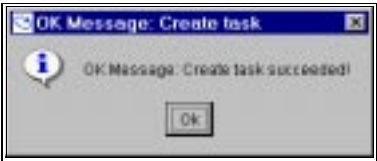


Figure 58: “OK Message” after creating a task

- Click *Ok* to exit the dialog box.

You will now find the new task in the object list. It exists exclusively on the “timeout” node at present.

Tasks							
	Node	Comment	Monthly	Daily	Weekly	Hourly	Minutely
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin..."	all	all	all	23	15
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	all	48
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	all	28,58
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	Sun,Tha	2	0
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	5	15
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	3,4,8,12,16,20	0
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	all	0
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	8	0
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	Tha-Fri	11	9
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	Sun	6	17
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	Mon-Fri	11,14	48
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	22	45
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	all	48
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	all	28,58
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	Sun,Tha	2	0
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	5	15
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	3,4,8,12,16,20	0
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	all	0
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	8	0
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	Sun	6	17
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	Mon-Fri	11,14	48
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	22	45
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	all	48
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	all	28,58
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	Sun,Tha	2	0
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	5	15
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	3,4,8,12,16,20	0
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	all	0
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	all	8	0
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	Tha-Fri	11	9
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	Tha-Fri	11	9
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	Tha-Fri	11	9
☒	createops	Update task: createops.suite -> "varlib/ops/psuadmin.poll -> /dev..."	all	all	Mon-Fri	22	45

Figure 59: Example of an object list after creating a task

Transferring the new task to other nodes in the domain

The new task should now be transferred to other nodes in the domain. To do this, proceed as follows:

- ▶ Select the task to be cloned in the object list.
- ▶ Click the *Tasks* option in the menubar. Choose the *Clone* action from the menu.

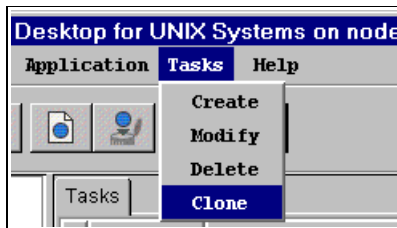


Figure 60: Menu for cloning a task

The following dialog box is displayed:

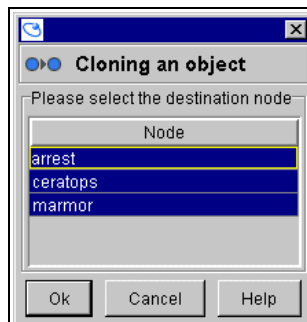
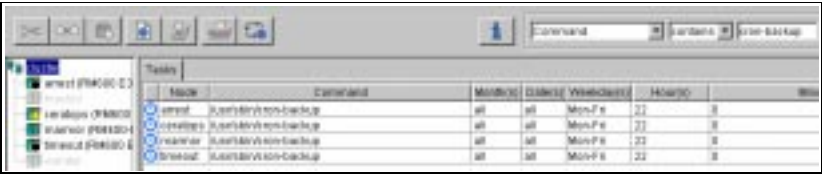


Figure 61: Example of a dialog box for cloning objects

- ▶ Select all required nodes and click *Ok* to exit the dialog box.

You will now find the cloned task on all selected nodes:



The screenshot shows a software interface with a tree view on the left and a table on the right. The tree view shows a hierarchy of nodes, with 'Server1 (P8400-03)' selected. The table on the right is titled 'Tasks' and contains the following data:

Task	Command	Mode	Owner	Priority	Frequency	Next Run
Server1 (P8400-03)	Server1 (P8400-03)	all	all	Mon-Fri	22	0
Server2 (P8400-04)	Server2 (P8400-04)	all	all	Mon-Fri	22	0
Server3 (P8400-05)	Server3 (P8400-05)	all	all	Mon-Fri	22	0
Server4 (P8400-06)	Server4 (P8400-06)	all	all	Mon-Fri	22	0

Figure 62: Example of an object list after cloning a task

The other selected nodes can now execute the new task.

5 Solving problems

When working with applications under *WebSysAdmin*, a wide variety of faults and problems may arise. These problems may originate in the browser being used, in the WSA interface itself or in the individual applications (i.e. in the *Domain-Admin* add-on product). Some problems from these three groups that we know from experience may recur are highlighted below. We also illustrate possible workarounds or ways to avoid such problems.

Whatever causes problems may well have, they always make their presence felt in the user's working medium, i.e. in the interface. If the screen does not respond at all, or only sluggishly or with error messages and if this adversely affects the user's work, the user is not interested on who or what may be the cause rather he or she just wants to be able to continue working as normal. The following problem scenarios and their proposed solutions are thus structured more in terms of their effect, since the same cause (e.g. high storage requirement) may have different effects (irregular interface or performance). We will look in greater detail at

- Problems using WSA
- Performance problems
- Problems managing domain nodes



In general, help texts are available for using *WebSysAdmin*, which generally also reference problems and constraints with regard to specific actions. If possible, you should consult these help texts when in doubt.

5.1 Problems with Netscape or WSA

It is a basic necessity when working with *WebSysAdmin* that you have the best possible hardware and software configuration. Details regarding the relevant technical data can be found in the section “Administration desktop requirements” on page 7.

The following list of problems is structured as follows: First, the problem is described briefly (indicated by •), then a solution is proposed (indicated by ►).

- The WSA interface does not respond, there are no plausible messages and output.

WSA may have been started more than once in Netscape.

► Terminate and restart Netscape.

- When you start WSA, no connection is established, the plug-in is not loaded.

The Apache server may not be running on the node.

► Check whether the Apache server is running. Enter the `ps -ef | grep http` command. If the result is negative, start the Apache server using the following commands:

1. `cd /etc/rc2.d`
2. `sh S97SIapache`

- The WSA screen is not correct, applications are missing.

► Change the browser settings for the cache and proxy.

1. Navigate to *Edit → Preferences → Advanced → Cache* and manually empty the two caches. Furthermore, under the heading “Document in cache is compared to document on network”, select the *Every time* entry.
2. Navigate to *Edit → Preferences → Advanced → Proxies* and select the “Direct connection to the Internet” or “Automatic proxy configuration” entry. In the latter case, the specified proxy file must ensure that a proxy server is excluded (no proxy).

- When WSA is started, a message appears to the effect that the selected application is already running and it is only possible to work in read mode, although no other user is working with WSA.

► Check whether the “WSAserver” process is still running on any node in the domain. Enter the `ps -ef | grep WSAserver` command. If the result is positive, i. e. a `/opt/wsa/bin/WSAserver` process is located, quit this process with the following command: `kill -14 PID`

If the problem persists, start the following commands on the node on which your WSA server is to run:

```
clos_unlock -f -p WSA_vconf
```

```
clos_unlock -f -p WSA_vdiag
```

```
clos_unlock -f -p WSA_vuser
```

See also the *clos_unlock*(1M) manual page for the command.

- If WSA is used intensively for several days, there may be a bottleneck in the PC's virtual memory.
 - ▶ Use the task manager to check the resource usage and terminate and restart the Web browser if necessary.
- There are locations on the WSA screen where it appears that buttons cannot be clicked with the mouse.
 - ▶ Click various areas of the button or reposition the WSA window.
- The automatic column width calculation does not always work properly.
 - ▶ Manually correct the column width using the mouse.
- Messages are sometimes truncated in the case of the confirmation and error box with the result that they cannot be read fully.
 - ▶ Use the mouse to enlarge the boxes.
- When an event is confirmed (e.g. node status change) the data is not always automatically reloaded.
 - ▶ Perform a *Refresh* using the *View* menu option on the menubar.
- When the Web browser has been terminated, the Windows temp directory (*C:\windows\temp*) still contains files called *jar_cachexxx.jar*. This may cause a memory bottleneck on the relevant drive.
 - ▶ Delete the files.

5.2 Performance problems

In order to exclude the possibility of performance problems, you should make sure initially that hardware requirements are fulfilled optimally. In other words, you should use a PC system with at least a 200-MHz processor and 128-Mbyte memory as the administration desktop.

Furthermore, you should be aware that the performance is not dictated by the number of nodes in a domain rather by the number of objects being managed (number of cabinets, disks and vdisks).

- A domain is under pressure and the response times are much too slow.

WSA always waits until the last node in a domain has supplied its data (rule: the slowest node in the domain determines the overall speed).

- ▶ If you want to perform actions even though a node is heavily loaded, you can only avoid the problem by disabling the node in question (*Nodes* tab, *Disable* action).



If you disable a node, configurations that relate to this node are no longer possible.

- The network load is high with the result that communication in the domain is also impaired.

- ▶ You should set up a separate network for data communication (second LAN controller).

- After you added a node to a domain, the response times are slow.

With the increase in the number of objects to be managed, the memory and the performance of the PC are impaired.

- ▶ Increase the performance of the PC (main memory extension, faster processor).

- After a lengthy session with many actions, the response time behavior deteriorates.

The Web browser has a high memory requirement and the output and messages in the WSA interface are not plausible.

- ▶ Terminate Netscape and all its applications and then restart it.

5.3 Problems with node administration

Node administration, i.e. the creation and removal of nodes, is the basic function of domain administration. Should problems arise in this context, these are often serious and have a global effect. However, problems that are only to be avoided **before** nodes are created or deleted should also be noted. The following information is therefore subdivided into “up-front” problems, problems creating nodes and problems removing nodes.

Up-front problems

- You want to change the name of a node retrospectively.
 - ▶ You can only change a node name before each configuration (using *SYSADM*). It is not possible afterwards.
- You want to change the IP net address or the net alias name for domain communication.
 - ▶ Proceed as follows:
 1. Stop the Closmon monitor, in other words the interface that controls communication between the nodes in a domain. For this purpose, start the following command on all nodes in the domain:

```
node_adm -K nodename
```

2. On every node in the domain, change the IP net address in the */etc/saf/CADMIN/_pmtab* file using an editor (e.g. *vi*).

3. Start the Closmon monitor again. To do this, use the

```
node_adm -E nodename
```

command.

4. Update and distribute the domain database. To do this, invoke the

```
cod_sync
```

command.

Please refer to the manual page of the same name for information about the *node_adm* and *cod_sync* commands.

- A node in a domain permanently has the status “not available” (*NOT_AVAIL*). This may happen if
 - a) the node is locked
 - b) the IP net address of the node has changed
 - c) the network connection is lost
 - ▶ In case a) you have to release the node (*Nodes* tab, *Enable* action).
 - ▶ In case b) you have to edit the *_pmtab* file (see above for how to do this: stop the Closmon monitor etc.).
 - ▶ In case c) you have to restore the network connection.

Problems creating/after creating nodes

- When you create a node, the error message `COD archive is not consistent` or `COD archive busy` appears.

An inconsistency message means that a difference has been noted between the archived domain database (COD) and the local configuration.

- ▶ First perform a consistency check by invoking the

`cod_check -e`

command. If the check confirms the inconsistency, update and distribute the COD archive with the

`cod_sync`

command. Please refer to the manual page of the same name for information about the `cod_check` and `cod_sync` commands.

- After you create a new node, the local vdisks for the node you added are no longer displayed.

When you create a new node, the local configuration files are overwritten and the domain configuration data is valid. If the vdisks were still configured locally, inconsistencies will arise between the new vdisk database and the vdisks configured in the kernel.

- ▶ You have to deconfigure the local vdisks before you create the new node.

After you create the node, the problem can only be resolved by rebooting.

Problems after deleting nodes

- The following query is issued when a node is booted:

```
This node seems to be removed from a cluster while being down!  
It cannot boot with shared vdisks.  
There are two options:
```

- 1: Remove this node from the CADMIN cluster and boot into multi user mode.
In this case all shared vdisks will be removed.
- 2: This option is for experts only!
Leave this node inconsistent and fall back to single user mode.
While inconsistent, this node cannot boot to multi user mode.

```
enter your choice (1,2) [1]
```

The node was integrated in the administration domain and deleted when it had the status “not available” (*NOT_AVAIL*).

- In this case, alternative 1 should be chosen for booting to restore the consistency of the database.

Abbreviations

CD-ROM	Compact Disk-Read Only Memory
CDS++	C++ Development System
CLODB	Cluster Object Database
CLOS	Cluster Operating System
COD	Cluster Object Database (archival file)
HA	High Availability
ID	Identifier
IP	Internet Protocol
JENV	Java Runtime Environment
JRE	Java Runtime Environment
JVM	Java Virtual Machine
LAN	Local Area Network
LAR	Lockout Auto Recovery
NFS	Network File System
OPS	Oracle Parallel Server
OPSRES	Oracle Parallel Server Resilvering
RCS II	Reliant Cluster Server II
RMS	Reliant Monitor Software

Abbreviations

SCSI	Small Computer Systems Interface
SPS	Siemens Parallel Server
TCP/IP	Transmission Control Protocol/Internet Protocol
URL	Uniform Resource Locator
vdisk	virtual disk
VGA	Video Graphics Adapter
WSA	Web based System Administration

Related publications

Please apply to your local office for ordering the manuals.

You can also view a number of the manuals on the Internet. The Web address is as follows: http://www.fujitsu-siemens.com/servers/man/man_us.htm

- [1] **Reliant UNIX 5.45**
Reliant UNIX Installation
Installation Guide

Content

Describes the new and update installation for Reliant UNIX as well as initial installation following integration of a new system disk.

- [2] **Reliant UNIX 5.45**
Reliant UNIX Operation
System Administrator Guide

Content

Describes the basics of Reliant UNIX, the SYSADM and WSA user interfaces as well as the uninterruptible power supply (UPS).

- [3] **Reliant UNIX 5.44**
User's Guide

Content

Provides an overview of the Reliant UNIX 5.44 operating system and an introduction to the general principles.

- [4] **Reliant UNIX 5.45**
Commands. User's Reference Manual

Content

Describes the general Reliant UNIX commands in alphabetical order.

- [5] **Reliant UNIX 5.45**
System Administrator's Reference Manual

Content

Describes the commands and user programs for system maintenance as well as file formats and special files for performing system administration on the Reliant UNIX operating system.

[6] **Reliant UNIX 5.45
Networking Reference Manual**

Content

Describes the commands, functions, utilities and system files required for working with nodes that communicate with one another in a computer network.

[7] **Reliant UNIX 5.44
System Administrator's Guide**

Content

Provides an introduction to UNIX system administration. Among other things, it describes the tools used by the system administrator, the configuration and support required for a UNIX system, the calculation of system usage as well as aspects of system security. The manual is relevant both for MIPS-based UNIX and for Intel-based UNIX.

[8] **Reliant UNIX 5.45
Diagnostics and Troubleshooting
System Administrator Guide**

Content

Contains information about analyzing and resolving errors at system startup and during operation of RM200, RM300, RM400 and RM600 models. In addition, the user interfaces for system diagnostics are described and errors in relation to I/O, the effects of a power failure, the LAR concept and selected error messages are explained.

[9] **Reliant UNIX 5.44
System Administration and Hardware Configuration Using the
SYSADM User Interface
System Administrator Guide**

Content

Describes hardware configuration and system administration using SYSADM: Managing file systems and network services, installing software, configuring systems, managing users.

- [10] **Reliant UNIX 5.45**
Configuring Virtual Disks with VDisk Lite
System Administrator Guide

Content

Describes the configuration of virtual disks using VDisk Lite. The most important menus and dialog boxes in VDisk Lite are presented and a detailed configuration example is demonstrated.

- [11] **Reliant UNIX**
OBSERVE V1.2B
System Administrator Guide

Content

Describes OBSERVE V1.2B. The manual is written for system administrators responsible for configuring and supporting the operation of OBSERVE, as well as for programmers who want to use the C interfaces supplied with OBSERVE.

- [12] **Reliant Monitor Software V3.1D**
Installing RMS
(Reliant UNIX, Solaris)
System Administrator Guide

Content

Provides guidelines for installing Reliant Monitor Software (RMS) on Siemens RM systems.

- [13] **Reliant Monitor Software V3.1D**
Administering RMS Using RCVm
(Reliant UNIX, Solaris)
System Administrator Guide

Content

Describes the RCVm Motif-based graphical user interface, via which the Reliant Monitor Software (RMS) resources are administered.

- [14] **Reliant Monitor Software V3.1D**
RMS Configuration and Administration
(Reliant UNIX, Solaris)
System Administrator Guide

Content

Describes the functionality as well as the configuration and administration tasks to be performed by RMS.

[15] **Reliant Monitor Software V3.1D**

RMS Concepts

(Reliant UNIX, Solaris)

System Administrator Guide

Content

Describes the concept of the RMS product.

[16] **Reliant UNIX 5.45**

High Availability

System Administrator Guide

Content

Outlines the hardware and software high-availability components of the RM system family, for example OBSERVE, RMS, OPS, SPS, RCS II etc.

[17] **Siemens Parallel Server**

SPS Installation, Configuration, and Administration

(Reliant UNIX)

System Administrator Guide

Content

Describes the software installation process for Reliant UNIX SPS as well as administration of clusters that use these features.

Index

A

- activating
 - event definition 52
 - reaction definition 52
- administration desktop 4, 7
 - requirements 7
- administration domain
 - advantages 3
 - definition 3

C

- CADMIN 5, 18
- checking packages
 - for consistency 56
- CLODB
 - domain administration database 6, 23, 24, 31
- cloning
 - event definition 50
 - reaction definition 52
 - tasks 61
 - user 27
- CLOS
 - operating system for domain administration 5
- consistency checking
 - of packages 56
- creating
 - domain 18
 - event definition 49
 - node 21
 - reaction definition 51
 - tasks 59
 - user 24
 - vdisk 31

D

- deleting
 - domain 21
 - node 23
 - user 28

- diagnostics 41
- domain
 - creating 18
 - deleting 21
- DomainAdmin
 - add-on product 1
 - installing 5
- domain-wide
 - configuring events 49
 - creating events 49

E

- event definition
 - activating 52
 - cloning 50
 - creating 49

H

- hierarchical browser (WebSysAdmin) 14

M

- menubar (WebSysAdmin) 12

N

- NFS access, domain-wide 53
- node
 - creating 21
 - deleting 23

O

- object list (WebSysAdmin) 16
- object table (WebSysAdmin) 16

P

- password
 - global 29
- problems
 - Netscape or WSA 63
 - node administration 66
 - performance 65

Index

R

- reaction definition
 - activating 52
 - cloning 52
 - creating 51

S

- screen resolution for WebSysAdmin 11
- selection configurator (WebSysAdmin) 14
- setting up tasks 58
- software versions
 - checking 56
 - displaying 56
- starting
 - WebSysAdmin 10
- sub-tab bar (WebSysAdmin) 16

T

- tab bar (WebSysAdmin) 16
- tasks
 - cloning 61
 - creating 59
- toolbar (WebSysAdmin) 12
- transferring tasks 58
- troubleshooting
 - general 42
 - object-specific 45

U

- user
 - cloning 27
 - creating 24
 - deleting 28

V

- vdisk
 - configuring on shared resources 37
 - creating (local) 31

W

- WebSysAdmin
 - client 4
 - elements 12
 - hierarchical browser 14
 - menubar 12
 - object list 16
 - object table 16
 - screen resolution 11
 - selection configurator 14
 - server 4
 - starting 10
 - sub-tab bar 16
 - tab bar 16
 - toolbar 12

Fujitsu Siemens Computers GmbH
User Documentation
33094 Paderborn
Germany

Fax: 0 700 / 372 00001

e-mail: manuals@fujitsu-siemens.com
<http://manuals.fujitsu-siemens.com>

Submitted by

Comments
Suggestions
Corrections

Comments on Reliant UNIX 5.45
System Administration within a Domain



Fujitsu Siemens Computers GmbH
User Documentation
33094 Paderborn
Germany

Fax: 0 700 / 372 00001

e-mail: manuals@fujitsu-siemens.com
<http://manuals.fujitsu-siemens.com>

Submitted by

Comments
Suggestions
Corrections

Comments on Reliant UNIX 5.45
System Administration within a Domain



Siemens Business Services
Training Center, Consulting Service
81730 Munich
Germany

Fax: (++49) 89636-52003

Courses Consulting Self-tuition media

The Siemens Training Center offers consulting for in-service training, courses and self-tuition media on almost all aspects of information technology, either on your premises or at a Training Center near you. Write to us for information and course programs.

Submitted by

send me information/course programs on the following topics



Siemens Business Services
Training Center, Consulting Service
81730 Munich
Germany

Fax: (++49) 89636-52003

Courses Consulting Self-tuition media

The Siemens Training Center offers consulting for in-service training, courses and self-tuition media on almost all aspects of information technology, either on your premises or at a Training Center near you. Write to us for information and course programs.

Submitted by

send me information/course programs on the following topics



