

Barco Controlroom Division



R5976144

Osiris 1.3

Owner's Manual

R5976144, Current Version

Title: **Osiris 1.3 Owner's Manual**

ID-no.: **R5976144**

Date: sept-2004

	Main issue	update
chapter 1	new	
chapter 2	new	
chapter 3	new	
chapter 4	new	
chapter 5	new	
chapter 6	new	
chapter 7	new	
chapter 9	new	
chapter 10	new	
chapter 11	new	
chapter 12	new	
chapter 13	new	
chapter 14	new	
chapter 15	new	
chapter 16	new	
chapter 17	new	
chapter 18	new	
chapter 19	new	
chapter 20	new	

new: The corresponding chapters are new or completely revised.

corr.: Passages of the corresponding chapter were corrected; see modification bars.

add.: Passages of the corresponding chapter were added; see modification bars.

Document History

Modifications which result in a new version are indicated by a vertical bar.

Keep this sheet!

Trademarks

Brand and product names mentioned in this manual may be trademarks, registered trademarks or copyrights of their respective holders. All brand and product names mentioned in this manual serve as comments or examples and are not to be understood as advertising for the products or their manufacturers.

DLP and DMD are trademarks of Texas Instruments, Inc.

Metaframe and UIS are trademarks of Citrix, Inc.

Windows NT TSE is a trademark of Microsoft, Inc.

Osiris is a trademark of BARCO inc.

X-Windows is a trademark of Massachusetts Institute of Technology

UNIX and OPEN LOOK are registered trademarks of UNIX System Laboratories, Inc.

Copyright © 2000 by Barco Controlroom Division

All rights reserved. No part of this document may be copied, reproduced or translated. It shall not otherwise be recorded, transmitted or stored in a retrieval system without the prior written consent of Barco Controlroom Division.

Guarantee and Compensation

Barco Controlroom Division provides a guarantee relating to perfect manufacturing as part of the legally stipulated terms of guarantee. On receipt, the purchaser must immediately inspect all delivered goods for damage incurred during transport, as well as for material and manufacturing faults. Barco Controlroom Division must be informed immediately in writing of any complaints.

The period of guarantee begins on the date of transfer of risks, in the case of special systems and software on the date of commissioning, at the latest 30 days after the transfer of risks. In the event of justified notice of complaint, Barco Controlroom Division can repair the fault or provide a replacement at its own discretion within an appropriate period. If this measure proves to be impossible or unsuccessful, the purchaser can demand a reduction in the purchase price or cancellation of the contract (redhibition). All other claims, in particular those relating to compensation for direct or indirect damage, and also damage attributed to the operation of software as well as to other services provided by Barco Controlroom Division, being a component of the system or independent services, will be deemed invalid provided the damage is not proven to be attributed to the absence of properties guaranteed in writing or due to the intent or gross negligence on the part of Barco Controlroom Division.

If the purchaser or a third party carries out modifications or repairs on good delivered Barco Controlroom Division, or if the goods are handled incorrectly, in particular if the systems are commissioned or operated incorrectly or if, after the transfer of risks, the goods are subject to influences not agreed upon in the contract, all guarantee claims of the purchaser will be rendered invalid. Not included in the guarantee coverage are system failures which are attributed to programs or special electronic circuitry provided by the purchaser, e. g. interfaces. Normal wear as well as normal maintenance are not subject to the guarantee provided by Barco Controlroom Division either.

The environmental conditions as well as the servicing and maintenance regulations specified in this manual must be complied with by the customer.

Revision sheet

To:

Barco Controlroom Division
Noordlaan 5
8520 Kuurne
Belgium
Tel.: +32-56-368.211
Fax: +32-56-368.251
E-mail: support.bcd@barco.com
WWW: www.barco.com

From: _____

Date: _____

Please correct the following points in this documentation (**R5976144**):

page	wrong	correct

Contents

1	Introduction	1-1
1.1	How This Manual Is Organized	1-2
1.2	Styles And Symbols	1-3
2	Architecture	2-4
3	Minimum requirements	3-5
3.1	Hardware	3-5
3.2	Software	3-5
4	First-time installation	4-6
4.1	Prerequisites	4-6
4.2	Installation and Activation	4-7
5	System Configuration	5-8
5.1	First time Configuration	5-8
5.1.1	Hardware Configuration: The configuration drawing	5-8
5.1.2	Configuration usage	5-8
5.1.3	General Configuration Settings	5-9
5.1.4	Overview of possible modules to integrate in configuration	5-10
5.1.5	Licensing	5-28
5.1.6	X-Server configuration	5-29
5.2	Applying the changes to the configuration	5-30
5.3	Upgrade procedure	5-30
5.3.1	Prerequisites	5-30
5.3.2	Upgrade Procedure	5-30
5.3.3	Upgrade from a Redhat 6.x system	5-31
6	VISU Settings	6-32
6.1	VISU Source Settings	6-32
6.1.1	Video Source settings	6-32
6.1.2	RGB Source settings	6-33
6.2	VISU General Settings	6-36
6.2.1	Phase settings	6-36
6.2.2	OPU settings	6-37
6.3	Force VISU Reset	6-38
7	Osiris Startup	7-39
7.1	Osiris Login	7-39
7.2	Control Panels	7-40
7.2.1	General Osiris Control Panel	7-40
7.2.2	Administration Control Panel	7-41
7.2.3	Multiple Osiris GUI session	7-41
7.2.4	Customizing the Control Panel	7-42
7.2.5	Syntax for the GUI configuration file	7-42
7.2.6	Example of multi-user control panel setup	7-43
7.3	Osiris GUI color usage	7-45
8	Control Management	8-46

8.1	Introduction.....	8-46
8.2	Object Definition	8-46
8.2.1	Alias.....	8-47
8.2.2	VISU window.....	8-53
8.2.3	Layout.....	8-56
8.2.4	Scenario	8-65
8.3	Layout Stack.....	8-72
8.3.1	Organizing	8-72
8.3.2	Unloading objects	8-73
8.3.3	Saving Active Layout	8-73
8.3.4	Starting Osiris commands.....	8-74
8.4	Copying objects	8-74
9	Projector Control	9-76
9.1	Supported range of projectors.....	9-76
9.2	Connection.....	9-76
9.3	Projector Commands.....	9-77
9.4	Integration of commands within layouts/scenarios.....	9-77
9.4.1	Layout.....	9-77
9.4.2	Scenario	9-77
9.5	Projector Control utility (GUI)	9-78
9.5.1	Barco CRT range of projectors.....	9-78
9.5.2	Barco LCD range of projectors.....	9-78
9.5.3	Overview mL (SVGA or XGA).....	9-79
9.5.4	Overview mP	9-79
9.5.5	Overview DG.....	9-80
9.5.6	Overview DME.....	9-81
10	Zoom-X	10-82
10.1	Overview.....	10-82
10.1.1	Functional description	10-82
10.1.2	Architecture	10-82
10.1.3	Task of each module.....	10-83
10.2	Configuration Options	10-84
10.2.1	Zoomxserver options	10-84
10.2.2	Zoomxclient options	10-87
10.2.3	ZoomX configuration for Multiple Logical Screens.....	10-88
11	Test patterns.....	11-89
11.1	Range of patterns	11-89
11.2	Application.....	11-90
12	Remote Pointer (Octopus)	12-91
12.1	Functionality.....	12-91
12.2	Startup and usage of the remote pointer client.....	12-91
12.3	multi-user-mode.....	12-92
12.4	single-user-mode.....	12-93
12.5	Special options for the remote pointer Client.....	12-95
12.6	Logical screens	12-97
13	User Administration.....	13-98

13.1	General info	13-98
13.2	Creation of new users.....	13-98
13.3	Deleting existing users.....	13-98
13.4	Modifying Passwords.....	13-99
14	Device Control.....	14-100
14.1	Functionality	14-100
14.2	Configuration	14-100
14.3	Syntax of Device Control Configuration	14-100
14.4	Example	14-102
15	Error Log	15-104
16	On-line Help	16-105
17	System Administration	17-106
17.1	Console Window.....	17-106
17.2	17-106	
17.3	UCL Control	17-107
17.4	Osiris Restart.....	17-107
17.5	System restart.....	17-107
17.6	System halt.....	17-108
17.7	Backup/Restore customizations	17-108
17.8	Specific Osiris customizations	17-110
17.9	Specific X-server customizations.....	17-110
17.10	Using Osiris in stand-alone mode (off-line).....	17-110
17.10.1	Purpose	17-110
17.10.2	Configuration.....	17-111
17.10.3	Restrictions	17-111
18	Software Integration Issues.....	18-112
18.1	Integration of X-windows based applications.....	18-112
18.1.1	EOS configuration	18-112
18.1.2	Application's host configuration	18-112
18.2	Integration of NT based applications	18-113
18.2.1	TSE/Metaframe/UIS	18-113
18.3	Miscellaneous application integration.....	18-114
18.3.1	VMS based applications	18-114
18.4	Integration of 3 rd party serial devices using GCI functionality	18-115
19	Osiris Directory Tree overview	19-116
20	Osiris Command List.....	20-117
20.1	General.....	20-117
20.2	wall	20-117
20.2.1	enablesenario	20-117
20.2.2	alarm.....	20-118
20.2.3	dumpdatabase	20-118

20.2.4	infoscenario	20-118
20.2.5	loadalias	20-118
20.2.6	loadlayout	20-119
20.2.7	loadscenario	20-119
20.2.8	loadvisu	20-119
20.2.9	disablesscenario	20-120
20.2.10	restoredatabase	20-120
20.2.11	stopscenario	20-120
20.2.12	ucldisable	20-120
20.2.13	uclenable	20-120
20.2.14	uclstatus	20-121
20.2.15	unloadalias	20-121
20.2.16	unloadlayout	20-121
20.2.17	unloadvisu	20-122
20.3	server	20-122
20.3.1	stopserver	20-122
20.4	tty	20-122
20.4.1	write	20-122
20.5	sad	20-123
20.5.1	reboot	20-123
20.5.2	shutdown	20-123
20.6	crt	20-123
20.6.1	polling	20-123
20.6.2	readsource	20-124
20.6.3	remotecontrol Unit	20-124
20.6.4	toallprojectors	20-125
20.6.5	writesource	20-126
20.7	lcd	20-126
20.7.1	adjpossible	20-126
20.7.2	polling	20-127
20.7.3	setprojectoroff	20-127
20.7.4	setprojectoron	20-127
20.7.5	toallprojectors	20-128
20.7.6	readsource	20-128
20.7.7	writesource	20-129
20.8	dmd	20-129
20.8.1	poll	20-129
20.8.2	toallprojectors	20-129
20.8.3	setprojectoroff	20-130
20.8.4	setprojectoron	20-130
20.8.5	readlampruntime	20-130
20.8.6	readbrightness	20-131
20.8.7	readcontrast	20-131
20.8.8	writebrightness	20-131
20.8.9	writecontrast	20-132
20.9	dme	20-132
20.9.1	polling	20-132
20.9.2	toallprojectors	20-133
20.9.3	setprojectoroff	20-133
20.9.4	setprojectoron	20-133
20.9.5	readlampruntime	20-134
20.9.6	readvalues	20-134
20.10	oml	20-134
20.10.1	polling	20-134
20.10.2	toallprojectors	20-135

20.10.3	setprojectoroff.....	20-135
20.10.4	setprojectoron	20-136
20.10.5	readlampruntime	20-136
20.11	omp	20-136
20.11.1	poll.....	20-136
20.11.2	toallprojectors	20-137
20.11.3	setprojectoroff.....	20-137
20.11.4	setprojectoron	20-137
20.11.5	readlampruntime	20-138
20.12	autopatch.....	20-138
20.12.1	logicalswitch.....	20-138
20.13	gsw.....	20-138
20.13.1	logicalswitch.....	20-138
20.14	gci	20-139
20.14.1	closepipe	20-139
20.14.2	openpipe	20-139
20.14.3	closettyport	20-139
20.14.4	openttyport.....	20-139
21	References	21-141
22	Remarks and Comments	22-143

1 Introduction

The name Osiris originates from Ancient Egyptian mythology.

Osiris is based on the former „Control Manager“ product. It's main purpose was to provide users with a possibility to automate the startup of applications and grouping the resulting application windows in different layouts.

This functionality has been broadened by a number of additional features, as there are:

- the control of projectors making part of the Graphics Wall
- the control of external devices such as AMX, Crestron, ...
- the controlled starting/stopping of applications which are not directly involved with displaying on the display wall but play an important role in the management of the control center as well
- integration of UPS device in order to allow a controlled behavior when power is lost or restored.
- creation of automated Scenarios.

As opposed to the former SW system, where all items were individual SW products, this is a more integrated environment where different modules, responsible for different functions, are grouped round a common mediator. All components communicate to each other via the common TCP/IP communication protocol.

This offered functionality should allow a seamless integration of the Graphics Wall components within the customer's Control Room environment.

1.1 How This Manual Is Organized

This manual is divided into eighteen chapters:

Preliminary Remarks

explains the structure of the manual itself and the used typographic styles and symbols.

Architecture

explains the build-up of Osiris in its totality without going to much in detail

Minimum Requirements

provides details on the minimum hardware and software requirements.

First-time installation

explains the installation process.

System Configuration

explains how the configuration is done.

Osiris Startup

explains the startup of Osiris.

Control Management

all Control Management related topics.

Projector Control

explains how projectors are controlled and which types of projectors are supported.

Test Patterns

software tools for projector alignment.

Remote Pointer (Octopus)

access to the Graphics Wall using a Software Remote Pointer utility.

User Administration

how to administer different users within Osiris.

Device Control

explains the setup and usage of the device control utility.

Error Log

utility to monitor the Osiris behaviour

On-Line Help

how to get help when using Osiris (not implemented yet)

System Administration

how to administer the complete Osiris software environment

Software Integration Issues

covers specific details on how to integrate customer-applications within the Osiris environment

Osiris Command List

lists all external Osiris commands which can be used by external parties to control the Osiris system

References

references to more detailed information sources

Chapters, pages, figures and tables are numbered separately. Chapters are indicated by a »point syntax«, e. g. **4.2.3**, pages by a »dash syntax«, e. g. **2-1**, as figures and tables are, e. g. **Figure 5-4**.

1.2 Styles And Symbols

The typographic styles and the symbols used in this document have the following meaning:

Helvetica bold

Labels, menus and buttons are printed in the **Helvetica bold** font.

Condensed

Links to both other chapters of this manual and to sites in the Internet are printed **condensed**. In the on-line version of this manual all hyperlinks appear **teal**.

Courier

Names of files and parts from programs are printed in the `Courier` font.

Courier bold

Inputs you are supposed to do from the keyboard are printed in **Courier bold** font.



Within a piece of programming code this arrow marks a line, that must be made up in two lines, though meant to be one line.



This arrow marks tips and notes.



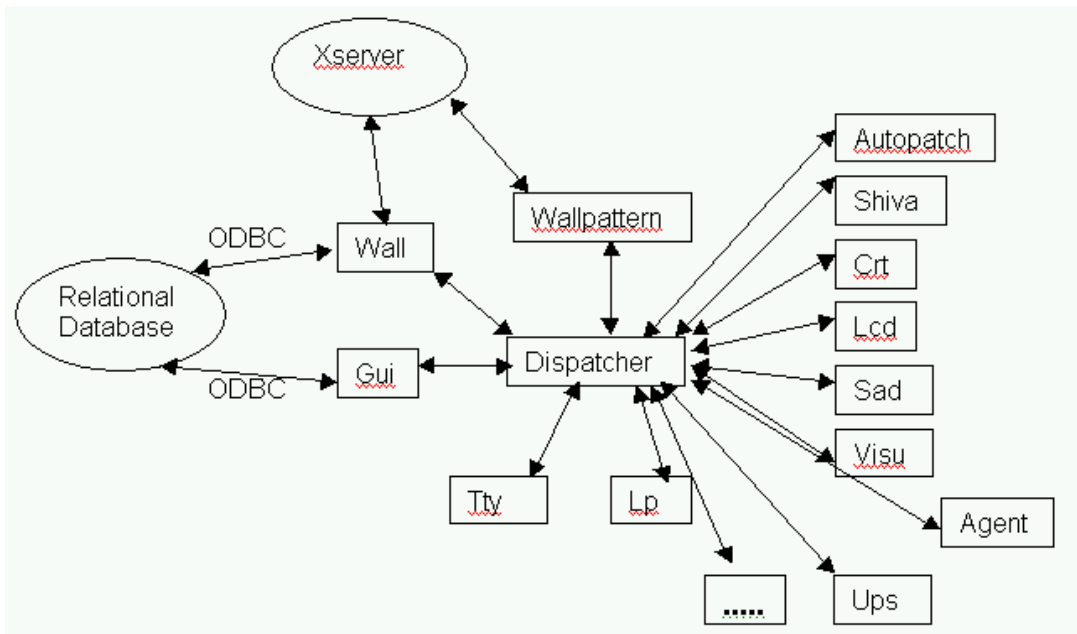
If you do not heed instructions indicated by this symbol there is a risk of damage to the software integrity!



If you do not heed instructions indicated by this symbol there is a risk of electrical shock and danger to personal health!

2 Architecture

Osiris consists of a number of different modules. The central module is the dispatcher which sends and receives events from/to the other modules. An overview is given in the graphic below:



Each functionality within Osiris is managed by some individual module which may be activated or not depending on it's need within the configuration. For example, the VISU Osiris module would not need to be active within a configuration not using any VISU devices.

One module may be depending on a number of other modules. For example, the wall module can only start when the server-module is active. Likewise the VISU-module depends on the TTY and LP module being active. And so on...

This dependency is automatically resolved, based on the Osiris hardware configuration. All needed modules are derived from this configuration and started. However if necessary it is always possible to manually force or suppress the start of specific modules.

As already mentioned before, the communication between the different modules is done using some proprietary interfacing protocol over the standard TCP/IP networking protocol. This perfectly allows you to scatter Osiris modules over a number of different machines. As TCP/IP is the underlying communication's protocol, the different modules will be able to reach one-another over the LAN. This however requires in-depth understanding of how to manually configure a standard Osiris configuration.

As Osiris allows you to set up a number of definitions with respect to applications, layouts, etc... This information needs to be stored in some way. Therefore a relational database engine is running in background, communicating between the actual database and the wall module. Only the wall module gui module and gci module may provide or extract information to/from the database.

Osirs

Within the next chapters, this architecture is being looked at in more detail, specifying for each module what it's functionality entails and how to start and manage it.

3 Minimum requirements

The next subsections clarify the minimum requirements needed for Osiris to be installed and run without problems. Both SW and HW requirements are covered.

3.1 Hardware

The following minimum HW requirements are put forward:

Internal System's Memory	64Mb or higher
CPU	Pentium II, 350Mhz or higher
Free space of disk storage (prior to installation)	100Mb

This does not mean that Osiris will not run on EOS platforms that do not meet these requirements. There is however no guarantee that it will.

For upgrading existing EOS systems with the Osiris SW, an analysis will be done beforehand in order to determine the needs for upgrading the HW platform.

Osiris, and more in particular the X-server is designed to be compatible with the following range of graphical output cards:

Type of Card	Supplier
OVT-2978-2(2-output digital card, Panel-link)	Barco Nv
OVT-2978-1 (2-output digital card, LVDS) (old)	Barco Nv
OVT-2978-0 (2-output analog card MX)	Barco Nv
OVT-2978-3 (2-output digital card, Panel-link, Full Color Card)	Barco Nv
OVT-2824-4 (2-output analog card, synchronized outputs, VX)	Barco Nv
IMSGP Graphical cards (Master and Slave)	Barco Nv

The following digitizing equipment is supported by Osiris:

Type of Card	Supplier
R9830600 VISU sub-unit	Barco Nv
R9830601 VISU+ sub-unit	Barco Nv
R9830602 VISU+D sub-unit	Barco Nv

3.2 Software

Osiris 1.3 is designed to run on the following operating systems:

OS Platform	Supplier
Linux RedHat 7.x. with Barco EOS-X-Server 2.4	RedHat

4 First-time installation

Within the next subsections, the complete installation process is covered. Be it the installation from scratch or the upgrading of an existing Osiris version to a higher version.



All new EOS systems supplied by Barco or Barco partner, are equipped with the Osiris base software. First-time installation of Osiris will therefore not be needed in the majority of cases. To upgrade a system with a new version of Osiris, go to section 5.3

4.1 Prerequisites

In order to install Osiris 1.3, the following is required:

Item	Supplier
EOS system meeting the mentioned minimum requirements	Barco Nv
Barco EOS-X-Server 2.4	Barco Nv
Osiris installation/upgrade CD	Barco Nv
X-windows capable system (X-terminal or PC running X-windows emulator for the client GUI.	Not specified

The following terminology is used within the rest of this documentation:

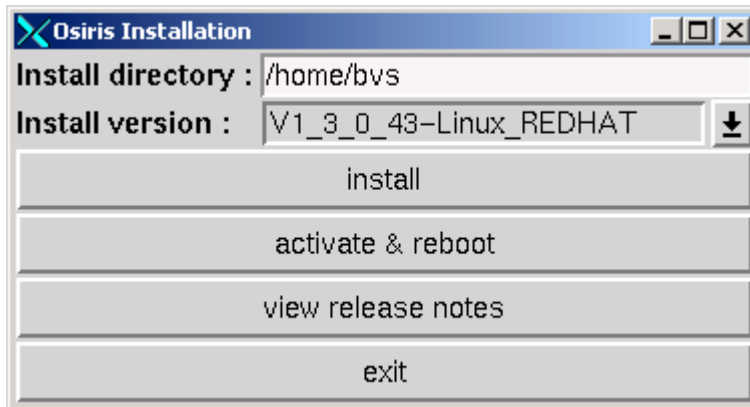
Name	Description
mgvs	hostname of the EOS machine
mgvs_pc	hostname of the default X-windows capable machine which displays the Osiris GUI
BVS_HOME	environment variable referring to the home directory for Osiris. Commonly, /home/bvs' is used as home directory
GW_DISPLAY	environment variable referring to display of the wall. Normally set to mgvs:0. Specified in /etc/rc.d/init.d/Osiris
DISPLAY	environment variable referring to the default X-Windows capable machine which displays the Osiris GUI. Normally set to mgvs_pc:0.0. Specified in /etc/rc.d/init.d/osiris

4.2 Installation and Activation

When an EOS system is supplied, Osiris comes preinstalled with the configuration and licensing already in place.

When installing Osiris for the first time, follow the steps described in table below.

Following GUI should appear:



- It is advised to use **'/home/bvs'** as the install directory (BVS_HOME).
- Install version: preferably, select the non-debug version rather than the debug version
- **Select 'Install Osiris'** => Installation takes place unattended. Message of completion will appear in the session from where installation/upgrade took place.

Task	Description
Installation of Linux Redhat 7.x and Barco EOS-X XServer 2.4	see EOS X Terminal user's manual
Start of Osiris installation	<ul style="list-style-type: none"> - Insert the Osiris CDROM in the CDROM drive. - Login or telnet to the mvgs system as ,ofc' user - Become super-user (su) - On a Redhat Linux platform, you mount the cdrom by using the command: <code>mount /mnt/cdrom</code> as super-user.
Setup of display for installation-Gui	Redirect DISPLAY to some X-windows platform (for ex. Mvgs_pc): <code>DISPLAY=mvgs_pc:0</code> <code>export DISPLAY</code>
Installation startup	<code>cd /mnt/cdrom</code> (for Redhat Linux) <code>./setup</code>
Activation	After installation has been completed, Osiris still remains to be activated before usage. To do this select the activate & reboot button.

5 System Configuration

This section describes the process of configuring the Osiris environment. Most of this will have been completed upon the pre-installing of the system.

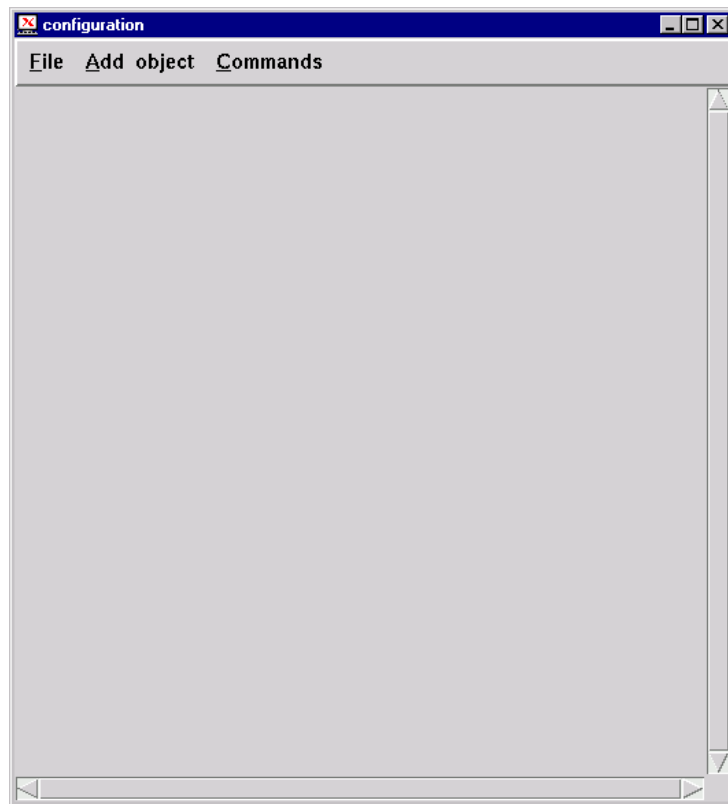
For users, this section may be seen as informational in order to understand the Osiris system better.

Section 5.3 describes what needs to be done in order to upgrade an existing Osiris system with a new release.

5.1 First time Configuration

5.1.1 Hardware Configuration: The configuration drawing

After restart of the system when Osiris was installed from scratch, the hardware configuration window will automatically pop up on the mvgs_pc console:



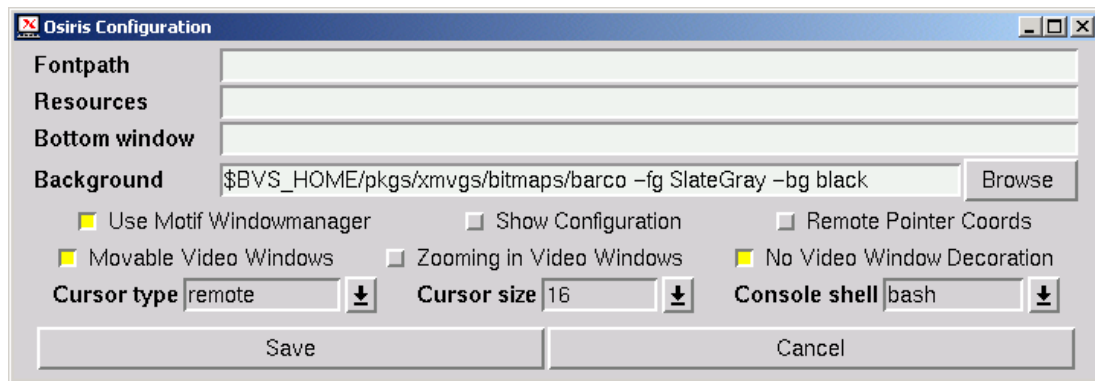
5.1.2 Configuration usage

Purpose	Action
Adding a configuration module	Select [Add Object] in menu and select appropriate module
Interconnection of objects	Each object may have 0 or more inputs and 0 or more outputs depending on it's functionality. Interconnection of objects is done by first clicking on some object's output lead and then on some object's input lead. To cancel while connecting to leads, right-click.
Structuring wires	When simply clicking in and output, a wire will be drawn by the system between the 2 selected points. To make the wiring diagram somewhat more structured, one can hold the

	<CTRL> key down while clicking, which will result in the drawing of separate wire-sections.
Deleting some object	Double click the object or wire to delete.
Editing of objects	Upon publication of this document, an object may not be edited. The only way to modify some object is to remove it and recreated it with revised properties.
Checking configuration	In order to check the configuration set up, select ,Check Configuration‘ within Commands menu.
Checking X-server connection	Select ,Check X-server connection‘ within Commands menu
Moving some defined object	In order to move some object, right-click on it and drag to new position

5.1.3 General Configuration Settings

When you exit the configurator via the exit command in the file menu the following dialog pops up an allows you to modify some general settings:

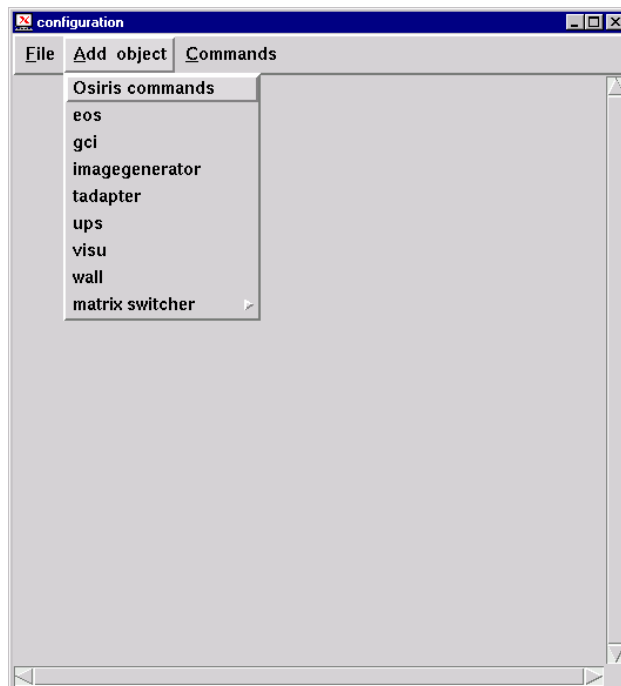


The following items may be modified/completed:

Item	Description
Fontpath	When a specific fontpath should be appended to the normal default fontpath, this can be completed here.
Resources	When a specific X-windows resource file is to be used, it can be specified here.
Background	A customized background picture can be specified here. It should be a X11 compatible bitmap.
Console shell	Specifies the console shell. When the console window is opened, the shell specified here will be used within that X-terminal.
Motif WindowMgr	When checked the Motif window manager will be activated. When unchecked, no window manager will be started.
Movable Video Windows	Specified if the video & RGB window will be moveable using the remote pointer.
Zooming in Video Windows	Specify whether zooming in video-windows is allowed or not.
No Video Window Decoration	Disables the Window Title-bar etc for visu Windows. Temorary re-enabling by middle mouse button (e.g. to move window).

Show Configuration	The configuration will be visible (readonly) to users without admin rights.
Remote Pointer Coords	Specifies if the remote pointer windows display the x and y location of the cursor on the wall.
Cursor Type	remote: sprite mouse pointers are used - recommended if multiple users will access the wall via the remote pointer. local : real mouse pointer is used - recommended if only a single user will access the wall via the remote pointer. If multiple users try to access the wall at the same time they will be warned that the system cursor is already in use.
Cursor Size	specifies the size in pixels of the cursor on the wall

5.1.4 Overview of possible modules to integrate in configuration



Osiris Command

Allows the user to create some customized Osiris commands apart from the already pre-defined set of commands.

The pre-defined commands are basically some commands to control different types of projectors connected to one of the serial ports of the controlling system running Osiris.

An Osiris command is built up out of the following components:

Component	Purpose
Osiris module	Name of the Osiris module where the specified command is passed to.
Osiris command	The command itself
Parameter(s)	1 or more parameters linked to the command. Each parameter can be given a default value so when the parameter is not explicitly specified, it gets that default value.

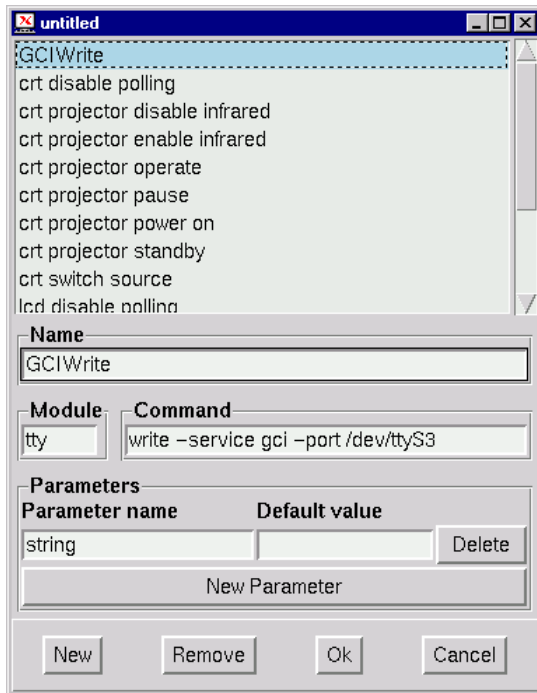
Actions one can perform on such a command definition are: delete, new and modify.

The commands created in this way can be used within different areas of Osiris:

- from within the Activate Layout section by right clicking and selecting ,Execute Command‘
- from within a Layout definition: upon the opening or closing of a Layout for example
- from within a Scenario definition

Example of a defined Osiris command:

The following command will send a string out on the GCI-port /dev/ttyS3



EOS



The EOS is the building-block for the graphical generator (being an EOS Compact, Power or Ultra system).

Item	description
Serial Port Name(s)	List of serial ports associated with this system. For a system running the Linux OS, the serial ports are named /dev/ttyS0, /dev/ttyS1,... For an EOS system running Solaris, the serial ports will be name /dev/ttya, /dev/ttyb,... For special serial cards used within the system, other device names may be used as well (non-standard configuration)
Parallel Port Name(s)	List of parallel ports associated with this system. In most cases, only /dev/lp1 will be used (parallel port on board)
EOS Nbr. Of Channels	Specifies the number of graphical output channels associated with the EOS



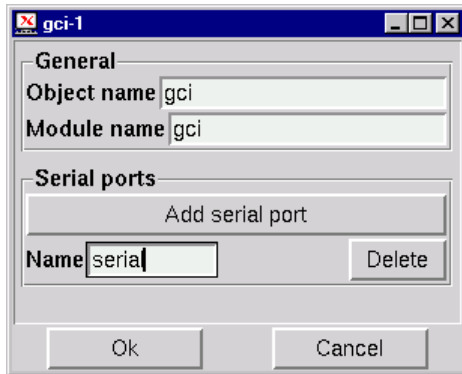
Only 1 EOS system per configuration is supported. 1 EOS may of course contain 1 master and several PCI-extension slave boxes, or one central device and several rendering devices (e.g. EOS Power)

Engine

The X-Server may be spited on several rendering engines.



GCI object



- GCI stands for ‚Generic Command Interface‘
- A GCI object handles Osiris commands that are passed to it via either:

Telnet session	When opening a telnet session to the EOS on port 5701 , one can thereafter issue an Osiris command, which will be handled by the gci-module.
Serial port connection	A serial device connected to one of the serial ports of the EOS, may generate some Osiris command which will be handled by a GCI object connected to that serial port. One or more serial ports may be controlled by 1 GCI object. For example, a device-control system like Crestron may „control“ the Osiris environment in this way.
Redirection to gci_pipe	One can redirect an Osiris command the the pipe-file: <code>\$BVS_HOME/logging/gci_pipe</code>

- The following items need to be specified when defining such an object:

Item	Description
Module Name	Name of GCI module managing this gci-object. In most cases this will be ‚gci‘
Serial port Name	If the GCI object is to be linked to some serial port(s) of the EOS system, each serial port should be given a unique serial-port name. For ex. CresSer.

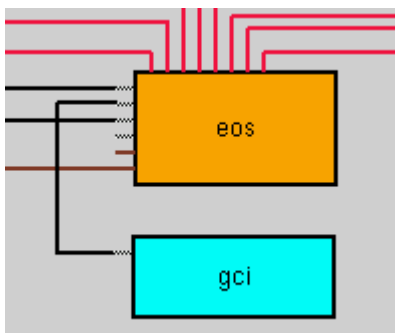
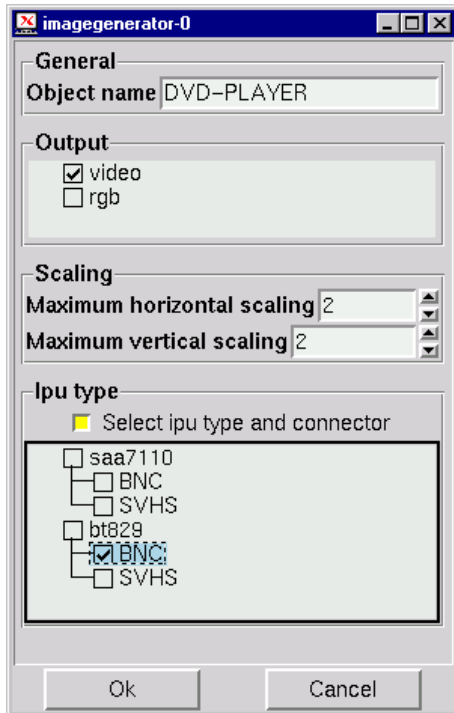


Image Generator

- Defines some source, supplying some object with a Video or RGB signal.
- Can be used to connect to:
 - VISU input
 - Video/RGB matrix switcher
 - T-adapter
- Currently, connection up to a 2nd input of some wall component is not supported.
- Definition of **Video signal source**:



The following items need to be specified when defining such an object:

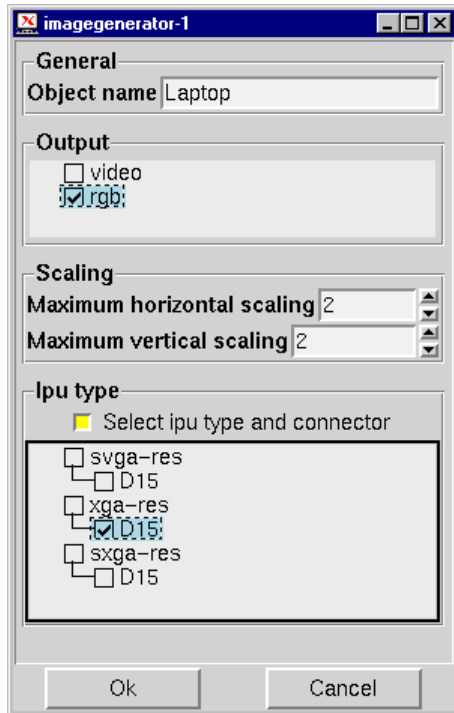
Item	Description
Output	Specifies the type of image generator. In this case ,video‘
Max. hor. Scaling	Horizontal scaling factor applied when needed (cfr (*))
Max. vert. Scaling	Vertical scaling factor applied when needed (cfr (*))
Input type	<p>Definition of VISU IPU-type and connector-type where this source signal will be connected to. Currently, 2 types of video-input boards are available:</p> <p>Saa7110: older type of VISU video input</p> <p>Bt829: new, current type of VISU video input</p> <p>The 2 names refer to the chipset used within the video input board HW.</p> <p>A video input has 2 connections:</p> <p>BNC: Standard composite video input</p>

	SVHS: Y/C SVHS signal
--	-----------------------



It is important to select the correct connection type for the video source in order to be able to connect it to the appropriate input of the video input board on the VISU

- Definition of **RGB signal source**:



The following items need to be specified when defining such an object:

Item	Description
Output	Specifies the type of image generator. In this case ,rgb‘
Max. hor. Scaling	Horizontal scaling factor applied when needed (cfr (*))
Max. vert. Scaling	Vertical scaling factor applied when needed (cfr (*))
Input type	Definition of VISU IPU-type and connector-type where this source signal will be connected to. Currently, 3 types of RGB-input boards are available: SVGA: handles input resolution up to 800x600 pixels XGA: handles input resolution up to 1024x768 pixels SXGA: handles input resolution up to 1280x1024 pixels

(*) Scaling: This is to avoid/compensate certain VISU limitations:

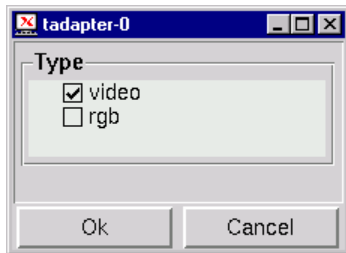
- ⇨Can't fit video
- ⇨Insufficient Transfer Bandwidth

If any of these errors appear in the error-log, increase the max. horizontal and vertical scaling number.

When set to 2, the input source will only be sampled at half of the sampling rate when required, thus gathering only half as much information.

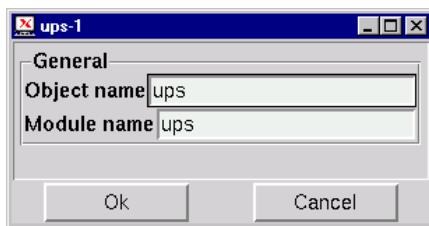
T-adapter

- used to split-up a signal into 2 identical signals: video or RGB source
- can be video or RGB splitter. No object name is required.



- For ex. a Barco Quad splitter 350 would be integrated using 3 of these T-adapter objects.

UPS



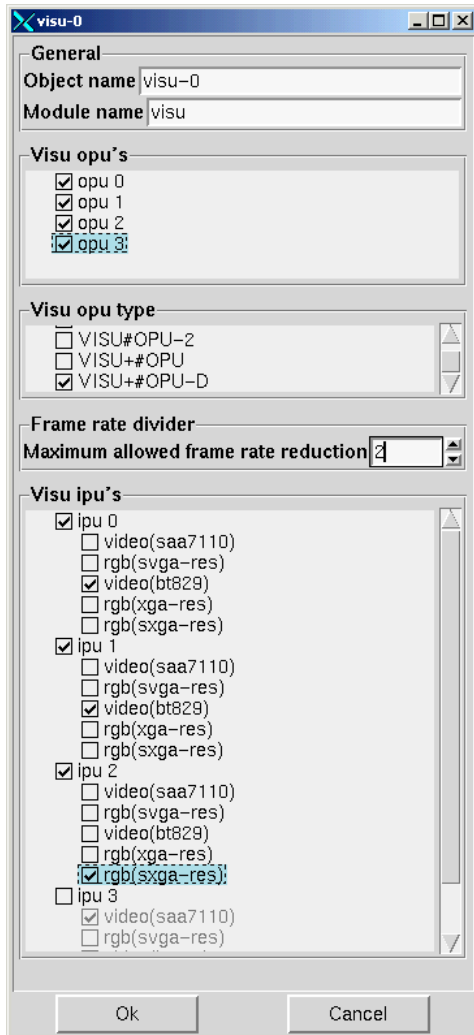
- Object used for integration of an Uninterruptable Power Supply unit
- The following items need to be specified when defining such an object:

Item	Description
Module Name	Name of module managing this gci-object. In most cases this will be 'ups'

- Upon different events resulting from the ups device, an action may be triggered within an Osiris scenario. For ex., upon power failure, Osiris may close down properly after some time unattended and shut-down the system it is running on.
- The UPS changes are used in Osiris as events that activate the ups triggers:
 - ⇒ Line operation (normal operation)
 - ⇒ Battery operation (a failure has occurred, battery power is used)
 - ⇒ Backup time (remaining backup time in minutes)
- The command "getstatus" can be used to query the current status of the ups:
 - ⇒ NORMAL OPERATION
 - ⇒ ABNORMAL OPERATION
 - ⇒ empty string: there is no connection between the UPS module and the ups device
- More about the different scenario-triggers based on the UPS object later in the scenario section.
- The UPS object is compatible with the following UPS device:

NetUPS SE Rack-Mount UPS 1000-3000 VA (supports the PowerRite protocol)

VISU



- VISU stands for ,Video InSertion Unit
- Device used to allow video or RGB insertion within the rest of the X-windows application environment.
- It is a proprietary system built around a dedicated processor board and a number of input and output modules (4 max. of each)
- VISUs may be looped through when required.
- For more detailed information, we refer to the VISU's owner's manual
- The following items need to be specified when defining such an object:

Item	Description
Module Name	Name of module managing this visu-object. In most cases this will be ,visu'.
VISU OPUs	Each VISU system may consist out of minimum 1 and maximum 4 OPUs (Output Processing Units). Each OPU number mentioned.

	corresponds with the slotnumber where the OPU is inserted in. Slots are numbered from top to bottom
VISU OPU type	Types of OPUs: OPU V1: old type. Still contains the manual color-key alignment trimmers (no longer used within new systems) OPU V2: Normal VISU OPU-type. Digital colorkey alignment OPU VISU+: OPU type used within a VISU+ system OPU VISU+D with digital output (latest)
Hardware Handshaking	To be checked when RTS/CTS HW handshaking is used during communication. (VISU+ technology and higher). When HW handshaking is used, only 1 type of serial cable is used between EOS and VISU and between VISU-VISU (straight RS232 cable fem-male)
VISU IPU's	Selection of which type of IPU is present in which input slot. Again, slots are numbered from top to bottom. Currently, 5 types of input boards are supported: <i>Video input SAA7110</i> <i>Video input BT829</i> <i>SVGA input</i> <i>XGA input</i> <i>SXGA input</i>
Frame rate divider	Factor specifying how many times the frame rate of RGB images may be reduced to save bandwidth on internal bus of the VISU.

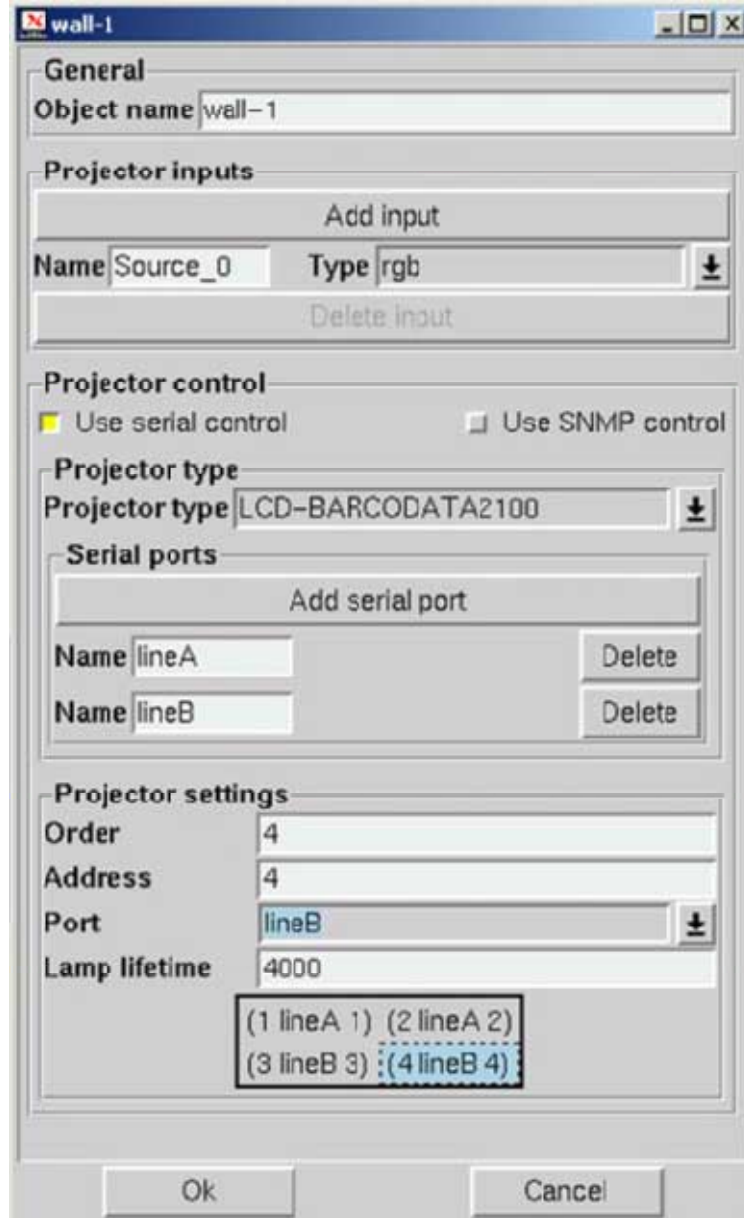
- Each VISU has a serial input and output and a parallel input and output (for reset). When these signals are not properly connected, a configuration error will result
- Upon time of publication, only **1 chain of VISUs** is supported.
- When the first VISU is defined, the subsequent created VISUs will be forced to contain the same type of OPUs (greyed out)!



For more detailed information on the VISUs functionality, please refer to the VISU's owner's manual.

Wall

- A Wall is a definition of the Graphics Wall matrix itself
- Contains a number of display devices (projectors, monitors,...)
- The dimension (x:y) of the wall is extracted from the configuration of the X-Server running on the EOS



Item	Description
Projector inputs	One projector may have different inputs. An input may take an RGB or Video. Currently, only 1 input is used. Future Osiris versions may support the automatic switching of the projector to the source selected to some particular input. Currently, this is not the case.
Use serial control	Serial control of projectors. May be enabled/disabled. All projector types used within the Barco Controlroom Division line of applications are supported.
Serial ports	Arbitrary name of serial port where the controller may connect to. One wall can have different chains of projectors. Therefore, more than 1 serial port may be defined.
Projector Settings/Address	Each projector must have a specific address in order to be uniquely addressable. Some projectors may be given an address using HW dip-switches, most others are given an address via SW setup.
Projector Settings/Port	Serial port associated with the chain of projectors. As specified previously, different chains of projectors may be set up.
Projector Settings/Order	Each projector is given an order which determines the place it occupies within the chain. It is important to set this up correctly, as this allows Osiris to bypass a projector whenever it gets off-line or is out-of-order for some reason.
Use SNMP control	This option is only available for projectors of type: <ul style="list-style-type: none"> - Overview-pSi using the BARCO OMP projector protocol - Overview-DG using the BARCO LCD protocol - using LCD/DLP communication protocol. Use this if the projectors are not directly connected to the EOS via a serial line but are controlled via an SNMP-Agent accessible via the LAN.

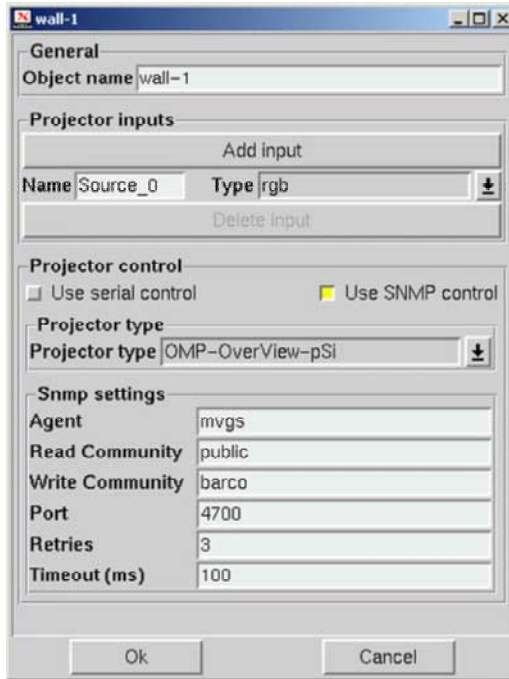
SNMP settings:

For projector control Osiris can communicate with the BARCO RC-Agent (ver. 1.5SR2)

The following projectors are supported via SNMP:

- Overview-Psi (using the BARCO OMP protocol)
- Overview-DG (using the BARCO LCD protocol)
- all LCD/DLP type of projectors

If SNMP control is enabled you can configure the following parameters for the RC-Agent. The default values correspond to the default values of the RC-Agent, thus you normally only need to specify the parameter 'Agent'. By default the Agent is considered to run on the Eos or Argus system (mvgs).



Item	Description
Agent	IP-Address or Hostname of the machine the SNMP-Agent is running on.
Read Community	Authorization to read SNMP-Values from the agent.
Write Community	Authorization to write SNMP-Values to the agent.
Port	TCP/IP Port the agent is listening to. (default: is 4700)
Retries	Number of retries to repeat an-SNMP requests before an SNMP-read error is indicated.
Timeout (ms)	Time in ms to wait until a time-out is indicated.

Please make sure that the dimensions of the wall defined in the X-Server running on the EOS correspond to the configuration of the RC-Agent. Further you do not have to specify the position and order of the modules any more in Osiris instead the RC-Agent is responsible to handle this (see manual of RC SNMP Agent for details).

The following parameters must be setup correctly in the RC Agent :

Projector Addresses:

The RC-Agent will auto-assign addresses to detected projectors.

Use of SNMP Traps:

The Barco RC-SNMP-Agent is able to send confirmation messages back to Osiris. These confirmation messages are called “traps”. Osiris currently makes use of traps only for LCD-Projectors!

So if Osiris is controlling LCD-Projectors via SNMP, you must explicitly configure in the SNMP Agent to send traps to the Osiris systems.

Example:

If Osiris is running on machine 150.158.181.204 you need to modify barcorc.ini in the following way:

```
barcorc.ini:  
[SNMP]  
trap_destination=150.158.181.204/4701
```

Osiris expects traps always on port 4701!

If a trap is missing Osiris will produce an corresponding Error.log entry.

The trap-timeout time is set to 2 seconds and can not be changed.

Overview-Psi projectors

For these projectors Osiris will ask the agent about the projector positions. Use the tool breconf of RC-Agent to configure the projector-order.

Overview-D projectors

These projectors have a fixed projector-location (increasing from top-left to bottom right). Osiris will refer to projector locations as shown in the following scheme:

1	2	3	4
5	6	7	8
9	10	11	12

Furthermore Osiris expects projector addresses, used by the RC agent, to be in the range 192.168.1.x and supports only a single serial chain.

The RC agent assigns the highest projector address to be the first one in the chain (so the wire order is important). This assignment can be enforced by changing the [IP Address Map] in the /opt/barcorc/etc/barcorc.ini file, as shown in the example configuration below:

```
[init]  
loglevel=3  
rctable=/opt/barcorc/etc/barcorc.tab  
walltable=/opt/barcorc/etc/barcorc.csv  
Fix_Address_Count_1.8.0.0=192.168.1.1  
  
[SNMP]  
port=4700  
read_community=public  
write_community=barco  
logfile.linux=/var/log/barcorc.log  
error_log=0  
warning_log=0  
event_log=0  
info_log=0  
debug_log=0  
table_access=yes
```



```
trap_destination=150.158.192.103/162
```

```
[lib.OverViewD]  
pollcount=4  
updatecount=50  
wallcols=2  
interfaces=OverViewD0
```

```
[lib.OverViewD.if.OverViewD0]  
port.linux=/dev/ttyS1  
speed=9600,8,n,1  
netaddr=1.8.0.0
```

```
[IP_Address_Map]  
OverView-D:1=192.168.1.4  
OverView-D:2=192.168.1.3  
OverView-D:3=192.168.1.2  
OverView-D:4=192.168.1.1
```

LCD Projectors

These projectors have a fixed projector-address. Osiris will only address the projectors correctly if they correspond to the following scheme:

Assign these addresses to the projector by aid of the infrared control:

1	2	3	4
5	6	7	8
9	10	11	12

At start-up the barco RC-Agent will assign these “virtual IP-addresses” to the projectors:

192.168.1.12	192.168.1.11	192.168.1.10	192.168.1.9
192.168.1.8	192.168.1.7	192.168.1.6	192.168.1.5
192.168.1.4	192.168.1.3	192.168.1.2	192.168.1.1

So in a 4x3 wall Osiris always expects projector(3,2) to have the IP Address: 192.168.1.6

Matrix Switchers

Osiris provides in the integration of matrix switchers to allow the seamless usage of these devices in combination with the VISU systems Barco supplies.

Autopatch 1YDM

- 3rd party supplied matrix switcher (Autopatch).
- 2 different type of switchers are typically used: Video or RGB matrix switchers

- The following items need to be specified when defining such an object:

Item	Description
Module name	Specification of module managing this matrix switcher object
Logical Inputs	Definition of each input for the switcher. Each input signal has a Name and Type associated.
Logical Outputs	Definition of each output for the switcher. The same applies as for the inputs

Autopatch 4YDM

- 4YDM version of the 3rd party supplied matrix switcher
- differs from 1YDM version in the possibility of specifying a number of enclosures. An enclosure is a separate unit, which is linked to the previous and/or next unit by means of a serial interconnection cable. (Provided by Autopatch). Osiris will check upon startup whether all enclosures are present before continuing. Whenever one of the enclosures is either not present, not powered, not connected or malfunctioning, an error will be produced and Osiris will not fully initialize.
- Refer to the previous definition of entry-fields for more details on how to complete this form. Complete the number of enclosures additionally.

Autopatch 8YDM

As above but then for the 8YDM version which allows a more extensive switcher configuration.

Generic Switch

- Allows the integration of other switchers aside the mentioned ones.

- Switching commands are provided in a configuration file for that specific switcher device.
- For ex. when using the Barco Quad 350 switcher, one uses the supplied ,Quad350.cfg‘ file containing the serial commands to be used when a switch to a particular input source needs to be done. The contents of this file is as follows:

```
# resource file
mode binary
command-1-1 /0x02/0x01/0xFB/0x12/0x01/0x01/0x01/0x01/0x0C
command-2-1 /0x02/0x01/0xFB/0x13/0x01/0x01/0x01/0x01/0x0D
command-3-1 /0x02/0x01/0xFB/0x14/0x01/0x01/0x01/0x01/0x0E
command-4-1 /0x02/0x01/0xFB/0x15/0x01/0x01/0x01/0x01/0x0F
```

So whenever there is a need for Osiris to select for example input 3, the 3rd command is sent to the serial port connected to that switcher device.

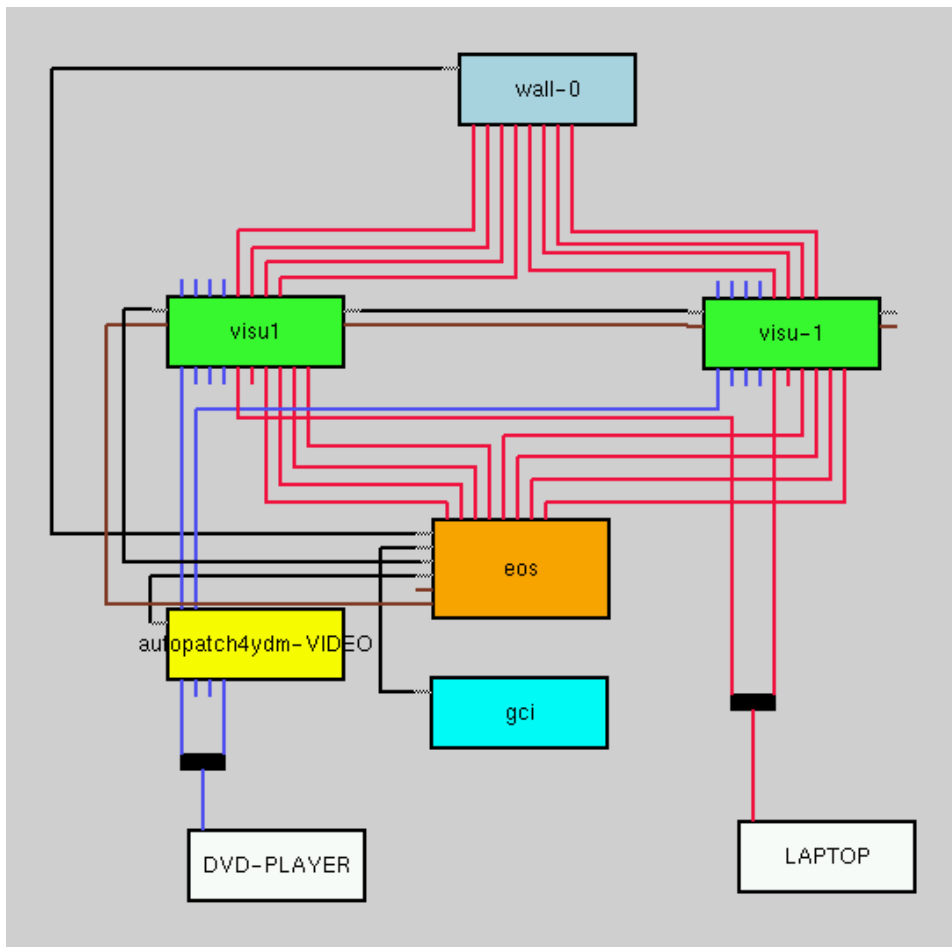
Either **binary** or **ascii** commands may be sent. This is defined by the mode keyword in the definition file.

The GSW module will also be used for some Autopatch configurations. For ex., when a 1YDM setup is used containing 2 separate levels to be switched independently, a GSW configuration will need to be prepared in order for this to work. An excerpt from such a file is shown here:

```
#
# Switching commands for combined RGB/Video Switcher
#
mode ascii
command-1-1 CL10I1T
command-2-1 CL10I2T
command-3-1 CL10I3T
command-4-1 CL10I4T
command-5-1 CL10I5T
command-6-1 CL10I6T
command-7-1 CL10I7T
command-8-1 CL10I8T
command-1-2 CL102I1T
...
command-9-9 CL20I1T
command-10-9 CL20I2T
command-11-9 CL20I3T
command-12-9 CL20I4T
command-13-9 CL20I5T
command-14-9 CL20I6T
command-15-9 CL20I7T
command-16-9 CL20I8T
command-17-9 CL20I9T
```

```
command-18-9 CL201I10T
command-19-9 CL201I11T
command-20-9 CL201I12T
command-21-9 CL201I13T
command-22-9 CL201I14T
command-23-9 CL201I15T
...
```

Configuration check



- After configuration completion, the 'Check Configuration' menu option may be used to do a cross-check of the setup.
- Does not guarantee a fault-free configuration however! Merely looks at the proper interconnection of objects.

5.1.5 Licensing

- Licensing system for specific optional Osiris modules
- Licenses are provided by Barco Controlroom Division depending on the configuration.
- Licenses may be provided for the following functionality:

Control Manager
VISU
Autopatch
UPS
Device Control
Switcher
Projector Control
ZoomX

- Whenever a new functionality is provided within Osiris, it may be that a new license will be needed as well.
- The licenses are kept in the following file:
\$BVS_HOME/config/osi_licenses
- the licenses are based on the MAC-address of the network card within the EOS.
- Proper functionality is only possible when proper licenses are in place.



Whenever a network card needs to be replaced. A new set of licenses will have to be provided in order to restore proper functionality.

5.1.6 X-Server configuration

In order to allow applications to connect to the X-windows environment of the EOS controller, the X-server needs to be set up properly. In the next sections, a split-up is done between the different environments that are running Osiris.

Redhat 7.x on EOS with IMSGP card

The IMSGP XServer will start up automatically when there is no /etc/rc.d/init.d/mmtsर्व file present. This file is created by running MMTinstall/MMTConfigure and should be removed if you use IMSGP cards.

Server configuration file

```
$BVS_HOME/config/XF86Config.eos
```

Multiple Logical Screens

To configure the X-server to use multiple logical screens, the XF86Config.eos file should contain multiple "Screen" sections. For example, if a 4x1 X-server area were to be split up in 2 logical screens of each 2x1 in size, the screen sections would be:

```
Section "Screen"
    Driver      "EOS"
    Device      "Node 0"
    Monitor     "MAG"
    OrbitSize   0
    OrbitTime   10
    Position    0 0
    Number      2 1
    BlankTime   0
    SuspendTime 0
    OffTime     0
    SubSection "Display"
        Depth    8
        Visual    "PseudoColor"
    EndSubSection
EndSection
```

```
Section "Screen"
    Driver      "EOS"
    Device      "Node 0"
    Monitor     "MAG"
    OrbitSize   0
    OrbitTime   10
    Position    2 0
    Number      2 1
    BlankTime   0
    SuspendTime 0
    OffTime     0
    SubSection "Display"
        Depth    8
        Visual    "PseudoColor"
    EndSubSection
```

EndSection

Mode-files

The mode files describe the timings for the selected resolution + specific VISU related parameters. They are contained within the directory:

```
$BVS_HOME/pkgsg/xmvgsg/modes
```

The mode-file used is defined with the 'Mode' keyword within the **XF86Config.eos** configuration file.

Redhat 7.x on EOS with OVT-card

Please refer to the EOS X Terminal users manual.

5.2 Applying the changes to the configuration

- In order to apply the changes which have been done to Osiris (HW configuration and X-server configuration), one has to login to the Osiris user interface as 'admin' user and go to the 'Administration' control panel section. (Cfr. Section 7.1) Then, click the 'Osiris restart' button and confirm.
- Check the messages generated within the 'Errorlog' window or the 'Console' window for more information on the restart on Osiris. This may supply you with key information as to why a configuration does not seem to end up in a fully functional Osiris environment.

5.3 Upgrade procedure

5.3.1 Prerequisites

- Before starting an upgrade, make a backup of the current customizations (cfr. Section 17.7).
- It is advised to have all Osiris users log out before an upgrade is started.
- One will need the upgrade CD for Osiris.

5.3.2 Upgrade Procedure

Follow the next steps one by one to guarantee a successful upgrade:

Action	Details
Startup	<ul style="list-style-type: none">- Insert the supplied Osiris update CDROM into the CDROM-drive- Login to Osiris using the 'admin' account- Select the 'System Administration' button- Select the 'Upgrade Osiris' button
Start upgrade	<ul style="list-style-type: none">- Click [OK] to start the upgrade procedure- Specify the directory to upgrade (default set to /home/bvs)- Select the Label of the software to upgrade to. Be sure to select the appropriate Linux one.- Select [OK] to confirm the upgrade- Select whether the existing directory where to install to needs to be moved or not: Yes => current directory is set aside as a backup No => current directory is overwritten

Safeguarding settings	<ul style="list-style-type: none"> - Settings can be kept from previous version. Select either [OK] for keeping previous settings or [Skip] for going back to default settings. - The same can be done for the database of Osiris. Select [OK] to safeguard your database or [Skip] for going back to default settings.
Reboot	System will prompt you to reboot in order to activate the upgrade.

When previous procedure has been followed, the upgrade is completed and the system should be fully functional again.

5.3.3 Upgrade from a Redhat 6.x system

Action	Details
Backup	save current configuration to floppy (see backup procedure)
Install Redhat 7.x and EOS-X Server 2.4	see EOS X Terminal users manual
Install Osiris 1.3	see above
Restore	restore saved configuration from floppy (see restore procedure)

6 VISU Settings

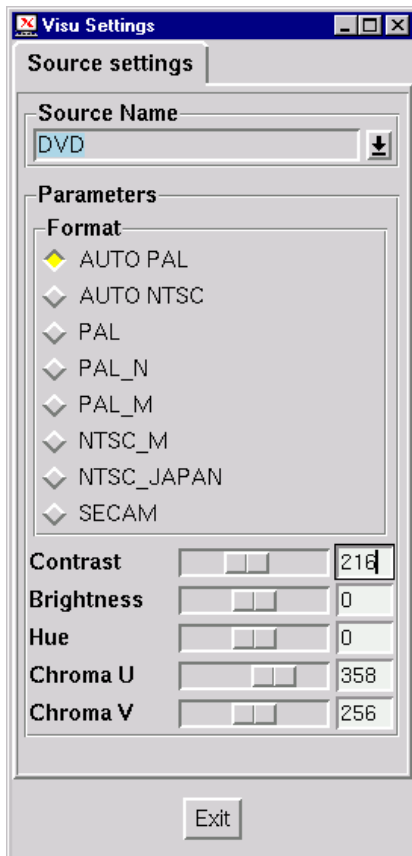
In order to set up the VISU's environment, a number of VISU related settings may need to be tuned. This is described within the next sections. Some settings are of a general nature for the complete VISU stack, some other settings are specific for some Video or RGB source.

6.1 VISU Source Settings

6.1.1 Video Source settings

Most settings for video sources connected are set up automatically and match up to the desired result in most cases.

The following settings may still be modified to your own preference:



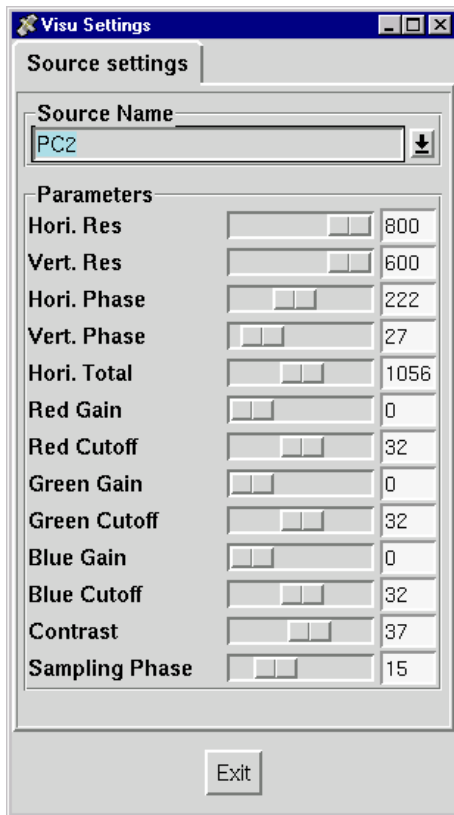
6.1.2 RGB Source settings

Setting up the RGB sources takes somewhat longer than setting up a video source. 3 types of RGB insertion cards are available each with their own set of settings. The next 3 sub-sections describe the way to set up each of these sources.



It is important to note that the 3 sections refer to the type of input-board in the VISU and not the type of source!

SVGA RGB Input

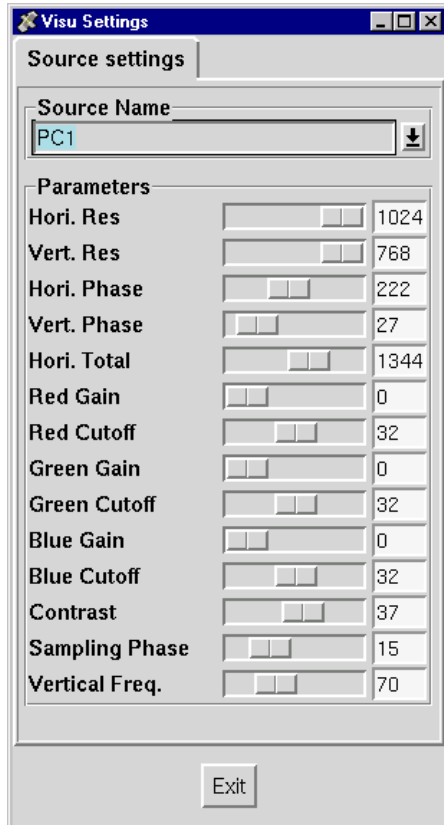


The order in which the settings are to be modified should be:

Item	Description
Hor. Res	Total number of visible pixels horizontally
Vert. Res	Total number of visible pixels vertically
Hori. Total	Total number of pixels (including invisible pixels). For an 800x600 source, this is typically set to 1056
Hori. Phase	Positioning of information within the frame horizontally
Vert. Phase	Positioning of information within the frame

	vertically
Sampling Phase	Used to fine-tune the picture stability. It is used to remove the little instabilities which might show up.
Contrast	Set-up of the contrast for this particular source
Red, Green and Blue gain	Color alignment of overall RGB inserted image
Red, Green and Blue cutoff	Color alignment of overall RGB inserted image

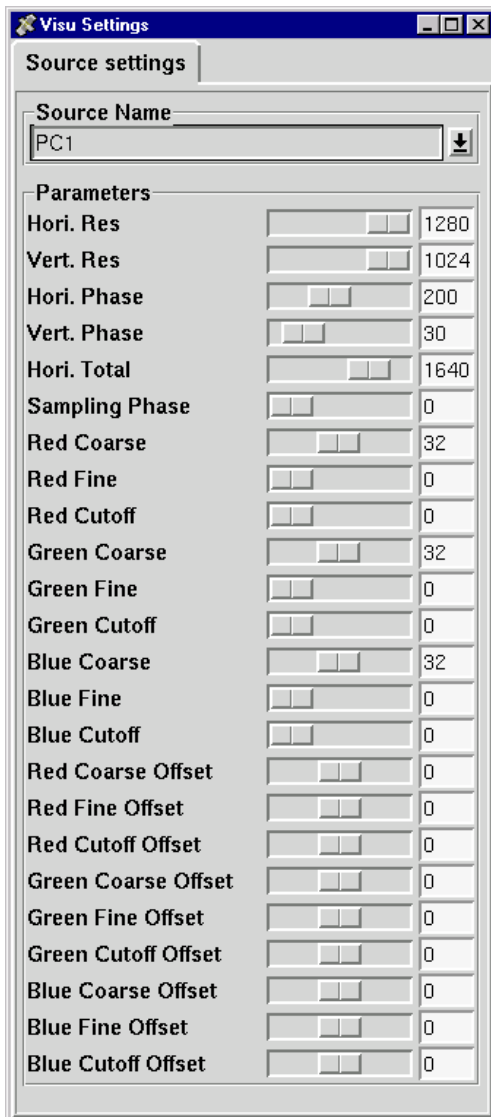
XGA RGB Input



Item	Description
Vertical Freq.	It may be that the vertical frequency is set to a frequency which is too low. This may result in an image which is completely not in sync. Therefore, set this freq. To a frequency higher than 60 Hz, matching the sources frequency as closely as possible.
Hor. Res	Total number of visible pixels horizontally
Vert. Res	Total number of visible pixels vertically
Hori. Total	Total number of pixels (including invisible pixels). For a 1024x768 source, this is typically set to 1344
Hori. Phase	Positioning of information within the frame horizontally
Vert. Phase	Positioning of information within the frame

	vertically
Sampling Phase	Used to fine-tune the picture stability. It is used to remove the little instabilities which might show up.
Contrast	Set-up of the contrast for this particular source
Red, Green and Blue gain	Color alignment of overall RGB inserted image
Red, Green and Blue cutoff	Color alignment of overall RGB inserted image

SXGA RGB Input

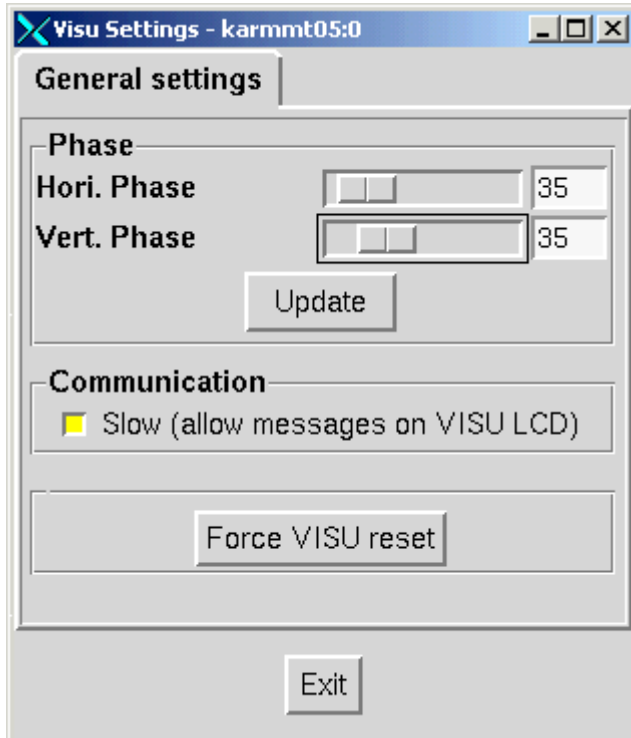


Item	Description
Vertical Freq.	It may be that the vertical frequency is set to a frequency which is too low. This may result in an image which is completely not in sync. Therefore, set this freq. To a frequency higher than 60 Hz, matching the sources frequency as

	closely as possible.
Hor. Res	Total number of visible pixels horizontally
Vert. Res	Total number of visible pixels vertically
Hori. Total	Total number of pixels (including invisible pixels). For a 1280x1024 source, this is typically set to 1640
Hori. Phase	Positioning of information within the frame horizontally
Vert. Phase	Positioning of information within the frame vertically
Sampling Phase	Used to fine-tune the picture stability. It is used to remove the little instabilities which might show up.
Red, Green, Blue Coarse	Color settings
Red, Green and Blue Fine	Color settings
Red, Green and Blue Cutoff	Color settings
Red, Green, Blue Coarse Offset	Color settings
Red, Green, Blue Fine Offset	Color settings
Red, Green, Blue Cutoff Offset	Color settings

6.2 VISU General Settings

6.2.1 Phase settings



Item	Description
Communication Slow	fast is recommend, select slow for trouble solving only

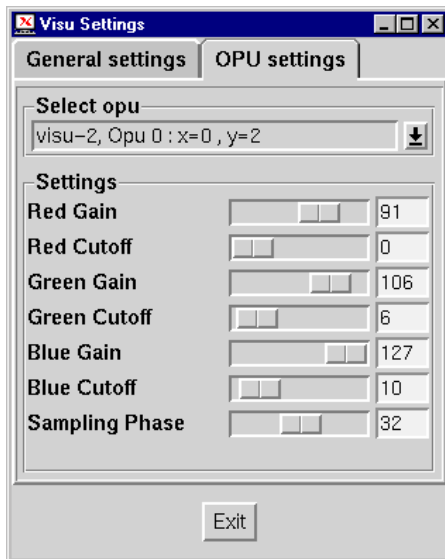
The general horizontal and vertical phase settings of the VISU may be modified in this section. The default settings normally suffice and should not be changed. However, when one has black borders between 2 adjacent screens within the VISU window or when part of 1 VISU window shows up in an underlying VISU window or the other way round, a further tuning of the hphase and/or the vphase would be required.

The menu above will contain entries for each logical screen. When multiple logical screens use the same VISU timing the entries refer to the same information. This means that changing a value in one logical screen may, at the same time, change the value for another logical screen.

6.2.2 OPU settings

Each individual VISU OPU has a set of settings which may be tuned. This may be required in order to:

- do the color alignment of the VISU picture
- attain a stable RGB inserted picture



The following parameters are available for each OPU that may be selected

Item	Description
Red Gain	Color alignment
Red Cutoff	Color alignment
Green Gain	Color alignment
Green Cutoff	Color alignment
Blue Gain	Color alignment
Blue Cutoff	Color alignment
Sampling Phase	Stability of the inserted image over that OPU can

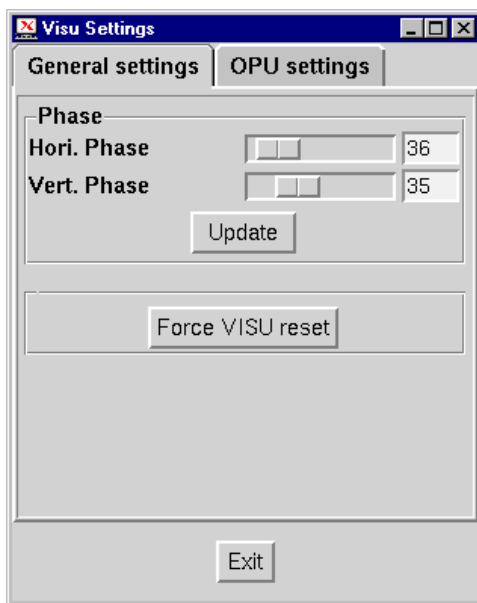
	still be tuned somewhat using this parameter. The default position is set to 30. This value should only be modified slightly.
--	--



It is important to make sure that the projectors are properly aligned with respect to geometry and color on the EOS picture prior to the VISU window color alignment.

6.3 Force VISU Reset

In isolated conditions it may be needed to force a VISU-stack reset. This may be achieved by clicking the Force VISU Reset button.

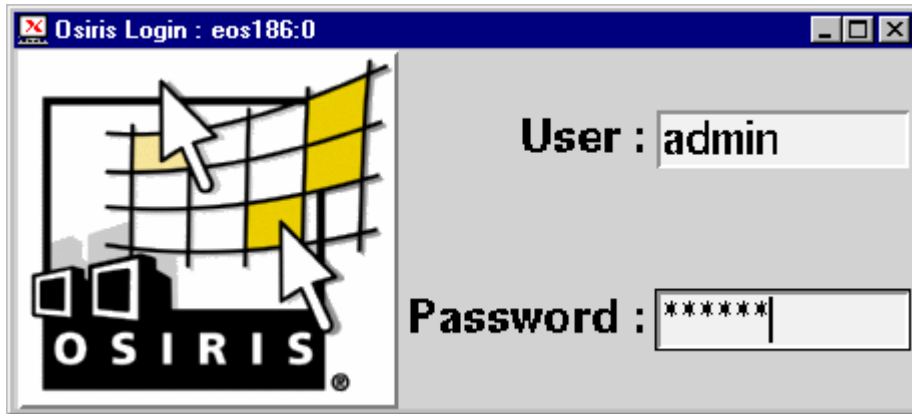


When an explicit VISU reset was performed, a automatic restart of Osiris will be initiated!

7 Osiris Startup

As Osiris is now installed and configured for usage, we will have a look at the startup of the Osiris User Interface.

7.1 Osiris Login



Upon the startup of Osiris, the Osiris Login box appears, which requests you to logon, in order to access the rest of the Osiris GUI.

The default system where the display of this Login box is exported to, is the system defined as `,mvgs_pc'` in the `,etc/hosts'` file of the system running Osiris. This will be the system that displayed the installation menu during activation of Osiris.

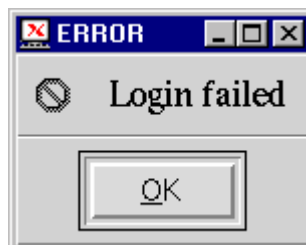
Two user accounts are predefined:

Username	Password	Usage
ofc	glad01	Normal user
admin	secret	Administrator



Be aware that the username and password is case-sensitive. Also leading and trailing spaces are seen as separate characters and may prevent you from logging in.

If a user account is denied, the following message appears:



Possible reasons why the Osiris Login box would not appear:

Possible cause	Remedy
The display-console is not the ,mvgs_pc‘ system	Check the ,/etc/hosts‘ file and verify that the ,mvgs_pc‘ entry is present and points to the correct IP-address
The GUI configuration file ,gui.cfg‘ contains syntax errors.	Check the ,gui.cfg‘ for possible errors, fix, and retry. Osi_gui process will have to be killed by hand in order for it to restart.
The display-console is not connected to the LAN or is on a different sub-net.	Check whether communication can take place between the ,mvgs_pc‘ and the Osiris system (mvgs). Use ping to check this.
The display-console does not have X-windows display capabilities	Verify that X-windows access is allowed from another system. If a Windows PC is used, verify that a X-window emulator SW packet is running. The standard ,xdpyinfo‘ application can be used from another host to check whether X-access is allowed.
The environment variable DISPLAY is not set correctly	check the /etc/rc.d/init.d/osiris file
No GUI is configured for startup	check the file \$BVS_HOME/config/osi_daemon.cfg to verify that the gui will be started on \$DISPLAY

7.2 Control Panels

7.2.1 General Osiris Control Panel

When login was successful, the following control panel will be displayed when logged in as ,admin‘ user (providing no changes were made to the default configuration):



Button functionality:

- 1st Control Manager
- 2nd Active Layout
- 3rd Projector Control
- 4th Alignment Tools
- 5th Remote Pointer
- 6th Zoom X
- 7th VISU sources setup
- 8th On-line Help
- 9th Administration Control Panel
- 10th Errorlog window
- 11th Exit

A detailed discussion on these different sections will be explained in the next chapters.



7.2.2 Administration Control Panel

The Administration Control panel is displayed when selecting the appropriate button on the previous control panel.

Button functionality:

- 1st System Console
- 2nd User Administration
- 3rd VISU general settings
- 4th UCL Enable/Disable
- 5th Osiris restart
- 6th System Reboot
- 7th System Halt
- 8th Backup/Restore the system
- 8th Hardware Configuration
- 10th Upgrade of Osiris

7.2.3 Multiple Osiris GUI session

Different instances of the Osiris GUI may be displayed on different systems or on the same system at any one time. To start up a new instance, the following command is to be used:

```
$BVS_HOME/bin/osi_startmodule osi_gui -display $DISPLAY
```

Where:

\$BVS_HOME = the root for the Osiris installation (/home/bvs by default)

\$DISPLAY= the display console on which the user-interface has to appear.

To configure Osiris to always bring up multiple GUIs after restart you need to modify the configuration of the osi_daemon process:

File:

```
$BVS_HOME/config/osi_daemon.cfg
```

Add entry:

```
karmmt05:gui:gui1:1:respawn:/home/bvs/bin/osi_startmodule osi_gui -display <DISPLAY>
```

Where <DISPLAY> is the XDisplay you want the GUI to be displayed on. The third entry 'gui1' may be any string and is only necessary to distinguish different entries referring to the same type of module.



By default, a maximum of 7 different GUI sessions may be active at any one time.
When more sessions should be needed, contact a Barco Controlroom Division representative.

7.2.4 Customizing the Control Panel

For all possible GUIs for all possible users, 1 file defines the configuration:

```
$BVS_HOME/config/gui.cfg
```

7.2.5 Syntax for the GUI configuration file

This configuration file is parsed in TCL, and this means that it is evaluated in the same way. This means it is line-oriented, "#" at the beginning of a line denotes a comment-line, strings can be quoted by a brackets ({} or double-quotes (""), backslash (\) at the end of a line means that the next line is a continuation of the current line, all commands are case-sensitive... Each line defines a button on the control-panel. The syntax for the commands is as follows:

MODULE <userspecs> <name> <buttondecoration>: this button starts a gui-module (this is a Tix-class defined as *gui_<name>*). <buttondecoration> defines the view of the button (see below). <userspecs> defines which user can see this button (see below).

COMMAND <userspecs> <id> <user> <command> <buttondecoration>: this makes the agent-module executing command <command>. This command is run on the EOS as user <user>. If id is specified (not empty), the button will be configure as a toggle-button, the application will stop automatic when the user exits the GUI. In <id> you can specify a \${BVS_HOME}, \${DISPLAY} or \${GW_DISPLAY}; these parameter will be replaced with the corresponding environment variables.

OSICOMMAND <userspecs> <module> <command> <confirm> <message> <decoration>: This sends a command "<command>" to the OSIRIS-module "<module>". <confirm> is a boolean value (yes/no) which specifies whether or not a confirmation is due before executing the command. If yes is specified then the message <message> will be used to ask the user for configuration.

OSITOGGLECOMMAND <userspecs> <module> <command1> <command2> <statuscommand> <event1> <event2> <confirm> <message> <decoration1> <decoration2>:

This does the same thing as OSICOMMAND except that it is used as a toggle button: it switches between <command1> and <command2>. The status is shown with <decoration1> and <decoration2>. The state is changed by two events: <event1> and <event2> switches between state 1 and 2. <statuscommand> sends a command that should generate the appropriate event at startup. Note:

these events are NOT generated by the GUI-module, the controlling module (the wall for instance) should handle them.

MENU <userspecs> <buttondecoration> {list of commands}: This generates a submenu of the controlpanel. The list of commands is also a list of line-separated commands corresponding to one button. These lines can be any of the above commands.

DEVCTRL <userspecs> <filename> <buttondecoration> : This generates a separate window (if the button is pressed) with buttons, labels and sliders. This is mainly used for device control (via a crestron or amx device). The filename specified is the name of the configuration file of the menu. This file should be located in the \$BVS_HOME/config directory.

ACTIVATE

When placed as the last argument,

The button-decoration is defined by a list of elements. This will generate a combination of text, bitmaps, images ... Following elements are known:

text <text>: displays a text

bitmap <bitmapfile>: displays a text

image <imagefile>: displays an images (mostly a .gif-file).

newline: let the next element start on a new line

space <hspace>: puts a space between the previous element and the next

vspace <vspace>: puts a vertical space

font : changes the font for following text elements (already defined text is left alone).

color <color>: changes the default foreground for following elements.

balloon <balloontext>: if this is specified, then moving over this button will cause a balloon to pop up. <balloontext> is the message that will be shown. This message can be or the actual text or a string denoting a string in the registry file (eg STACK will get the message "Control Manager" from the registry file if the language is English).

The first field in each command defines which user can access the button (other users just do not see that button). It is a tcl-list (strings separated by spaces enclosed between brackets { }) of "allowed" users. For instance {ofc admin} allows users ofc and admin. If the keyword "all" is specified then all users are allowed. Names preceded with an exclamation mark are not allowed to use the button, even when "all" is specified.

For instance {all !ofc} allows all users except the user ofc.

7.2.6 Example of multi-user control panel setup

The following file shows a configuration file gui.cfg which brings up a control panel either for the ,admin' user, where all buttons are accessible, or for a ,user' user where a more restricted set of buttons is accessible:

```
# GUI.CFG file : designs the control panel
#
# Modules :
# MODULE {<userspecs>} <modulename> {<button-decorations>}
```

```

# possible button-decorations :
#   text "text" : text to display
#   space number : space to insert
#   newline : a new line to begin a new line of text/images/....
#   image "image" : a image (gif file) to display
#   userspecs = list of allowed users for a button, all allows all users, !user means
#               allows not this user
# Eos Commands :
# COMMAND {<userspecs>} "id" "user" "commandline" {<button-decorations>}
# Osiris Commands :
# OSICOMMAND {<userspecs>} "module" "commandline" "confirm(yes/no)" "confirm-message" {<button-dec.>}
# SubMenu :
# MENU {<userspecs>} "title" {<button-decorations>} {<New buttons ...>}

MODULE {all !user} stack {image {gui_stack} balloon {STACK}}
MODULE {all} activelayout {image {gui_activelayout} balloon {ACTIVELAYOUT}}
MODULE {all !user} projcontrol {image {gui_projcontrol} balloon {PROJCONTROL}}
MODULE {all !user} projadjust {image {gui_projadjust} balloon {PROJADJUST}}
COMMAND {admin} "remote-$DISPLAY" "ofc" "$BVS_HOME/bin/osi_startmodule osi_remptrclient -user
$USERNAME -size 64" {image {gui_remptr} balloon {REMOTEPTR}}
COMMAND {all !admin} "remote-$DISPLAY" "ofc" "$BVS_HOME/bin/osi_startmodule osi_remptrclient -user
$USERNAME -size 64" {image {gui_remptr} balloon {REMOTEPTR}}
MODULE {all !user} visu {image {gui_visusettings} balloon {VISUSOURCE}}
MODULE {all !root !admin} user {image {gui_user} balloon {PASSWORD}}
COMMAND {all} "help-$DISPLAY" "ofc" "$BVS_HOME/bin/netscape file:///BVS_HOME/doc/main.html" {image
{gui_help} balloon {DOCUMENTATION}}

MENU {root admin} "System administration" {image {gui_sysadmin} balloon {SYSADMIN}} {
COMMAND {all} "CONSOLE-$DISPLAY" "ofc" {xterm -T Console -sb -e $BVS_HOME/bin/osi_createconsole} {image
{gui_console} balloon {CONSOLE}}
MODULE {all} user {image {gui_user} balloon {PASSWORD}}
MODULE {all} visuadmin {image {gui_visusettings} balloon {VISUGENERAL}}
OSITOGGLECOMMAND {all} wall "uclenable" "uclisable" "uclstatus" "wall_uclOff" "wall_uclOn" no "" {image
{gui_ucloff} balloon {UCLOFF}} {image {gui_uclon} balloon {UCLON}}
OSICOMMAND {all} server "stopServer -origin \[$DISPLAY\]" yes "Restart Osiris" {image {gui_restart} balloon
{RESTARTOSI}}
OSICOMMAND {all} sad "reboot -delay 0" yes "Reboot machine" {image {gui_reboot} balloon {REBOOT}}
OSICOMMAND {all} sad "shutdown -delay 0" yes "Shutdown machine" {image {gui_shutdown} balloon
{SHUTDOWN}}
MODULE {all} backup {image {gui_backup} balloon {Backup/Restore the system}}
COMMAND {all} "EDITCONFIG-$DISPLAY" "root" "$BVS_HOME/bin/osi_startmodule osi_wall -edit" {image
{gui_editconfig} balloon {EDITCONFIG}}
MODULE {all} upgrade {image {gui_upgrade} balloon {UPGRADE}}
}

```

7.3 Osiris GUI color usage

The Osiris GUI may be displayed using either 16 or 256 colors. In some conditions, running the GUI using 256 colors results in color-allocation problems for the rest of the X-windows applications displaying to that same DISPLAY.

To change the color usage:

- for changing to **16 colors**:

telnet to the EOS as ofc

become super-user:

```
cd /home/bvs/images
rm -f *
ln -s ../pkgs/tkgui/images_16/* .
```

- for changing to **256 colors**:

telnet to the EOS as ofc

become super-user

```
cd /home/bvs/images
rm -f *
ln -s ../pkgs/tkgui/images/* .
```



The previous modifications might be needed when color-allocation problems manifest themselves when displaying to some X-windows platform.

It is advised to use only 16-colors for the GUI in all circumstances.

8 Control Management

The real power of Osiris is held within the Control Management section. It basically allows a user to set up a number of applications, group them in layouts and assist in the automatic startup and stopping of these applications held within these layouts.

This just briefly describes the possibilities. More detailed information is found within the next subsections.

8.1 Introduction

As specified within the introduction of this manual, the Control Manager functionality is the core-business of the Osiris SW. It allows customers to define the way their applications are to be started and where they have to be displayed on the Display Wall.

The individually defined applications can then be started in an automated manner and grouped within the so-called ‚Layouts‘.

A layout, when loaded, initiates the automatic startup of the applications comprised within that layout and, when unloaded, closes the applications again in a secure automated fashion.

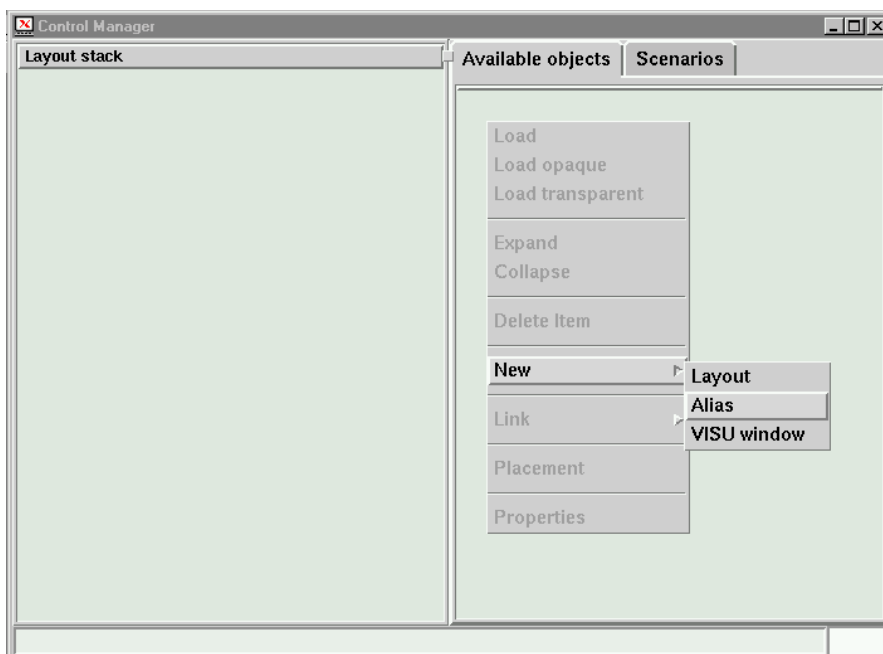
In order to be able to define all this, one will need some knowledge with respect to:

- Location of applications to be started (which servers?)
- user(s) which should start up the application(s)
- super-user password of servers on which applications are to be started
- UNIX , in general (basic knowledge) and UNIX security issues specifically (basic knowledge)
- X-windows principles (basic knowledge)

8.2 Object Definition

To create an object within Osiris, you need to open the Control Manager section of the Control Panel, then Right-click in the Available Objects section of the window and select ‚New‘.

Within the ‚New‘ section, one can create a new Layout, Alias or VISU window:



8.2.1 Alias

An alias is a definition of an application. It defines where to start it up, with what user, from which server, etc...

The screenshot shows a dialog box titled 'xclock'. It contains the following fields and options:

- Name:** xclock
- New name:** xclock
- Commands:**
 - Start command: /usr/X11R6/bin/xclock -update 1
 - Stop command: (empty)
 - Clean command: (empty)
- Credentials:**
 - User: ofc
 - Agent: agent
 - Host: (empty)
- Methods:**
 - Restart:** Never, 5 Times (selected), Forever
 - Start method:** local (selected), rsh, ssh
 - Stop method:** command, break (selected), terminate, kill
 - Magic cookie:** Cookie (selected), Display, No Cookie/No Display
 - Fixed cookie value:** (empty)

Buttons: Apply, Ok, Cancel

Properties

The next table describes each entry within the alias definition window:

Object component	Purpose
Alias Name	Unique Alias name
Start Command	Definition of startup command. This is the exact name + necessary options to start the application
Stop Command	If needed, a specific stop-command can be specified. In most cases, this is not required. It depends on the application whether this is required or not.
Clean Command	One can even further specify a Clean command, which is executed, when the stop command has been executed and the

	application has stopped running.
Username	Name of the user with which the application is to be started, either locally or on the remote host.
Agent	By default, the ,agent‘ module is responsible for starting this particular when needed. It would be possible to specify a different agent when such an agent would be available.
Host	Host on which the application is to be started. This may be either local on the system (,mvgs‘ or blank) or on a remote host. The list of hosts one is presented is originating from the local /etc/hosts file.
Restart Method	The automatic restart of the alias can be either: <u>Never</u> : no restart <u>5 times</u> : Osiris will attempt to restart the application 5 times. If the 5 th time fails all further attempts will cease. <u>Forever</u> : Osiris will retry continuously to start the application
Start Method	Defines the way the application is to be started: <u>local</u> : Alias to be started locally on the system <u>rsh</u> : Alias to be started on a remote host using the Remote Shell method <u>ssh</u> : Alias to be started using a secure shell.
Stop Method	Method to use when stopping the alias: <u>Command</u> : Will then use the stop-command specified to stop the application <u>Break</u> : Will effectively remove the X-windows link between the application (X-client) and the X-server. This is the default stop-method. The application is expected to cleanly shut down when this is done. <u>Terminate</u> : This method makes sure that any intermediate startup script is stopped as well. <u>Kill</u> : Sends a ,kill -2‘ signal to the application, requesting it to shut down. Each one is a checkmark which can either be on or off. So combinations are also possible
Magic Cookie	This section defines the way in which the X-server differentiates between the incoming application (X-client) connections: <ul style="list-style-type: none"> - Cookie: Standard X-windows authorization mechanism is used. (default method) - Display: Different way of differentiation between application connections - No Cookie/No Display: none of the above (exotic)
Fixed Cookie	In some cases, an application may only make a connection using a fixed Cookie. One can specify that cookie within the entry-field. An application for this might be when tight security does not allow the agent-module to remote start up an application on a remote host. A layout could still be built using dummy aliases, each with a separate fixed cookie. The applications are then started from the server(s) using the

	different fixed cookies. In this way, proper placement can still be guaranteed.



The properties of some alias may only be modified when it is not loaded on the Active Layout stack.

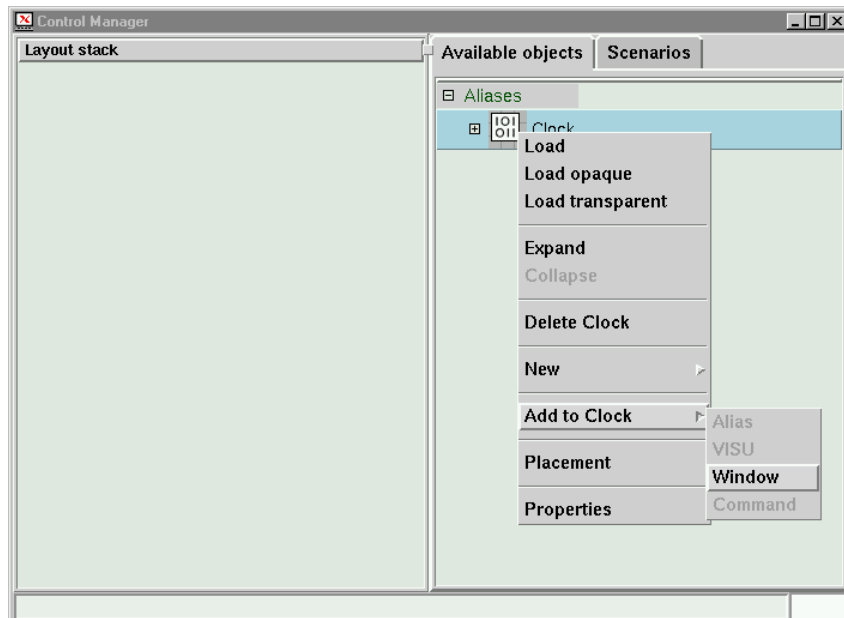
Default Windows

An alias has a Default Windows area associated with it, which defines the area in which all windows are to be displayed that do not match any other separately defined windows for this alias. The default window area, as the other separately defined window areas, can be assigned some properties:

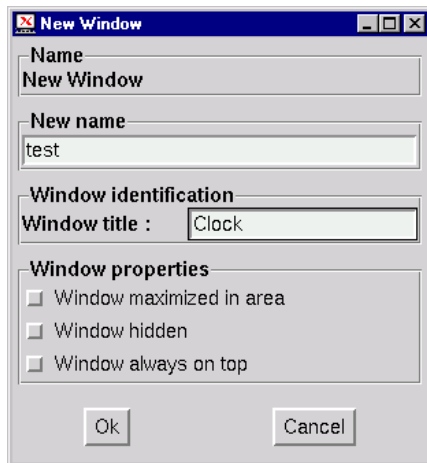
Property	Purpose
Maximized	Will maximize the application's window(s) in the designated area (default area)
Hidden	The area will be hidden (in an off-screen location)
Always on Top	Application windows within this area will be forced on top when another application's window overlap.

Separate Windows

Apart from the Default Area, associated with an application, one can define some separate application window areas. This can be achieved by selecting the alias, right-clicking on it and selecting the ,Add to <alias-name>' /Window option:



Selecting this option will present you with the form for specifying the separate application's window:



The following may be specified:

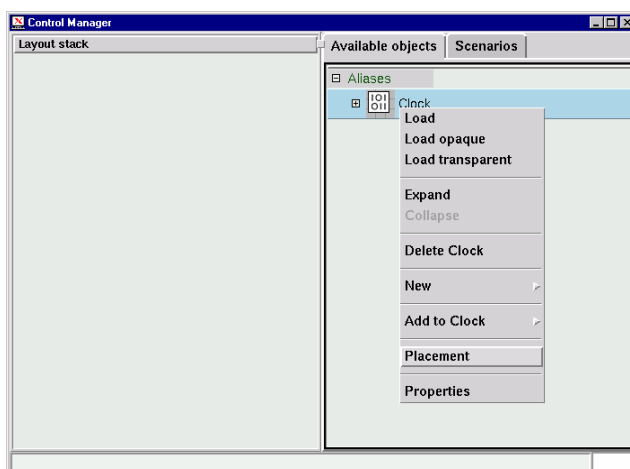
Property	Purpose
Window name	Just a name for the window-object.
Window Identification	Name of the window as seen in the window's title-bar. One may also use the normal Unix regular expressions as *,?,[],etc.. to specify the name of the window.
Window Properties	Same as with the Default Area

When such a separately defined window is used within some Layout, upon the loading of the layout, when the application connects and displays it's windows, the window matching the specifications defined with respect to the title-bar name will be forced into the designated area.

This is the method to differentiate between the different resulting application windows. In this way one can easily set up which windows one is interested in and show these separately. The rest, not defined, will end up in the Default Area, which may be set as being ,Hidden'.

Placement

The placement option is used to organize the default placement for the application Window(s):



The placement window allows you to place the windows related to the Alias and resize them. For this purpose, one can either use the mouse or the Positioning entry-fields by hand.

By default, a newly created alias is placed in the top-left corner of the screen.

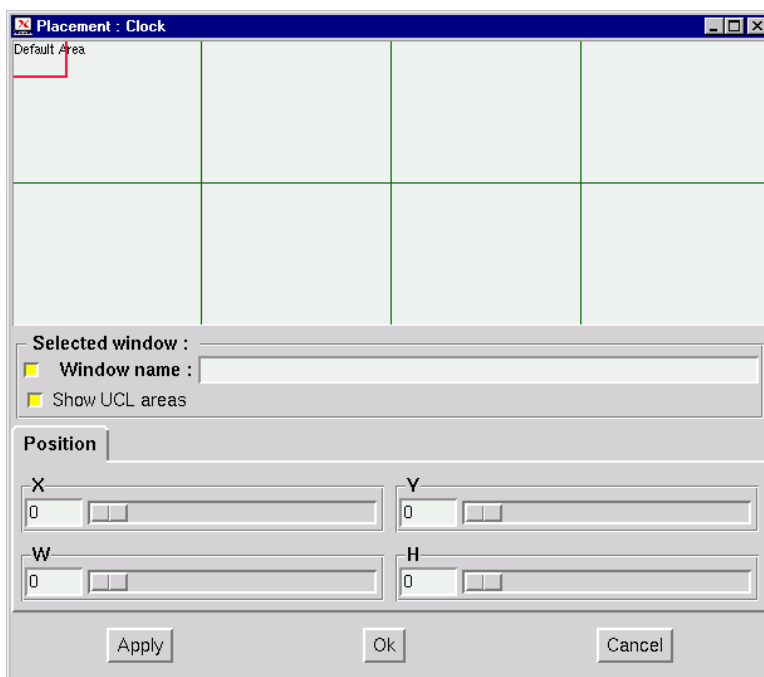
To operate on some area, one need to select the area first by left clicking on the are-name or one of the sides of the area. When the title changed color from black to blue, the area is selected.

Right clicking and holding while you move the area to the new location does moving an area. Left clicking on one of the corners or sides and dragging to modify the aspect ratio does resizing.

To confirm the changes, use either the Apply or the OK button.



It is important to note that the placement definition for an alias is only related to that one alias and not to the placement of that alias in the different layouts where it is used.



Loading an Alias

There are 2 ways of loading or starting an alias:

Opaque: before application comes up, all application windows already up on the screen are hidden first. The applications that were running are not stopped however!

Transparent (default): screen layout is left as it is, alias is loaded on top of what was already there.

When the application is loaded, the Transparent method is used. One may however explicitly load an alias transparent or opaque.

Deleting an Alias

To delete an alias from the Available Objects section, select the alias from the list, right-click and select the 'Delete' option.



Deleting an alias which makes part of some layout or scenario will not succeed.

You will need to remove the alias from the involved layouts and scenarios first in order for this to succeed.

Multiple Logical Screens

When the X-server is set up to have multiple logical screens, one will be able to select the default logical screen on which the alias windows are to appear:

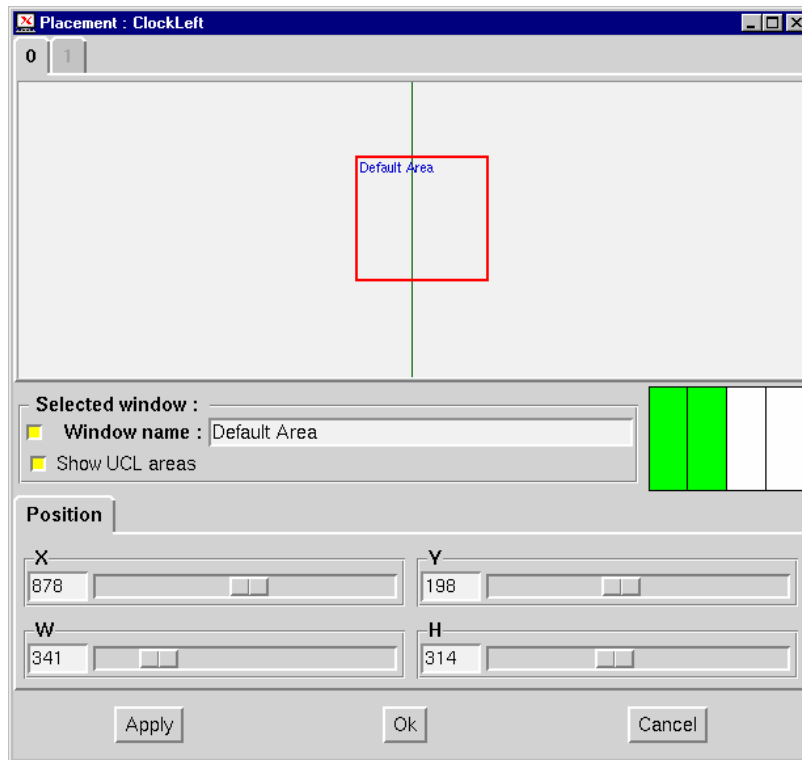
The screenshot shows a dialog box titled "ClockRight". It contains the following fields and options:

- Name:** ClockRight
- New name:** ClockRight
- Commands:** Start command: /usr/X11R6/bin/xclock
- Credentials:** User: ofc, Agent: agent, Host: (empty)
- Screen:** Screennumber: 1
- Methods:** Restart: Never, 5 Times (selected), Forever; Start method: local (selected), rsh, rexec; Stop method: command, break (selected), terminate, kill
- Magic cookie:** Cookie (selected), Display, No Cookie/No Display; Fixed cookie value: (empty)

Buttons: Apply, Ok, Cancel

The [ScreenNumber] entry field denotes the logical screen to display on (0,1,...)

The placement window will have tabs for selecting the logical screen to organize:



8.2.2 VISU window

A VISU window defines a link between a certain VISU source and an X-window. The X-window is treated as any other X-window on the screen.

To create a new VISU window, right-click on the right-hand side of the Control Manager window and select 'New/Window'

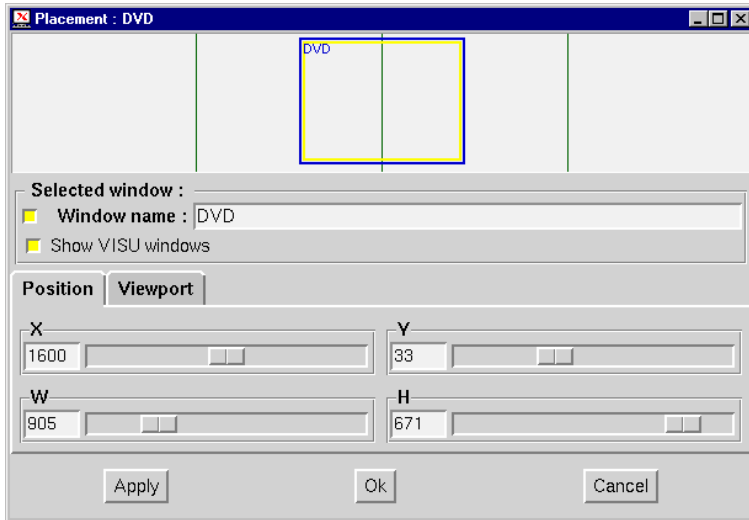
Properties



Object component	Purpose
New Name	Unique VISU window name
Screen	Selection of screen number (cfr. Section 132.3926.490)
Source	Source to be linked to the VISU window. This source is to be defined in the Hardware Configuration section.
Window always on top	To be checked when the VISU window should be displayed on top of all other X-windows

Placement

The placement allows you to define the default position and size of the VISU window:



Loading a VISU window

To load a VISU window, select it from the list of available VISU windows, right-click and select 'Load'. By default, the VISU window will be loaded **transparent**. One can explicitly specify to load the VISU window opaque when this would be required.

Deleting a VISU window

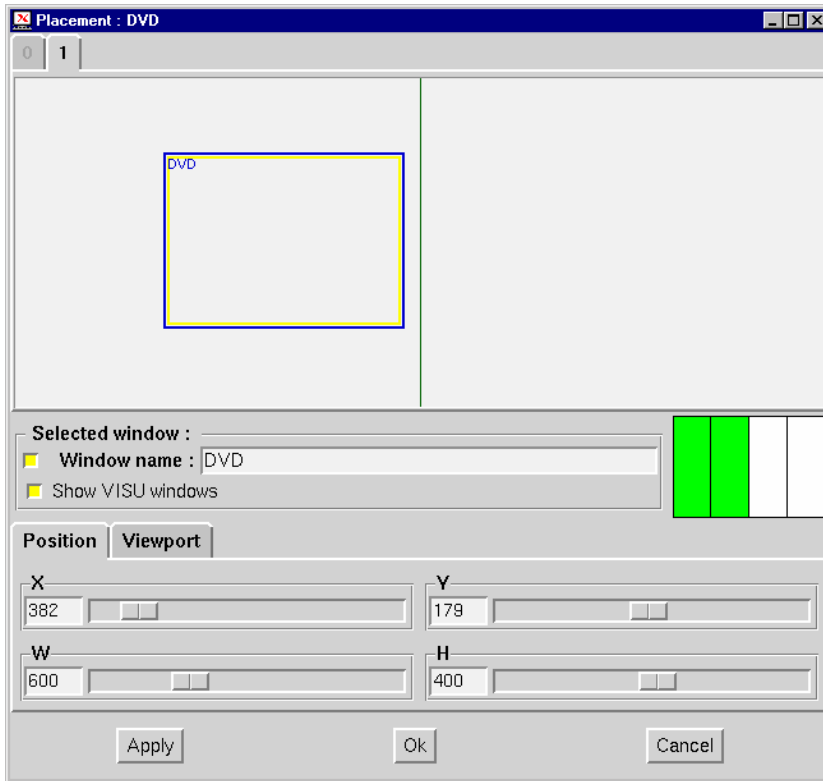
To delete a VISU window, select it from the list of available windows, right-click and select 'Delete'.



Deleting a VISU window which makes part of some layout or scenario will not succeed. You will need to remove the alias from the involved layouts and scenarios first in order for this to succeed.

Multiple Logical Screens

When the X-server is configured to display multiple logical screens, the placement window will look slightly different from the former placement window:



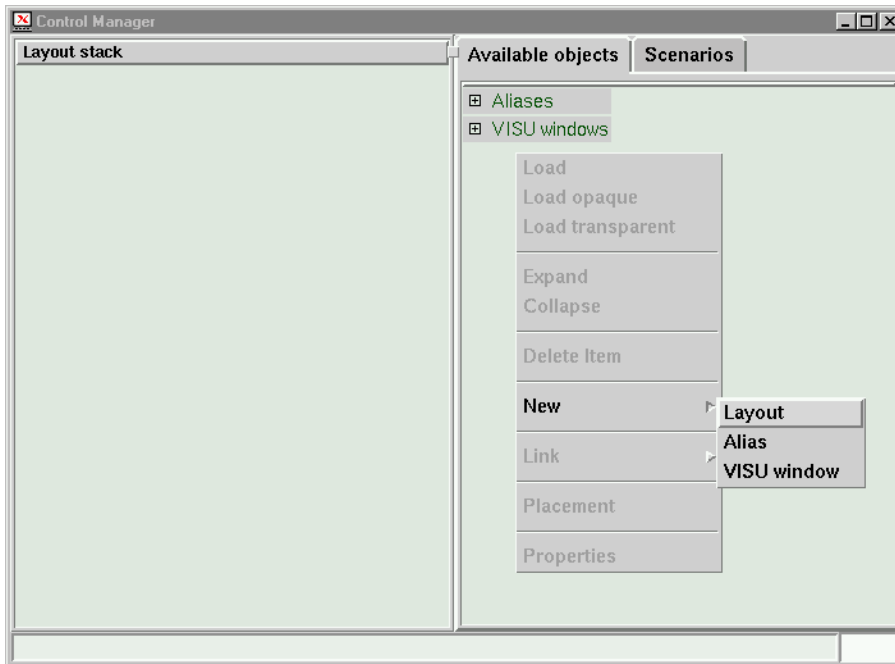
8.2.3 Layout

Properties

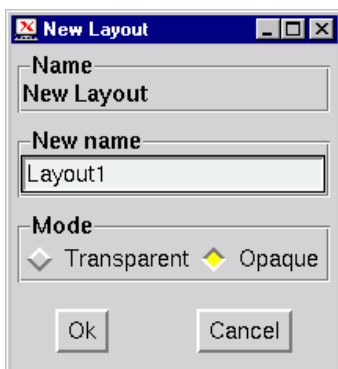
A layout is an object, which may hold:

- a number of Alias Default Areas
- a number of Alias specific Windows
- a number of VISU windows
- a number of Osiris commands to be executed upon the starting and/or stopping of the layout

Creation



As with the creation of other objects, when right-clicking in the Available Objects section, one can select New/Layout:



Following properties need to be completed:

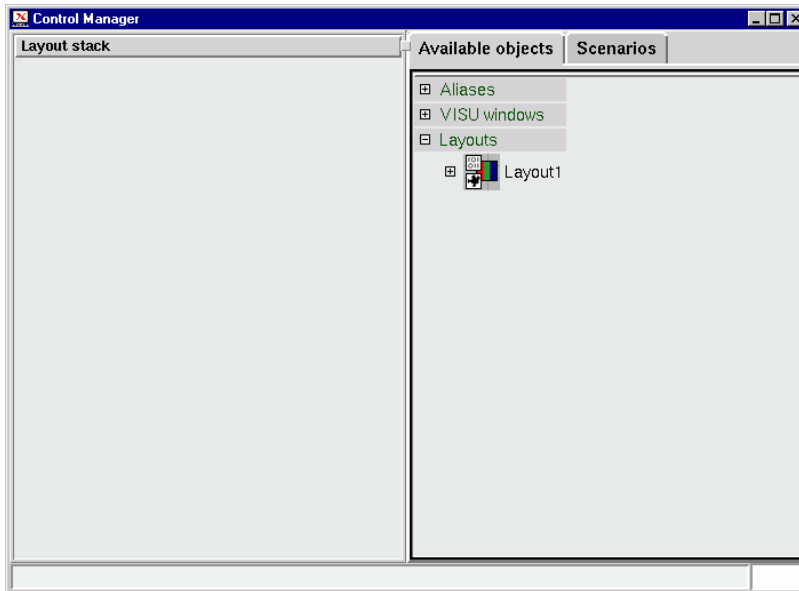
Property	Purpose
Object Name	Unique name for the layout
Layout mode	As with the alias, a layout may be Transparent or Opaque. By default, a layout is loaded opaque. All application windows present on the display wall are made hidden and the new windows from the new layout are displayed.



Items that were already on the Active Layout stack are not stopped when an opaque layout is loaded additionally. Only the application windows are moved to an offscreen location (hidden). The application(s) will still run in the background!

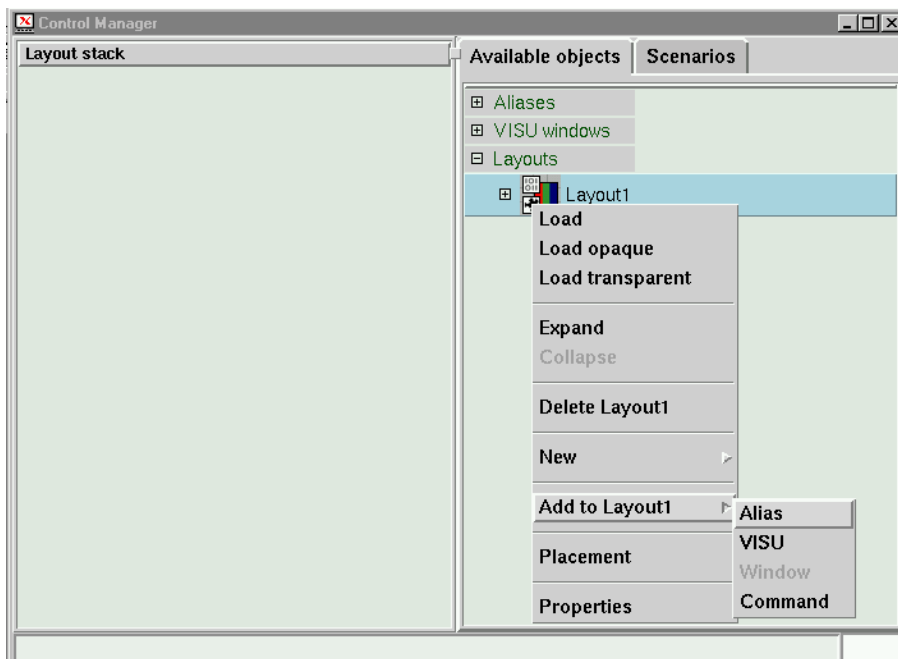
Types of Layouts

As already mentioned, a layout can be opaque or transparent. Loading a layout in the default way will load it as defined in its properties. However, one can force a layout to be loaded opaque or transparent.



Adding objects

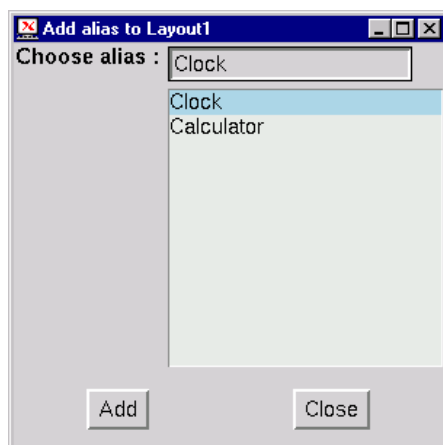
To add objects to a newly created layout:



Select the layout and select the 'Add to <layout-name>' option in the drop-down menu options. Then select one of the items in the list.

8.2.3...1 Adding an Alias

When selecting the Add Alias selection, a list-box will pop up with the available aliases in the database:

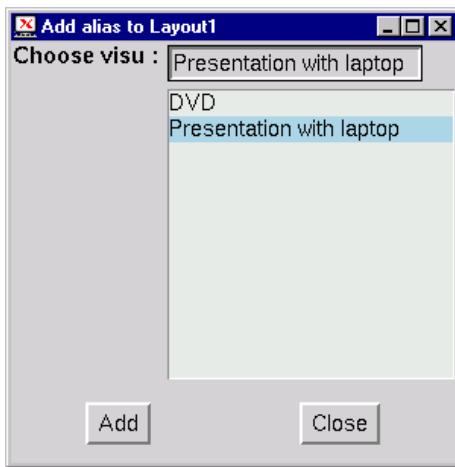


From the list box, select the required aliases and select the [Add] button. The alias will have been inserted in the layout, using the default placement properties it was given.

To complete the adding of aliases, select the [Close] button.

8.2.3...2 Adding a VISU window

When selecting the Add VISU selection, a list-box will pop up with the available VISU windows in the database:

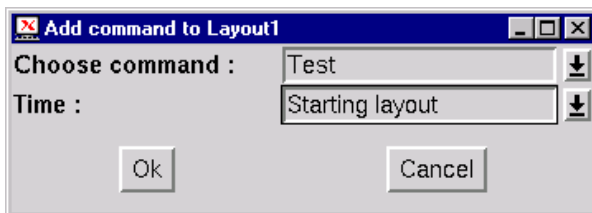


From the list box, select the required VISU windows and select the [Add] button. The alias will have been inserted in the layout, using the default placement properties it was given.

To complete the adding of VISU windows, select the [Close] button.

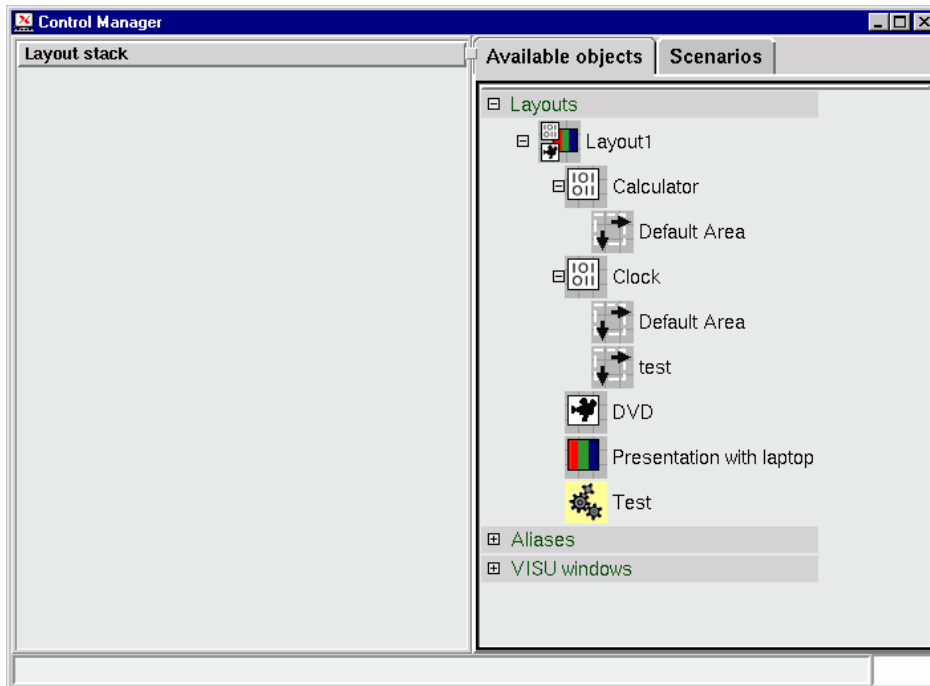
8.2.3...3 Adding an Osiris Command

When selecting the Add Command selection, the following entry form will appear:

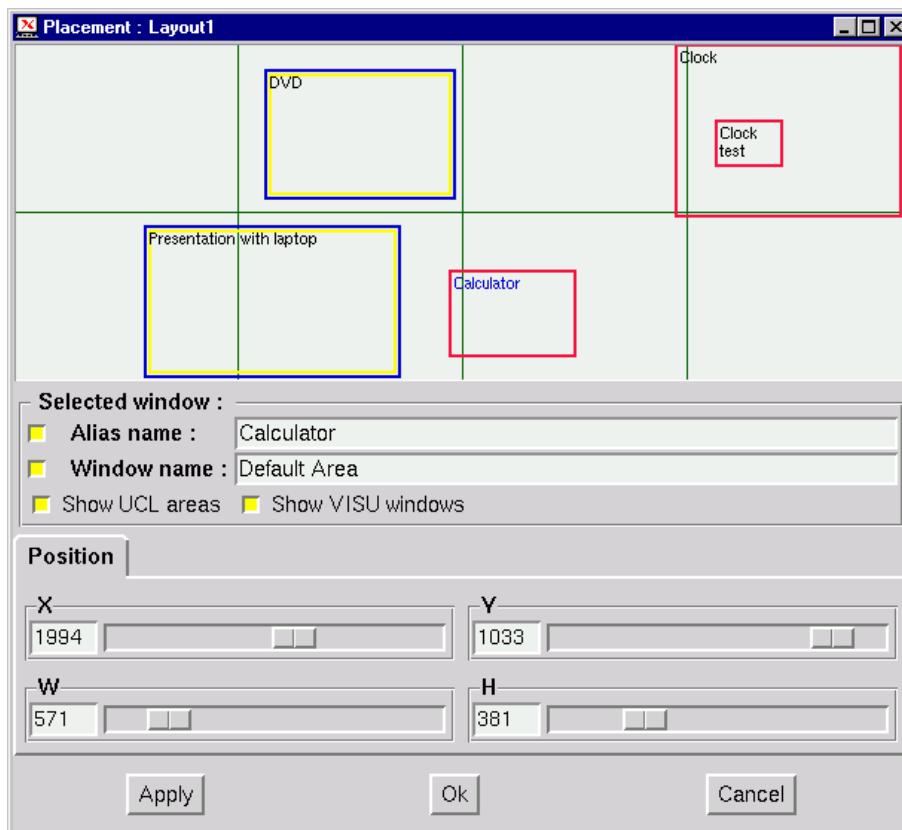


Select the required Osiris command from the 'Choose command' listbox. Then select the time upon which the command is to be executed. Either upon Starting the layout or upon Stopping the layout. Select [Ok]

Grouping objects



In order to define which object goes where, one still needs to define the placement of the layout:



Each object is represented with some area outline. In the next table, an overview is given of what outline color corresponds to what function:

Outline	Functionality
Red	Application windows area. Either the default windows area or area of specific applications window. These areas can be moved and resized
Blue	VISU window outline. The corresponding window will fit into this area when loaded
Yellow	Outlined area within the VISU window which corresponds to the viewport on the information displayed within the VISU window (zooming)
Orange	Corresponds to the actual X-windows that are visible on the Graphics Wall.

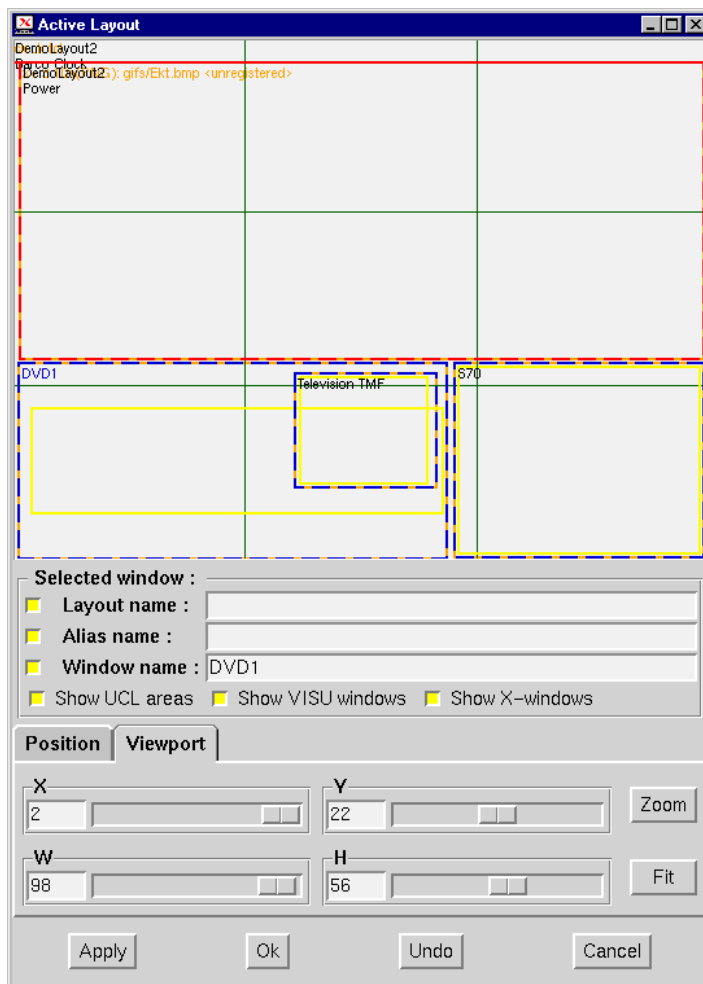
One can hide or show the different areas by checking the checkmarks:

Show UCL areas

Show VISU windows

Show X-windows

An area, which is highlighted, has its title displayed in blue.



When highlighted, the area may be moved, resized by means of the mouse pointer, by means of the slider-bars or by means of the entry-fields. Press [Enter] when entering a new value.

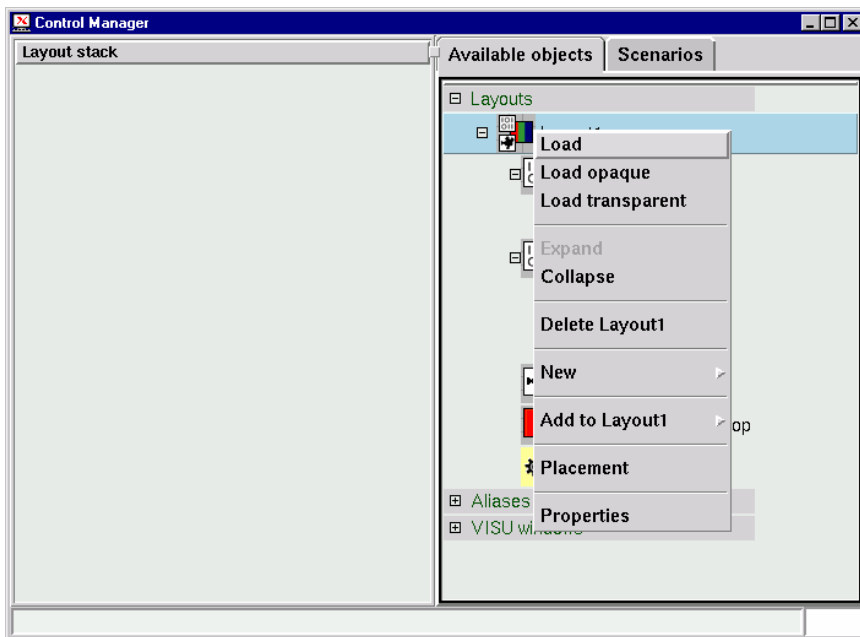
An area can be selected by

- pressing the left mouse button on the label displayed in the area.
- pressing the left mouse button on the border of the area.
- using the drop-down list in the selected window pane.

When a VISU area has been selected it is possible to modify the Viewport for the displayed information within the VISU window. A viewport is a definition of the portion of the information, which is being digitized. This allows you to zoom in on the image.

Loading/Unloading

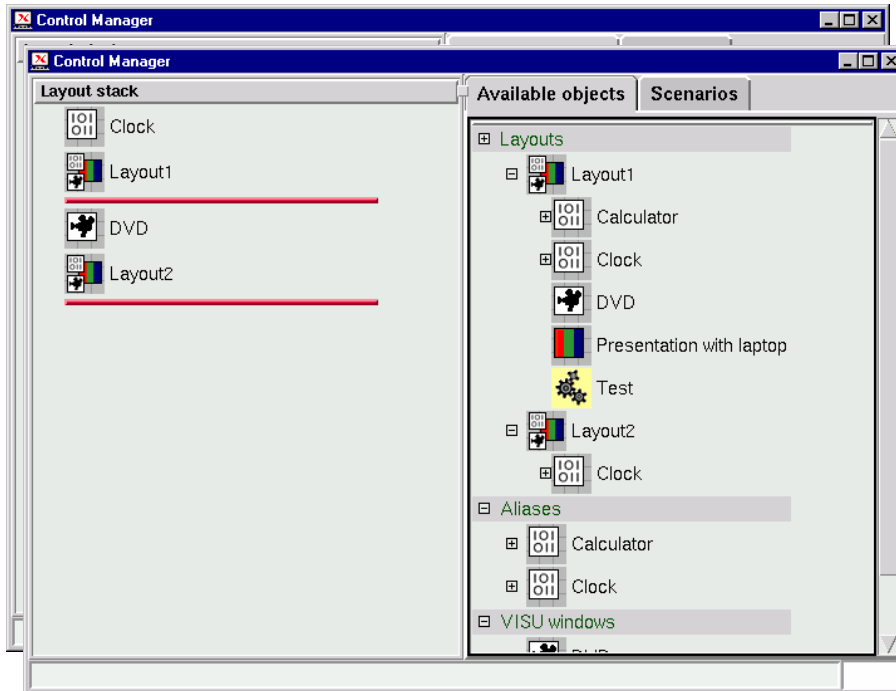
To load a layout: right-click the layout and select ,Load‘



All Aliases, VISU windows and Osiris commands making part of this layout are initiated to start. When a new layout is tested in this way, it is important to check the ,Console‘ window to verify possible errors that might prevent a layout to load successfully.

When loaded opaque (default), it will show up at the Layout Stack side with a red separator underneath, obscuring everything that may already have been showing on the screen.

To unload the layout, select it, right-click and select ,Unload Selection‘. This will remove the layout from the screen and stop all necessary alias objects making part of the layout.



As mentioned, different objects may be loaded on the Layout stack as in the following example:
Some objects are loaded opaque and others are loaded transparent.

To select multiple objects on the stack, use the standard method:

- clicking while holding down the [Ctrl] key selects individual items
- clicking while holding the shift key, selects a range of items



It is important to restate that when 2 layouts Layout1 and Layout2 are loaded, one on top of the other and where alias Alias1 makes part of both layouts, only 1 instance of the alias is active. Unloading 1 of the layouts will therefore not stop the Alias1 object.

Deleting

To delete a layout from the Available Objects section, select the layout from the list, right-click and select the ‚Delete‘ option.

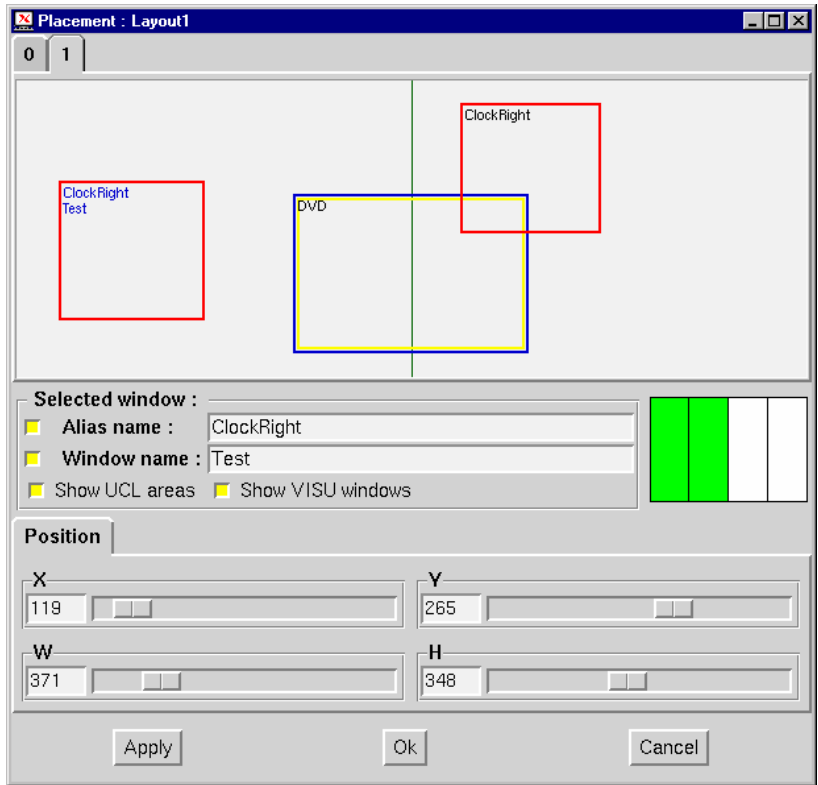


Deleting a layout which makes part of some scenario will not succeed.

You will need to remove the layout from the involved scenarios first in order for this to succeed.

Multiple Logical Screens

When X-server is configured to use multiple logical screens, the Layout placement window will contain some Tabs for selecting the desired screen:



8.2.4 Scenario

Purpose

In order to provide even more flexibility, a ‚Scenario‘ object was introduced. The Scenario provides in an even higher level of integrating the customer environment with the Graphics Wall systems installed.

A scenario could handle:

- the activation of some day-time layout between 8.00am and 6.00pm
- the activation of some night-time layout for the remainder of the day
- The automatic raising of the curtains of some presentation room when a button is pressed on a touch-panel and then showing a presentation DVD displayed within a VISU window and at the same time on a presentation projector.
- Automatic shutdown of the system or reorganization of the most important display-information when a power failure is detected by the UPS system connected to the system
- Performing an automated backup every day.
- And many more possibilities.

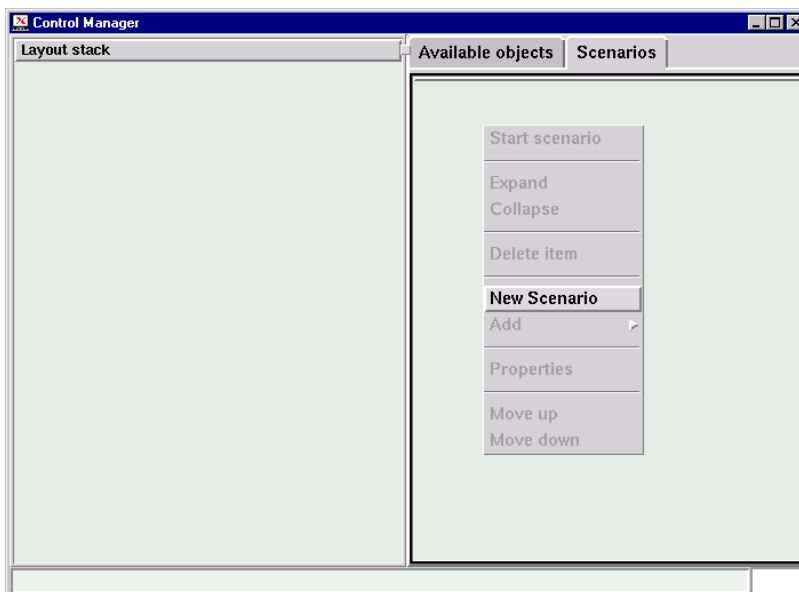
All of this can be programmed within some ‚Main‘ scenario that takes care of all these possible events that might come by. This scenario would then run day in, day out, managing the complete Control Room environment.

Users may even use this scenario functionality for their own, non-Graphics Wall, related purposes. An alias, starting up specific customer applications, is indeed perfectly possible.

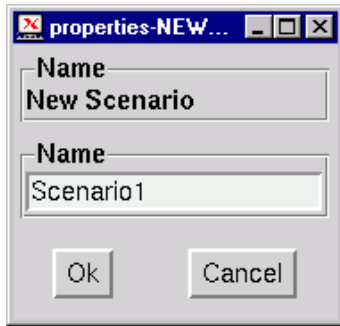
A scenario is built out of a number of triggers and possibly a number of sequence steps. The triggers drive the functionality of the Scenario.

Creation

Select the [Scenarios] Tab, right-click in the right window-area and select [New Scenario]



The resulting window will prompt you for the Scenario details:

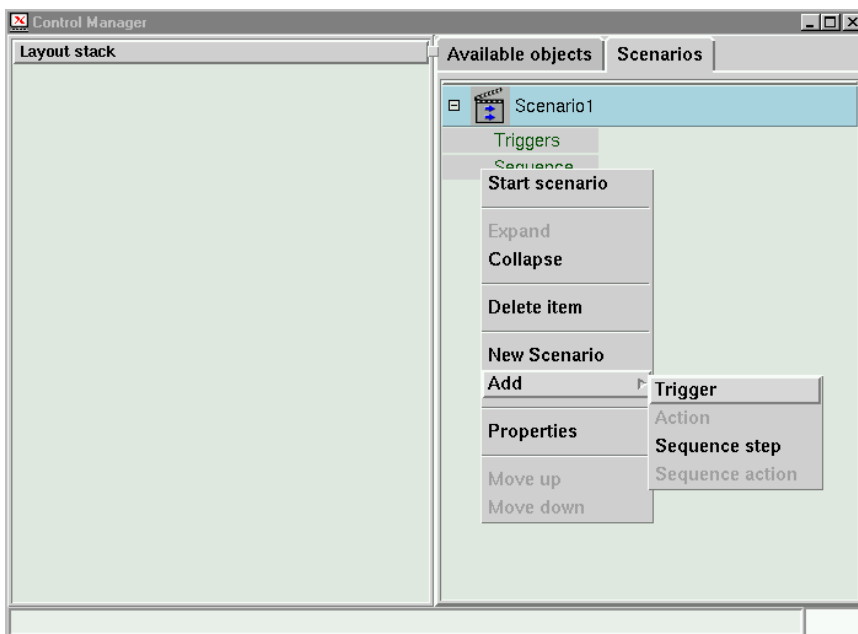


Complete the name of the scenario, using a unique object name and select [OK].

Trigger

Creation

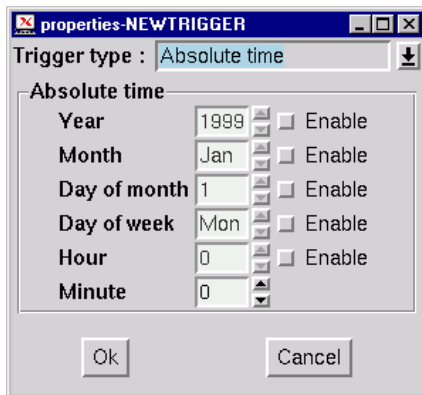
To create a new trigger, select the corresponding Scenario, right-click and select ,Add Trigger‘



The triggers are listed in order of creation. In order to group triggers, one can use the ,Move Up‘ and ,Move down‘ menu-options when a particular trigger, which you would like to move, is selected.

Types of trigger

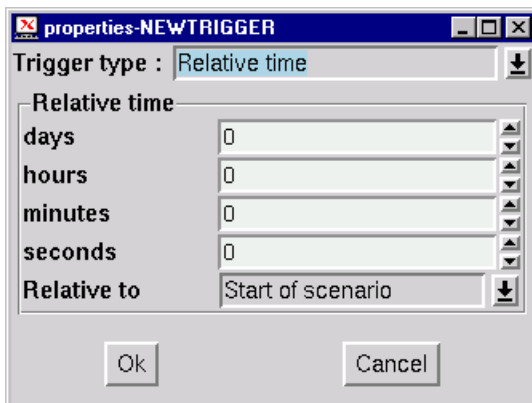
8.2.4...1 Absolute Trigger



An absolute trigger is triggered at a specific time. Checking one or more of the ,Enable‘ checkmarks will narrow down the time of the trigger.

For example, having a trigger activated every year, every 1st day of the month at 16h00, the ,Enable‘ checkmark of ,Day of month‘ must be checked.

8.2.4...2 Relative Trigger



Defines a trigger, which is to be triggered so many days, hours, minutes and seconds after a certain Event specified in the ,Relative to‘ field.

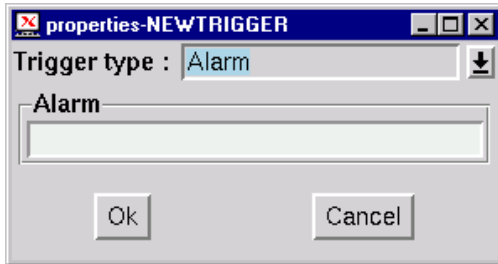
This event may be one of the following:

Event	Parameters	Description
Start of Scenario	None	Triggered the specified time after the scenario has been started
Trigger	Trigger number	Triggered when the trigger n has been triggered
Window Creation	Name of window	Triggered when the specified window has been created by some application
Window Destroy	Name of window	Triggered when the specified window has disappeared from the screen.
Projector Down	Projector address	Triggered when projector # went off
Projector Up	Projector address	Triggered when projector # restarted
UPS Down		Triggered when the UPS detects a power failure

UPS Up		Triggered when the UPS detects that the power has been restored
--------	--	---

8.2.4...3 Alarm

An alarm is a trigger-type which, when triggered externally, executes the actions defined for it. To raise and cancel an alarm condition, one will need to create 2 separate alarms, for ex. an alarm ‚AlarmOn‘ which executes appropriate actions and a second ‚AlarmOff‘ trigger which cancels the previous actions.



For an alarm, one only has to specify a unique trigger alarm-name.

To activate an Alarm trigger, the following methods can be used:

Method	Details
osi_alarm utility	<p>Syntax for starting this utility:</p> <pre>\$BVS_HOME/bin/osi_alarm -host <hostname> -name <AlarmName></pre> <p><hostname> will normally be ‚mvgs‘</p>
telnet session	<p>By opening a telnet-session to the mvgs on port 5701, one opens a connection to the Osiris command interface. After connection, Osiris commands may be entered, for example to trigger some alarm trigger:</p> <pre>alarm -name <AlarmName></pre>
Redirection to the gci_pipe	<p>Redirection of an Osiris command string to the file</p> <p>\$BVS_HOME/logging/gci_pipe is another possibility. For starting some Alarm Alarm1On, the following command could be used:</p> <pre>echo „alarm -name Alarm1On“ >\$BVS_HOME/logging/gci_pipe</pre> <p>This will of course only work when this gci_pipe file (pipe) is in place.</p>
RS232 command to serial port managed by gci object	<p>Connect a terminal device connected to one of the serial ports, which is connected to a GCI object in the configuration tool.</p> <p>Connection should be done using a crossed serial connection.</p>

8.2.4...4 Repeat Time

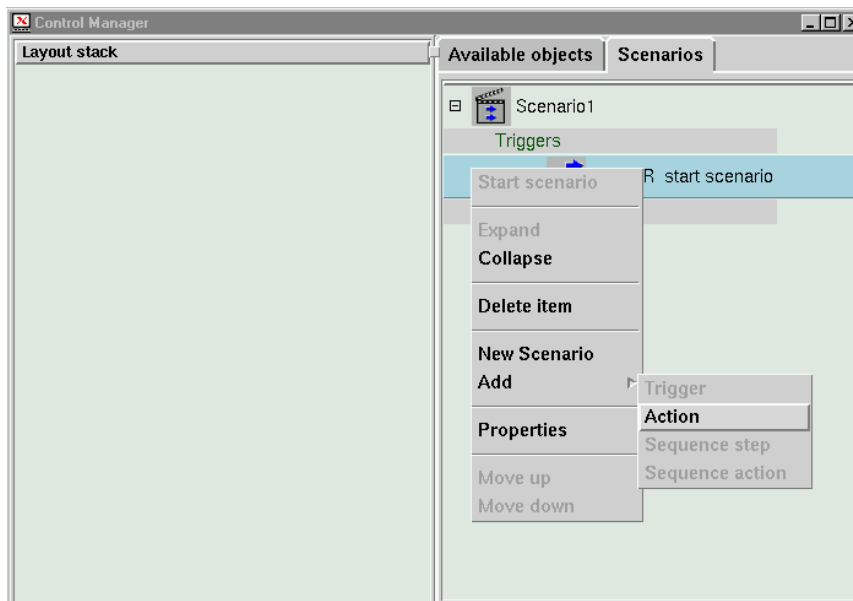
A 'Repeat time' trigger type will repeatedly execute a number of actions every now and then, determined by the specified number of days, hours, minutes and seconds.



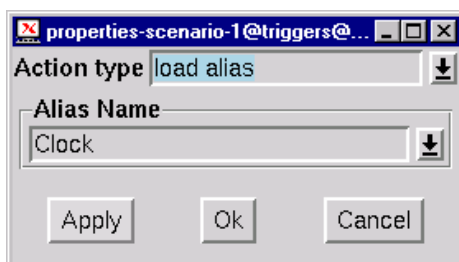
Trigger Action(s)

For each trigger defined, 1 or more actions may be specified to be executed upon the occurrence of that particular trigger.

To add an Action to some trigger, select the Trigger, right-click and select Add/Action.



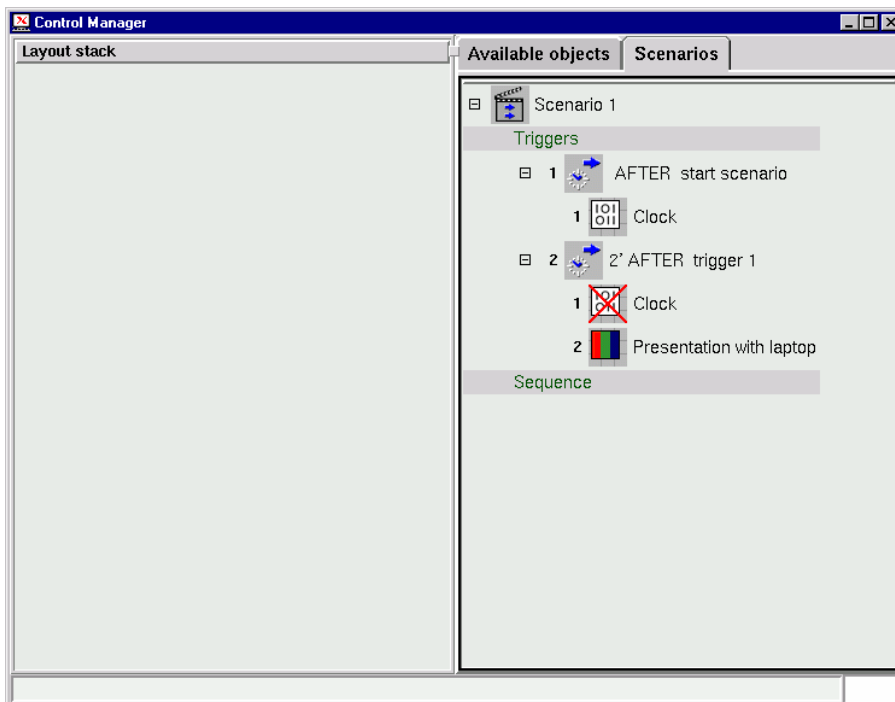
The next entry form will pop-up:



The following actions can be specified to start when the trigger is activated:

Action Type	Parameters
Load alias	Alias name
Unload Alias	Alias name
Load Layout	Layout name
Load VISU window	VISU window name
Unload Alias	Alias name
Unload Layout	Layout name
Unload VISU window	VISU window name
Execute Command	Command name + associated command parameters
Goto	Number of particular Sequence step
Next	Go to the next sequence step. If the pointer is at the last sequence step, the first sequence step is executed again
Disable Trigger	Number of the trigger to disable. It will from this point onwards not be possible to trigger the trigger until it gets re-enabled.
Enable Trigger	Number of the trigger to enable.

The actions defined are executed in the order they are created. Each action is numbered. Using the ‚Move Up‘ and ‚Move Down‘ pop-up menu options, one can change the order in which these actions are started.



Sequence

Description

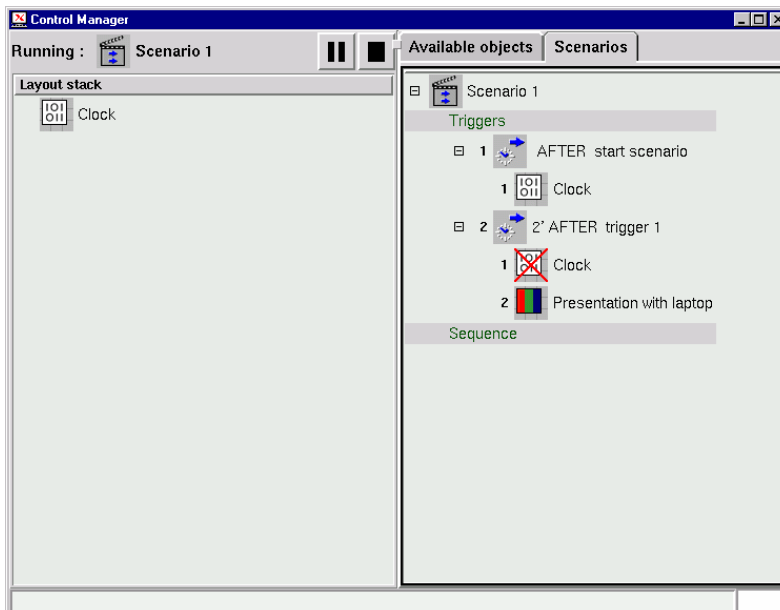
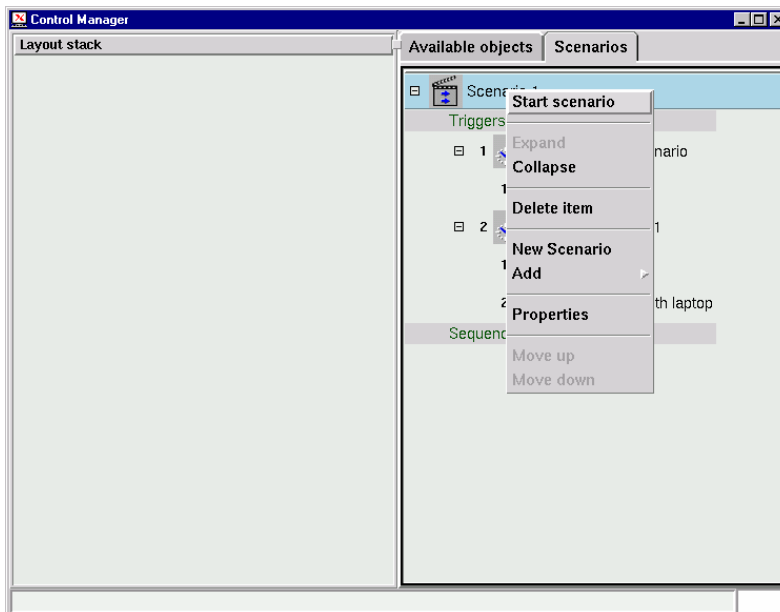
A sequence defines a number of actions that are to be executed together in one step. These actions will be executed within that one sequence-step.

Sequence Action(s)

A sequence action is comparable to a trigger action. (cfr. Section 191.3985.431)

Starting/Stopping a Scenario

A scenario is started by selecting it, right-click, select 'Start Scenario':



To Pause or Stop the scenario, use the appropriate buttons at the left side of the 'Available objects' Tab.



Only 1 scenario can be started at any one time.

8.3 Layout Stack

8.3.1 Organizing

Comparable to the layout-placement as discussed in section 219.4013.403, the Active Layout Stack may be organized in the same way.

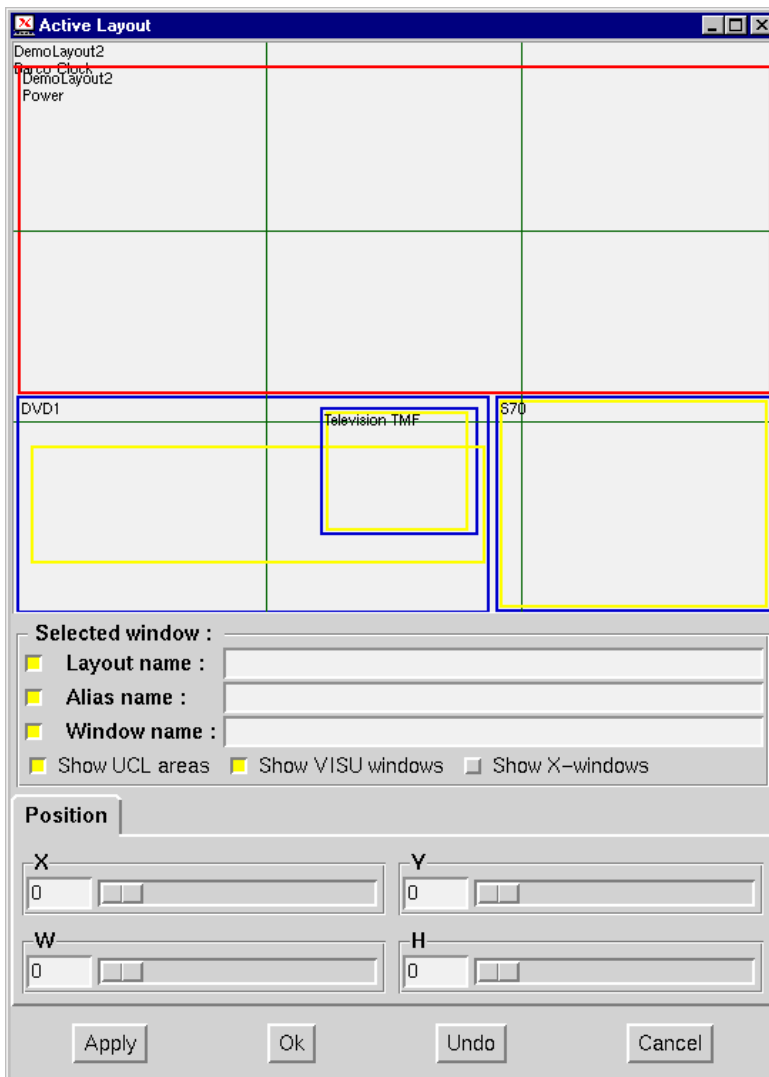
To access the Active Layout:

- click the 'Active Layout' button in the Control Panel

Or

- right-click in the 'Control Manager' left side and select 'Edit Active Layout'

A window pops up as in the following example:



An area can be selected by

- pressing the left mouse button on the label displayed in the area.
- pressing the left mouse button on the border of the area.
- using the drop-down list in the selected window pane.

When a VISU object has been selected, one can

- Freeze the contents being displayed or release a frozen contents – select a visu area and press the button Freeze/Release on the Position pane. This button is only visible after selection of a VISU object
- Temporarily hide the window or show a temporarily hidden window – select a visu area and press the button Hide/Show on the Position pane. This button is only visible after selection of a VISU object
- modify the Viewport for the displayed information within the VISU window. A viewport is a definition of the portion of the information, which is being digitized. This allows you to zoom in on the image. The Viewport tab is only visible after selection of a VISU object.

To operate on some area, one need to select the area first by left clicking on the are-name or one of the sides of the area. When the title changed color from black to blue, the area is selected.

Right clicking and holding while you move the area to the new location does moving an area. Left clicking on one of the corners or sides and dragging to modify the aspect ratio does resizing.

To confirm the changes, use either the Apply or the OK button.

Use the Undo button to go back to the status of the last confirmed change (which may be executed by another operator).

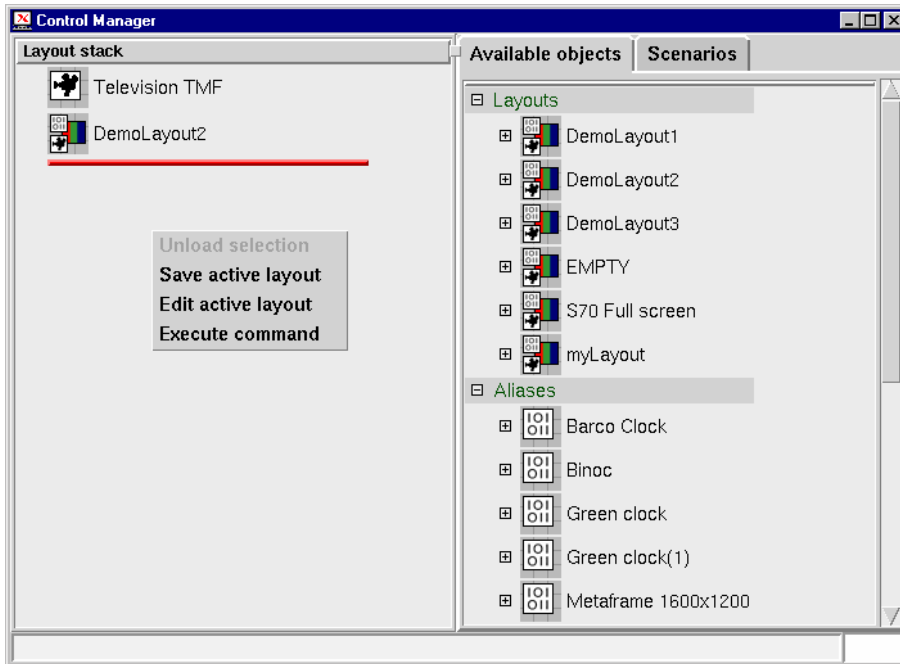
Press the Cancel button to leave the active layout menu without confirming the changes done.

8.3.2 Unloading objects

To remove some object from the Layout Stack, you need to select the items in the left-hand area of the 'Control Manager' window, right-click and select 'Unload'.

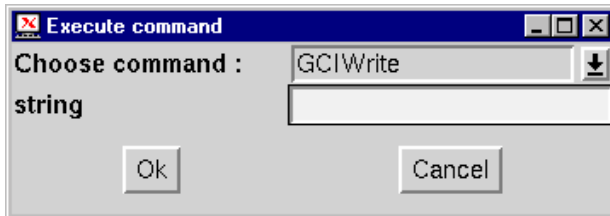
8.3.3 Saving Active Layout

To save a layout that was prepared in the Active Layout, you right-click



8.3.4 Starting Osiris commands

Selecting the 'Execute Command' from the list-box may start an Osiris command. A list of available commands will be presented. After selecting one of the commands, it may be that one or more additional parameters are requested depending on the definition of the command:



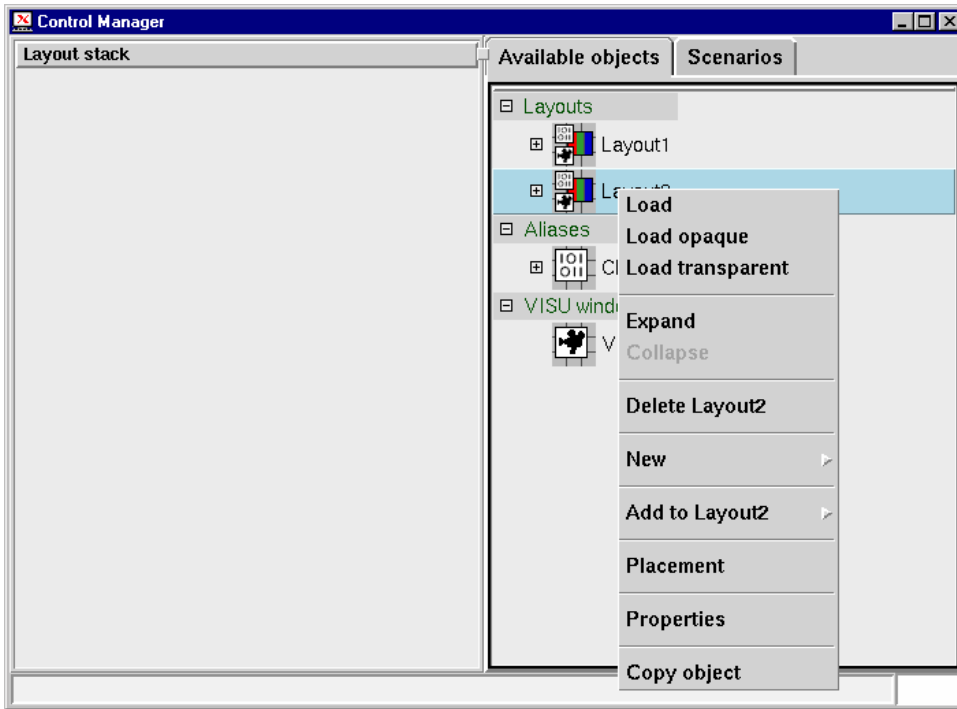
8.4 Copying objects

The following objects may be copied:

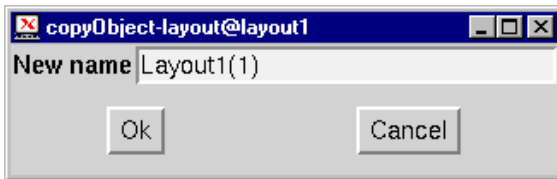
Aliases

Visu windows

Layouts



To copy such an object, select it with the left-mouse button, right-click and select Copy Object. The following entry will appear:



Change the name and select [OK]

9 Projector Control

9.1 Supported range of projectors

The following range of projectors are supported by Osiris:

Projector Type	Osiris module	Technology
BD2100	lcd	LCD
BG2100	lcd	LCD
70x	crt	CRT
80x	crt	CRT
120x	crt	CRT
Overview mL	oml	LCD
Overview DMD	dmd	DLP Double Chip
Overview DME	sdmd	DLP Single Chip
Overview mP	omp	Polysilicon

All mentioned projectors are Barco supplied. Other display devices with serial control capabilities are not supported.

9.2 Connection

Projectors are interconnected using straight RS232 cables (D9 Male to D9 Female).



With reference to the table in previous section 9.1, all projectors of the ,Overview' range should additionally have a terminator connected to the last projector in the chain.

The way (order) in which the projectors are interconnected can be specified in Osiris upon configuring the system (Wall object definition, cfr. Section 232.4026.390 for more details on this).

By defining the proper order in which the looping through is done, a failure of one of the projectors may be bypassed intelligently when possible by Osiris.

9.3 Projector Commands

In the Osiris command database, a range of standard projector commands is present. This includes command for starting/stopping projectors, switching sources (where possible), etc...

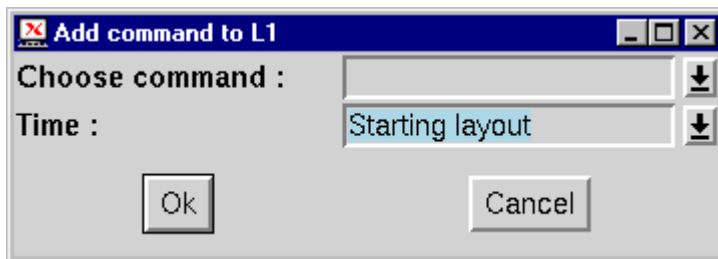
These commands can be invoked within layouts and scenarios (cfr. Next section)

9.4 Integration of commands within layouts/scenarios

9.4.1 Layout

Defined commands can be integrated within a layout definition by selecting the Layout in the Available Objects section, right-click and select **Add to [LayoutName]** and select **Command**.

This results in the following entry to be displayed:



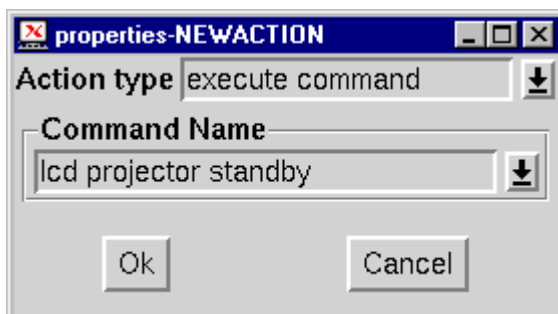
The first entry defines the command to execute at the specified time. The pull-down menu will present all commands defined in the [Edit the hardware configuration] section.

The time when the command should be executed is defined in the second entry. The following time-events are available:

Time	Purpose
Starting layout	Will be invoked when the layout is loaded. The commands are executed in the order of appearance under the Layout definition
Stopping layout	Will be invoked when the layout is unloaded.
Hiding	Will be invoked upon the hiding of the layout. In practice this means that when another item is loaded opaque on top of this layout, this command will be executed.
Showing	Will be invoked when the layout is unhidden so when an overlapping (opaque) object is unloaded, rendering the layout visible again

9.4.2 Scenario

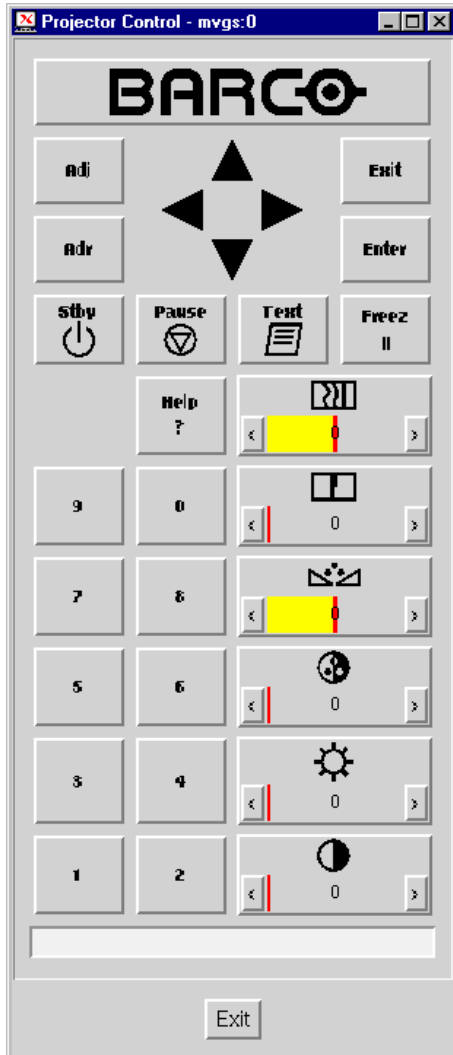
Likewise as within the Layout integration, one can integrate Osiris commands within a scenario. Within a trigger or Sequence step, a possible action could be the execution of a command. E.g.:



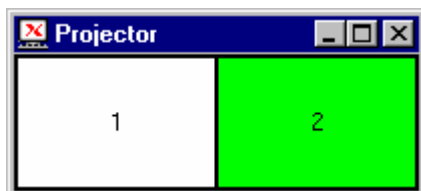
9.5 Projector Control utility (GUI)

For the different types of projectors used, a specific GUI is available in order to control some settings specific to that type of projector. In the next sub-sections, the different GUI's are displayed.

9.5.1 Barco CRT range of projectors



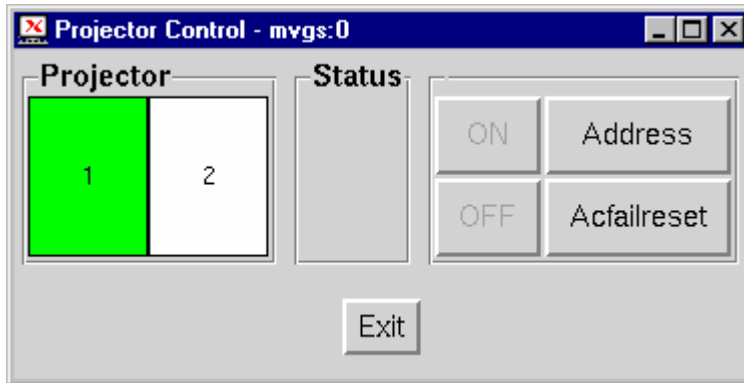
In order to tune the settings of one of the projectors in the matrix, select the projector using the Address button [Adr] and select the projector:



9.5.2 Barco LCD range of projectors

With the LCD range of projectors the same user-interface is displayed.

9.5.3 Overview mL (SVGA or XGA)

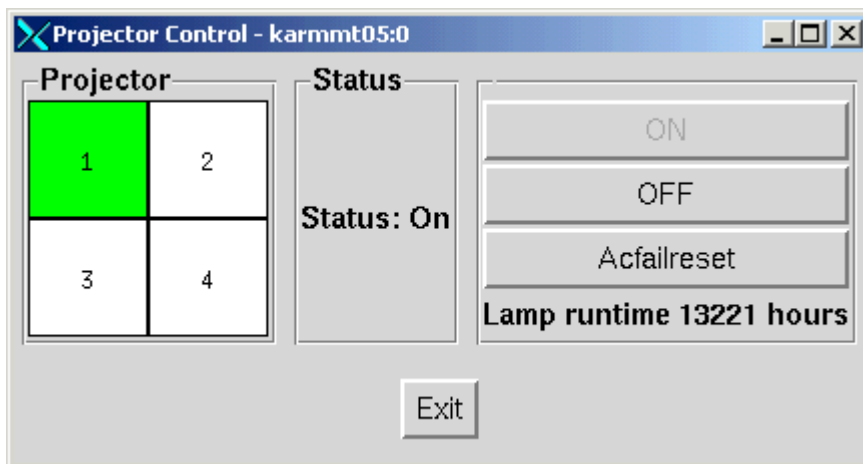


After selecting a projector within the matrix of projectors, the following options are at your disposal:

Item	Description
ON	Turn specific projector ON
OFF	Turn specific projector OFF
Address	Re-address the chain of projectors (1)
Acfailreset	Reset a specific projector (2)

- (1) When some projector is turned off for some reason, it will be necessary to, when projector is on line again, press this button before all projectors can be controlled again!
- (2) This might be needed when one experiences difficulties to fire the engine up again.

9.5.4 Overview mP

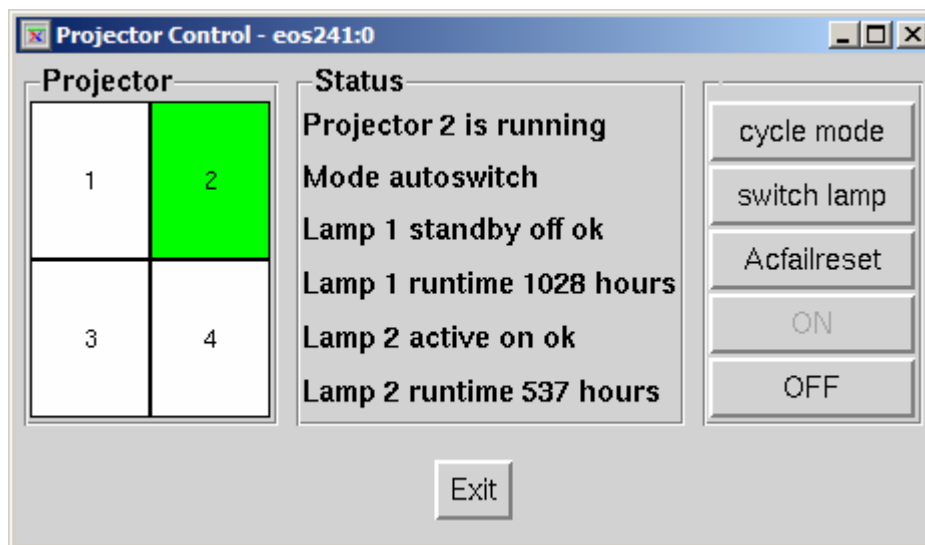


Item	Description
ON	Turn selected projector ON
OFF	Turn selected projector OFF

Address	Re-address the chain of projectors (does not appear when SNMP Support is selected)
Acfailreset	Reset selected projector
Lamp runtime	Runtime off the lamp

As with the mL line of projectors, the same options are valid.

9.5.5 Overview DG

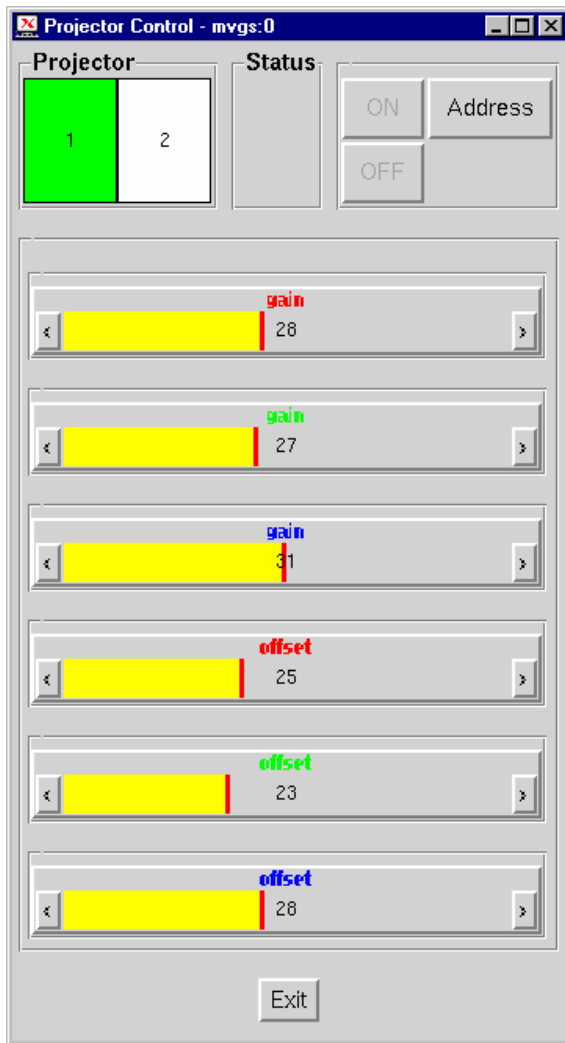


Item	Description
Cycle mode	Switch between different modes supported and enable by the projector. Modes are autoswitch – coldstandby - hotstandby
Switch lamp	Switches between the 2 lamps in the projector.
Acfailreset	Reset selected projector
ON	Turn selected projector ON
OFF	Turn selected projector OFF

The status field provides the operator with the following information:

1. The status of the projector: running, standby, unknown
2. The lamp mode for the projector: this can be one of autoswitch, coldstandby, hotstandby (if option was purchased with the projector).
3. The status of each of the two lamps in the projector, consisting of
 - a. Active/standby/down – is the lamp expected to be active or not.
 - b. On/off/unknown/cooling/starting– the actual status of the lamp.
 - c. Ok/Failed – the lamp condition

9.5.6 Overview DME



For each projector the following settings may be altered:

Red, Green and Blue Gains

Red, Green and Blue Offsets

The address button is used again for the automatic re-addressing of projectors.

10 Zoom-X

10.1 Overview

The ZoomX programs are automatically installed during Osiris installation and is available as a button in the control panel. As such each operator gets his own instance of the zoomx application. The zoomx application is a licensed application. A demo license may be provided on request. See licensing.

10.1.1 Functional description

Sometimes an operator is too far away from the wall display in a control room to read all the screen details. The zoomx application gives the ability to an operator to tear out an arbitrary part of a specified wall display and show the chosen part on the screen in front of him. Consequently, he now can read all necessary details he's interested in except for video-data from VISU units.

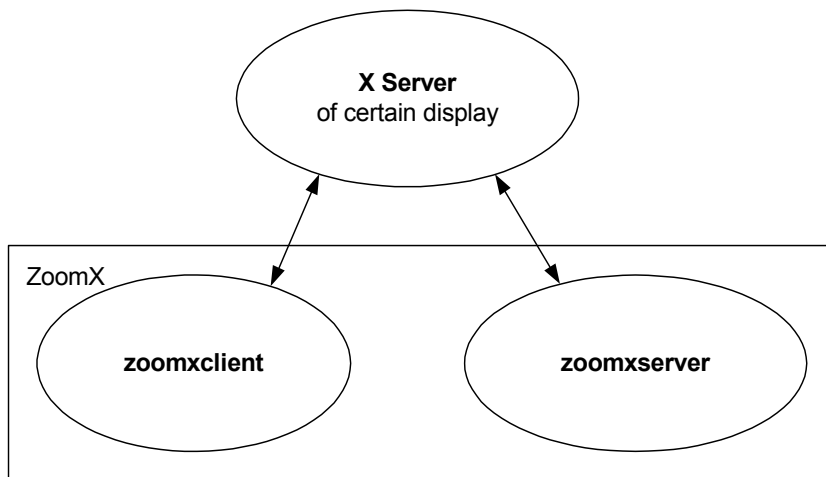
Zoomx is started up from within Osiris by a licensed module in the same way as Osiris starts the XServer.

10.1.2 Architecture

For reasons of flexibility, zoomx is implemented as a client/server application. It runs with the help of three basic parts:

- A zoomxserver application
- A zoomxclient application
- An XServer running on a certain display (XServer)

To see the functionality of zoomx, at least one XServer, one zoomxclient and a zoomxserver must



be running.

Figure 3.1 – Basic parts of the zoomx application.

It is possible to start the zoomxserver and the zoomxclient separately with their own customisable command line options. Both zoomxserver and zoomxclient connect as client application to the XServer of a display specified by the user.

The operator can stop the zoomxserver and the zoomxclient independently. If the XServer stops running, then both the zoomxserver(s) and/or zoomxclient(s) using the XServer are stopped.

Osiris starts a zoomx session using the \$BVS_HOME/bin/zoomx script which can be further customised.

10.1.3 Task of each module

The zoomxserver and the zoomxclient of zoomx together with an XServer must cooperate to let the zoomx application function properly. Each part has its own task in the zoomx application.

The Zoomxserver tears out a user specified part of a wall display and show it in an X window on a display of his choice. The selected view is updated according to the user actions in the zoomx GUI.

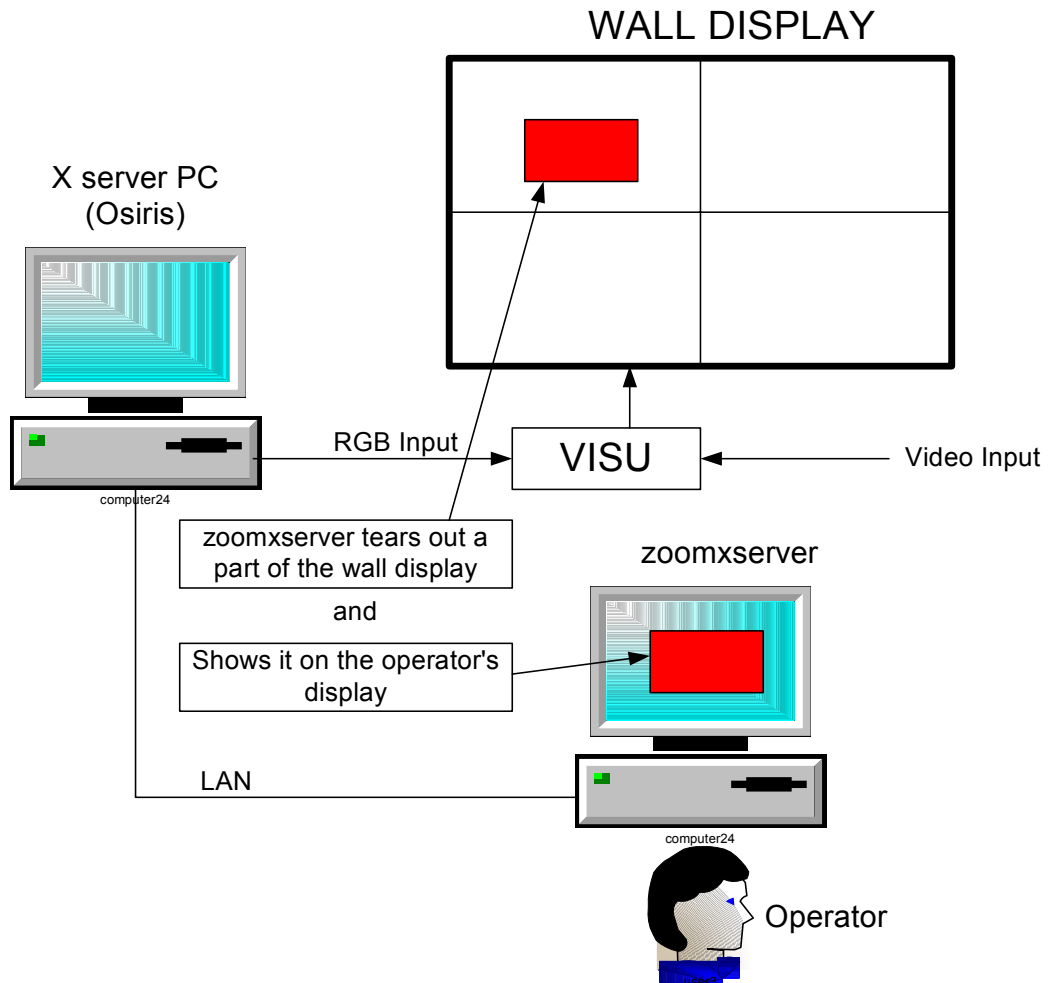


Figure 4.4 – The zoomx server tears out a part of the wall display and shows it on the operator's display.

The Zoomxclient starts the zoomx GUI where the operator can select or do some user actions.

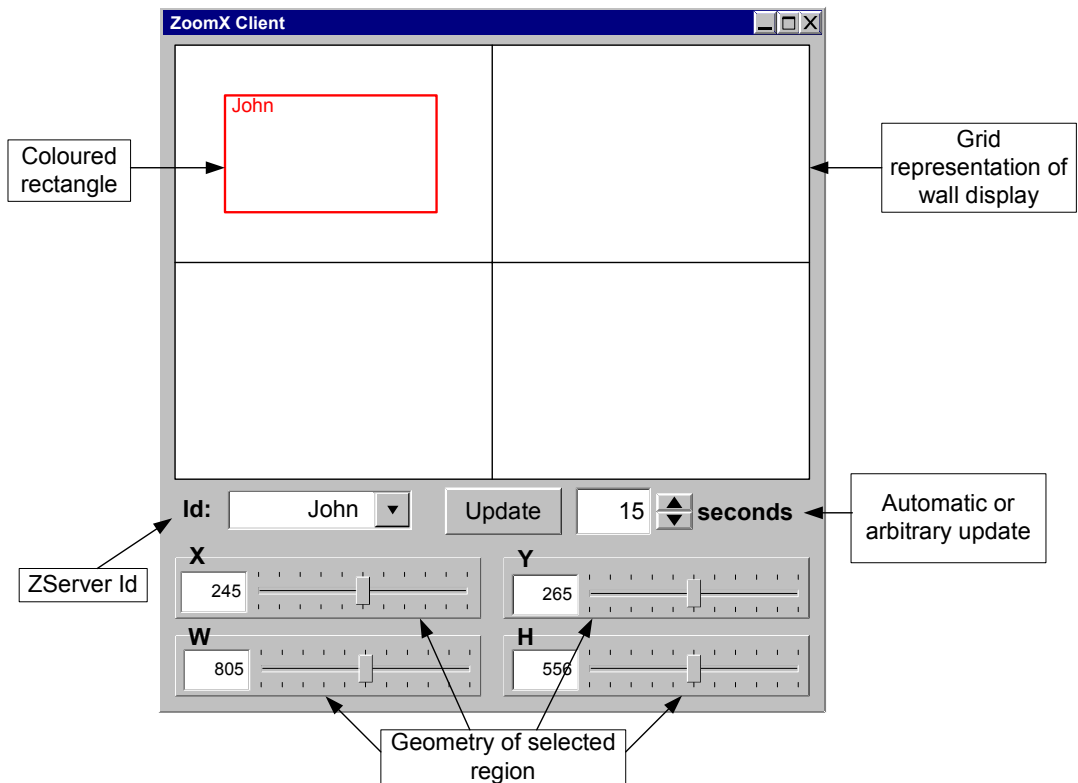


Figure 4.5 – The zoomx GUI.

The XServer of a display functions as the communication channel between the zoomxserver(s) and the zoomxclients. Remark that the XServer is not started by zoomx. It's an application already running on a computer where the zoomxserver and the zoomxclient connect.

10.2 Configuration Options

The Zoomxserver and the Zoomxclient are started by specifying options on the command line. All possible options for Zoomxserver and Zoomxclient are here explained in detail. Some options are mandatory while others are optional.

Sometimes it is possible to specify an option in several ways which is indicated by the '|' (OR) sign. In this case, the user can only specify one of them otherwise an error message is generated.

10.2.1 Zoomxserver options

Mandatory options

-id <Id>

This option specifies the Id of the Zoomxserver. The Id of the Zoomxserver must be unique otherwise an error is generated.

Optional options

`-wall <host:dpy> | -w <host:dpy> | -trusted`

This argument allows you to specify the wall display (i.e. the XServer) to connect to, see *X (1)*.

The “trusted” option is used in conjunction with Osiris. If UCL is enabled, this option is needed to be able to display the selected view on the wall display.

`-display <host:dpy>`

This argument allows you to specify the display where the selected view must be shown. It can be any XServer display.

`-click | -region <reg> | -root | -windowid <wid> | -name <name>`

These options specify what to take as selected view.

The “click” option indicates that the operator will select a window by clicking the pointer in the desired window.

For the rest of the options, specified above, the view is selected without requiring the user to select a window with the pointer.

The “region” option indicates that the specified region should be taken from the root window as selected view.

The “root” option indicates that the root window should be taken as selected view.

The “windowid” option indicates that the window with the specified resource id should be selected for the window dump.

The “name” option indicates that the window with the specified WM_NAME property should be taken as selected view.

`-autoupdate <secs>`

This argument specifies the amount of seconds for automatic update of the selected view. If not specified, the view is not updated unless the user changes the autoupdate value in the zoomx GUI.

`-help | -?`

Print out of the “Usage:” command syntax summary. A short description of the allowable options is given.

`-nobdrs`

This argument specifies that the window dump should not include the pixels that compose the X window border. This is useful in situations where you may wish to include the window contents in the Zoomxserver window.

`-xy`

This option applies to colour displays only. It selects ‘XY’ format dumping instead of the default ‘Z’ format.

`-screen`

This option indicates that the GetImage request used to obtain the image should be done on the root window, rather than directly on the specified window. In this way, you can obtain pieces of other windows that overlap the specified window, and more importantly, you can capture menus or other popups that independent windows but appear over the specified window.

-icmap

Normally the colour map of the chosen window is used to obtain RGB values. This option forces the first installed colour map of the screen to be used instead.

-add <value>

This option specifies a signed value to be added to every pixel.

-frame

This option indicates that the window manager frame should be included when manually selecting a window.

-fg <fgcolor>

If a bitmap image (or a single plane of an image) is displayed, this option can be used to specify the colour to display for the “1” bits in the image.

-bg <bgcolor>

If a bitmap image (or a single plane of an image) is displayed, this option can be used to specify the colour to display for the “0” bits in the image.

-geometry <widthxheight+x+y>

This option allows you to specify the size and position of the window. Typically you will only want to specify the position, and let the size default to the actual size of the image.

-inverse | -rv

If a bitmap image (or a single plane of an image) is displayed, this option forces the foreground and background colours to be swapped. This may be needed when displaying a bitmap image which has the colour sense of pixel values “0” and “1” reversed from what they are on your display.

-new

This option forces creation of a new colour map or displaying the image. If the image characteristics happen to match those of the display, this can get the image on the screen faster, but at the cost of using a new colour map (which on most displays will cause other windows to go Technicolor).

-noclick

Clicking any button in the window with the selected view will terminate the Zoomxserver, unless this option is specified. Termination can always be achieved by typing ‘q’, ‘Q’ or ctrl-c.

-plane <number>

You can select a single bit plane of the image to display with this option. Planes are numbered with zero being the least significant bit. This option can be used to figure out which plane to pass to *xpr(1)* for printing.

-raw

This option forces the image to be displayed with whatever colour values happen to currently exist on the screen. This option is mostly useful when undumping an image back onto the same screen

that the image originally came from, while the original windows are still on the screen, and results in getting the image on the screen faster.

-std <maptype>

This option causes the image to be displayed using the specified Standard Colourmap. The property name is obtained by converting the type to upper case, prepending “RGB_”, and appending “_MAP”. Typical types are “best”, “default”, and “gray”. See *xstdcmap(1)* for one way of creating Standard Colourmaps.

-vis <vis-type-or-id>

This option allows you to specify a particular visual or visual class. The default is to pick the “best” one. A particular class can be specified: “StaticGray”, “GrayScale”, “StaticColor”, “PseudoColor”, “DirectColor”, or “TrueColor”. Or “Match” can be specified, meaning use the same class as the source image. Alternatively, an exact visual id (specific to the server) can be specified, either as a hexadecimal number (prefixed by “0x”) or as a decimal number. Finally, “default” can be specified, meaning to use the same class as the colour map of the root window. Case is not significant in any of these strings.

-version

Displays the version of the zoomx software.

-debug

Option used for debugging purposes.

10.2.2 Zoomxclient options

Mandatory options

None.

Optional options

-id <Id>

This option specifies the Id of the Zoomxserver to connect to. If the Id is not specified at start up, an Id of an active Zoomxserver can be selected afterwards in the zoomx GUI.

-wall <host:dpy> | -w <host:dpy> | -trusted

This argument allows you to specify the wall display (i.e. the XServer) to connect to, see *X(1)*.

The “trusted” option is used in conjunction with Osiris. If UCL is enabled, this option is needed to be able to display the selected view on the wall display.

-display <host:dpy>

This option specifies the display where the zoomx GUI must be displayed.

-region <reg>

This option indicates the specified region to be taken from the root window as selected view. This value overwrites the region value specified by the Zoomxserver, so Zoomxclient is always master in defining the region.

-autoupdate <secs>

This argument specifies the amount of seconds for automatic update of the selected view. If not specified, the view is not updated unless the user changes the autoupdate value in the zoomx GUI. Also here, the autoupdate value possibly specified by the Zoomxserver, will be overwritten by the autoupdate value specified by the Zoomxclient.

-help | -?

Print out of the “Usage:” command syntax summary. A short description of the allowable options is given.

-version

Displays the version of the zoomx software.

-debug

Option used for debugging purposes.

-timeval <secs>

This option specifies the time interval to check from Zoomxclient side if the Zoomxserver is still alive.

10.2.3 ZoomX configuration for Multiple Logical Screens

When using multiple logical screens the default configured Zoomxclient will only show the first logical screen. To view other logical screens additional buttons should be configured in the control panel.

The following example shows how the properties of the ZoomX buttons in the \$BVS_HOME/config/gui.cfg file can be set for 2 logical screens.

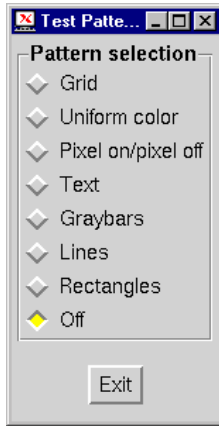
```
COMMAND {all} “zoomx screen :0” “ofc” “$BVS_HOME/bin/zoomx mvgs $USERNAME 0”  
image {gui_zoom} balloon {ZOOMX}}  
COMMAND {all} “zoomx screen :1” “ofc” “$BVS_HOME/bin/zoomx mvgs $USERNAME 1”  
image {gui_zoom} balloon {ZOOMX}}
```

Further customization in terms of look and feel is possible as described in section [Customizing the Control Panel](#)

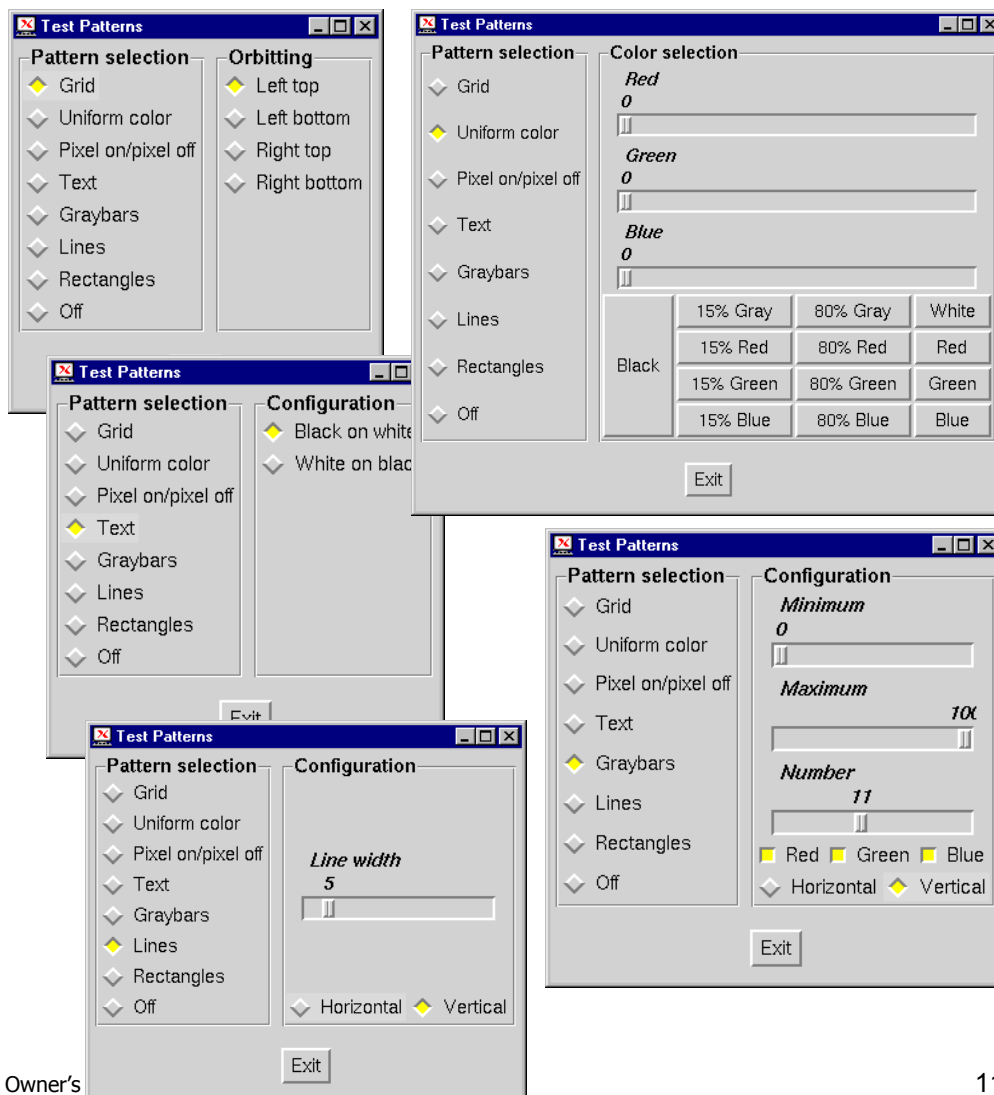
11 Test patterns

11.1 Range of patterns

In order to assist the installation and maintenance of the display devices installed, a range of alignment patterns is available. When activating the appropriate button in the default control panel (4th from the left), the following selection box will pop-up:



Each selection displays a different kind of alignment pattern on the Graphics Wall:



11.2 Application

The different subsets of the Alignment patterns are used for different tasks within the alignment process and are only of use for servicing the wall.

12 Remote Pointer (Octopus)

12.1 Functionality

The remote pointer functionality has been developed to allow users to interact with the Graphics Wall screen, as they would do on their local workstation's environment.

After the startup of the remote pointer, the user can go back and forth from local environment to remote environment (Graphics Wall) using a hotkey on the keyboard.

All keyboard and mouse events are, when remote pointer is activated, passed from the local environment to the remote X-windows environment.

On the remote side (Graphics Wall), a remote pointer server is running, referred to as 'Octopus Server'. The client side can either be the Octopus Client running on a native NT4.0 or Windows 2000 environment or a remote pointer client running on the EOS/X platform, displaying on the client machine being a genuine X-windows platform or a Windows platform running some form of X-windows emulator.

The Octopus Packages are outside the scope of this manual. Please refer to the documentation related to this SW product.



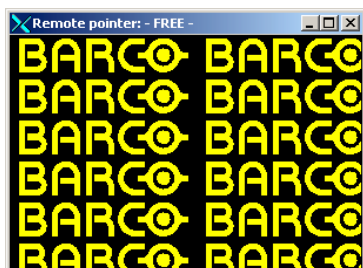
It is only from the Octopus Packages V2.0 onwards that the remote pointer functionality from a genuine Windows environment is allowed to connect to the Octopus server running within the Osiris environment.

12.2 Startup and usage of the remote pointer client

The normal way of starting the remote pointer is by clicking on the Remote pointer button on the control panel:

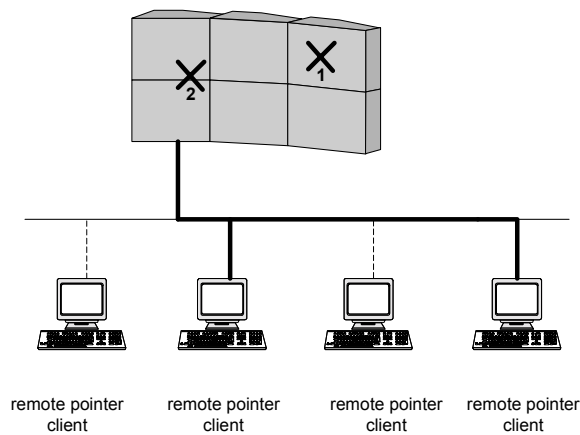


This will bring up the following window:



When you click into the Remote Pointer window, the mouse and keyboard is warped onto the screen. In order to return to the local screen, one has to define a hotkey. A possible hotkey that is used by hardly any application is F12. After the hotkey is linked, it will become possible to toggle between local and remote screen using that assigned hotkey.

12.3 multi-user-mode



Per default multiple clients can connect to wall at the same time. Each user will get it's own mouse cursor. However when multiple operators work to the wall at the same time they might disturb each other as soon as they try to use keyboard on different applications at the same time, because per default all keyboard inputs are sent to the window that currently has the focus.

To overcome this problem the `-focus` option can be used. With this option enabled keyboard inputs will always be sent to the window below the operators mouse pointer (in fact a click event is sent before each key-board event). Thus two operators can interact with two different applications at the same time. However this kind of control requires a certain degree of discipline between the operators because there still is disturbance:

- During operations which depends on constantly having the focus e.g. dragging or resizing a window or opening a context menu, mouse and/or keyboard inputs of all other operators are blocked. While an operator A is moving a window all mouse events of an operator B are blocked and all keyboard events of operator B are buffered. They will be send to the window as soon as operator A releases finishes his move-operation.
- Further the operator is always forced to position his mouse-pointer correctly, over the window he wants to send key-board events to. This could be especially annoying e.g. in dialogs with little edit boxes which have their own windows. Depending on the application this might become quite annoying.

How to enable the focus option

Config file:

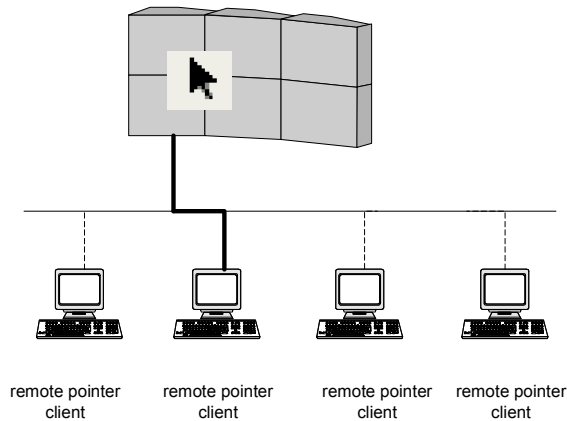
```
$BVS_HOME/config/osi_daemon.cfg
```

Change entry:

```
from: karmmt05:remptr:remptr:0:respawn:/home/bvs/bin/osi_startremptrsrv
```

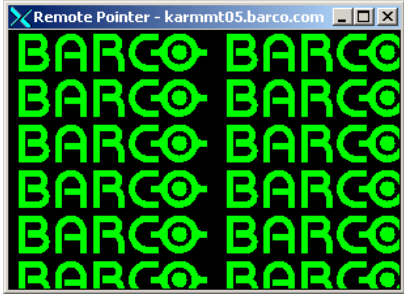
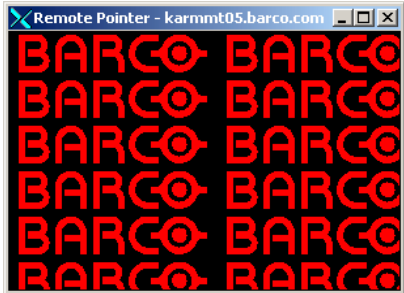
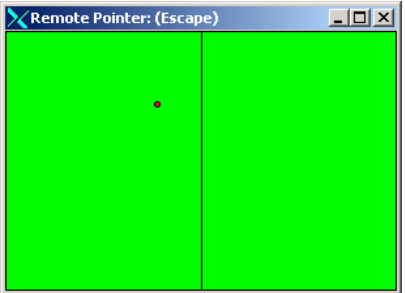
```
to: karmmt05:remptr:remptr:1:respawn:/home/bvs/bin/osi_startremptrsrv -focus
```

12.4 single-user-mode



To avoid the disturbance that might be caused in multi-user mode the remote pointer can be switched to single user mode. In this mode only one user can control the wall at a time. Other users need to wait until the active user explicitly releases control or until he gets idle. The remote pointer can be in one of the following 'Activation-States':

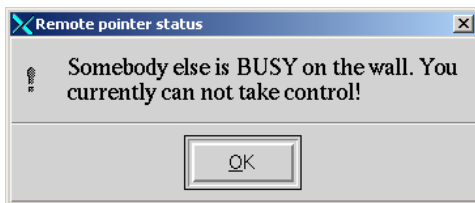
Activation States

Activation State	Remote Pointer Window of all inactive users	Remote Pointer Window of the active user
<p>FREE</p> <p>Nobody has activated the remote pointer on the wall. All remote pointer users will see a green Barco Logo. The next user that tries to activate will get control immediately.</p>	 <p>green logo</p>	<p>nobody is active</p>
<p>BUSY</p> <p>Somebody has activated the remote pointer and is currently busy on the wall. Nobody else is able to take control while he is busy and the wall. The wall is blocked for all other users. Users that are interested in taking control will be informed when the wall is available (idle or free) again.</p>	 <p>red logo</p>	 <p>green background</p>

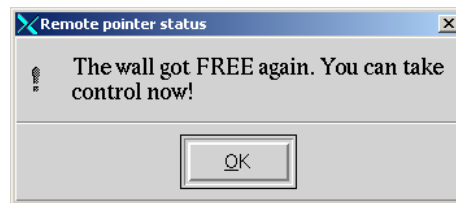
<p>IDLE</p> <p>Somebody has activated the remote pointer, but has been idle (no mouse or keyboard interaction) for a configured period of time (default 15sec). The next user that tries to activate the remote pointer will kick out the currently active user and will get control.</p>	 <p>yellow logo</p>	 <p>yellow background</p>
--	--	--

Notifications

If a user tries to activate the remote pointer while another user is busy on the wall, he'll get the message:



A user that tried to activate while somebody else was busy, will automatically be informed about all further changes of the activation state of the remote pointer until he got control over the wall himself (or restarted the octopus client).



Default cursor usage modes

When selecting cursor type 'remote' during Osiris configuration (see section 5.1.3 - **General Configuration Settings**) the cursor usage mode will be set by default to multi user mode. When selecting cursor type 'local' the cursor usage mode is automatically set to single user mode.

How to override default cursor usage modes

To switch between multi user and single user mode you need to modify the Osiris configuration manually. Per default the remote-pointer server starts in multi-user or single user mode dependant on the cursor type requested (remote or local). The switch

`-singleuser`

turns it into single-user mode. In single user mode the option

`-idletimeout <seconds>`

allows to define the period of time the active user must be idle before he is considered to be idle (default is 15 seconds).

Config file:

```
$BVS_HOME/config/osi_daemon.cfg
```

Change entry:

```
from: karmmt05:remptr:remptr:0:respawn:/home/bvs/bin/osi_startremptrsrv
```

```
to: karmmt05:remptr:remptr:1:respawn:/home/bvs/bin/osi_startremptrsrv singleuser  
-idletimeout 10
```

12.5 Special options for the remote pointer Client

To start an individual remote pointer client, the following command is to be used:

```
$BVS_HOME/bin/osi_startmodule osi_remptrclient [-options]
```

The following environment settings are necessary:

```
TCL_LIBRARY= $(BVS_HOME)/lib/tcl
```

```
TIK_LIBRARY= $(BVS_HOME)/lib/tix
```

```
TK_LIBRARY= $(BVS_HOME)/lib/tk
```

```
DISPLAY=clientworkstation:0.0
```

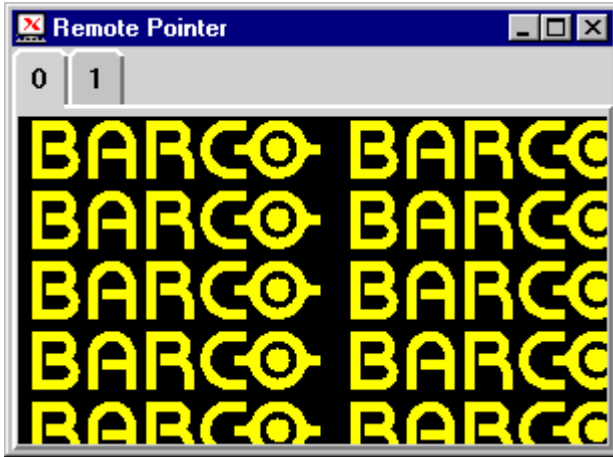
Options for the remote pointer client:

Option	Purpose	Default
display	display where the client user interface should appear	\$DISPLAY
user	Defines the user who starts up the remote pointer. This defines the default shape of the cursor The cursor size/shape may be related to the account with which one logs in to the Osiris environment: - bitmaps and mask files for the different possible pointers are located in the directory \$BVS_HOME/pkgs/xmvg/octopus/resources the maximum size of the remote pointer is set up in the Osiris configuration dialog. The <code>-keeppointer</code> option of the XServer assures that the pointer bitmap is not changed by the application.	ofc
activate	When used, the remote pointer will be activated when the remote pointer client is started.	0
hotkey	Predefines the hotkey.	<none>
squid	This option activates the „squid“ functionality, meaning that the pointer and KB will be warped to the Graphics Wall when moving to the top of the screen. The mouse will go off-screen and appears on the Graphics Wall screen.	0
bitmap	Defines the bitmap to be used as background for the local remote pointer window.	gui_barcologo
host	Defines the host-name of the system where the Octopus server is running to which one wants to connect.	mvgs
geom	Defines the geometry of the local window	300x200+100+

		100
port	Defines the tcp-port via which the communication will take place between Octopus server and clients.	6060
onscreen*	Specifies the logical screen to connect to. (only when connecting to X-server split up in multiple logical screens)	0
inactiveforeground	Foreground of bitmap in local control window in inactive mode.	yellow
inactivebackground	Background of bitmap in local local control window in inactive mode.	black
activeforeground	Foreground of pointer in local control window in active mode.	red
activebackground	Background of bitmap in local control window in active mode.	white
key	Code that has to match the access code of an alias to allow manipulation of the aliases windows by the operator using the remote pointer	<none>

12.6 Logical screens

When the X-server for IMSGP cards is configured for multiple logical screens, the logical screen to interact with, using the remote pointer, can be selected using a Tab on top of the remote pointer window:



Select the Tab for the logical screen to interact with and then click into the Remote Pointer window.

13 User Administration

Osiris is a multi-user environment. Different users may be created each having their own Control Panel look-and-feel.

13.1 General info

The users created and managed under Osiris are not related to the Unix/Linux users. This is a completely separate set of users.

The user-administration for all users is by default located in the Administration control panel, 2nd button from the left.

When logged in as the default ,ofc‘ user, the user is allowed to modify his own password.

All information on users and passwords is stored within the Osiris database.

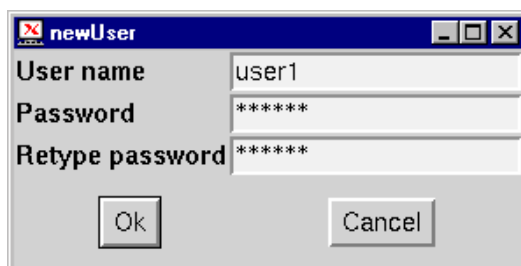
13.2 Creation of new users

When selecting the User Admin button, as Administrator, the following window appears:



In order to be able to modify/create or delete user accounts, one always has to specify the old password of the ,admin‘ user.

Then, select ,New user‘ which opens the following window:



Complete the details and select OK. New user is created.

13.3 Deleting existing users

As with the creation of a new user, the deletion of an existing user requires you to specify the password of the admin user prior to deleting another existing user.



13.4 Modifying Passwords

As with the creation of a new user, the modification of the password requires you to specify the password of the admin user prior to changing the password.



14 Device Control

14.1 Functionality

The device control functionality has been implemented in Osiris to allow the control of external devices by means of a customizable user-interface.

By means of a configuration file, a user-interface consisting of Tabs, Buttons, Text entry fields, etc... is created.

Pushing a button, a command may be sent to Osiris or to an external device (Crestron, AMX,...) via a serial connection.

14.2 Configuration

The Osiris module responsible for the Device Control functionality is the `,osi_devctrl'` module. To enable this module, a license is required.

The default configuration file for the device control software is `,devctrl.cfg'`.

The device control utility can be started either via the Control Panel, (cfr. Gui configuration file) or via the command-line:

```
$BVS_HOME/bin/osi_startmodule osi_devctrl -display <display> ...
```

Next to the `,-display'` option, a number of additional options may be used:

Option	Use
display	Describes the X-windows capable display. For example <code>mvgs pc:0</code>
configspec	Specifies the configuration file for the device control interface. By default, the file is <code>'devctrl.cfg'</code>

14.3 Syntax of Device Control Configuration.

This configuration is stored in a separate file (the file specified in the DEVCTRL command of the control panel configuration). There can be more than one such file (one for each device-control button on the control panel). These files are parsed in. A menu consists of different frames with one or more widgets (buttons, labels and slider-bars). These widgets are ordered in a grid-like way.

Following commands are known:

- **BACKGROUND** <value>: <value> becomes the default value for the background color.
- **ACTIVEBACKGROUND** <value>: <value> becomes the default value for the active background color.
- **FOREGROUND** <value>: <value> becomes the default value for the foreground color (text colors/ bitmap colors).
- **FONT** <value>: <value> becomes the default font. can be a X11-like format ("fixed", "*_*_*-12-1-*_*") or the TCL-way of specifying fonts ({fixed 22 bold})
- **FRAMETYPE** <value>: <value> can be either "plain" or "notebook". This specifies how the different frames are ordered. "plain" just puts the frame next to each other, "notebook" creates a notebook with a tab for each frame (see the FRAME command).

- **NODECORATION** : when this keyword is specified, the device control menu will be shown without window border. This allows to define a layout which covers the entire local X-display, thus emulating a touch-screen-like interface.
- **FRAME** [NEWROW] [TITLE <title>]: creates a new frame. Each new widget will be placed in this new frame until the next frame is created. NEWROW starts a new row of frames if the frametype is “plain” otherwise this option is ignored. TITLE specifies the title on top of the frame if the frametype is “notebook” otherwise it is ignored. Title may contain plain text or images.
- **MINWIDTH** <value>: specifies the minimal width of a widget in the grid.
- **MINHEIGHT** <value>: specifies the minimal height.
- **PADX** <value>: specifies the space between widgets in the X-direction.
- **PADY** <value>: specifies the space in the Y-direction.
- **ROW**: starts a new row of widgets in the frame. Widgets are placed from left to right until a “ROW” command is specified, then new widgets appear on the next row of the grid.
- **SPACE** [<number>]: skips one or more horizontal spaces before placing the next widget. If no number is specified one place in the grid is skipped.
- **LABEL CONFIG** <configuration> [COLUMNS <number>]: puts a label in the frame. The configuration of the label is done in just the same way (and therefore with the same protocol) as the configuration of the buttons in the control panel (see previous chapter). If COLUMNS is specified then it represents the number of grid-places this widget takes.
- **BUTTON CONFIG** <configuration> \
 - [SYSCOMMAND <ID> <user> <command>] \
 - [OSICOMMAND <command>] \
 - [DEVICECOMMAND <command>] \
 - [OSIRELEASECOMMAND <command>] \
 - [DEVICERELEASECOMMAND <command>] \
 - [COLUMNS <number>].

This command creates a button. CONFIG and COLUMNS are the same parameters as for the LABEL command.

SYSCOMMAND specifies a Unix command to be executed, the <ID> allows to distinguish between 2 identical process names.

OSICOMMAND specifies a command sent to any OSIRIS-module, the syntax for these commands is exactly the same as commands sent to the GCI module (e.g.: “>autopatch logicalswitch -input 1 -output 2”).

DEVICECOMMAND specifies a string or hex-code to send to the serial port connected to the gci-module.

These 3 commands send a string when the button gets pressed.

SYSRELEASECOMMAND, OSIRELEASECOMMAND and DEVICERELEASECOMMAND are three commands specified in the same way as SYSCOMMAND, OSICOMMAND and DEVICECOMMAND but these commands get executed when a button is released instead of when it is pressed. All commands can be specified at the same time.

- **SLIDER** [COLUMNS <number>] \
 - [MIN <number>] [MAX <number>] \
 - [SYSCOMMAND <user> <command>] \
 - [OSICOMMAND <command>] \
 - [DEVICECOMMAND <command>] \
 - [OSIQUERY <command>] \
 - [DEVQUERY <command>] \

[RETURNEVENT <event>]

This command creates a sliderbar. COLUMNS again specifies the number of spaces the widget takes. MIN and MAX specify the minimal and maximal value of the slider.

SYSCOMMAND, OSICOMMAND and DEVICOMMAND are commands sent with each change of the slider bar (like for the BUTTON command). The only difference is that all occurrences of “%d” get replaced with the current value of the sliderbar.

OSIQUERY and DEVQUERY are commands that query for the current value of the sliderbar. These commands should generate an OSIRIS event. RETURNEVENT specifies this event. This event can have several options. One of these options should have as value “%d”. The actual value sent by the event will be used as the value for the sliderbar. An example: in the configuration file an event is specified like this: “lightoutput –light right –value %d”, this means that an event like “lightoutput –light right –value 20” will put the sliderbar on value 20. If one the options are incorrect the value is ignored. This means that “lightoutput –light left –value 10” is completely ignored (at least for the slider in question). The commands that get executed by SYSCOMMAND, OSICOMMAND and DEVICOMMAND should also generate this event if you want the sliderbars from different gui’s to follow.

- **COMBOBOX** NAME <name> [COLUMNS <number>] \
VALUES { <values> } \
[SYSCOMMAND {<ID> <user> <command>}] \
[OSICOMMAND . <command>] \
[DEVCOMMAND <command>]

This command creates a pull-down list. Each value in the list is a string in the parameter <values>. When an option is selected form the list, the command string will be executed. The selected string-value can be used in the list of arguments with a “%s” notation.

- **SYSRAISECOMMAND** <Name> <User> <Command>. This specifies the default action which needs to be performed when a frame is activated
- **SYSLOWERCOMMAND** <name> <User> <Command>. This specifies the default action which needs to be performed when a frame is no longer used
- **OSIRASECOMMAND** <osiris command> : this specifies the Osiris command which needs to be performed when a frame is activated
- **OSILOWERCOMMAND** <osiris command> : this specifies the Osiris command which needs to be performed when a frame is no longer used
- **DEVRAISECOMMAND** <command> : this specifies the string sent to the gci-module which needs to be performed when a frame is activated.
- **DEVLOWERCOMMAND** <command> : this specifies the string sent to the gci-module which needs to be performed when a frame is no longer used.

14.4 Example

```
#####  
# general commands  
#####  
FRAMETYPE notebook  
FONT {times 16}  
PADX 5  
PADY 5  
MINWIDTH 100  
MINHEIGHT 20
```

```

#GEOM 500x500+0+0
#NODECORATION

#####
#####

FRAME TITLE "Layouts" newline image layout_icon
ROW
ROW
SPACE 1
  BUTTON CONFIG {font {times 18 bold} text "General 1"} OSICOMMAND "alarm -name Layout1"
SPACE 1
  BUTTON CONFIG {font {times 18 bold} text "General 2"} OSICOMMAND "alarm -name Layout2"
SPACE 1
ROW
ROW
SPACE 1
  BUTTON CONFIG {font {times 18 bold} text "Esprit"} OSICOMMAND "alarm -name Layout3"
SPACE 1
  BUTTON CONFIG {font {times 18 bold} text "Rocade"} OSICOMMAND "alarm -name Layout4"
SPACE 1
ROW
ROW
COMBOBOX NAME LoadAlias COLUMNS 1 VALUES { \
      Value_1 \
      Value_2 \
      Value_3 \
      Value_4 \
      Value_5 \
      Value_6 \
      Value_7 \
      Value_8 \
} \
  SYSCOMMAND {LOADALIAS root /home/bvs/config/switch_source "%s" 1}

```


15 Error Log

The error-log serves as a first-line-troubleshooting tool. It will offer you the most basic information as to what the Osiris SW system is busy doing and which kind of problems it is experiencing.

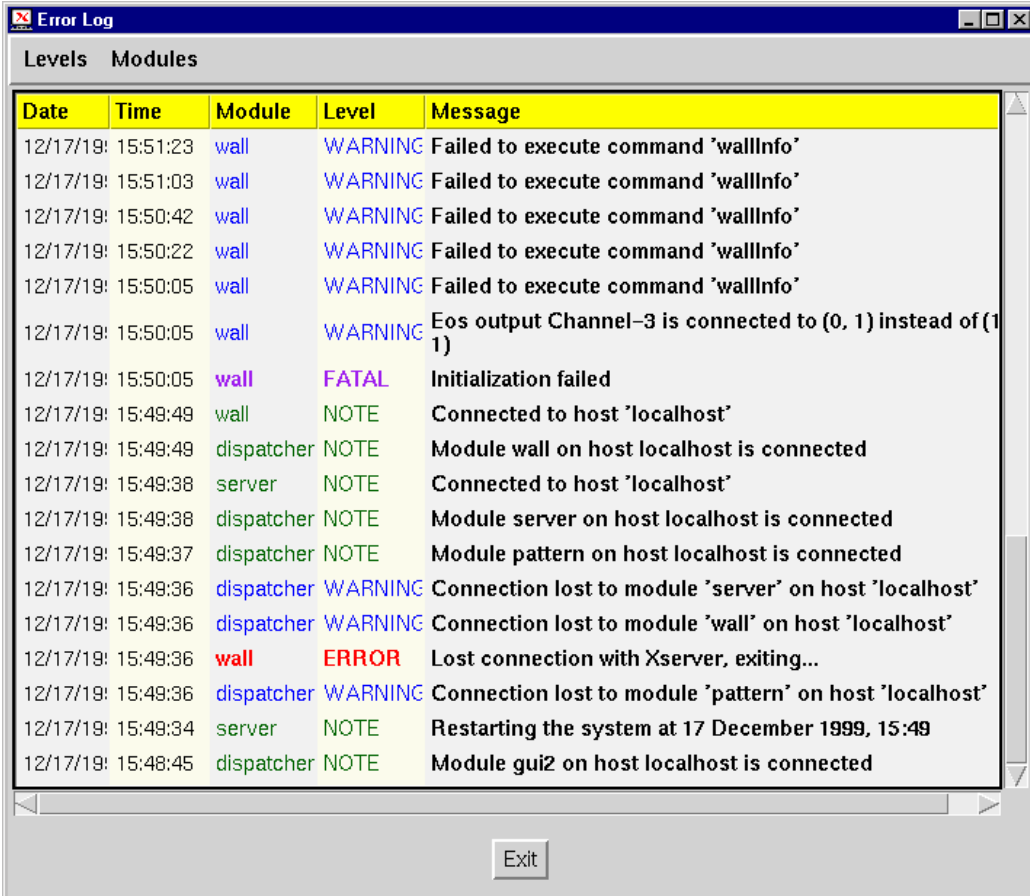
The view only displays the most recent messages. Also, one can view the information using different filters:

- filter on severity:
 - **Fatals**
 - **Errors**
 - **Warnings**
 - **Notes**
- filter on message originating module:

Lists all modules that have been spawning messages into this buffer. When checked, messages are displayed, when unchecked, these messages are suppressed.

This list is NOT stored within a file on the file-system. More detailed messages are displayed in the Console-window (cfr. System Administration) and are stored in the ‚ctrlpanel.log‘ and ‚ctrlpanel.oldlog‘ files.

The error-log should always be used whenever the system does seem to come up correctly or behaves inconsistently. For more details, always check the **console-window** or go and have a direct look in the logging files.



Date	Time	Module	Level	Message
12/17/19:	15:51:23	wall	WARNING	Failed to execute command 'wallInfo'
12/17/19:	15:51:03	wall	WARNING	Failed to execute command 'wallInfo'
12/17/19:	15:50:42	wall	WARNING	Failed to execute command 'wallInfo'
12/17/19:	15:50:22	wall	WARNING	Failed to execute command 'wallInfo'
12/17/19:	15:50:05	wall	WARNING	Failed to execute command 'wallInfo'
12/17/19:	15:50:05	wall	WARNING	Eos output Channel-3 is connected to (0, 1) instead of (1, 1)
12/17/19:	15:50:05	wall	FATAL	Initialization failed
12/17/19:	15:49:49	wall	NOTE	Connected to host 'localhost'
12/17/19:	15:49:49	dispatcher	NOTE	Module wall on host localhost is connected
12/17/19:	15:49:38	server	NOTE	Connected to host 'localhost'
12/17/19:	15:49:38	dispatcher	NOTE	Module server on host localhost is connected
12/17/19:	15:49:37	dispatcher	NOTE	Module pattern on host localhost is connected
12/17/19:	15:49:36	dispatcher	WARNING	Connection lost to module 'server' on host 'localhost'
12/17/19:	15:49:36	dispatcher	WARNING	Connection lost to module 'wall' on host 'localhost'
12/17/19:	15:49:36	wall	ERROR	Lost connection with Xserver, exiting...
12/17/19:	15:49:36	dispatcher	WARNING	Connection lost to module 'pattern' on host 'localhost'
12/17/19:	15:49:34	server	NOTE	Restarting the system at 17 December 1999, 15:49
12/17/19:	15:48:45	dispatcher	NOTE	Module gui2 on host localhost is connected

16 On-line Help

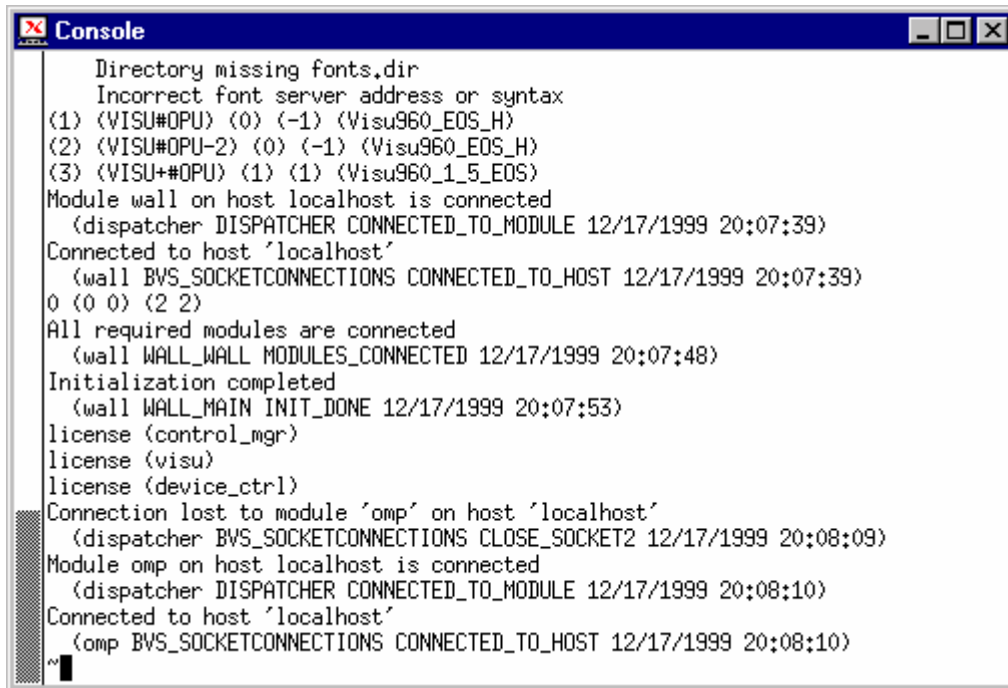
On-line Help opens the Acrobat Reader displaying the Osiris manual (this document).
The on-line help is not context-sensitive and will always display the complete document.

17 System Administration

A set of system administration options are available from the separate control panel appearing when clicking the appropriate button in the regular control panel. This section may, by default, only be accessed by the ,admin' user.

17.1 Console Window

Selecting the Console window button in the Administration Control Panel opens up an X-terminal which is an interactive terminal to the EOS system. This is a session started as the ,ofc' user.

A screenshot of a Windows-style X-terminal window titled "Console". The window contains a series of system messages and status reports. The text is as follows:

```
Directory missing fonts.dir
Incorrect font server address or syntax
(1) (VISU#OPU) (0) (-1) (Visu960_EOS_H)
(2) (VISU#OPU-2) (0) (-1) (Visu960_EOS_H)
(3) (VISU+#OPU) (1) (1) (Visu960_1_5_EOS)
Module wall on host localhost is connected
(dispatcher DISPATCHER CONNECTED_TO_MODULE 12/17/1999 20:07:39)
Connected to host 'localhost'
(wall BVS_SOCKETCONNECTIONS CONNECTED_TO_HOST 12/17/1999 20:07:39)
0 (0 0) (2 2)
All required modules are connected
(wall WALL_WALL MODULES_CONNECTED 12/17/1999 20:07:48)
Initialization completed
(wall WALL_MAIN INIT_DONE 12/17/1999 20:07:53)
license (control_mgr)
license (visu)
license (device_ctrl)
Connection lost to module 'omp' on host 'localhost'
(dispatcher BVS_SOCKETCONNECTIONS CLOSE_SOCKET2 12/17/1999 20:08:09)
Module omp on host localhost is connected
(dispatcher DISPATCHER CONNECTED_TO_MODULE 12/17/1999 20:08:10)
Connected to host 'localhost'
(omp BVS_SOCKETCONNECTIONS CONNECTED_TO_HOST 12/17/1999 20:08:10)
```

Next to being a regular X-terminal, it shows all the Osiris system messages and is therefore a valuable source of information when looking for causes of things not happening as expected.

All messages that are redirected to the following file are being displayed within this console session:

```
$BVS_HOME/logging/LoggingPipe
```

The messages are stored within the file `$BVS_HOME/logging/ctrlpanel.log`. When this file has reached a total size of 1Mb, it gets copied to a `$BVS_HOME/logging/ctrlpanel.oldlog` file and the `ctrlpanel.log` is blanked out and new messages are stored.



When experiencing difficulties within Osiris of any nature, please e-mail a zipped version of the `ctrlpanel.log` and possibly the `ctrlpanel.oldlog` with a detailed description of what is not working as expected to 'support.bcd@barco.com' for further analysis.

17.2

17.3 UCL Control

- UCL stands for **User Control Language**. It is used as an interfacing language between the different Osiris components and the X-server.
- It is an extension of the X-server. Within the X-server environment, a UCL table is maintained and holds information with respect to the placement of application windows, as defined by the loaded layouts.
- By means of the toggle button in the Administration control panel, one can activate/deactivate the UCL control, meaning that when turned off, any X-windows based client-application is allowed to display on the display wall managed by the X-server. When it is on, the X-server will only accept a connection from an application that has been started from within the Osiris Control Manager environment. (aliases). Only trusted clients, as the Remote Pointer application is, are granted to connect to the X-server.
- The complete UCL functionality for the X-server may be turned off by using the `,-no_ucl` option in the startup of the X-server application. This has as a consequence that the application windows, started by Osiris can no longer be placed using the Osiris tools.

17.4 Osiris Restart

Whenever some modification has been done to the configuration of the system (wall or X-server configuration), an Osiris restart is to be done in order to invoke the changes.

This will restart the Wall module and all other modules communicating to the Wall module.



The Osiris kernel module, `osi_dispatcher`, is not restarted by this operation. Also, the `osi_gui` processes are not reset.

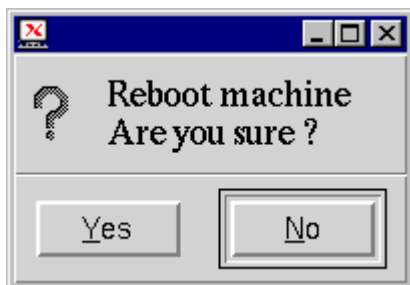
After the 'Initialization completed' message when Osiris is fully restarted, it will reload the items that were previously on the Active Layout stack. Also, when a scenario was active before the Osiris restart was initiated, that scenario will be restarted again automatically.

17.5 System restart

In order to restart the EOS machine safely, this is the advised method to do so.

The Osiris database will be closed properly first prior to the restart of the Operating System.

When selecting the System Restart button, a confirmation-message will pop up.



After confirming the restart, wait for the system to restart. It may take some time before the Osiris Gui to close down.

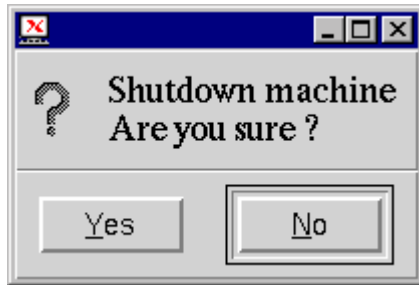


Rebooting the system via the reset or power on/off button directly without following this procedure may put the Osiris database integrity in danger and is therefore not advised.

17.6 System halt

To safely shutdown the system, the 'Shut the system down' button may be used. As with the 'Restart' button, this will properly close off the Osiris database and halt the Operating System.

Upon selecting the button, a confirmation will be requested:



After confirming the restart, wait for the system to restart. It may take some time before the Osiris GUI to close down.



Halting the system via the power off button directly without following this procedure may put the Osiris database integrity in danger and is therefore not advised.

17.7 Backup/Restore customizations

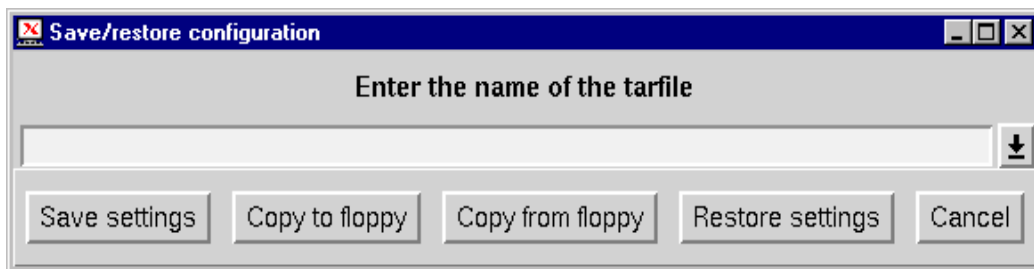
To safeguard the specific settings on the EOS, a backup/restore facility has been provided. During the backup procedure, the following files/directories will be included in the backup by default:

Files/Directories	Purpose
\$BVS_HOME/wall.cfg	Hardware configuration file
\$BVS_HOME/XF86Config.eos	X-server configuration for Linux EOS
\$BVS_HOME/XF86Config.mmt	X-server configuration for EOS Compact/Power
\$BVS_HOME/mvgs.cfg	X-server configuration file for Solaris EOS
\$BVS_HOME/backupfiles.cfg	List of files to be backed up additionally
\$BVS_HOME/gui.cfg	Configuration file for the GUI

\$BVS_HOME/osi_licenses	File containing the Osiris licenses for all licensed Osiris modules
\$BVS_HOME/database.dump	Dump of the Osiris database. Contains definitions like alias, layout, ...
\$BVS_HOME/osi_customserver	Script which contains specifics with respect to the X-server configuration. All non-standard X-windows related things are to be specified in this file, for ex.: specific fonts to use, window manager to use, ...
\$BVS_HOME/osiris.cfg	This configuration file contains specific environment variables used within the startup of the different Osiris modules (handled by the 'osi_startmodule') script
\$BVS_HOME/wall.resource	Contains specific resources to be set for the wall-module.

To specify additional files to be backed up, these files are to be specified in the `$BVS_HOME/backupfiles.cfg` file.

When selecting the Backup/Restore button, the following entry appears:



To backup:

- Fill in a name
- Select [**Save settings**] to create the backup file within the `$BVS_HOME/logging` directory. The file will have the extension `.tar` and will contain the following 2 sub-files: `root.tar` and `bvs_home.tar`
- To copy the file to a floppy, make sure an empty DOS formatted floppy is present in the floppy-device and select the 'Copy to floppy' button. The backup file previously created with the [Save settings] button will be saved to the floppy

To restore:

- To restore some previously made backup, copy that tar-file to the `$BVS_HOME/logging` directory and select [**Restore settings**]
- To restore some file from a floppy, make sure the floppy contains the proper backup file (as saved previously), insert in the floppy drive device and select the [**Copy from floppy**]. Then select the [**Restore settings**] button.

Execute specific tasks after the backup/restore operation:

- For the backup process:
When creating an executable script

```
$BVS_HOME/config/backup.script
```

The script will be executed after the normal backup operations have been concluded.

- For the restore process:

When creating an executable script

```
$BVS_HOME/config/restore.script
```

The script will be executed after the normal restore operations have been concluded.

17.8 Specific Osiris customizations

In order to keep specific Osiris related customizations together, one can specify these settings within:

```
$BVS_HOME/config/osiris.cfg
```

This file is created each time the Osiris configuration tool is used.



This is only supported from Osiris V1.1.2 onwards

17.9 Specific X-server customizations

In order to keep specific Osiris related customizations together, one can specify these settings within:

```
$BVS_HOME/config/osi_customserver
```

For Osiris versions V1.1.2 and higher

Specific X-server related things may be specified within these files. Possible customizations are:

- Background logo
- Specific X-Windows resources
- Specific Font-paths
- Window manager to use

These files will make part of the backup when a backup operation is performed!

17.10 Using Osiris in stand-alone mode (off-line)

17.10.1 Purpose

In some conditions, it may be handy to use Osiris in an off-line mode. By 'off-line' is meant a situation where the normal X-server can not be started. This may be on a system where just a normal graphical card is inserted rather than the specialized OVT or IMSGP boards.

This can be handy for the following purposes:

- **Practice** : Almost the complete functionality of Osiris can be used in this off-line configuration.

- **Testing** : in order to test out a particular real-life configuration of a customer.
- **Demo** : to demonstrate the possibilities of Osiris without the need for a complete Wall-setup
- **VISU Stand-Alone** : to configure a Linux console driving a Visu stand-alone unit.

17.10.2 Configuration

The XServer should be configured for standalone operation (see EOS X Terminal user's manual). Osiris will detect standalone use automatically.

- the local graphical environment will be display "unix:0"
- a dummy X-server running as "unix:1" will simulate the presence of a "wall".

17.10.3 Restrictions

- Dummy X-server needs sufficient memory to "simulate" a real frame buffer. E.g. for a 3x2 / XGA / 8 bits, approx. 4.5Mb is needed.
- If only the Osiris user interface is needed, instead of running the dummy server, the `osi_startmodule` may refer to the local X-server (unix:0) also. Any applications (incl. Visu windows) will display on the local desktop.
- With the dummy X-server, the actual output is stored in memory.

To store the entire screen :

```
xwd -root -display unix:1 > dumpfile
```

To display this :

```
xwud -in dumpfile -display unix:0 -geometry +0+0
```


18 Software Integration Issues

During the integration of Osiris within a Control Room environment, the guidelines specified in this chapter are very important. They will assist you in the integration of a variety of customer applications within the Osiris system.

18.1 Integration of X-windows based applications

18.1.1 EOS configuration

In order to integrate X-windows based applications, the following steps should be taken at the EOS side:

Action	Involved files
Add hostname of host where application is to be started	/etc/hosts
Create the alias in the Osiris environment to start the application.	

18.1.2 Application's host configuration

Action	Involved files
Create an entry for the EOS in the host-table of the application host. You will need the system's administration password for doing this! The name of the host should be ,mvgs' by default. When more than one EOS is connected to the same LAN, different names are to be used (mvgs1, mvgs2,...)	/etc/hosts
Change the ,rhosts' file of the user with whom the application will be started. If the file does not exist, create a new one and create the following line: mvgs root This will grant the EOS permission to do a ,rsh' and ,rhp' to the application host as the specified user from the root-environment on the EOS. To test whether this works, login on the EOS, become super-user, execute a rsh command on the remote host as the specific user. For ex.: rsh -l <username> <hostname> date	\$USER_DIR/.rhosts



Doing a rsh on a remote host does not start the user-specific login files as there are the ,.profile', ,.login' or ,.cshrc'. Be aware of this when starting an application in this way.

It may result in the application not starting in the way it would start up after

logging in to the application host with that specific user.

The advice is to make a separate script containing the application's startup and also include the specific user-specific scripts prior to startup of the applic.

18.2 Integration of NT based applications

18.2.1 TSE/Metaframe/UIS

NT TSE

A server running the Windows NT TSE software without the **Metaframe** and possibly **UIS** software, can not be used in conjunction with the Osiris environment as this is a X-windows only environment which is not supported by the NT TSE alone.

Metaframe

Citrix Metaframe is installed on top of NT TSE. The communication between server and display client takes place via an intermediate protocol ICA. Therefore, an ICA-client is required on the display-side (EOS)

A number of steps are required to integrate a TSE/MetaFrame session within Osiris :

Step	Details
ICA Client installation	Install the latest ICA client on the Linux platform. The ICA client can be found at the following location: http://download.citrix.com/#client_ica
Startup of wfcmgr	Start the wfcmgr application as super-user
Do some customizations	Make a directory ./ICAClient Then make a link between ./ICAClient/appsrv.ini and /root/.ICAClient/appsrv.ini ln -s /root/.ICAClient/appsrv.ini ./ICAClient/appsrv.ini
Create a session	Create a named session using wfcmgr as super-user: # cd /usr/lib/ICAClient # wfcmgr -display mvgs_pc:0 Specify : - Network : the TSE server to connect to - Window : the desired (max.) window size and the number of colors - if possible, select 16 colors - if 256 colors are needed, always select the "sharing" option - application : if not specified, a complete desktop is desired
Check session	Start up the session manually : # ./wfcmgr -desc <description> -display mvgs:0

	<p>Verify the actual process which is started</p> <pre># ps -ef grep wfica</pre>
Create the alias	<p>Create an alias, start method “local” to start the ICA-client.</p> <p>Given the number of parameters needed by the ICA-client, it may be advisable to create a local script with the complete start command :</p> <pre>/usr/lib/ICAClient/wfica <params></pre> <p>For the local script, put the following line at the end of the script in order to keep the script alive:</p> <pre>tail -f \$0 > /dev/null</pre>



Please, refer to the user's manual of the Citrix Metaframe product for more detailed information.

UIS

UIS (Unix Integration Services), sits on top of the Metaframe environment and allows the display of NT-sessions of applications using genuine the X-windows protocol. To integrate the startup of a UIS session within Osiris, an alias is to be created starting a script residing somewhere on the EOS and contains the following:

```
#!/bin/csh

set mcookie=`xauth list $DISPLAY | grep MAGIC`
rsh metaframe x11 -display $DISPLAY -mcookie $mcookie[3] \
-user administrator -passwd glad01 -depth 8 -geom +0+0
tail -f $0 > /dev/null
```

Specify ,Cookie‘ within the ,Magic Cookie‘ subsection of the Alias definition.

More options may be used for the ,x11‘ application but this is outside the scope of this manual.



Please, refer to the user's manual of the Citrix UIS product for more detailed information.

18.3 Miscellaneous application integration

18.3.1 VMS based applications

To integrate VMS applications in Osiris :

The syntax on VMS is quite different from Unix : create a local script on the EOS which does the remote execution

The X-authority mechanism is not supported by VMS. The port-mechanism should be used instead when defining the alias.

For configuration of remote execution privileges :

```
Applic      : UCX
Command : add proxy ofc /host=mvgs /remote_user=ofc
```

To set up the display variable :

```
set display /node=mvgs /transport=tcPIP /create /server=0
```

18.4 Integration of 3rd party serial devices using GCI functionality

Within a system's configuration, one may regularly find the usage of serial devices which may deliver triggers for the Osiris environment to react to. Regular used devices are ,Crestron' or ,AMX'.

These devices are called ,device control' systems which may be programmed to deliver an Osiris command string to one of their serial ports which is connected to one of the serial ports of the EOS system. For more information on how to connect such a device, refer to section 233.4027.389.

Using the standard commands, as specified in section 19 or any of the custom-created ones, external events may be integrated within the Osiris environment in a seamless manner.

An example:

A Crestron touch-panel is programmed to have a user-interface which allows a user to press a button to load/unload a layout. A string will be produced by the Crestron SW and is sent to the serial port of the EOS connected, where the GCI module will handle the incoming command and pass it to the Osiris wall-module which will initiate the loading and unloading of the requested layout.

An example of a string that might be sent from a device controlling system as Crestron would be:

```
alarm -name Layout1\n
```

to activate a trigger Layout1 defined within the running scenario. The '\n' is required!

19 Osiris Directory Tree overview

```
$BVS_HOME          : selected at installation time
Sub-directories    : symbolic links to packages :
  bin
  config
  images
  include
  lib
  registry
  tk

Database :
  velocis

Packages :
  pkgs  agent
        alarm
        autopatch
        ...
  tkgui  library
        tk
        config
        images
        images_16
  xmvgs  bin
        config
        keyboards
        modes
        octopusresources
```

20 Osiris Command List

20.1 General

As specified before, Osiris can be controlled using specific Osiris commands (cfr. Section 291.4085.331)

When connecting to the Osiris connection port (5701) via telnet, the commands entered at that point are directed by default to the ,wall‘ module.

To direct Osiris commands to another module, use the following syntax to do so:

>module OsirisCmd <options>

module=name of module (cfr. List below)

<options> are options specified in the commands in next sections

The ,>module‘ part may be omitted when using wall-related commands. (default module=wall)

List of available modules:

Module
Wall
Crt
Lcd
Dmd
Dme
Oml
Omp
Gci
Gsw
Sad
Tty
Agent
Server

20.2 wall

20.2.1 enablescenario

Parameters

No parameters.

Comments

Activate the paused scenario

20.2.2 alarm

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Name	name of the alarm	string			

Comments

Generate an alarm

20.2.3 dumpdatabase

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Name	name of the dump file	string			\$WALL_SESSION/databas e.dump

Comments

Dumps the contents of the database in a file

20.2.4 infoscenario

Parameters

No parameters.

Comment

Returns the status of a running scenario.

The value returned is an empty string if there is no running scenario.

- The value returned for a running scenario is a string with three parts:
 - name
followed by the name of the scenario
 - state
followed by the keyword running or paused
 - step
followed by a number indicating the current scenario sequence step

20.2.5 loadalias

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
name	name of the alias	string			
transparent	transparent mode	bool			ON
opaque	opaque mode	bool			OFF

Comments

- The alias has to be defined in the database.
- Only one option {transparent, opaque} should be specified.
- Opaque means that first a separator is loaded.

20.2.6 loadlayout

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Name	name of the layout	string			
Transparent	transparent mode	bool			OFF
Opaque	opaque mode	bool			OFF

Comments

- The layout has to be defined in the database.
- Only one option {transparent, opaque} should be specified, which overrides the default layout mode.
- Opaque means that first a separator is loaded.

20.2.7 loadscenario

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
name	name of the scenario	string			

Comments

20.2.8 loadvisu

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
name	name of the visu window	string			
transparent	transparent mode	bool			ON
opaque	opaque mode	bool			OFF

Comments

Load a visu window

- The visu window has to be defined in the database.
- Only one option {transparent, opaque} should be specified.
- Opaque means that first a separator is loaded.

20.2.9 disablescenario

Parameters

No parameters.

Comments

Pause the running scenario

20.2.10 restoredatabase

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Name	name of the dump file	string			\$WALL_SESSION/databas e.dump
Init	initialise flag	int	0	1	1

Comments

restores the contents of the database from a file

By default, all data is deleted before restoring.

20.2.11 stopscenario

Parameters

No parameters.

Comments

stop the running scenario

20.2.12 ucldisable

Parameters

No parameters.

Comments

disable ucl control for external user

20.2.13 uclenable

Parameters

No parameters.

Comments

enable ucl control for external users

20.2.14 uclstatus

Parameters

No parameters.

Comments

returns the status of the UCL control

The value returned is a string with following values:

- enabled
- disabled

20.2.15 unloadalias

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Name	name of the alias	string			
Options	unload option	string			onevisible

Comments

- The unload options are
 - one
 - all
 - onevisible
 - allvisible
 - scenario

20.2.16 unloadlayout

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Name	name of the layout	string			
Options	unload option	string			onevisible

Comments

- The unload options are
 - one
 - all
 - onevisible
 - allvisible
 - scenario

20.2.17 unloadvisu

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Name	name of the visu window	string			
Options	unload option	string			onevisible

Comments

- The unload options are
 - one
 - all
 - onevisible
 - allvisible
 - scenario

20.3 server

20.3.1 stopserver

Parameters

No parameters.

Comments

stops the Xserver and exits the server module

20.4 tty

20.4.1 write

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
service	name of the module that has opened the port	string			
port	the port to connect to	string			
string	the string to send	string			
code	the string to send but in a hexadecimal or decimal format	string			

Comments

write a string to a serial port

- port is an optional parameter. If it is not specified then the tty module sends the string to every tty-port that is opened by the specified service (this is the osiris modulename).
- only one of string and code needs to be specified. If they are both specified, string is used. String is just an ASCII string, while code specifies the same string but as the hexadecimal or decimal values separated by a slash (/).
- Examples :

- write `-service gci -string [put the lights off]` : this sends the string "put the lights off" to every port that the gci module has opened (for instance to a crestron device). Note : there is no newline or carriage-return automatically added to the string. If you need this, you must specify it in the string itself.
- write `-service gci -port /dev/ttya -code /0x0A/0xFF/0xE3/` : sends 3 characters to the port ttya IF the gci has opened it. The characters send have an hex-value of 0A, FF and E3.

20.5 sad

20.5.1 reboot

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Delay	number of minutes the module waits before rebooting the system	int			0

Comments

shuts the machine down

20.5.2 shutdown

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Delay	number of minutes the module waits before shutdown	int			0

Comments

20.6 crt

20.6.1 polling

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Name	name of the polling log file	String			\$WALL_SESSION/pr oj_polling.log
Interval	polling interval in seconds	Int	0		60

Comments

control polling of projector settings

- Polling is ON if the interval specified is not zero.
- Polling is OFF if the interval specified is zero.
- The interval specified is the time between the polling of two projectors. Only one projector is polled in one polling interval.

20.6.2 readsource

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Id	projector logical id	int	1	99	
Address	projector hardware address	int	1	99	
Port	serial port	string			
Module	name of the tty module	string			tty
Type	projector type	string			

Comments

read the source of the projector

Specify either {id} or {address, port, module, type}

20.6.3 remotecontrol Unit

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Id	Projector logical id	int	1	99	
Address	Projector hardware address	int	1	99	
Port	Serial port	string			
Module	Name of the tty module	string			tty
Type	Projector type	string			
num_0	Numeric key 0	bool			OFF
num_1	Numeric key	bool			OFF
num_2	Numeric key	bool			OFF
num_3	Numeric key	bool			OFF
num_4	Numeric key	bool			OFF
num_5	Numeric key	bool			OFF
num_6	Numeric key	bool			OFF
num_7	Numeric key	bool			OFF
num_8	Numeric key	bool			OFF
num_9	Numeric key	bool			OFF
addr	address key	bool			OFF
arrow_down	arrow down key	bool			OFF
arrow_left	arrow left key	bool			OFF
arrow_right	arrow right key	bool			OFF
arrow_up	arrow up key	bool			OFF
inc_balance	increment balance	bool			OFF
dec_balance	decrement balance	bool			OFF
inc_bass	increment bass	bool			OFF
dec_bass	decrement bass	bool			OFF

inc_brightness	increment brightness	bool			OFF
dec_brightness	decrement brightness	bool			OFF
inc_color	increment color	bool			OFF
dec_color	decrement color	bool			OFF
inc_contrast	increment contrast	bool			OFF
dec_contrast	decrement contrast	bool			OFF
enter	enter key	bool			OFF
exit	exit key	bool			OFF
freez	freez key	bool			OFF
mono	toggle mono/stereo	bool			OFF
expand	toggle normal/expand	bool			OFF
mute	mute key	bool			OFF
adjust	adjust key	bool			OFF
text	text	bool			OFF
stdby	standby key	bool			OFF
pause	pause key				
inc_sharpness	increment sharpness	bool			OFF
dec_sharpness	decrement sharpness	bool			OFF
inc_tint	increment tint	bool			OFF
dec_tint	decrement tint	bool			OFF
inc_treble	increment treble	bool			OFF
dec_treble	decrement treble	bool			OFF
inc_volume	increment volume	bool			OFF
dec_volume	decrement volume	bool			OFF
togg_a	toggles from picture to sound	bool			OFF
sel_sound	force to sound	bool			OFF
sel_pict	force to picture	bool			OFF

Comments

simulate the functions on the remote control unit

- Specify either {id} or {address, port, module, type}
- Only one option should be specified.

20.6.4 toallprojectors

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
command	command name	string			
list	list of projector id's	list			

Comments

send a command to a list of projectors

- If the list is not specified, all known projectors are addressed.

- The list separator can be any non-digit character
- The available command names are:
 - on
 - standby
 - pause
 - operate
 - irdisable
 - irenable

20.6.5 writesource

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	Projector logical id	int	1	99	
address	Projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			
source	the new source	int	1		1

Comments

modify the source of the projector

- Specify either {id} or {address, port, module, type}
- The range of the source is projector type dependant.

20.7 lcd

20.7.1 adjpossible

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Id	projector logical id	int	1	99	
Address	projector hardware address	int	1	99	
Port	serial port	string			
Module	name of the tty module	string			Tty
Type	projector type	string			
Setting	the adjustment setting	string			COLOR

Comments

test if a projector adjustment is possible in the current projector mode

- Specify either {id} or {address, port, module, type}
- Returns a string with 0 or 1

- The possible settings are:
 - COLOR
 - TINT
 - PHASE

20.7.2 polling

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
name	name of the polling log file	string			\$WALL_SESSION/pr oj_polling.log
interval	polling interval in seconds	int	0		60

Comments

control polling of projector settings

- Polling is ON if the interval specified is not zero.
- Polling is OFF if the interval specified is zero.
- The interval specified is the time between the polling of two projectors. Only one projector is polled in one polling interval.

20.7.3 setprojectoroff

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.7.4 setprojectoron

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	
port	serial port	string			

module	name of the tty module	string			tty
type	projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.7.5 toallprojectors

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
command	command name	string			
list	list of projector id's	list			

Comments

send a command to a list of projectors

- If the list is not specified, all known projectors are addressed.
- The list separator can be any non-digit character
- The available command names are:
 - on
 - standby
 - pause
 - operate

20.7.6 readsource

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			

Comments

read the source of the projector

Specify either {id} or {address, port, module, type}

20.7.7 writesource

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			
source	the new source	int	1		1

Comments

modify the source of the projector

- Specify either {id} or {address, port, module, type}
- The range of the source is projector type dependant.

20.8 dmd

20.8.1 poll

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Name	name of the polling log file	string			\$WALL_SESSION/pr oj_polling.log
Interval	polling interval in seconds	int	0		60

Comments

control polling of projector settings

- Polling is ON if the interval specified is not zero.
- Polling is OFF if the interval specified is zero.
- The interval specified is the time between the polling of two projectors. Only one projector is polled in one polling interval.

20.8.2 toallprojectors

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
command	command name	string			
list	list of projector id's	list			

Comments

send a command to a list of projectors

- If the list is not specified, all known projectors are addressed.
- The list separator can be any non-digit character
- The available command names are:
 - on
 - standby
 - pause
 - operate

20.8.3 setprojectoroff

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.8.4 setprojectoron

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.8.5 readlampruntime

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	

address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.8.6 readbrightness

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.8.7 readcontrast

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	Projector logical id	int	1	99	
address	Projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.8.8 writebrightness

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	

address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			
brightness	the new brightness value	int			32

Comments

modify the brightness value of the projector

- Specify either {id} or {address, port, module, type}
- The range of the brightness value is projector type dependant.

20.8.9 writecontrast

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Id	projector logical id	int	1	99	
Address	projector hardware address	int	1	99	
Port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			
contrast	the new contrast value	int			0

Comments

modify the contrast value of the projector

- Specify either {id} or {address, port, module, type}
- The range of the contrast value is projector type dependant.

20.9 dme

20.9.1 polling

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Name	name of the polling log file	string			\$WALL_SESSION/pr oj_polling.log
Interval	polling interval in seconds	int	0		60

Comments

control polling of projector settings

- Polling is ON if the interval specified is not zero.
- Polling is OFF if the interval specified is zero.
- The interval specified is the time between the polling of two projectors. Only one projector is polled in one polling interval.

20.9.2 toallprojectors

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
command	command name	string			
list	list of projector id's	list			

Comments

send a command to a list of projectors

- If the list is not specified, all known projectors are addressed.
- The list separator can be any non-digit character
- The available command names are:
 - on
 - standby
 - pause
 - operate

20.9.3 setprojectoroff

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.9.4 setprojectoron

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty

type	projector type	string			
------	----------------	--------	--	--	--

Comments

Specify either {id} or {address, port, module, type}

20.9.5 readlampruntime

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Id	projector logical id	int	1	99	
Address	projector hardware address	int	1	99	
Port	serial port	string			
Module	name of the tty module	string			tty
Type	projector type	string			

Comments

read the run time of the lamp

Specify either {id} or {address, port, module, type}

20.9.6 readvalues

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			

Comments

read the settings of the projector

Specify either {id} or {address, port, module, type}

20.10 oml

20.10.1 polling

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Name	name of the polling log file	string			\$WALL_SESSION/pr oj_polling.log
Interval	polling interval in seconds	int	0		60

Comments

control polling of projector settings

- Polling is ON if the interval specified is not zero.
- Polling is OFF if the interval specified is zero.
- The interval specified is the time between the polling of two projectors. Only one projector is polled in one polling interval.

20.10.2 toallprojectors

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Command	command name	string			
List	list of projector id's	list			

Comments

send a command to a list of projectors

- If the list is not specified, all known projectors are addressed.
- The list separator can be any non-digit character
- The available command names are:
 - on
 - standby
 - pause
 - operate

20.10.3 setprojectoroff

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
Id	projector logical id	int	1	99	
Address	projector hardware address	int	1	99	
Port	serial port	string			
Module	name of the tty module	string			tty
Type	projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.10.4 setprojectoron

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	Projector logical id	int	1	99	
address	Projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	Projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.10.5 readlampruntime

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.11 omp

20.11.1 poll

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
name	name of the polling log file	string			\$WALL_SESSION/pr oj_polling.log
interval	polling interval in seconds	int	0		60

Comments

control polling of projector settings

- Polling is ON if the interval specified is not zero.
- Polling is OFF if the interval specified is zero.

- The interval specified is the time between the polling of two projectors. Only one projector is polled in one polling interval.

20.11.2 toallprojectors

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
command	command name	string			
list	list of projector id's	list			

Comments

send a command to a list of projecto

- If the list is not specified, all known projectors are addressed.
- The list separator can be any non-digit character
- The available command names are:
 - on
 - standby
 - pause
 - operate

20.11.3 setprojectoroff

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.11.4 setprojectoron

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	

port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.11.5 readlampruntime

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
id	projector logical id	int	1	99	
address	projector hardware address	int	1	99	
port	serial port	string			
module	name of the tty module	string			tty
type	projector type	string			

Comments

Specify either {id} or {address, port, module, type}

20.12 autopatch

20.12.1 logicalswitch

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
TtyPort	serial port	string			
Ttymodule	name of the tty module	string			tty
Input	input name	string			
Output	output name	string			

Comments

make a connection between an input and an output

20.13 gsw

20.13.1 logicalswitch

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
master	master id	int	1	99	1
resource	resource file	string			
port	serial port	string			
module	name of the tty module	string			tty

input	input name	string			
output	output name	string			

Comments

make a connection between an input and an output

20.14 gci

20.14.1 closepipe

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
pipe	full name of the pipe	string			

Comments

20.14.2 openpipe

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
pipe	full name of the pipe	string			

Comments

20.14.3 closettyport

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
module	name of the tty module	string			tty
port	name of the serial port	string			

Comments

20.14.4 openttyport

Parameters

NAME	DESCRIPTION	TYPE	MIN	MAX	DEFAULT
------	-------------	------	-----	-----	---------

module	name of the tty module	string			tty
port	name of the serial port	string			

Comments

21 References

Documentation:

Ref	Contents
R5976144	Osiris Owner's Manual
DOC-2740	MMT-X User's Manual
R5975547	EOS Owner's Manual
R5975817	VISU Owner's Manual
	Windows NT TSE Owner's Manual
	Citrix Metaframe Owner's Manual
	Citrix UIS Owner's Manual
	Octopus Packages V2.0.0 Owner's Manual

Websites:

Quite a lot of information can be found on the Internet with respect to different general topics and 3rd party products. The next table covers just a few of these pools of information.

Ref	Contents
www.barco.com	More info on Barco in general.
www.seufert.com	More info on Barco-Karlsruhe
www.citrix.com	<p>More info on Citrix and it's products and more in particular the location: http://www.citrix.com/products/metaframe/mfnt.htm On the Metaframe 1.8 for Windows NT TSE 4.0 And the location: http://www.citrix.com/products/unix/default.htm On the Citrix Unix Integration Services product. All on-line documentation can be found on: http://www.citrix.com/services/productdocs.asp</p>
www.microsoft.com	<p>More info on Microsoft products and in particular the following location: http://www.microsoft.com/ntserver/terminalserver/default.asp</p>
www.redhat.com	Official Linux Redhat web-site.
Usage of the 'vi' editor, the most prominent editor within Unix/Linux environment	<p>http://docs.sun.com/ab2/@LegacyPageView?toc=SUNWab_8_29%3A%2Fsafedir%2Fspace3%2Fcoll2%2FSUNWabe%2Ftoc%2FADVOSUG%3A1096;bt=Solaris+Advanced+User%27s+Guide;ps=ps%2FSUNWab_8_29%2FADVOSUG%2F06.Using_the_vi_Editor&Ab2Lang=C&Ab2Enc=iso-8859-1 This is an interesting link to the Sun site where the vi-editor is</p>

	<p>explained in more detail.</p> <p>Interesting for non or inexperienced vi-users.</p>
Tcl/tk references	<p>http://dir.yahoo.com/Computers_and_Internet/Programming_Languages/Tcl_Tk/</p> <p>More info on Tcl/Tk which is the scripting language used in different GUI related scripts. (\$BVS_HOME/pkg/tkgui/tk/*)</p>

