


1.0.3	Released	2013-01-08	<ul style="list-style-type: none"> -Itrack TSM0017-86(CRP: Allow for multiple simultaneous TCP/IP connections to CRP) implemented -WallOff, WallOn, WallSetSource1Preferred, WallSetSource2Preferred implemented as a wall level command as per BCMC server supported in their newer versions (Previously command was on projector basis but for newer version of BCMC server, it is possible to execute these commands at wall level). It will be executed at projector level if the BCMC server version won't support them. -GetWallHealthSummary, GetWallAlarms commands introduced for BCMC. -CMS version1 calls updated with version2. 	ASHG	PML	PML
1.0.2	Released	2012-10-08	<ul style="list-style-type: none"> Below commands added -GetPerspectiveAppliedTile -GetDisplayAppliedTile -UpdateDisplayTile -UpdatePerspectiveTile -CreateAndApplyTileOnDisplay -CreateAndApplyTileOnPerspective -RemoveSourceFromPerspectiveTile Bug fixed for create perspective command. 	ASHG	PML	PML
1.0.1	Released	2012-01-15	<ul style="list-style-type: none"> -Mgs Server Incorporated -LoadVideoSourceOnPerspectiveWithMgs introduced -ApplyDecoratorToASourcePerspective introduced -RemoveDecoratorFromASourcePerspective introduced - RunCmsMgsSyncUtility introduced 	ASHG	PML	PML
1.0.0.snapshot	Released	2010-10-21	Original issue	ASHG	PML	PML
Version	Status	Date (yyyy-mm-dd)	Comments / changes	Prepared	Checked	Approved

Project name :

Barco Control Room Proxy

Doc. title :

Barco Control Room Proxy User Manual

Size	A4	<i>This document is the exclusive property of Barco and shall not be used, copied or communicated to third parties without its prior authorization</i>		Barco nv President Kennedypark, 35 8500 Kortrijk Belgium www.barco.com	 Visibly yours
Doc. type	Word	Cust. doc. ID	-		
Sheet n°	1	Country	Belgium		
N° of sheets	49	City	-		

Computer file :	Control Room Proxy - ver 1.0.3.doc
-----------------	------------------------------------

Table of Contents

- 1 ABOUT THIS MANUAL.....3**
- 2 CONTROL ROOM PROXY DESCRIPTION3**
- 3 INSTALLATION4**
 - 3.1 PREREQUISITES4
 - 3.2 INSTALLING THE CRP(CONTROL ROOM PROXY).....4
- 4 CONFIGURING THE CRP APPLICATION GUI6**
 - 4.1 FILE MENU DESCRIPTION:6
 - 4.2 CONFIG MENU DESCRIPTION:6
 - 4.1.1 *CMS*:7
 - 4.1.2 *BCM*:9
 - 4.1.3 *NETWORK*:10
 - 4.3 LICENSE MENU DESCRIPTION:11
 - 4.2.1 *LICENSE PIN*:11
 - 4.2.2 *LICENSE IDENTIFICATION*:11
 - 4.4 RESTART MENU DESCRIPTION:11
 - 4.5 HELP MENU DESCRIPTION:12
 - 4.4.1 *HELP CONTENTS*:12
 - 4.4.2 *ABOUT CONTROL ROOM PROXY*:12
- 5 SUPPORTED COMMANDS13**
 - 5.1 BASIC COMMAND SET14
 - 5.2 SIMPLE TEXT COMMAND SET20

1 About this manual

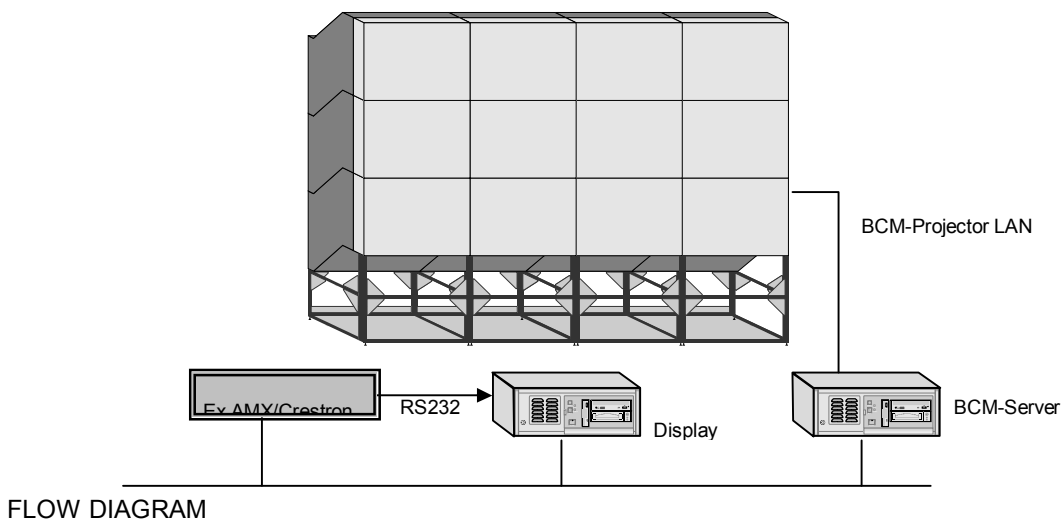
This manual describes the installation and the configuration of Control Room Proxy(CRP).

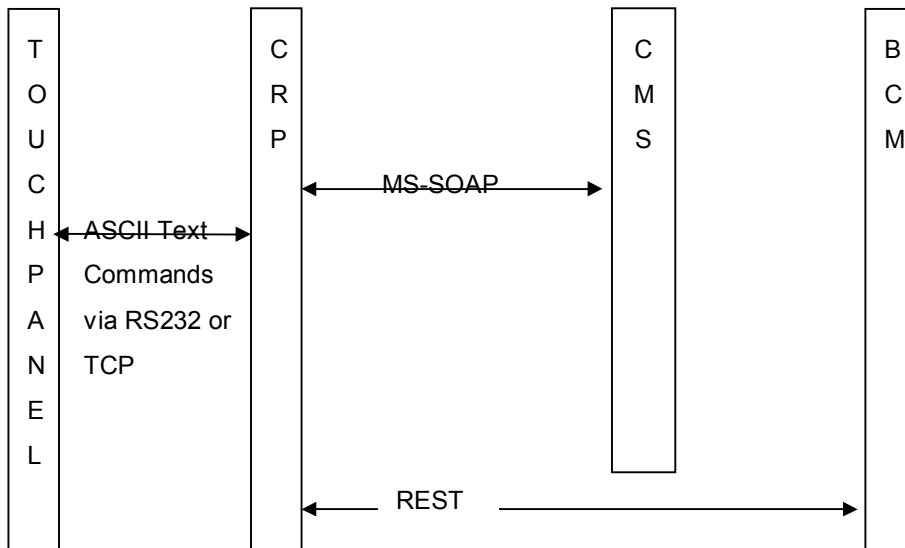
2 Control Room Proxy Description

The CRP acts as touch panel interface and allows the control of:

- 1) Layouts, Perspectives, & Sources
- 2) Projectors

The main requirement of this tool is to act as static command interface between a touch panel system and it's connected hard- and software devices. This allows different BARCO devices to have a compatible command set towards the touch panel system. (Ex. BCM, CMS)





3 Installation

3.1 Prerequisites

The CRP is developed using the java framework and as such requires the java6 or above to be installed. For installations where layout management is required the CMS Wall Management software needs to be installed.

For installation where projector control is required the BCM-Server needs to be installed.

Typically the CRP is installed on the BARCO-Transform-N display wall controller. The CRP can also be installed on a separate PC.

Software listing:

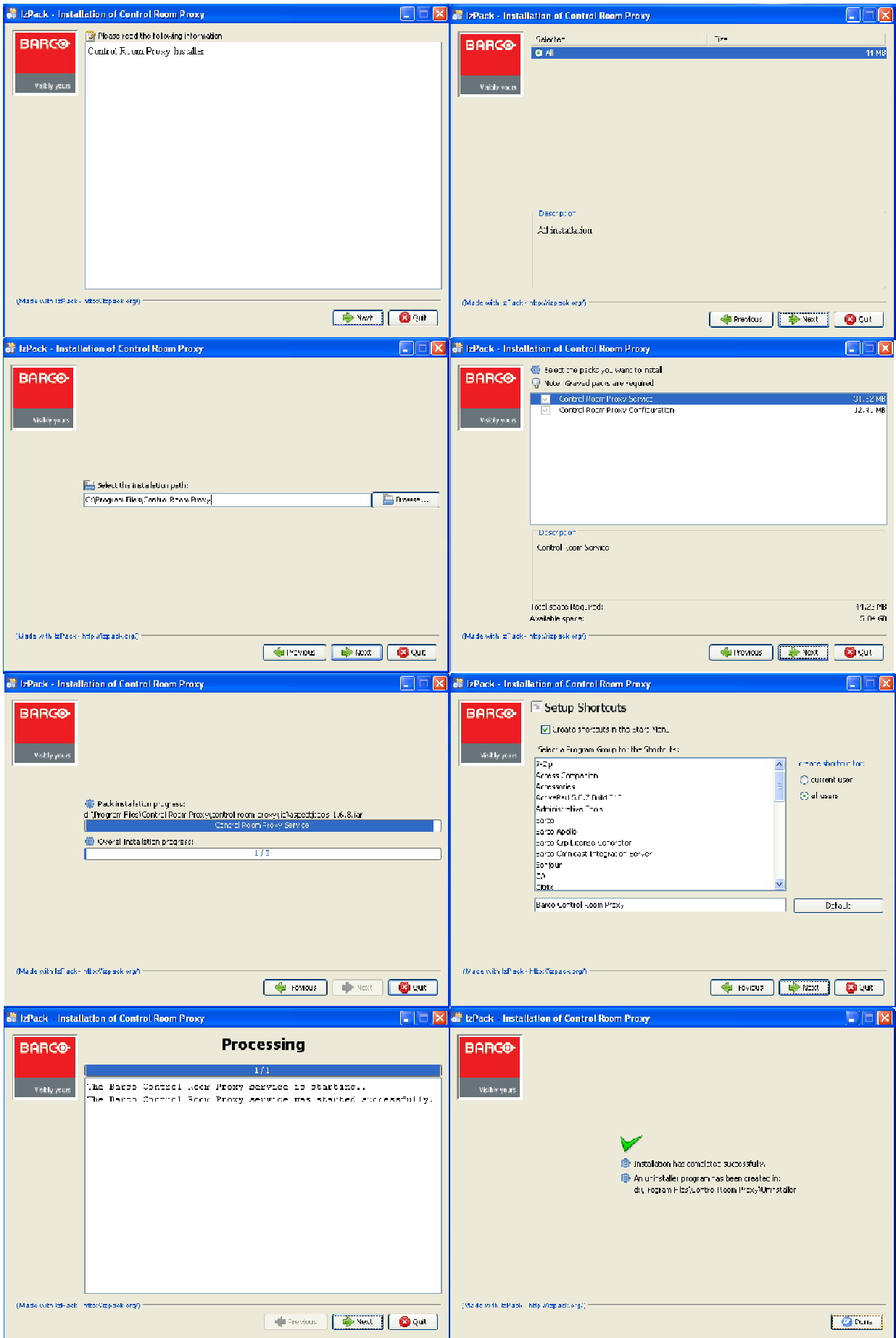
Java6 or above

Barco Wall Management Software CMS 2.x or above

Barco Wall Control Manager (BCM-Server)

3.2 Installing the CRP(Control Room Proxy)

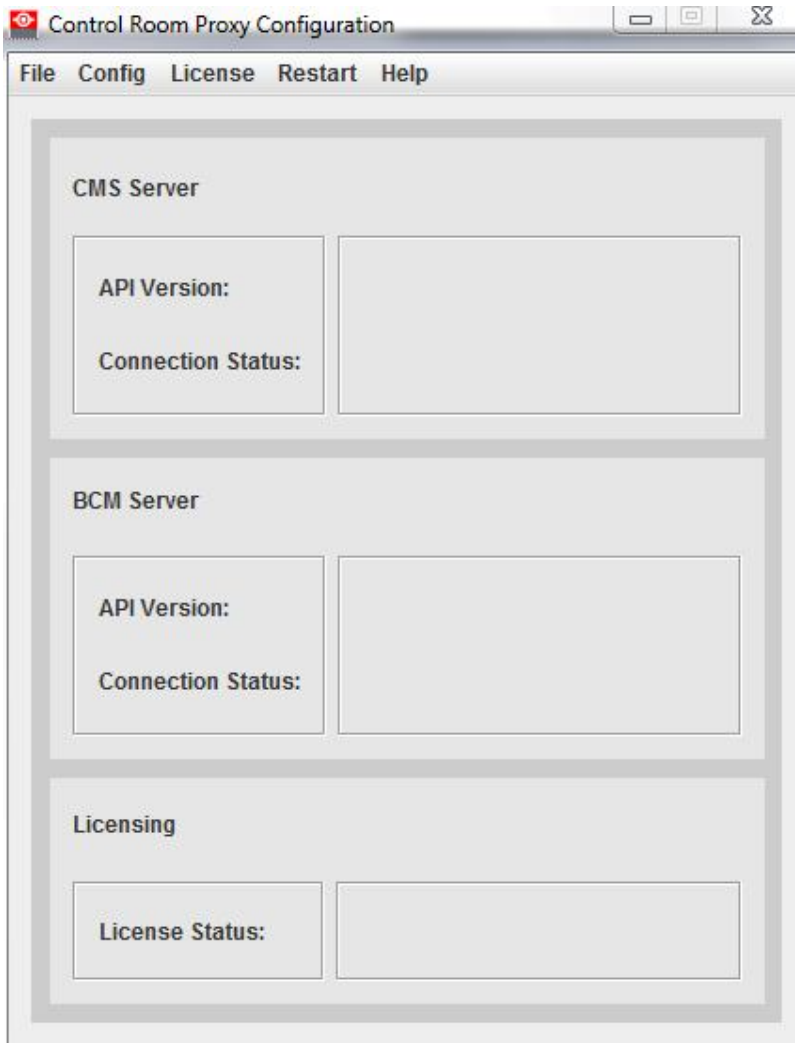
CRP application comes with setup.exe. Click on the setup.exe and follow the installation steps.



4 Configuring the CRP Application GUI

Start the CRP configuration interface from "start | All Programs | Barco Control Room Proxy | Barco Control Room Proxy Configuration".

It will open the configuration interface like as below image:-



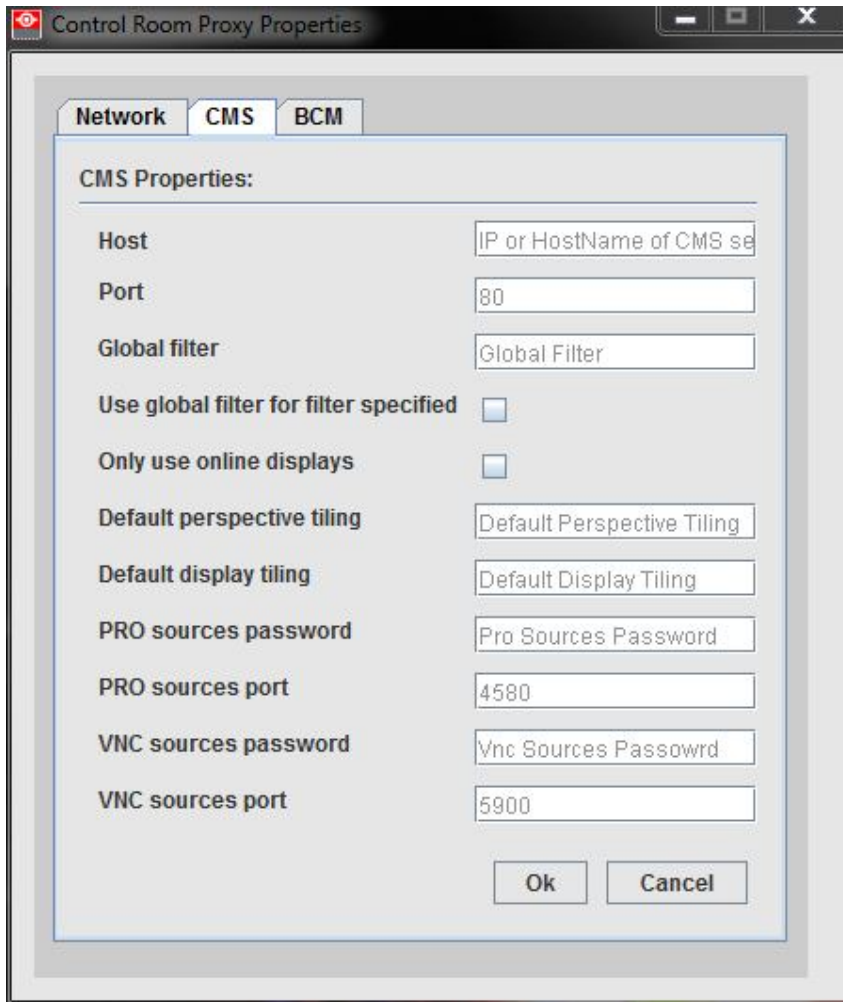
4.1 File Menu description:

It contains the "Exit" submenu to quit the application.

4.2 Config Menu description:

The configuration interface has three tabs:-

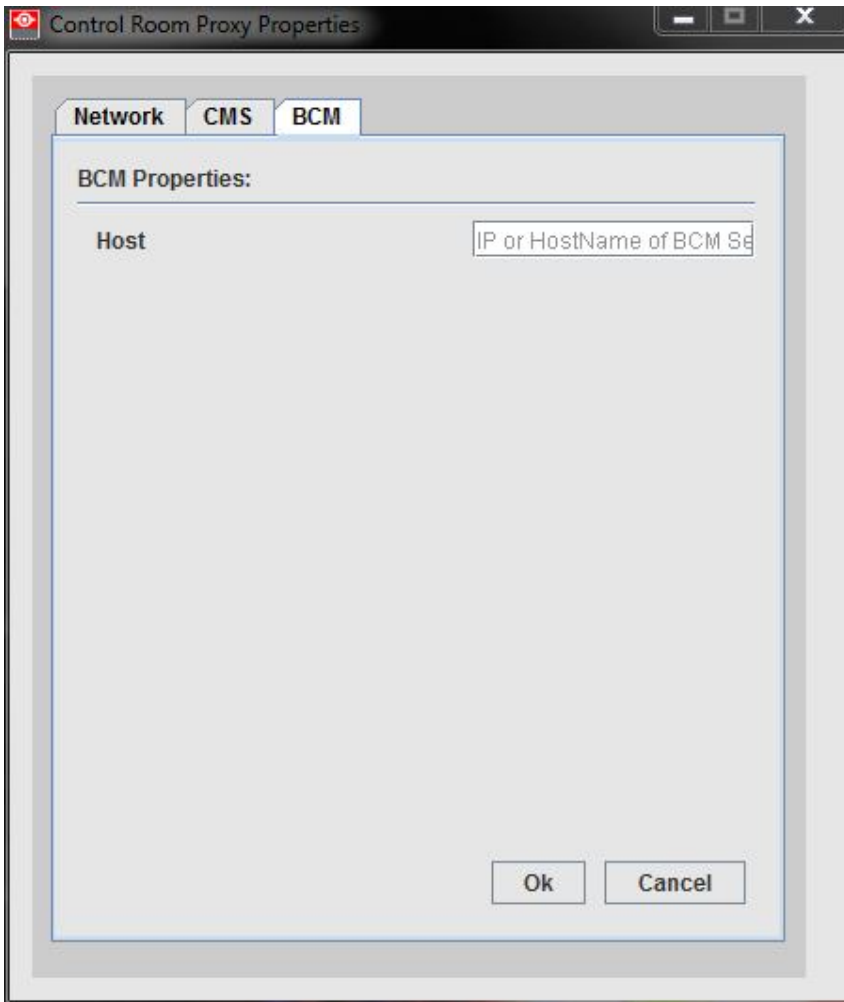
4.1.1 CMS:



- **Host:** Host name or ipaddress of cms server.
- **Port:** Port number to connect to cms server.
- **Global Filter:** When using the CRP to control cms software the global filter can be set so that only displays, source and perspectives which have the filter text within their name will be returned on the command calls. This should always be empty if you want to make sure you always receive a complete list.
- **Use Global filter when filter is specified:** Check if you want to use the filter specified by property above.
- **Only use online displays:** If selected, returns only online displays list for command. This property is typically used by "GetDisplayList" and "_GetDisplayListLong" command.
- **Default Perspective Tiling:** Default tiling will be applied on the perspective to load the source, if no tiling is applied on the perspective. This property is typically used by "LoadSourceOnPerspectiveCommand" command that will use the default perspective tiling for loading the source, if no tiling has been applied on the perspective.

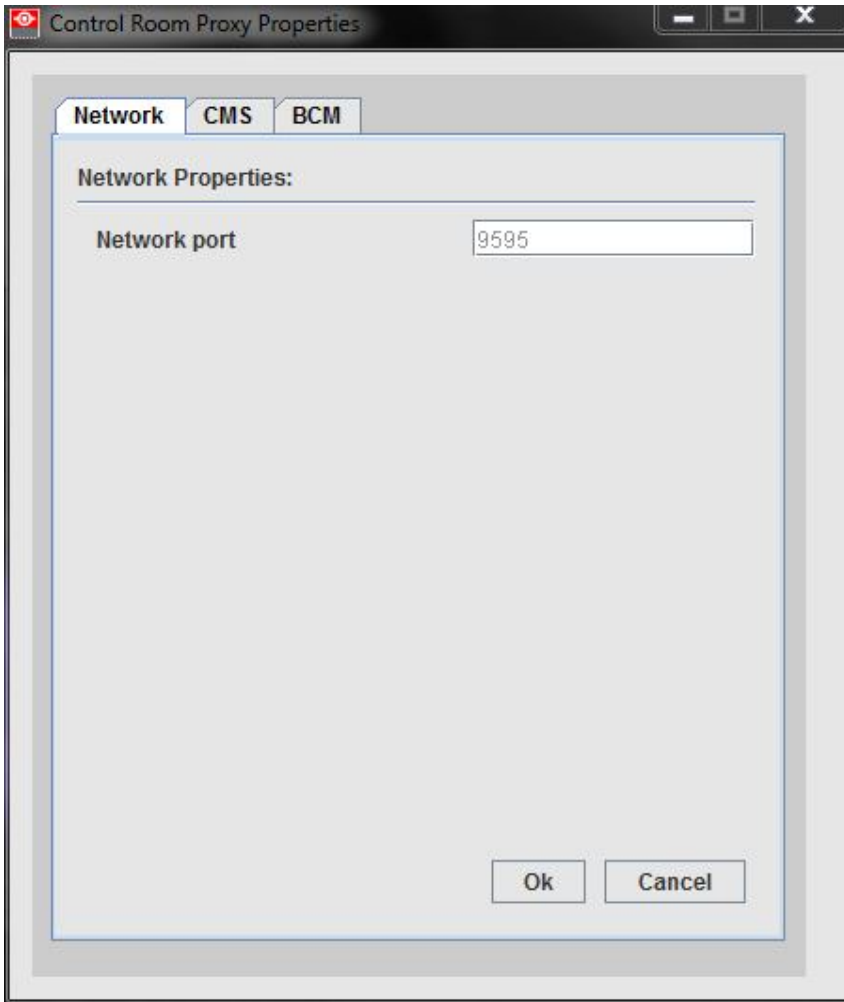
- **Default Display Tiling:** Default tiling will be applied on the display to share the perspective, if no tiling is applied on the display.
This property is typically used by "LoadPerspectiveOnDisplayCommand" command that will use the default display tiling for sharing the perspective, if no tiling has been applied on the display.
- **PRO Sources Password:** Password for PRO Server.
This property is typically used by " CreateScreenSource " command. Please see the command syntax for more details.
- **PRO Sources Port:** Port for PRO video sources. Default is 4950.
This property is typically used by " CreateScreenSource " command. Please see the command syntax for more details.
- **VNC Sources Password:** Password for VNC Server.
This property is typically used by " CreateScreenSource " command. Please see the command syntax for more details.
- **VNC Sources Port:** Port for VNC video sources. Default is 5900.
This property is typically used by " CreateScreenSource " command. Please see the command syntax for more details.

4.1.2 BCM:



- **Host:** Host name or ipaddress of bcm server.

4.1.3 Network:



- **TCP Port:** Type the TCP-IP port on which the CRP will listen for network connections from the Touchpanel system.

4.3 License Menu description:

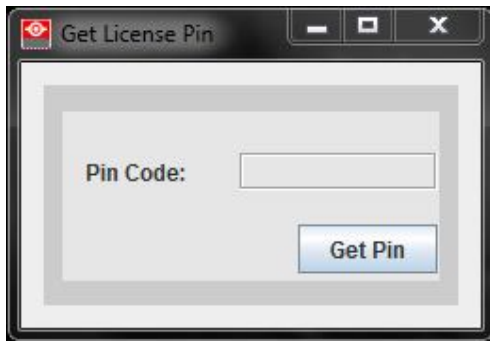
A license is required to use the CRP software. Barco will provide the license file for the software based on the two inputs provided by the client:-

License Key:- A key can be obtained from the CRP installation CD.

License Pin:- A pin can be obtained from CRP by clicking on "License Pin" submenu. See section 4.2.1

4.2.1 License Pin:

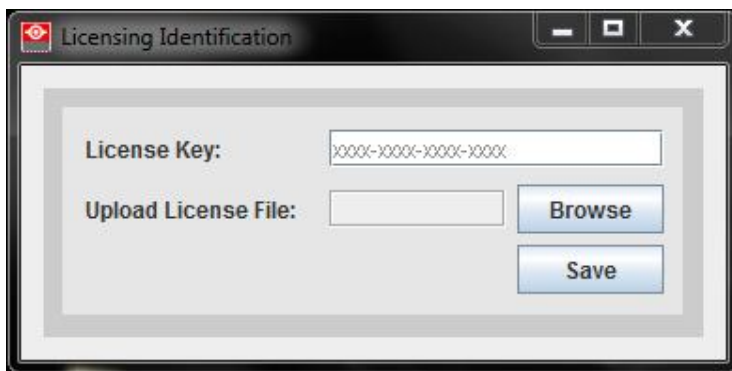
Click on "License Pin" submenu of the "License" menu. It will open the below GUI.



Click on "Get Pin" to obtain the pin code to be used by barco to generate the license file.

4.2.2 License Identification:

This tab is to provide the license information for the CRP. Client has to provide the license key and a valid license file from here.



Note: For every installation of CRP, client has to provide the license-key and license file.

4.4 Restart Menu description:

Any property change from "Config" tab or providing license information from the "License Identification" tab requires restart of service to apply the effect of changes for CRP.

Click on "Restart Service" submenu from the "Restart" menu to restart the CRP service.

4.5 Help Menu description:

4.4.1 *Help Contents:*

This submenu will open the help file.

4.4.2 *About Control Room Proxy:*

This submenu will open the dialog box stating the basic information about CRP.

5 Supported Commands

The CRP support two command types.

- 1) Basic Command Set
- 2) Simple Text Command Set

Both Basic and Simple Text can be used.

Note: Commands are case-insensitive however the arguments are case-sensitive.

5.1 Basic Command Set

Command format: Text Parameters Terminator

Sample: WallOn 0x0D

Command Format: Command + Parameter + EndDelimiter one of 0x0D of 0x0A

Command Reply Format:

Head: <

Body: Command 0x0D

Body: Parameter separated by 0x0D

Tail: >

Sample: < WallOn 0x0D OK 0x0D >

The following commands are supported:

For the CMS Layout Management:

GetDesktopList	Returns list of cms displays.
GetWindowList	Returns a list of windows with geometries being displayed
GetSelectedDesktop	Returns the currently selected desktop name.
SetSelectedDesktop	Set the selected display for other commands like "GetLayoutList","LoadLayout"
etc	
GetLayoutList	Returns a list of cms layouts
GetActiveLayoutList	Returns a list of currently loaded cms layouts
LoadLayout aLayoutName	Loads the given cms layout
UnloadLayout aLayoutName	Unloads the give cms layout
GetRegionList	Return a list of cms perspectives.

For the Projector Control(BCM):

WallOn	Turn the entire wall on
WallOff	Turn the entire wall off
ProjectorOn aProjectorNr	Turns a single projector with Nr. On
ProjectorOff aProjectorNr	Turns a single projector with Nr. Off
NumberOfProjectors	Returns the number of configured projectors
ProjectorRows	Returns the number of rows in the wall configuration
ProjectorColumns	Returns the number of columns in the wall configuration
GetPreferredSourceList	Returns a list of currently selected projector inputs.
ProjectorSetSource1Preferred aNr.	Switched the projector with Nr. To Input 1
ProjectorSetSource2Preferred aNr.	Switched the projector with Nr. To Input 2
WallSetSource1Preferred	Switches the entire wall to Input 1
WallSetSource2Preferred	Switches the entire wall to Input 2
NumberOfWalls	When the BCM-Server controls more than 1 wall this returns

	the number of Wall being controlled.
SetSelectedWall aWallID	If the BCM-Server is controlling more than 1 wall this allows You to select which wall to control. aWallID=1,2,3... etc.
GetSelectedWall	Return the number of the selected Wall. Default is 1.
GetLampRuntimes*	Returns a list of LampRuntimes. Not supported.
GetLampSerialNumbers*	Returns a list of Lamp Serial Numbers currently installed. Not supported.
GetWallStatus	Returns the operational status of each cube within the wall
GetProjectorStatus aProjectorNr	Returns the operational status of the given projector
GetLampPowerStatus*	Returns the On/Off lamp status of each projector. Not supported.
UpdateCrpBcmWall	To update the CRP with the current bcm wall
BcmHelp	Return the above list.

CMS:

Command	Parameter	Description
GetWindowList	none	Returns a text list of windows being displayed on the display wall with its geometries. The list consists of Windowtype, Windowid, X, Y, Width, Height, Zorder. Ex: Call: GetWindowList 0x0D Reply: <GetWindowList PerspectiveWindow,9,720,450,720,450,1 PerspectiveWindow,3,0,0,720,450,2 0x0D>
GetSelectedDesktop	none	On installations where multiple walls are configured. Commands like getLayoutlist or loadlayout are relevant per display. This allows you to recall which display or desktop will be used when executing those commands. Ex: Call: GetSelectedDesktop 0x0D Reply: <GetSelectedDesktop Display 0x0D>
SetSelectedDesktop	aDisplayName	Sets the display to be used when multiple displays are present in a CMS configuration. Ex: Call: SetSelectedDesktop Display 0x0D Reply: <SetSelectedDesktop OK 0x0D>
GetLayoutList	none	Returns a text list of configured layouts. Ex: Call: GetLayoutList 0x0D Reply: <GetLayoutList layout yg blank 0x0D>
GetActiveLayoutLi	none	Returns a text list of currently loaded/active layouts.

st

Ex:

Call: GetActiveLayoutList 0x0D

Return: < getActiveLayoutList layout 0x0D >

LoadLayout

aLayoutName

Loads the given layout. aLayoutName is the text layout name as returned from the getlayoutlist.

Ex:

Call: LoadLayout yg 0x0D

Replay: <LoadLayout OK 0x0D>

UnloadLayout

aLayoutName

Unloads the layout with the give name from the display.
aLayoutName is the name returned from GetActiveLayoutList.

Ex:

Call: UnLoadLayout yg

Reply: <UnLoadLayout OK 0x0D>

GetRegionList

none

Returns a list of Perspectives.

Ex:

Call: GetRegionsList 0x0D

Reply: <GetRegionList yahoo gmail test 0x0D>

GetDesktopList

none

Returns a list of cms displays.

Ex:

Call: GetDesktopList 0x0D

Reply: <GetDesktopList Display Operator [NOICLT22815] 0x0D>

BCM:

Command	Parameter	Description
WallOn	none	Turns the entire display wall on Ex: Call: WallOn 0x0D Reply: < WallOn STATE_REQUEST_DONE 0x0D> WallOn reply will consists of all the status of projectors from the row to columns. If the request gets completed, the CRP will reply with STATE_REQUEST_DONE string.
WallOff	none	Turns the entire display wall off Ex: Call: WallOff0x0D Reply: < WallOff STATE_REQUEST_DONE 0x0D> WallOff reply will consists of all the status of projectors from the row to columns. If the request gets completed, the CRP will reply with STATE_REQUEST_DONE string.

ProjectorOn aProjectorNumber Turns the projector with the give number on.
aProjectorNumber is the cube number. 1---32
Ex:
Call: ProjectorOn 1 0x0D
Reply: < ProjectorOn STATE_REQUEST_DONE 0x0D>
Alternative Reply: <ProjectorOn STATE_IN_PROGRESS 0x0D>
If the request gets completed, the CRP will reply with
STATE_REQUEST_DONE string.

ProjectorOff aProjectorNumber Turn the projector with the give number off
Ex:
Call: ProjectorOff 1 0x0D
Reply: < ProjectorOff STATE_REQUEST_DONE 0x0D>
If the request gets completed, the CRP will reply with
STATE_REQUEST_DONE string.

NumberOfProjectors none Returns the number of projectors within the configured wall.
Ex:
Call: NumberOfProjectors 0x0D
Reply: <NumberofProjectors 6 0x0D>

ProjectorRows none Returns the number of rows the display wall has.
Ex:
Call: ProjectorRows 0x0D
Reply: < getProjectorRows 0x0D 2 0x0D>

ProjectorColumns none Returns the number of columns the display wall has.
Ex:
Call: ProjectorColumns 0x0D
Reply: <ProjectorColumns 3 0x0D>

GetPreferredSourceList none Returns the currently active and selected Input as comma separated values
for each projector.
Values can be INPUT_NONE, INPUT_1, INPUT_2
Ex of the 2x2 display wall consisting of 4 cubes.
Call: GetPreferredSourceList 0x0D
Reply: < GetPreferredSourceList INPUT_NONE,INPUT_2
INPUT_NONE,INPUT_1 INPUT_1,INPUT_1 INPUT_1,INPUT_1 0x0D>

**ProjectorSetSource1P
referred** aProjectorNumber Sets the input of the give projector to input number 1
Ex:
Call: ProjectorSetSource1Preferred 6 0x0D
Reply: <ProjectorSetSource1Preferred 0x0D OK 0x0D>
Selected input number 1 of cube number 6

**ProjectorSetSource2P
referred** aProjectorNumber Sets the input of the give projector to input number 2
Ex:

Call: ProjectorSetSource2Preferred 6 0x0D

Reply: <ProjectorSetSource2Preferred 0x0D OK 0x0D>

Selected input number 2 of cube number 6

Command	Parameter	Description
WallSetSource1Preferred red	none	Switches all the projectors to input number 1 Ex: Call: WallSetSource1Preferred 0x0D Reply: < WallSetSource1Preferred STATE_REQUEST_DONE STATE_REQUEST_DONE STATE_REQUEST_DONE 0x0D> WallSetSource1Preferred reply will consists of all the status of projectors from the row to columns. If the request gets completed for an projector, the projector will have STATE_REQUEST_DONE status.
WallSetSource2Preferred red	none	Switches all the projectors to input number 2 Ex: Call: WallSetSource2Preferred 0x0D Reply: < WallSetSource2Preferred STATE_REQUEST_DONE STATE_REQUEST_DONE STATE_REQUEST_DONE 0x0D> WallSetSource2Preferred reply will consists of all the status of projectors from the row to columns. If the request gets completed for an projector, the projector will have STATE_REQUEST_DONE status.
NumberOfWalls	none	In installation where the BCM-Server is controlling multiple Walls this returns the number of walls configured within the BCM. Ex.: Call: NumberOfWalls 0x0D Reply: < NumberOfWalls 0x0D 1 0x0D> Note: Typically 1 BCM-Server only controls a single wall. Usually each display wall has its own BCM-Server.
SetSelectedWall	aWallNumber	When multiple wall are controlled from 1 BCM-Server this allows you to selected which Wall should be used when applying calls like WallOn, getWallStatus etc. Ex: Call: SetSelectedWall 1 0x0D Reply: < SetSelectedWall 0x0D OK 0x0D>
GetSelectedWall	none	Returns the wall be used when multiple walls are configured within 1 BCM- Server. Typically this will always be 1 Ex:

Call: GetSelectedWall 0x0D

Reply: < GetSelectedWall 0x0D 1 0x0D>

GetLampRuntimes*	none	Returns the Lamp runtimes of all the lamps within the Wall. Not Supported
GetLampSerialNumbers*	none	Returns the Lamp Serial numbers of all the lamps within the Wall. Not Supported
UpdateCrpBcmWall	none	Whenever the new wall is configured in BCM or the existing wall is updated, "UpdateCrpBcmWall" command needs to be sent to Control Room Proxy. Ex: Call: UpdateCrpBcmWall 0x0D Reply: < UpdateCrpBcmWall 0x0D OK 0x0D>

Command	Parameter	Description
GetWallStatus	none	Returns the status as comma separated values of each cube within the wall where the first attribute gives the operational status of the cube, second attribute gives the connection state of the cube, third attribute gives the health status of the cube, fourth attribute gives the used led hours and fifth attribute gives the projector hours. 1st attribute- Operational status Possible values: "OPERATIONSTATE_ON", "OPERATIONSTATE_IDLE" 2nd attribute- Connection status Possible values: "CONNECTIONSTATE_OK", "CONNECTIONSTATE_NOT_RESPONDING" 3rd attribute- Health status Possible values: 'HEALTHSTATE_OK', 'HEALTHSTATE_WARNING', 'HEALTHSTATE_ERROR' 4th attribute- Runtime led hours 5th attribute- Runtime projection unit hours Ex: Call: GetWallStatus Reply: < GetWallStatus OPERATIONSTATE_ON, CONNECTIONSTATE_OK, HEALTHSTATE_OK, 1237198, 5819 0x0D>
GetProjectorStatus	aProjectorNumber	Returns the status as comma separated values of a projector with the given number where the first attribute gives the operational status of the cube, second attribute gives the connection state of the cube, third attribute gives the health status of the cube, fourth attribute gives the used led hours and fifth attribute gives the projector hours. 1st attribute- Operational status

Possible values:

“OPERATIONSTATE_ON”,“OPERATIONSTATE_IDLE”

2nd attribute- Connection status

Possible values:

“CONNECTIONSTATE_OK”,“CONNECTIONSTATE_NOT_RESPONDING”

3rd attribute- Health status

Possible values:

'HEALTHSTATE_OK','HEALTHSTATE_WARNING','HEALTHSTATE_ERROR'

4th attribute- Runtime led hours

5th attribute- Runtime projection unit hoursEx:

Call: GetProjectorStatus 2 0x0D

Reply: < GetProjectorStatus

OPERATIONSTATE_ON,CONNECTIONSTATE_OK,

HEALTHSTATE_OK,5923,6527 0x0D>

GetLampPowerStatus none

Return the lamp On Off status for each cube

*

Not Supported

Note : If the BCM projector command replies with the "STATE_IN_PROGRESS" status, then this means that BCM is under processing of the command.

5.2 Simple Text Command Set

The Simple Text Command Set follows a simple protocol structure consisting of HostID, Class, Objects and Attributes where each indicates:

- HostID: Identifies the initiator of the request or respond. Typically consisting of Username and PC-Name from which the command or reply was initiated.
- Class: One of BCM for Projector control, CMS for CMS-Control
- Object: The object specifies that command or response of a command related to the Class
- Attributes: Attributes are parameters passed towards the a Class or reply of a read or Set command.

General format:

<I:HostID || K:Class || O:Object || A1:Attrib1 || Attrib2 ||A3:.....||>

Responses that consists of a list or array, Ex. myName1,myName2.. are returned with each entry having its own Attribute.

The following commands are supported:

For the CMS Layout Management:

GetDisplayList	Returns a list of CMS desktops
GetDisplayListLong	Returns a list of CMS desktops
GetPerspectiveList	Return a list of CMS-Perspectives
GetSharedPerspectiveListLong	Return a list of CMS-Perspectives
LoadPerspective	Loads the give CMS-Perspective onto a CMS-Desktop
CreatePerspective	Create a CMS-Perspective
UnloadPerspective	Unloads the give CMS-Perspective from a CMS-Desktop
UnloadAllPerspectives	Unloads all Perspective from the give CMS-Desktop
GetSourceList	Returns the configured CMS-Sources
CreateWebSource	Creates a CMS-Web source
CreateVideoSource	Creates a CMS-Video source
CreateScreenSource	Creates a CMS-VNC, Pro or SCN source
CreateExtendedProScreenSource	Creates a CMS PRO type of source with extended parameters.
CreateExtendedVncScreenSource	Creates a CMS VNC type of source with extended parameters.
CreateExtendedScnScreenSource	Creates a CMS SCN type of source with extended parameters.
GetDispletList	Returns a list of Source displayed within a CMS-Perspective
LoadSourceOnPerspective	Loads the give CMS-Source within a CMS-Perspective
LoadVideoSourceOnPerspectiveWithMgs	Loads the given CMS-Source within a CMS-Perspective with MGS server.
RunCmsMgsSyncUtility	Synchronize the mgs workers and cms mgs sources created with "LoadVideoSourceOnPerspectiveWithMgs" command.
RemoveSourceFromPerspective	Removes the give CMS-Source from a CMS-Perspective
RemoveSourceFromPerspectiveTile	Remove the source from a given tile of a perspective.
ApplyDecoratorToASourcePerspective	Applies a decorator to a source within a perspective.
RemoveDecoratorFromASourcePerspective	Remove a decorator from a source within a perspective.
GetPerspectiveAppliedTile	Returns the applied tile name of a given perspective
GetDisplayAppliedTile	Returns the applied tile name of a given display
UpdateDisplayTile	Updates the display with the given tile name
UpdatePerspectiveTile	Updates the perspective with the given tile name
CreateAndApplyTileOnDisplay	Creates a tile on a display with the given row and columns and then will apply that tiling on that display
CreateAndApplyTileOnPerspective	Creates a tile on a perspective with the given row and columns and then will apply that tiling on that perspective
GetTileList	Returns a list of CMS-Tiles

GetWindowList	Returns a list of Windows display on the cms display with Geometries
GetLayoutList	Returns a list of cms layouts
GetActiveLayoutList	Returns a list of currently loaded cms layouts
LoadLayout aLayoutName	Loads the give cms Layout
UnloadLayout aLayoutName	Unloads the give cms Layout

For the Projector Control:

WallOn	Turn the entire wall on
WallOff	Turn the entire wall off
ProjectorOn aProjectorNr	Turns a single projector with Nr. On
ProjectorOff aProjectorNr	Turns a single projector with Nr. Off
NumberOfProjectors	Returns the number of configured projectors
ProjectorRows	Returns the number of rows in the wall configuration
ProjectorColumns	Returns the number of columns in the wall configuration
GetPreferredSourceList	Returns a list of currently selected projector inputs.
ProjectorSetSource1Preferred aNr.	Switched the projector with Nr. To Input 1
ProjectorSetSource2Preferred aNr.	Switched the projector with Nr. To Input 2
WallSetSource1Preferred	Switches the entire wall to Input 1
WallSetSource2Preferred	Switches the entire wall to Input 2
NumberOfWalls	When the BCM-Server controls more than 1 wall this returns the number of Wall being controlled.
SetSelectedWall aWallID	If the BCM-Server is controlling more than 1 wall this allows you to select which wall to control. aWallID=1,2,3... etc.
GetSelectedWall	Return the number of the selected Wall. Default is 1.
GetLampRuntimes*	Returns a list of LampRuntimes.
GetLampSerialNumbers*	Returns a list of LampSerialNumbers currently installed.
GetWallStatus	Returns the operational status of each cube within the wall
GetProjectorStatus aProjectorNr	Returns the operational status of the given projector
GetLampPowerStatus	Returns the On/Off lamp status of each projector
UpdateCrpBcmWall	To update the CRP with the current bcm wall
UpdateBrightnessControl	Updates the brightness control parameters of the wall.
UpdateBrightnessMode	Updates the brightness mode of the wall.
RenewColorAdjustment	Renews the color adjustment. Need to call after UpdateBrightnessControl command.
GetBrightnessList	Returns the brightness parameters of the wall.
GetWallHealthSummary	Returns the wall health.
GetWallAlarms	Returns the alarms of a wall if any exist.

**Commands not supported for BCM-OL/OVL.*

CMS:

<u>GetDisplayList</u>		Description
Class	CMS	Applies to CMS.
Object	GetDisplayList	Returns a list of CMS-Display Agents.
Attrib 1	aFilterText (Optional)	A Filter text. Only display names that have aFilterText within their names will be returned. Default: If no filter is specified all the displays are returned.

Example:

Request <l:BARCO||K:CMS||O:GetDisplayList||>

Reply <l:BARCO@noict22815||K:CMS||O:REGetDisplayList||A0:Display||>

Request <l:BARCO||K:CMS||O:GetDisplayList||A1:Disp*||>

Reply <l:BARCO@noict22815||K:CMS||O:REGetDisplayList||A0:Display||>

Request <l:BARCO||K:CMS||O:GetDisplayList||A1:test||>

Reply <l:BARCO@noict22815||K:CMS||O:REGetDisplayList||>

Remark The reply consists of a list. Each display will be listed as a separate Attribute.

Ex: If no displays are present then no Attribute will be returned.

If multiple displays exist then each will have its own Attribute

A1:aDisplay1Name1||A2:aDisplay2Name||.

Filter string Disp* will return Display, Display etc otherwise filter string will try to make exact match.

<u>GetDisplayListLong</u>		Description
Class	CMS	Applies to CMS .
Object	GetDisplayListLong	Returns a list of CMS-Display Agents.
Attrib 1	aFilterText (Optional)	A Filter text. Only display names that have aFilterText within their names will be returned. Default: If no filter is specified all the displays are returned.

Example:

Request <l:BARCO||K:CMS||O:GetDisplayListLong||>

Reply <l:BARCO@noict22815||K:CMS||O:REGetDisplayListLong||A0:Display@1440x900||>

Remark The reply consists of a list of Attributes. Each Attribute has the DisplayName and Display resolution listed and follows the format

DisplayName@DisplayWidth x DisplayHeight

GetWindowList		Description
Class	Cms	Applies to Cms
Object	GetWindowList	Returns a list window Windows being displayed on the give Cms Display.
Attrib 1	aDisplayName (Optional)	Cms Display. If no display agent is specified, it will use the display set from " SetSelectedDesktop " command.
Attrib 2	aFilterText (Optional requires Attrib 1)	A Filter text. Only window that have aFilterText within their tags will be returned. Default: If no filter is specified all the windows of the displays are returned.

Example:

Request <l:BARCO||K:CMS||O:GetWindowList||A1:Display||>

<l:BARCO||K:CMS||O:GetWindowList||>

Reply <l:BARCO@noiclt22815||K:CMS||O:REGetWindowList||A0:PerspectiveWindow,3,309,740,400,400,1||>

Remark The reply consists of a list if windows each as a separate Attribute and consist of Windowtype, Windowid, X, Y, Width, Height, Zorder.
Filter will search for filter string in window tags.

GetLayoutList		Description
Class	CMS	Applies to CMS.
Object	GetLayoutList	Returns a list of layout names associated with the given display name.
Attrib 1	aDisplayName (Optional)	The CMS Display Agent Name. If no display agent is specified, it will use the display set from " SetSelectedDesktop " command.
Attrib 2	aFilterText (Optional requires Attrib 1)	A Filter text. Only layout names that have aFilterText within their names will be returned. Default: If no filter is specified all the layouts are returned.

Example:

Request <l:BARCO||K:CMS||O:GetLayoutList||A1:Display||>

<l:BARCO||K:CMS||O:GetLayoutList||>

Reply <l:BARCO@noiclt22815||K:CMS||O:REGetLayoutList||A0:layout||>

Request <l:BARCO||K:CMS||O:GetLayoutList||A1:Display||A2:lay*||>
 Reply <l:BARCO@noiclt22815||K:CMS||O:REGetLayoutList||A0:layout||>

<u>GetActiveLayoutList</u>		Description
Class	CMS	Applies to CMS.
Object	GetActiveLayoutList	Returns a list of Active Loaded Layouts
Attrib 1	aDisplayName (Optional)	CMS Display Agent Name. If no display agent is specified, it will use the display set from "SetSelectedDesktop" command.
Attrib 2	aFilterText (Optional requires Attrib 1)	A Filter text. Only layout names that have aFilterText within their names will be returned. Default: If no filter is specified all the loaded layouts are returned.

Example:

Request <l:BARCO||K:CMS||O:GetActiveLayoutList||A1:Display||>
 <l:BARCO||K:CMS||O:GetActiveLayoutList||>
 Reply <l:BARCO@noiclt22815||K:CMS||O:REGetActiveLayoutList||A0:layout||>

Remark Active layouts are checked against the window position. Thus if a layout is loaded and the windows are more. Then this loaded layout is no longer considered as being active.

<u>LoadLayout</u>		Description
Class	CMS	Applies to CMS .
Object	LoadLayout	Loads the give layout onto the selected display.
Attrib 1	aLayoutName	Specifies the layout name to be loaded.
Attrib 2	aDisplayName (Optional)	The display onto which the layout should be loaded.. If no display agent is specified, it will use the display set from "SetSelectedDesktop" command.

Example:

Request <l:BARCO||K:CMS||O:LoadLayout||A1:layout||A2:Display||>
 <l:BARCO||K:CMS||O:LoadLayout||A1:layout||>
 Reply <l:BARCO@noiclt22815||K:CMS||O:RELoadLayout||A1:OK||>
 Alternative <l:BARCO@noiclt22815||K:CMS||O:RELoadLayout||NOK:'layout' layout not found on display 'Display'.||>
 Reply <l:BARCO@noiclt22815||K:CMS||O:RELoadLayout||NOK:'Display' display not found.||>

<u>UnLoadLayout</u>	Description
---------------------	-------------

Class	CMS	Applies to CMS .
Object	LayoutUnload	Unloads the given layout from the display
Attrib 1	aLayoutName	The Layout name to unload.
Attrib 2	aDisplayName (Optional)	The display from which the layout should be unloaded. If no display agent is specified, it will use the display set from " SetSelectedDesktop " command.

Example:

Request <l:BARCO||K:CMS||O:UnloadLayout||A1:layout||>
Reply <l:BARCO@noiclt22815||K:CMS||O:REUnloadLayout||A1:OK||>

Request <l:BARCO||K:CMS||O:UnloadLayout||A1:layout||A2:Display||>
Reply <l:BARCO@noiclt22815||K:CMS||O:REUnloadLayout||A1:OK||>

Alternative <l:BARCO@noiclt22815||K:CMS||O:REUnloadLayout||NOK:'layout' layout is not active on display 'Display'.||>
Reply

GetPerspectiveList

Description

Class	CMS	Applies to CMS
Object	GetPerspectiveList	Returns a list of configured CMS-Perspectives.
Attrib 1	aFilterText (Optional)	Only Perspectives with aFilterText in their names, tags or description will be returned.

Example:

Request <l:BARCO||K:CMS||O:getPerspectiveList||>
Reply <l:BARCO@noiclt22815||K:CMS||O:REgetPerspectiveList||A0:googleP||A1:yahooP||A2:gmail||A3:test||>

Request <l:BARCO||K:CMS||O:getPerspectiveList||A1:goo*||>
Reply <l:BARCO@noiclt22815||K:CMS||O:REgetPerspectiveList||A0:googleP||>

Alternative <l:BARCO@noiclt22815||K:CMS||O:REgetPerspectiveList||>
Reply

GetSharedPerspectiveList

Description

Class	CMS	Applies only to CMS
Object	GetSharedPerspectiveList	Returns a list of loaded perspectives on the give CMSdisplay.
Attrib 1	aDisplayName (Optional)	The CMS-Display name. If no display agent is specified, it will use the display set from " SetSelectedDesktop " command.
Attrib 2	aFilterText (Optional requires Attrib 1)	Only Perspectives with aFilterText in their names, tags or description will be returned.

Example:

Request <l:BARCO||K:CMS||O:getSharedPerspectiveList||A1:Display||>
 <l:BARCO||K:CMS||O:getSharedPerspectiveList||>

Reply <l:BARCO@noiclt22815||K:CMS||O:REgetSharedPerspectiveList||A0:gmail||>

Request <l:BARCO||K:CMS||O:getSharedPerspectiveList||A1:myUnknownDisplay||>
 Reply <l:BARCO@noiclt22815||K:CMS||O:REgetSharedPerspectiveList||NOK:'myUnknownDisplay' display not found.||>

Request <l:BARCO||K:CMS||O:getSharedPerspectiveList||A1:Display||A2:gma*||>
 Reply <l:BARCO@noiclt22815||K:CMS||O:REgetSharedPerspectiveList||A0:gmail||>

<u>GetSharedPerspectiveListLong</u>		Description
Class	CMS	Applies to CMS
Object	GetSharedPerspectiveListLong	Returns a list of loaded perspectives on the given CMSdisplay.
Attrib 1	aDisplayName (Optional)	The CMS-Display name. If no display agent is specified, it will use the display set from " SetSelectedDesktop " command.
Attrib 2	aFilterText (Optional requires Attrib 1)	Only Perspectives with aFilterText in their names, tags or description will be returned.

Example:

Request <l:BARCO||K:CMS||O:getSharedPerspectiveListLong||A1:Display||>
 <l:BARCO||K:CMS||O:getSharedPerspectiveListLong||>

Reply <l:BARCO@noiclt22815||K:CMS||O:REgetSharedPerspectiveListLong||A0:gmail,600x450||>

Request <l:BARCO||K:CMS||O:getSharedPerspectiveListLong||A1:Display||>
 Reply <l:BARCO@noiclt22815||K:CMS||O:REgetSharedPerspectiveListLong||A0:gmail,600x450,gmail,1,0,0,150,150||A1:yahoo,600x450,yahoo,1,0,0,300,225||>

Remarks For tiling,
 Attrib 1+n is the name of the Perspective including its Displets.
 Each Perspective has the Parameter:
 PerspectivName, widthxheight, Source1Name,Source1TileID,X,Y,Width,Height,
 Source2Name,Source2TileID,X,Y,Width,Height,

The number of SourceName's depends upon how many displets are loaded within the Perspective.

If the source is not displayed within a Tile or if no Tile is active then SourceTileId will be -1.

<u>LoadPerspective</u>	Description
------------------------	-------------

Class	CMS	Only Applies to CMS
Object	LoadPerspective	Loads a Perspective on the give display.
Attrib 1	aPerspectiveName	The name of the Perspective as returned from GetPerspectiveList.
Attrib 2	aDisplayName	The „CMS Display “ name onto which the Perspective should be loaded.
Attrib 3	TileID (Optional)	Alternatively to the Geometry the TileID can be used to specify the location where to load the perspective. If no tile and geometry is specified, 1st tile will be used.
	X (Optional requires Attrib 4,5&6)	Specifies the Left pixel position Default: 0.
Attrib 4	Y (Optional requires Attrib 3)	Specifies the top pixel position Default: 0.
Attrib 5	Width (Optional requires Attrib 3)	Specifies the Width Default: The value as specified/saved within the perspective.
Attrib 6	Height (Optional requires Attrib 3)	Specifies the Height Default: The value as specified/saved within the perspective.

Example:

Request <l:BARCO||K:CMS||O:LoadPerspective||A1:yahoo||A2:Display||>

Reply <l:BARCO@KARCLT0409||K:CMS||O:RELoadPerspective||A1:OK||>

Remark First tile will be used on a display for sharing the perspective in the above command.

Request <l:BARCO||K:CMS||O:LoadPerspective||A1:yahoo||A2:Display||A3:2||>

Reply <l:BARCO@noiclt22815||K:CMS||O:RELoadPerspective||A1:OK||>

Alternative <l:BARCO@noiclt22815||K:CMS||O:RELoadPerspective||NOK:'yaho' perspective not found.||>

Reply

Request <l:BARCO||K:CMS||O:LoadPerspective||A1:yahoo||A2:Display||A3:0||A4:0||A5:100||A6:100||>

Reply <l:BARCO@noiclt22815||K:CMS||O:RELoadPerspective||A1:OK||>

- Remarks
- 1) If the tileID and geometry is not specified, then the 1st tile will be used for loading the perspective on a display.
 - 2) If no tile is active, then the default display tile as configured within the config page will be applied to the display and used.

CreatePerspective		Description
Class	CMS	Applies only to CMS
Object	CreatePerspective	Creates a Perspective with the given name.
Attrib 1	aPerspectiveName	Perspective Name
Attrib 2	aFilterText (Optional requires Attrib 3&4)	Specifies the Filter text to be added to the Perspective which can then be used within the Sidebar or read requests like GetPerspectiveList.
Attrib 3	Width (requires Attrib2)	Default Perspective-Width in pixel.

Attrib 4 Height (requires Attrib2) Default Perspective-Height in pixel.

Example:

Request <l:BARCO||K:CMS||O:CreatePerspective||A1:Sample||>

Reply <l:BARCO@noiclt22815||K:CMS||O:RECreatePerspective||A1:OK||>

Request <l:BARCO||K:CMS||O:CreatePerspective||A1:Sample1||A2:myFilter||A3:200||A3:200||>

Reply <l:BARCO@noiclt22815||K:CMS||O:RECreatePerspective||A1:OK||>

Alternative <l:BARCO@noiclt22815||K:CMS||O:RECreatePerspective||NOK: 'Sample1' already exists||>

Reply

UnloadPerspective

Description

Class	CMS	Applies to CMS
Object	UnloadPerspective	Unloads the give Perspective Name from the display
Attrib 1	aPerspectiveName	Perspective name to unload
Attrib 2	aDisplayName (Optional)	The „CMS Display Agent“ Name. If no display agent is specified, it will use the display set from " SetSelectedDesktop " command.

Example:

Request <l:BARCO||K:CMS||O:UnloadPerspective||A1:yahoo||A2:Display||>

Reply <l:BARCO@noiclt22815||K:CMS||O:REUnloadPerspective||A1:OK||>

UnloadAllPerspectives

Description

Class	CMS	Applies to CMS
Object	UnloadAllPerspectives	Unloads all Perspectives from the specified display
Attrib 1	aDisplayName (Optional)	The „CMS Display Agent“ Name If no display agent is specified, it will use the display set from " SetSelectedDesktop " command.

Example:

Request <l:BARCO||K:CMS||O:UnloadAllPerspectives||A1:Display||>

Reply <l:BARCO@noiclt22815||K:CMS||O:REUnloadAllPerspectives||A1:OK||>

Request <l:BARCO||K:CMS||O:UnloadAllPerspectives||A1:myUnknownDisplay||>

Reply <l:BARCO@noiclt22815||K:CMS||O:REUnloadAllPerspectives||NOK:myUnknownDisplay display not found.||>

GetSourceList		Description
Class	CMS	Applies only to CMS
Object	GetSourceList	Returns a list of available CMS-Sources
Attrib 1	aFilterText (Optional)	Only source with aFilterText within their Name, Tag or Description will be returned.
Example:		
Request	<l:BARCO K:CMS O:GetSourceList A1:sc* >	
Reply	<l:BARCO@noiclt22815 K:CMS O:REGetSourceList A0:scn >	
Request	<l:BARCO K:CMS O:GetSourceList >	
Reply	<l:BARCO@noiclt22815 K:CMS O:REGetSourceList A0:DefaultProSource[NOICLT22815] A1:yahoo A2:scn A3:gm ail A4:raaga A5:hdfc A6:vnc A7:pro >	

CreateWebSource		Description
Class	CMS	Applies to CMS
Object	CreateWebSource	Create a new source of type Webpage
Attrib 1	aSourceName	The Name that this source will have.
Attrib 2	URL	The WEB-URL or Link
Attrib 3	aFilterText (Optional)	Specifies a filter Text which can be used by read operations or the Sidebar.
Example:		
Request	<l:BARCO K:CMS O:CreateWebSource A1:barco A2:http://www.barco.com >	
Reply	<l:BARCO@noiclt22815 K:CMS O:RECreateWebSource A1:OK >	
Request	<l:BARCO K:CMS O:CreateWebSource A1:barco A2:http://www.barco.com >	
Reply	<l:BARCO@noiclt22815 K:CMS O:RECreateWebSource A1:OK >	
Alternative	<l:BARCO@noiclt22815 K:CMS O:RECreateWebSource NOK:Source barco already exists >	
Reply		

CreateVideoSource		Description
Class	CMS	Applies to CMS
Object	CreateVideoSource	Create a new source of type Video
Attrib 1	aSourceName	The Name that this source will have.
Attrib 2	videoUrl	Video Url of the source.
Attrib 3	audioUrl(Optional requires 4&5)	Audio Url of the source.
Attrib 4	deviceUrl(Optional requires 3&5)	Device Url of the source.
Attrib 5	hostRdc(Optional requires 3&4)	Host rdc. (This type of source has to be pre exist in cms)

Attrib 6 aFilterText (Optional requires either attrib 1,2,3,4,5 or 1,2) Specifies a filter Text which can be used by read operations or the Sidebar

Example:

Request <l:BARCO||K:CMS||O:CreateVideoSource||A1:videoSource1||A2:LocalFile://C:\Windows\clock.avi||A3:video||>

Reply <l:BARCO@noiclt22815||K:CMS||O:RECreateVideoSource||A1:OK||>

Request <l:BARCO||K:CMS||O:CreateVideoSource||A1:Traffic1||A2:mms://ecc103.earthchannel.com/cobbcolivecam||A3:mms://ecc103.earthchannel.com/cobbcolivecam||A4: mms://ecc103.earthchannel.com/cobbcolivecam ||A5:DefaultProSource[NOICLT22815]||>

Reply <l:BARCO@noiclt22815||K:CMS||O:RECreateVideoSource||A1:OK||>

Alternative <l:BARCO@noiclt22815||K:CMS||O:RECreateVideoSource||NOK:Source Traffic1 already exists||>

Reply

CreateScreenSource		Description
Class	CMS	Applies only to CMS
Object	CreateScreenSource	Create a new source of type VNC or Pro
Attrib 1	aSourceName	The Name that this source will have.
Attrib 2	aPCHostnameOrIPAddress	IP-Address or PCName on which the VNC-Server or Pro-Server is running
Attrib 3	aType	PRO or VNC or SCN
Attrib 4	aPassword(Optional requires 5)	VNC or PRO-Server or SCN Server Password
Attrib 5	aPort(Optional requires 4)	VNC or PRO-Server or SCN Server Network Port.. VNC-Default=5900 Pro-Server-Default=4580 <u>Note</u> : For scn type of source, attrib 5 and 6 is required

Attrib 6 aFilterText (Optional) Specifies a filter Text which can be used by read operations or the Sidebar

Example:

Request <l:BARCO||K:CMS||O:CreateScreenSource||A1:myPC1||A2:192.168.0.10||A3:PRO||>

Reply <l:BARCO@noiclt22815||K:CMS||O:RECreateScreenSource||A1:OK||>

Request <l:BARCO||K:CMS||O:CreateScreenSource||A1:ss2||A2:192.168.0.10||A3:PRO||A4:sc||>

Reply <l:BARCO@noiclt22815||K:CMS||O:RECreateScreenSource||A1:OK||>

Alternative <l:BARCO@noiclt22815||K:CMS||O:RECreateScreenSource||NOK:Source ss2 already exists||>

Reply

Request <l:BARCO||K:CMS||O:CreateScreenSource||A1:ss3||A2:192.168.0.10||A3:VNC||A4:myPassword||A5:5900||>

Reply <l:BARCO@noiclt22815||K:CMS||O:RECreateScreenSource||A1:OK||>

Request <l:BARCO||K:CMS||O:CreateScreenSource||A1:ss4||A2:192.168.0.10||A3:SCN||A4:myPassword||A5:5900||A6:fill|>
 Reply <l:BARCO@noiclt22815||K:CMS||O:RECreateScreenSource||A1:OK|>

CreateExtendedProScreenSource		Description
Class	CMS	Applies only to CMS
Object	CreateExtendedProScreenSource	Create a new source of type Pro with extended attributes. This command requires 19 attributes some of which are optional depending upon certain attribute values. If the attribute is optional, it should be left as blank. See the below request as an example.
Attrib 1	Source name	Name of the source
Attrib 2	description	Descrption
Attrib 3	tags	Tags
Attrib 4	hostname	Host machine where the proserver is installed
Attrib 5	Password	Password of the host machine of attribute 4
Attrib 6	Port	Port of pro server to connect to. Generally pro server port is 4580
Attrib 7	Vds	Enable/Disable Virtual Desktop Mode. Type: boolean (true/false)
Attrib 8	Target type	Target type can be "Area", "Desktop", "Window", "DesktopIdent"
Attrib 9	Desktop indent(optional)	If the attribute 8 is of target type "DesktopIdent", then this attribute is required. Type: Integer
Attrib 10	x(optional)	If the attribute 8 is of target type "Area", then this attribute is required. Type: Integer
Attrib 11	y(optional)	If the attribute 8 is of target type "Area", then this attribute is required. Type: Integer
Attrib 12	width(optional)	If the attribute 8 is of target type "Area", then this attribute is required. Type: Integer
Attrib 13	height(optional)	If the attribute 8 is of target type "Area", then this attribute is required. Type: Integer
Attrib 14	Window title(optional)	If the attribute 8 is of target type "Window", then this attribute is required. Type: String.
Attrib 15	Manual update	Real time update mode. Type: boolean(true/false)
Attrib 16	Update interval(optional)	If the attribute 15 is "true", this value is required. Type: Integer
Attrib 17	width	Width of the source Type: Integer
Attrib 18	height	Height of the source Type: Integer
Attrib 19	timezone	Time zone of the source Type: String

Example:

Request <l:BARCO||K:CMS||O:CreateExtendedProScreenSource||A1:prosource||A2:prodescr||A3:pro||A4:noiclt22815||A5:pwd||A6:4580||A7:false||A8:Desktop||A9:||A10:||A11:||A12:||A13:||A14:||A15:false||A16:||A17:200||A18:200||A19:Asia/Calcutta|>
 >
 Reply <l:BARCO@noiclt22815||K:CMS||O:RECreateExtendedProScreenSource||A1:OK|>

CreateExtendedVncScreenSource		Description
Class	CMS	Applies only to CMS
Object	CreateExtendedVncScreen	Create a new source of type VNC with extended attributes. This command requires

Source

11 attributes some of which are optional depending upon certain attribute values.

If the attribute is optional, it should be left as blank. See the below request as an example.

Attrib 1	Source name	Name of the source
Attrib 2	description	Description
Attrib 3	tags	Tags
Attrib 4	hostname	Host machine where the proserver is installed
Attrib 5	Password	Password of the host machine of attribute 4
Attrib 6	Port	Port of pro server to connect to. Generally pro server port is 4580
Attrib 7	Manual update	Enable/Disable Virtual Desktop Mode. Type: boolean (true/false)
Attrib 8	Update interval(optional)	Target type can be "Area", "Desktop", "Window", "DesktopIdent"
Attrib 9	width	If the attribute 8 is of target type "DesktopIdent", then this attribute is required. Type: Integer
Attrib 10	height	If the attribute 8 is of target type "Area", then this attribute is required. Type: Integer
Attrib 11	timezone	If the attribute 8 is of target type "Area", then this attribute is required. Type: Integer

Example:

Request <l:BARCO||K:CMS||O:CreateExtendedVncScreenSource||A1:vnc||A2:vncdescr||A3:||A4:noiclt22815||A5:pwd||A6:5900||A7:false||A8:||A9:200||A10:200||A11:Asia/Calcutta||>

Reply <l:BARCO@noiclt22815||K:CMS||O:RECreateExtendedVncScreenSource||A1:OK||>

CreateExtendedScnScreenSource

Description

Class **CMS** Applies to CMS

Object **CreateExtendedScnScreenSource** Creates a scn type of screen source. This command requires 9 attributes

Attrib 1 **Source name** Name of the source

Attrib 2 **description** Description

Attrib 3 **tags** Tags

Attrib 4 **hostname** Host machine where the scn server is installed

Attrib 5 **Password** Password of the host machine of attribute 4

Attrib 6 **Port** Port of scn server to connect to. Generally scn server port is 5900

Attrib 7 **width** Width of the source
Type: Integer

Attrib 8 **height** Height of the source
Type: Integer

Attrib 9 **timezone** Time zone of the source
Type: String

Example:

Request <l:BARCO||K:CMS||O:CreateExtendedScnScreenSource||A1:scnsource||A2:scndescr||A3:scn||A4:noiclt22815||A5:scn||A6:5900||A7:200||A8:200||A9:Asia/Calcutta||>

Reply <l:BARCO@noiclt22815||K:CMS||O:RECreateExtendedScnScreenSource||A1:OK||>

GetDispletList

Description

Class	CMS	Applies to CMS
Object	GetDispletList	Returns a list of sources which are displayed within the specified Perspective
Attrib 1	aPerspectiveName	The Perspective name

Example:

Request <l:BARCO||K:CMS||O:getDispletList||A1:yahoo||>
 Reply <l:BARCO@noiclt22815||K:CMS||O:REgetDispletList||A0:yahoo||A1:600x600||A2:yahoo,4,300,225,300,225||>

Request <l:BARCO||K:CMS||O:getDispletList||A1:gmail||>
 Reply <l:BARCO@noiclt22815||K:CMS||O:REgetDispletList||A0:gmail||A1:600x600||>

Remarks
 Attrib 1 is the Perspective Name
 Attrib 2 is the Reference Size of the Perspective
 Attrib 3+n Returns the sources with Details:
 SourceName,TileID,X,Y,Width,Height

If the source is not displayed within a Tile or if no Tile is active then SourceTileId will be -1.

LoadSourceOnPerspective		Description
Class	CMS	Applies only to CMS
Object	LoadSourceOnPerspective	Show the given source on the specified perspective
Attrib 1	aPerspectiveName	The Perspective name onto which the source should be loaded.
Attrib 2	aSourceName	The name of the source to be loaded
Attrib 3	TileID (Optional)	Alternatively to the Geometry the TileID can be used to specify the location where to load the source within the Perspective.
	X (Optional requires Attrib 4,5&6)	Specifies the Left pixel position Default: 0
Attrib 4	Y (Optional requires Attrib 3,5,6)	Specifies the top pixel position Default: 0.
Attrib 5	Width (Optional requires Attrib 3,4,6)	Specifies the Width
Attrib 6	Height (Optional requires Attrib 3,4,5)	Specifies the Height

Example:

Request <l:BARCO||K:CMS||O:LoadSourceOnPerspective||A1:yahoo||A2:barco||A3:2||>
 Reply <l:BARCO@noiclt22815||K:CMS||O:RELoadSourceOnPerspective||A1:OK||>

Request <l:BARCO||K:CMS||O:LoadSourceOnPerspective||A1:test||A2:barco||A3:0||A4:0||A5:300||A6:200||>
 Reply <l:BARCO@noiclt22815||K:CMS||O:RELoadSourceOnPerspective||A1:OK||>

Request <l:BARCO||K:CMS||O:LoadSourceOnPerspective||A1:myName||A2:myWebSource1||A3:2||>

Reply <l:BARCO@KARCLT0409||K:CMS||O:RELoadSourceOnPerspective||A1:NOK||A2: Perspective myName not found.||>

Remarks

- 1) If the geometry values are used and the perspective has a tile active then the closes X,Y Position tile will be used to load the source into.
- 2) If no tile is active then the Configured DefaultPerspectiveTile as specified within the configuration will be loaded and used.

<u>LoadVideoSourceOnPerspectiveWithMgs</u>		Description
Class	CMS	Applies to CMS & MGS
Object	LoadVideoSourceOnPerspectiveWithMgs	Show the given video source on the specified perspective
Attrib 1	aPerspectiveName	The Perspective name onto which the video source should be loaded.
Attrib 2	aSourceName	<p>The name of the source to be loaded.</p> <p><i>Important:</i></p> <p>The source will have the description to describe the mgs worker config and cms url separated with a ";" delimiter along with a placeholder for multicast address and port.</p> <p>For example :</p> <pre>nwwsvc://%s:%s/encoder=MGS_mpeg4_rtp; uri-in=rtsp://10.3.0.71/mpeg4/1/media.amp?user-pw=pass&user-id=root&protocols=2 video-uri-out=rtp://%s:%s</pre> <p>The above source description will create a virtual cms source with video-url "nwwsvc://%s:%s/encoder=MGS_mpeg4_rtp" by replacing the place holder %s with some multicast address and port and will start a mgs worker with config "uri-in=rtsp://10.3.0.71/mpeg4/1/media.amp?user-pw=pass&user-id=root&protocols=2 video-uri-out=rtp://%s:%s" by replacing the place holder of %s with same multicast address and port filled for above cms virtual source url.</p>

Attrib 3 **TileID**

TileID is used to specify the tile number to load the source within the Perspective.

Example:

Request <l:BARCO||K:CMS||O:LoadVideoSourceOnPerspectiveWithMgs ||A1:video||A2:barco||A3:2||>

Reply <l:BARCO@noiclt22815||K:CMS||O:RELoadVideoSourceOnPerspectiveWithMgs ||A1:OK||>

Remarks This command is used for fast switching of a source on a perspective and is only valid for video type of sources.

The encoder should be all the same and configured with some same parameters: meaning the codec (ex. mpeg4) and resolution (ex. 720*576).

The same codec is needed for MGS, the same resolution is needed for the decoder.

On first request, a virtual source will be created for the perspective tile and a mgs worker will started to send the stream towards CMS.

Subsequent requests will only change the mgs worker uri-in.

RunCmsMgsSyncUtility		Description
Class	CMS	Applies only to CMS
Object	RunCmsMgsSyncUtility	Synchronize the mgs workers and cms mgs sources created with "LoadVideoSourceOnPerspectiveWithMgs" command.
Example:		
Request	<I:BARCO K:CMS O:RunCmsMgsSyncUtility >	
Reply	<I:BARCO@noiclt22815 K:CMS O:RERunCmsMgsSyncUtility A1:OK >	
Remarks		

RemoveSourceFromPerspective		Description
Class	CMS	Applies only to CMS
Object	RemoveSourceFromPerspective	Remove the give source from the specified perspective
Attrib 1	aPerspectiveName	The Perspective name from which to remove the give source
Attrib 2	aSourceName	The Name of the source to remove
Example:		
Request	<I:BARCO K:CMS O:RemoveSourceFromPerspective A1:test A2:barco >	
Reply	<I:BARCO@noiclt22815 K:CMS O:RERemoveSourceFromPerspective A1:OK >	
Remarks		

RemoveSourceFromPerspectiveTile		Description
Class	CMS	Applies only to CMS
Object	RemoveSourceFromPerspectiveTile	Remove the source from the given tile of a perspective
Attrib 1	aPerspectiveName	The Perspective name from which to remove the given source
Attrib 2	atileId	The tile of a perspective from where the source has to be removed
Example:		
Request	<I:BARCO K:CMS O:RemoveSourceFromPerspectiveTile A1:test A2:4 >	

Reply <l:BARCO@noiclt22815||K:CMS||O:RemoveSourceFromPerspectiveTile ||A1:OK||>

Remarks

<u>ApplyDecoratorToASourcePerspective</u>	Description
CMS	Applies to CMS
ApplyDecoratorToASourcePerspective	Applies a decorator to a source within a perspective.
aPerspectiveName	A Perspective name.
aSourceName	A name of the source on which the decorator will be applied.

aDecoratorName Name of the decorator to be applied on a source within a perspective.

Example:

Request <l:BARCO||K:CMS||O:ApplyDecoratorToASourcePerspective||A1:perspective||A2:source||A3:decorator||>

Reply <l:BARCO@noiclt22815||K:CMS||O:ApplyDecoratorToASourcePerspective||A1:OK||>

Remarks

<u>RemoveDecoratorFromASourcePerspective</u>	Description
CMS	Applies to CMS
RemoveDecoratorFromASourcePerspective	Remove a decorator from a source within a perspective.
aPerspectiveName	A Perspective name.
aSourceName	A name of the source on which the decorator will be applied.

aDecoratorName Name of the decorator to be remove from a source within a perspective.

Example:

Request <l:BARCO||K:CMS||O:RemoveDecoratorFromASourcePerspective ||A1:perspective||A2:source||A3:decorator||>

Reply <l:BARCO@noiclt22815||K:CMS||O:RemoveDecoratorFromASourcePerspective||A1:OK||>

Remarks

<u>GetPerspectiveAppliedTile</u>	Description
Class	CMS Applies only to CMS
Object	GetPerspectiveAppliedTile Returns the applied tile name of a given perspective.
Attrib 1	aPerspectiveName The Perspective name

Example:

Request <I:BARCO||K:CMS||O:GetPerspectiveAppliedTile||A1:test||>

Reply <I:BARCO@noiclt22815||K:CMS||O:GetPerspectiveAppliedTile||A1:2x2||>

Remarks

GetDisplayAppliedTile

Description

Class	CMS	Applies only to CMS
Object	GetDisplayAppliedTile	Returns the applied tile name of a given display
Attrib 1	aDisplayName	The Display name

Example:

Request <I:BARCO||K:CMS||O:GetDisplayAppliedTile||A1:Display||>

Reply <I:BARCO@noiclt22815||K:CMS||O:GetDisplayAppliedTile||A1:2x2||>

Remarks

UpdateDisplayTile

Description

Class	CMS	Applies only to CMS
Object	UpdateDisplayTile	Updates the display with a given tile
Attrib 1	aDisplayName	The display name on which tiling has to be updated
Attrib 2	aTileName	The tile name to be updated on a display

Example:

Request <I:BARCO||K:CMS||O:UpdateDisplayTile||A1:Display||A2:3x3||>

Reply <I:BARCO@noiclt22815||K:CMS||O:UpdateDisplayTile||A1:OK||>

Remarks

UpdatePerspectiveTile

Description

Class	CMS	Applies only to CMS
Object	UpdatePerspectiveTile	Updates the perspective with a given tile
Attrib 1	aPerspectiveName	The Perspective name on which tiling has to be updated
Attrib 2	aTileName	The tile name to be updated on a perspective

Example:

Request <l:BARCO||K:CMS||O:UpdatePerspectiveTile||A1:perspective||A2:1x1||>
 Reply <l:BARCO@noicIt22815||K:CMS||O:UpdatePerspectiveTile||A1:OK||>

Remarks

<u>CreateAndApplyTileOnDisplay</u>		Description
Class	CMS	Applies only to CMS
Object	CreateAndApplyTileOnDisplay	Creates tile with the given name and applies that tiling on a given display
Attrib 1	aDisplayName	The display on which tile has to be applied
Attrib 2	aTileName	The name of a tile to be created
Attrib 3	aRows	The no of rows to be used for creating a tile
Attrib 4	aColumns	The no of columns to be used for creating a tile

Example:

Request <l:BARCO||K:CMS||O:CreateAndApplyTileOnDisplay||A1:Display||A2:3x3||A3:3||A4:3||>
 Reply <l:BARCO@noicIt22815||K:CMS||O:CreateAndApplyTileOnDisplay||A1:OK||>

Remarks

<u>CreateAndApplyTileOnPerspective</u>		Description
Class	CMS	Applies only to CMS
Object	CreateAndApplyTileOnPerspective	Creates tile with the given name and applies that tiling on a given perspective
Attrib 1	aPerspectiveName	The perspective on which tile has to be applied
Attrib 2	aTileName	The name of a tile to be created
Attrib 3	aRows	The no of rows to be used for creating a tile
Attrib 4	aColumns	The no of columns to be used for creating a tile

Example:

Request <l:BARCO||K:CMS||O:CreateAndApplyTileOnPerspective||A1:perspective||A2:10x10||A3:10||A4:10||>
 Reply <l:BARCO@noicIt22815||K:CMS||O:CreateAndApplyTileOnPerspective||A1:OK||>

Remarks

<u>GetTileList</u>		Description
Class	CMS	Applies to CMS

Object **GetTileList** Returns a list of configured Tiles.

Attrib 1 aFilterText (Optional) Only Tiles which have aFilterText within their names will be returned

Example:

Request <!:BARCO||K:CMS||O:GetTileList|>

Reply <!:BARCO@noict22815||K:CMS||O:REGetTileList||A0:1x1||A1:4x4||A2:2x2|>

Request <!:BARCO||K:CMS||O:GetTileList ||A1:2*|>

Reply <!:BARCO@KARCLT0409||K:CMS||O:REGetTileList ||A1:2x2|>

BCM:

<u>GetLampPowerStatus*</u>		Description
Class	BCM	Applies to BCM
Object	GetLampPowerStatus	Returns the LampPower Status. Not valid for BCM -OL/OVL

<u>GetLampRuntimes</u>		Description
Class	BCM	Applies to BCM
Object	GetLampRuntimes	Returns the LampRuntimes of each lamp. Not valid for BCM - OL/OVL.

<u>WallOn</u>		Description
Class	BCM	Applies to BCM
Object	WallOn	Turns the entire display wall ON. This command executed internally either at wall level or projector level depending on BCMC version. If it gets executed internally at wall level, it will return the wall command executing status otherwise it will return the each projector command executing status.
Example:		
Request	<!:BARCO K:BCM O:WallOn >	
Reply	<!:BARCO@noict22815 K:BCM O:REWallOn A1:STATE_IN_PROGRESS >	
Request	<!:BARCO K:BCM O:WallOn >	
Reply	<!:BARCO@noict22815 K:BCM O:REWallOn A1:STATE_REQUEST_DONE A2:STATE_REQUEST_DONE A3:STATE_REQUEST_DONE A4:STATE_REQUEST_DONE >	

Remarks	<p>If multiple Walls are configured within the BCM-Server then the Wall to be used needs to be pre-configured using „SetSelectedWall“. The default is that the first wall will be used.</p> <p>Reply will consist of all the status of projectors from the row to columns. If the request gets completed, the CRP will reply for a projector with STATE_REQUEST_DONE status. Reply with STATE_IN_PROGRESS status indicate that the BCM server is under execution of a command for that projector.</p>
----------------	---

WallOff		Description
Class	BCM	Applies to BCM
Object	WallOff	<p>Turns the entire display wall OFF.</p> <p>This command executed internally either at wall level or projector level depending on BCMC version.</p> <p>If it gets executed internally at wall level, it will return the wall command executing status otherwise it will return the each projector command executing status.</p>
Example:		
Request	<l:BARCO K:BCM O:WallOff >	
Reply	<l:BARCO@noict22815 K:BCM O:REWallOff A1:STATE_IN_PROGRESS >	
Request	<l:BARCO K:BCM O:WallOff >	
Reply	<l:BARCO@noict22815 K:BCM O:REWallOff A1:STATE_REQUEST_DONE A2:STATE_REQUEST_DONE A3:STATE_REQUEST_DONE A4:STATE_REQUEST_DONE >	
Remarks	<p>If multiple Walls are configured within the BCM-Server then the Wall to be used needs to be pre-configured using „SetSelectedWall“. The default is that the first wall will be used.</p> <p>Reply will consist of all the status of projectors from the row to columns. If the request gets completed, the CRP will reply for a projector with STATE_REQUEST_DONE status. Reply with STATE_IN_PROGRESS status indicate that the BCM server is under execution of a command for that projector.</p>	

ProjectorOn		Description
Class	BCM	Applies to BCM
Object	ProjectorOn	Turns the projector with the Specified ID ON
Attrib	alD	Number of the Projector Nr.1 or 2 or ...
Example:		
Request	<l:BARCO K:BCM O:ProjectorOn A1:1 >	
Reply	Reply1:<l:BARCO@noict22815 K:BCM O:REProjectorOn A0:STATE_REQUEST_DONE >	

	Reply2:<l:BARCO@noict22815 K:BCM O:REProjectorOn A0:STATE_IN_PROGRESS >
Remarks	<p>If multiple Walls are configured within the BCM-Server then the Wall to be used needs to be pre-configured using „SetSelectedWall“. The default is that the first wall will be used.</p> <p>If the request gets completed for a projector, the CRP will reply with a STATE_REQUEST_DONE status. Reply with STATE_IN_PROGRESS status indicate that the BCM server is under execution of a command for a projector.</p>

ProjectorOff		Description
Class	BCM	Applies to BCM
Object	ProjectorOff	Turns the projector with the Specified ID OFF
Attrib	alD	Number of the Projector Nr.1 or 2 or ...
Example:		
Request	<l:BARCO K:BCM O:ProjectorOff A1:5>	
Reply	<l:BARCO@noict22815 K:BCM O:REProjectorOff A0:STATE_REQUEST_DONE >	
Remarks	<p>If multiple Walls are configured within the BCM-Server then the Wall to be used needs to be pre-configured using „SetSelectedWall“. The default is that the first wall will be used.</p> <p>If the request gets completed for a projector, the CRP will reply with a STATE_REQUEST_DONE status. Reply with STATE_IN_PROGRESS status indicate that the BCM server is under execution of a command for a projector.</p>	

GetPreferredSourceList		Description
Class	BCM	Applies to BCM
Object	GetPreferredSourceList	Returns the currently active and selected Input as comma separated values for each projector. Values are INPUT_NONE, INPUT_1, INPUT_2
Example:		
Request	<l:BARCO K:BCM O:GetPreferredSourceList >	
Reply	<l:BARCO@noict22815 K:BCM O:REGetPreferredSourceList A1:INPUT_NONE, INPUT_2 A2:INPUT_NONE,INPUT_1 A3:INPUT_1,INPUT_1 A4:INPUT_1,INPUT_1 >	
Remarks	<p>If multiple Walls are configured within the BCM-Server then the Wall to be used needs to be pre-configured using „SetSelectedWall“. The default is that the first wall will be used.</p>	

WallSetSource1Preferred		Description
Class	BCM	Applies to BCM
Object	WallSetSource1Preferred	Switches all the projectors to input number 1. This command executed internally either at wall level or

	projector level depending on BCMC version. If it gets executed internally at wall level, it will return the wall command executing status otherwise it will return the each projector command executing status.
Example:	
Request	<l:BARCO K:BCM O:WallSetSource1Preferred >
Reply	<l:BARCO@noiclt22815 K:BCM O:REWallSetSource1Preferred A1:STATE_IN_PROGRESS >
Request	<l:BARCO K:BCM O:WallSetSource1Preferred >
Reply	<l:BARCO@noiclt22815 K:BCM O:REWallSetSource1Preferred A1:STATE_REQUEST_DONE A2:STATE_REQUEST_DONE A3:STATE_REQUEST_DONE A4:STATE_REQUEST_DONE >
Remarks	If multiple Walls are configured within the BCM-Server then the Wall to be used needs to be pre-configured using "SetSelectedWall". The default is that the first wall will be used. Reply will consist of all the status of projectors from the row to columns. If the request gets completed, the CRP will reply for a projector with STATE_REQUEST_DONE status. Reply with STATE_IN_PROGRESS status indicate that the BCM server is under execution of a command for that projector.

WallSetSource2Preferred		Description
Class	BCM	Applies to BCM
Object	WallSetSource2Preferred	Switches all the projectors to input number 2. This command executed internally either at wall level or projector level depending on BCMC version. If it gets executed internally at wall level, it will return the wall command executing status otherwise it will return the each projector command executing status.
Example:		
Request	<l:BARCO K:BCM O:WallSetSource2Preferred >	
Reply	<l:BARCO@noiclt22815 K:BCM O:REWallSetSource2Preferred A1:STATE_IN_PROGRESS >	
Request	<l:BARCO K:BCM O:WallSetSource2Preferred >	
Reply	<l:BARCO@noiclt22815 K:BCM O:REWallSetSource2Preferred A1:STATE_REQUEST_DONE A2:STATE_REQUEST_DONE A3:STATE_REQUEST_DONE A4:STATE_REQUEST_DONE >	
Remarks	If multiple Walls are configured within the BCM-Server then the Wall to be used needs to be pre-configured using „SetSelectedWall“. The default is that the first wall will be used. Reply will consist of all the status of projectors from the row to columns. If the request gets completed, the CRP	

	will reply for a projector with STATE_REQUEST_DONE status. Reply with STATE_IN_PROGRESS status indicate that the BCM server is under execution of a command for that projector.	
<u>ProjectorSetSource1Preferred</u>		Description
Class	BCM	Applies to BCM
Object	WallSetSource1Preferred	Sets the input of the give projector to input number 1
Attrib	aID	Number of the Projector Nr.1 or 2 or ...
Example:		
Request	<!:BARCO K:BCM O:ProjectorSetSource1Preferred A1:1 >	
Reply	<!:BARCO@noiclt22815 K:BCM O:REProjectorSetSource1Preferred A1:STATE_REQUEST_DONE >	
Remarks		
<p>If multiple Walls are configured within the BCM-Server then the Wall to be used needs to be pre-configured using „SetSelectedWall“. The default is that the first wall will be used.</p> <p>If the request gets completed for a projector, the CRP will reply with a STATE_REQUEST_DONE status. Reply with STATE_IN_PROGRESS status indicate that the BCMC server is under execution of a command for a projector.</p>		

<u>ProjectorSetSource2Preferred</u>		Description
Class	BCM	Applies to BCM
Object	WallSetSource2Preferred	Sets the input of the give projector to input number 2
Attrib	aID	Number of the Projector Nr.1 or 2 or ...
Example:		
Request	<!:BARCO K:BCM O:ProjectorSetSource2Preferred A1:1 >	
Reply	<!:BARCO@noiclt22815 K:BCM O:REProjectorSetSource2Preferred A1:STATE_REQUEST_DONE >	
Remarks		
<p>If multiple Walls are configured within the BCM-Server then the Wall to be used needs to be pre-configured using „SetSelectedWall“. The default is that the first wall will be used.</p> <p>If the request gets completed for a projector, the CRP will reply with a STATE_REQUEST_DONE status. Reply with STATE_IN_PROGRESS status indicate that the BCMC server is under execution of a command for a projector.</p>		

<u>NumberOfWalls</u>		Description
Class	BCM	Applies to BCM
Object	NumberOfWalls	<p>Returns the number of configured walls.</p> <p>Default is 1.</p> <p>In BCM-OL, only one wall can be configured. Therefore this command would always return one for the current version.</p>

Example:	
Request	<I:BARCO K:BCM O:NumberOfWalls >
Reply	<I:BARCO@KARCLT0409 K:BCM O:RENumberOfWalls A1:1 >

<u>GetSelectedWall</u>		Description
Class	BCM	Applies to BCM
Object	GetSelectedWall	Returns the wall be used when multiple walls are configured within 1 BCM-Server. Typically this will always be 1
Example:		
Request	<I:BARCO K:BCM O:GetSelectedWall >	
Reply	<I:BARCO@KARCLT0409 K:BCM O:REGetSelectedWall A1:1 >	

<u>SetSelectedWall</u>		Description
Class	BCM	Applies to BCM
Object	SetSelectedWall	When multiple wall are controlled from 1 BCM-Server this allows you to selected which Wall should be used when applying calls like WallOn, getWallStatus etc
Attrib	aID	The Wall number to be used. Ex. 1 or 2
Example:		
Request	<I:BARCO K:BCM O:SetSelectedWall A1:1>	
Reply	<I:BARCO@KARCLT0409 K:BCM O:RESetSelectedWall A1:OK >	
Remarks	When the application starts the first time or is restarted then the first wall within the BCM-Server will be set as the selected Wall.	

<u>GetWallStatus</u>		Description
Class	BCM	Applies to BCM
Object	GetWallStatus	Returns the status as comma separated values of each cube within the wall where the first attribute gives the operational status of the cube, second attribute gives the connection state of the cube, third attribute gives the health status of the cube, fourth attribute gives the used led hours and fifth attribute gives the projector hours. Possible values: "OPERATIONSTATE_ON", "OPERATIONSTATE_IDLE" 2nd attribute- Connection status Possible values: "CONNECTIONSTATE_OK", "CONNECTIONSTATE_NOT_RESPONDING"
Control Room Proxy –ver 1.0.0 doc Page 40 of 43		

		<p>3rd attribute- Health status</p> <p>Possible values: "HEALTHSTATE_OK", "HEALTHSTATE_WARNING", "HEALTHSTATE_ERROR"</p> <p>4th attribute- Runtime led hours</p> <p>5th attribute- Runtime projection unit</p>
Example:		
Request	<I:BARCO K:BCM O:GetWallStatus >	
Reply	<I:BARCO@noiclt22815 K:BCM O:REGetWallStatus A1:OPERATIONSTATE_ON,CONNECTIONSTATE_OK,HEALTHSTATE_OK,8550,9154 >	

GetProjectorStatus		Description
Class	BCM	Applies to BCM
Object	GetProjectorStatus	<p>Returns the status as comma separated values of a projector with the given number where the first attribute gives the operational status of the cube, second attribute gives the connection state of the cube, third attribute gives the health status of the cube, fourth attribute gives the used led hours and fifth attribute gives the projector hours.</p> <p><i>1st attribute- Operational status</i></p> <p>Possible values: "OPERATIONSTATE_ON", "OPERATIONSTATE_IDLE"</p> <p><i>2nd attribute- Connection status</i></p> <p>Possible values: "CONNECTIONSTATE_OK", "CONNECTIONSTATE_NOT_RESPONDING"</p> <p><i>3rd attribute- Health status</i></p> <p>Possible values: "HEALTHSTATE_OK", "HEALTHSTATE_WARNING", "HEALTHSTATE_ERROR"</p> <p><i>4th attribute- Runtime led hours</i></p> <p><i>5th attribute- Runtime projection unit hours</i></p>
Attrib	aProjectorID	The Projector number to be used. Ex. 1 or 2
Example:		
Request	<I:BARCO K:BCM O:GetProjectorStatus A1:2 >	
Reply	<I:BARCO@noiclt22815 K:BCM O:REGetProjectorStatus A1:OPERATIONSTATE_ON,CONNECTIONSTATE_OK,HEALTHSTATE_OK,8550,9154 >	

<u>NumberOfProjectors</u>		Description
Class	BCM	Applies to BCM
Object	NumberOfProjectors	Returns the number of projectors.
Example:		
Request	<!:BARCO K:BCM O:NumberOfProjectors >	
Reply	<!:BARCO@noiclt22815 K:BCM O:RENumberOfProjectors A1:4 >	

<u>UpdateBrightnessControl</u>		Description
Class	BCM	Applies to BCM
Object	UpdateBrightnessControl	Update the brightness of the wall.
Attrib1	maxBrightnessUserConstraint	Value between 0 to wall supported brightness value.
Attrib2	minBrightnessUserConstraint	Value between 0 to wall supported brightness value.
Example:		
Request	<!:BARCO K:BCM O:UpdateBrightnessControl A0:0 A1:20 >	
Reply	<!:BARCO@noiclt22815 K:BCM O:REUpdateBrightnessControl A1:OK >	
Remarks	The 0 value won't be updated as the brightness of the wall. So, if only the maxbrightness of the wall has to be updated, set attr1(maxbrightness) as the value to be updated and attr2 value as 0 and vice-versa.	

<u>UpdateBrightnessMode</u>		Description
Class	BCM	Applies to BCM
Object	UpdateBrightnessMode	Update the brightness mode of the wall.
Attrib	mode	Either 0 or 1.
Example:		
Request	<!:BARCO K:BCM O:UpdateBrightnessMode A0:1 >	
Reply	<!:BARCO@noiclt22815 K:BCM O:REUpdateBrightnessMode A1:OK >	
Remarks	Two brightness mode of the wall: <ul style="list-style-type: none"> • BRIGHTNESSMODE_ECO • BRIGHTNESSMODE_MAX 0(BRIGHTNESSMODE_ECO) or 1(BRIGHTNESSMODE_MAX)	

<u>RenewColorAdjustment</u>		Description
Class	BCM	Applies to BCM
Object	RenewColorAdjustment	Renews the color adjustment.
Example:		
Request	<!:BARCO K:BCM O:RenewColorAdjustment >	
Reply	<!:BARCO@noiclt22815 K:BCM O:RERenewColorAdjustment A1:OK >	

Remarks	This command needs to be called after UpdateBrightnessControl command.
---------	--

GetBrightnessList		Description
Class	BCM	Applies to BCM
Object	GetBrightnessList	Gives all the brightness parameters as comma separated values.
Example:		
Request	<!:BARCO K:BCM O:GetBrightnessList >	
Reply	<!:BARCO@noiclt22815 K:BCM O:REGetBrightnessList A0:BRIGHTNESSMODE_MAX, 91,100,20,95,56 >	
Remarks	Parameter1 : Brightness mode Parameter2 : Wall brightness Parameter3 : Max brightness user constraint Parameter4 : Min brightness user constraint Parameter5 : Max brightness control parameter value Parameter6 : Min brightness control parameter value	

UpdateCrpBcmWall		Description
Class	BCM	Applies to BCM
Object	UpdateCrpBcmWall	Whenever the new wall is configured in BCM or the existing wall is updated, "UpdateCrpBcmWall" command needs to be sent to Control Room Proxy.
Example:		
Request	<!:BARCO K:BCM O:UpdateCrpBcmWall >	
Reply	<!:BARCO@noiclt22815 K:BCM O:REUpdateCrpBcmWall A1:OK >	
Remarks	This command needs to be called to update the CRP with bcm wall.	

ProjectorRows		Description
Class	BCM	Applies to BCM
Object	ProjectorRows	Returns the number of rows in the wall configuration.
Example:		
Request	<!:BARCO K:BCM O:ProjectorRows >	
Reply	<!:BARCO@NOICLT22879 K:BCM O:REProjectorRows A1:1 >	

ProjectorColumns		Description
Class	BCM	Applies to BCM
Object	ProjectorColumns	Returns the number of columns in the wall configuration
Example:		

Request	<!:BARCO K:BCM O:ProjectorColumns >
Reply	<!:BARCO@NOICLT22879 K:BCM O:REProjectorColumns A1:1 >

<u>GetWallHealthSummary</u>		Description
Class	BCM	Applies to BCM
Object	GetWallHealthSummary	Returns the overall health status of a wall. Possible values: "HEALTHSTATE_OK", "HEALTHSTATE_WARNING", "HEALTHSTATE_ERROR"
Example:		
Request	<!:BARCO K:BCM O:GetWallHealthSummary >	
Reply	<!:BARCO@noiclt22815 K:BCM O:RE GetWallHealthSummary A1:HEALTHSTATE_OK >	

<u>GetWallAlarms</u>		Description
Class	BCM	Applies to BCM
Object	GetWallAlarms	Returns the alarms for a wall, if there are any otherwise returns with an empty attribute.
Example:		
Request	<!:BARCO K:BCM O:GetWallAlarms >	
Reply	<!:BARCO@noiclt22815 K:BCM O:RE GetWallAlarms A1: >	