grass valley

Encore

RCL NATIVE PROTOCOL CLIENT

GUI User Guide

VERSION 1.0

071862200
1 NOVEMBER 2007

THOMSON
images & beyond

CONTENTS

OVERVIEW

The RCL (Router Control Language) NP (Native Protocol) Client Application is a software application implementing the RCL/NP Protocol, it also doubles over as a test tool that talks to any Server implementing the RCL/NP protocol for the various operations that need to be performed, like, takes, queries etc. It also provides the capability to interface to, and control the routers, which implement either NP or RCL protocol.

The RCL Client Library provides a set of APIs, which is used by the RCL Client Simulator application to communicate with and control any server system implementing the RCL Protocol. The RCL Client Library contains the implementations of the RCL/NP protocol specifications. RCL Client Library supports Serial RS-232 and Ethernet as the external communication medium.

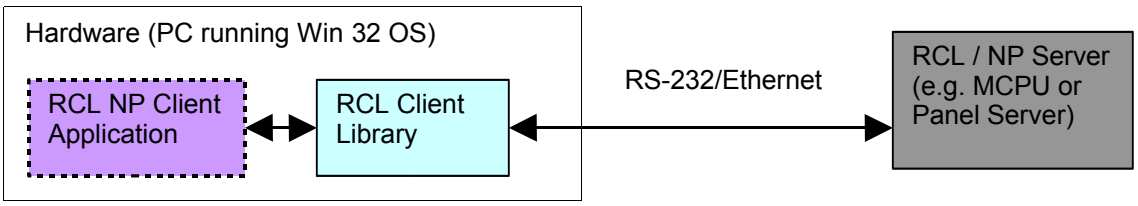
The RCL NP Client Application will be used to test/demonstrate the working of the RCL (Router Control Language) Protocol and NP (Native Protocol). It will test the RCL/NP Server (i.e. on the MCPU or Panel Server) and since it uses the RCL Client API to communicate with the MCPU, it serves as a mechanism to test the RCL Client API/Library as well.

Pre-Requisites:

You should know the Encore and SMS7000 systems as well as the NP and RCL Protocol.

Description

The RCL NP Client Application is a dialog-based application running on a PC with win32 based operating system. It will be used to test the functioning of the RCL/NP server (i.e. on the MCPU or Panel Server). The RCL client will interface with the SMS7000 MCPU/Panel Server over Serial and Ethernet.



The above diagram shows the system architecture where the RCL NP Client Application and the RCL Client & Server fit in. The RCL NP Client Application is nothing but a type of User Application as mentioned in the RCL Client FDD/HLD. The RCL NP Client Application interfaces to the RCL/NP Server through a set of APIs provided by the RCL library. The external interface could be one of RS-232 or Ethernet. The RCL NP Client Application and the RCL Client library will work together on a Win32 OS only.

Feature List

The RCL NP Client Application will have a dialog-based graphical user interface.

- The RCL NP Client Application will have a facility for connecting to the Router when it comes up. It will also provide the facility for the user to disconnect from the RCL/NP Server when necessary.
- The GUI will provide a mechanism whereby each API provided by the RCL Client Library can be invoked “individually” and with the help of dialog controls, the GUI will provide the user the facility of selecting different parameters for each command.

- The GUI will display the RCL/NP format of the message being sent to RCL/NP Server. It will also display the RCL/NP format of the message received from the RCL/NP Server as well as the response data received, if any.

- The GUI will also provide a mechanism whereby the user can send "multiple" queries at a time. This can be achieved by specifying a script file in which all the commands to be sent are included in a particular format as comma separated data. The user can also specify a file for logging all the responses received from the Server.

- There is another option provided by the GUI that is the Packet Generator. This is provided so that the users can enter the commands directly to be executed. Another alternative is to add a set of commands and store them in a file, which can be loaded later for execution. The commands can be executed as many number of times that the user wishes by specifying the iteration count. The Packet Generator gives the user the flexibility of manipulating the commands as per his/her requirements. This is also an excellent tool to test the server corresponding to the respective protocols.

DETAILS OF USAGE

The GUI for the RCL Client Simulator has been designed using MFC Property Sheets and Property Pages.

The user needs to establish a connection between this RCL Client Simulator and the RCL/NP Server running on the Router. The RCL Client Simulator can communicate with the RCL/NP server by using the RCL Protocol / NP Protocol.

When communicating to a server via NP Protocol, the client initially sends a BK, N command to RCL/NP Server. The connection is established between the client and server when client receives the appropriate response for BK, N from the server.

When communicating to server via RCL Protocol, the client initially sends the RC command to RCL/NP Server. The connection is established between the client and server when RA is received by the client as response to the RC command from the server.

The user also needs to specify the mode for testing. There are three modes available to the user for testing namely,

- User Interface
- Test Script
- Packet Generator

The entire set of NP / RCL Commands have been categorized into the following categories:

- Connections
- Query
- Settings
- Status
- Operations

- Subscription (Only in the RCL Mode)

Each of the above items has been represented in the GUI as a Property Page with tabs for each category. Since the Query commands are more in number, they have been spread over two Property Pages.

When the GUI comes up, the user is presented with only one Tab namely, the 'Connections' Tab. It is in this page that the user specifies the test mode. Depending on which mode the user specifies, the GUI will display the appropriate Tabs.

In the 'User Interface' mode, the GUI will bring up 6 additional Tabs namely,

- Query
- Query ctd.
- Settings
- Status
- Operations
- Subscriptions
- Help

In the 'Test Script' mode, only one tab will come up with the following heading

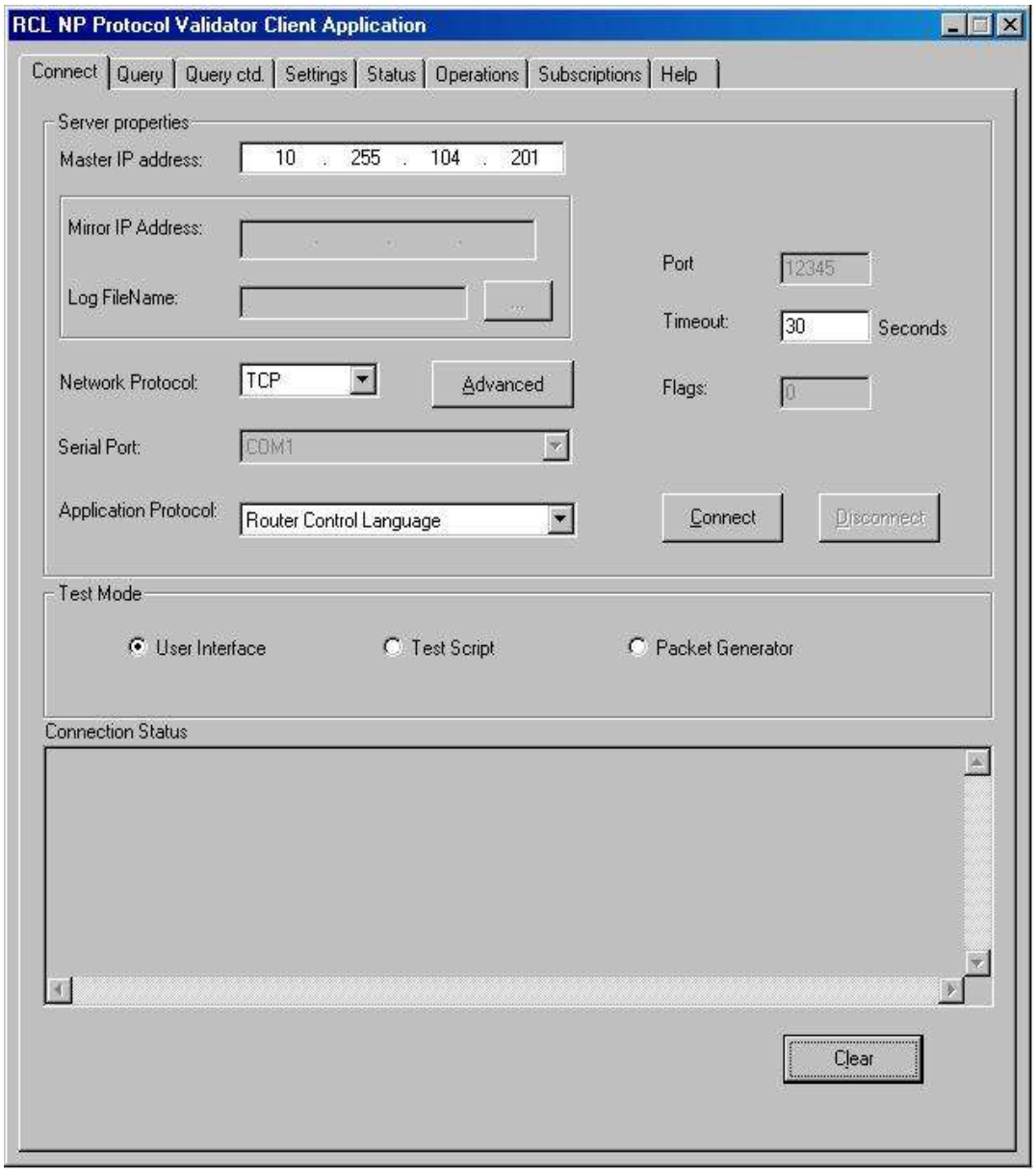
- Run script
- Help

In the 'Packet Generator' mode, the tab with the following heading comes up

- Packet Generator
- Help

RCL NP CLIENT APPLICATION

Connect Tab



The 'connect tab' is displayed at startup when the RCLNPClientSimulator comes up. In this screen the various configurable options are set, which indicates if it is a RCL/NP Client, whether it uses serial or ethernet communication, whether it runs in the user interface, test script or packet generator modes.

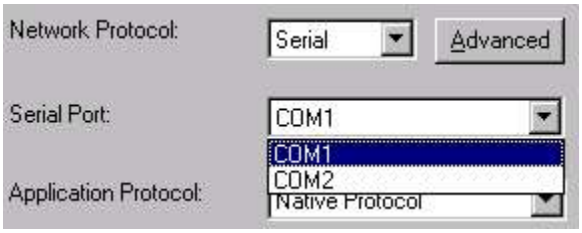
Description of Controls

- Master IP address: Accepts the IP address of the master server.
- Mirror IP address: Accepts the IP address of the redundant mirror server and also takes in the log file name. The response received from redundant server is logged into the file specified, as the GUI does not display these responses. The redundancy is provided only in the script mode.

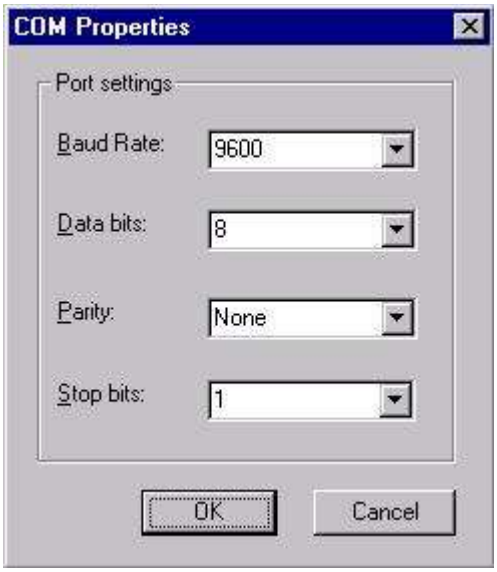
- Network Protocol: Selects the following TCP/UDP/Serial protocol options.

- Application protocol: Selects the following Native Protocol/Router Control Language application options.

- Port (using TCP): Non-editable port number '12345'.
- Serial Port: Selects the COM1/COM2 serial port options.

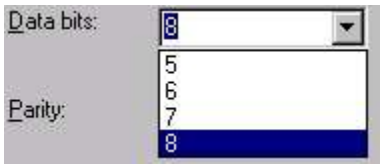


- Advanced: Pops up the following dialog box where the user can set the Baud rate, Data bits, parity and stop bits.



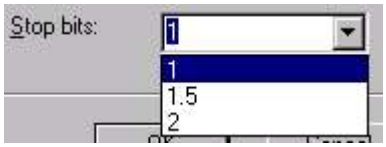
Baud Rate:

Data bits:

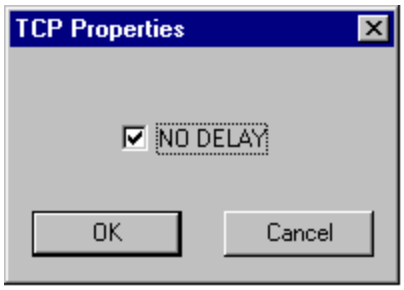


Parity:

Stop bits:



Advanced Feature Packet Delay:

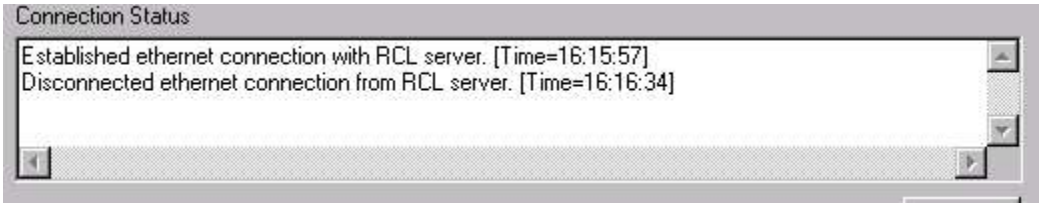


When the NO DELAY option is selected, the TCP_NODELAY option is enabled in the TCP properties, which in turn disables the Nagle algorithm. The process involves buffering send data when there is unacknowledged data already in flight or buffering send data until a full-size packet can be sent. The packets are no longer buffered and sent as and when a full packet is formed.

- Flags: Not Used
- Timeout: User can specify the timeout value (in seconds). The default time out value is 30 seconds.

- Test Mode: Toggles between the User Interface/Test Script/Packet Generator modes.

- Connection Status: Displays a message, if the connection has been established or is in the disconnected state or if an error has occurred.



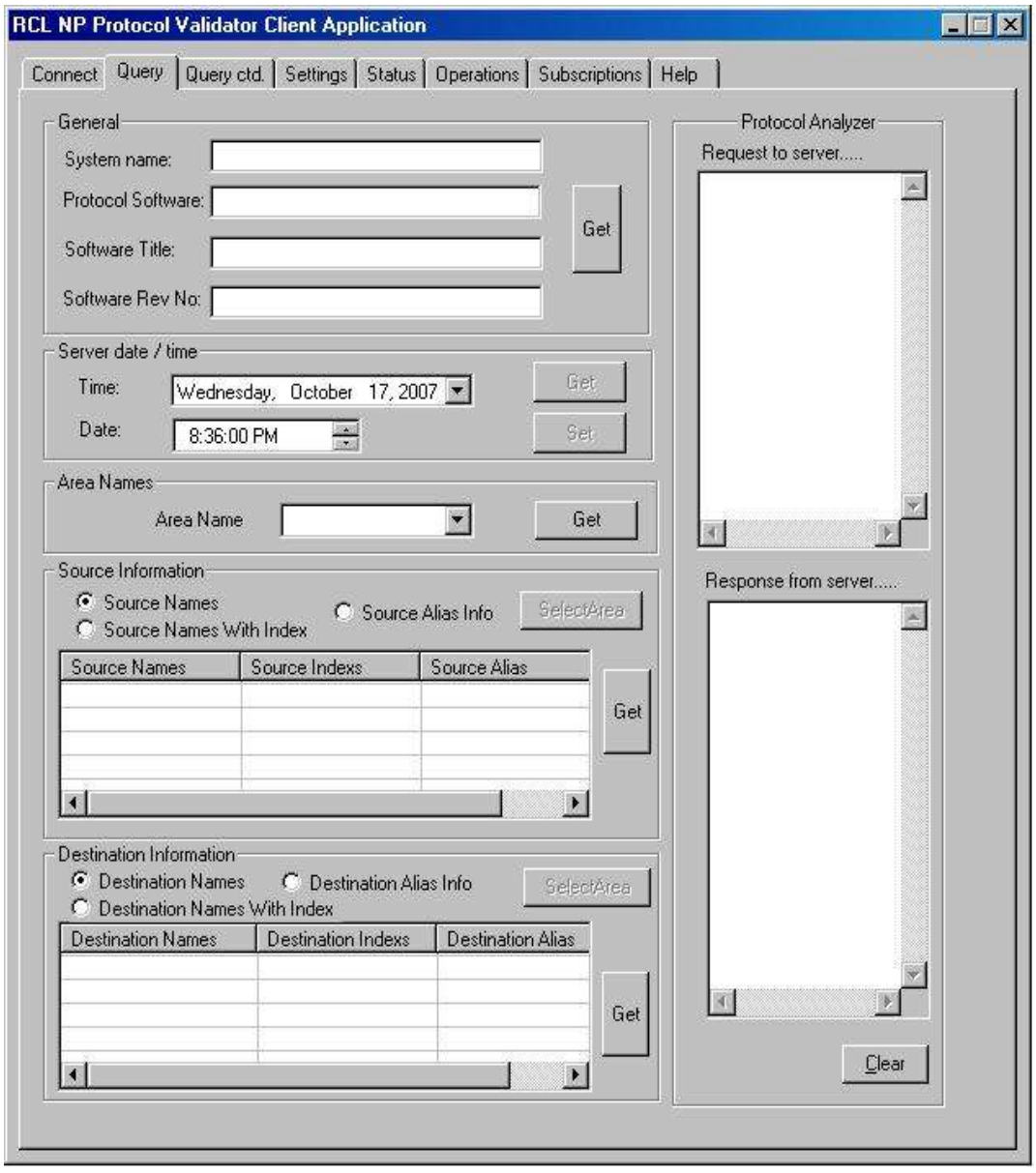
- Clear: Clears the contents of the Connection Status edit control.
- Connect: Establishes connection with the NP/RCL Server.
- Disconnect: Disconnects the established connection with the NP/RCL Server.
- About RCL…: Pops up a dialog box with details about the RCL Client Simulator.

User Interface

Overview:

The user interface mode helps the user to perform operations such as take, querying destination status etc., by a click of the button. Each of the tabs that are provided covers most of the operations allowed by the NP/RCL server. It helps the user to perform operations without bothering to know what commands these operations correspond to.

Query Tab



Upon choosing the user interface option we get 5 other tabs displayed to us. Let us discuss the first of the tabs, the query tab which is the screen captured above.

Description of Controls

- System Name/Protocol Software/Software Title/Software Revision: When the user clicks the Get button in the General group box, the name of the processor running on

the server is displayed against System Name. The Protocol Software command is not yet supported. The Software Title and Software Revision are NP commands, but currently they are not supported.

- Date/Time: The Get button retrieves the server date and time. Likewise the Set button sets the date and time on the server. Both these commands are supported only in the NP mode.

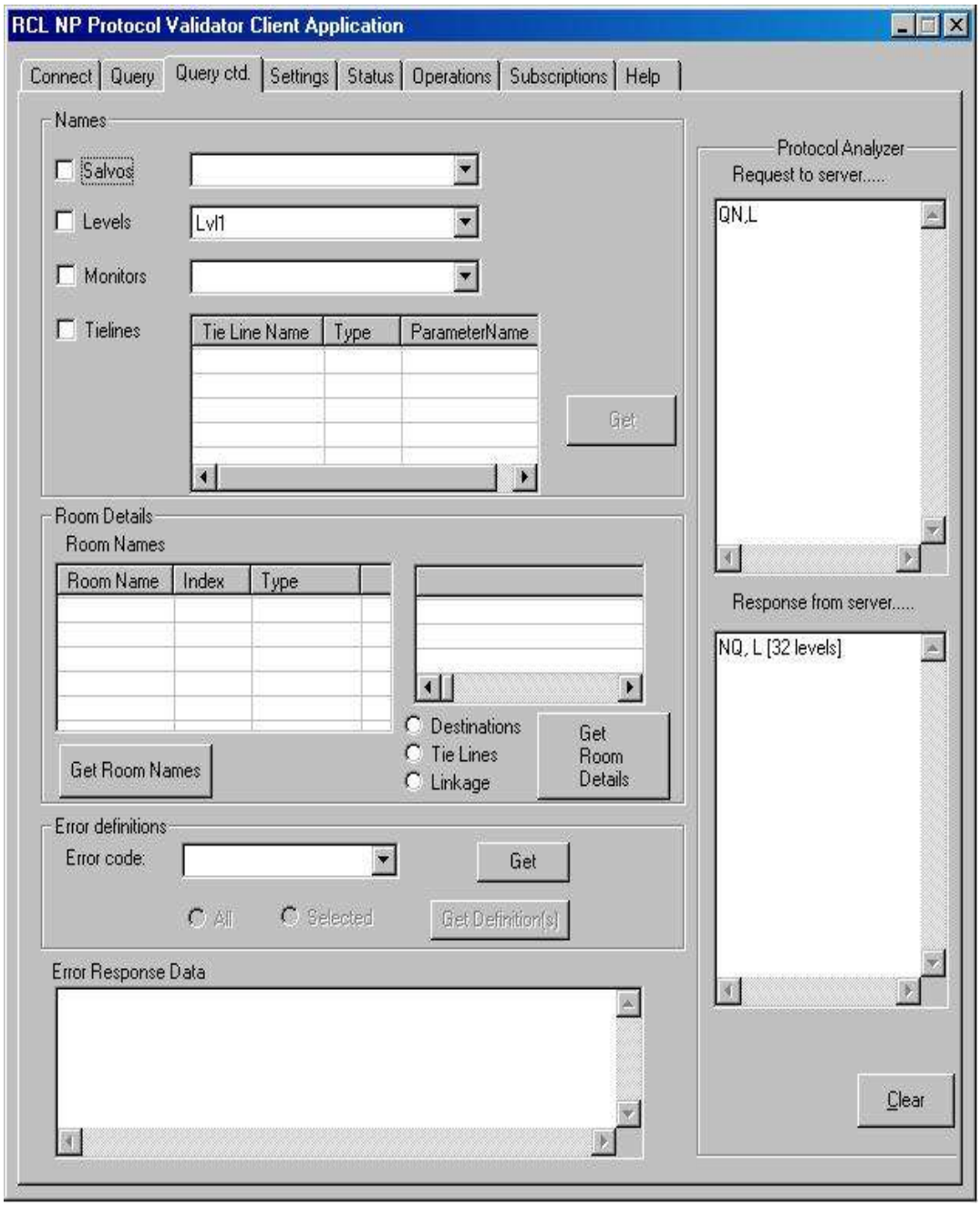
- Area: When user checks the check box and click on the “Get” button of the “Names/indices” group box, the list of area names available in system shown in the combo box. This command is supported only for RCL Protocol.

- Select Area: This button is enabled when the user queries for the area names. This button allows user to select the specific areas. This value is used when user queries for the source/destinations/source index/destination index/srcalias/destination alias info.

- Source Names/Destination Names/Source Names with Index/Destination Names With Index/Source Alias Info /Destination Alias Info: The user can choose the radio button that he wants to query. The user then has to click the Get button. The information about source/destination displayed in appropriate list box.

- Clear: Clears the contents of the Protocol Analyzer.

Query ctd. Tab



Description of Controls

- Salvos/Levels/Monitors/Tielines: The user can choose the check box that he wants to query. The user then has to click the Get button. Once clicked, the get button gets disabled. Any further selection among the check boxes will enable the Get button once again.

- GetRoomNames: This button is used to get available rooms' information in system.

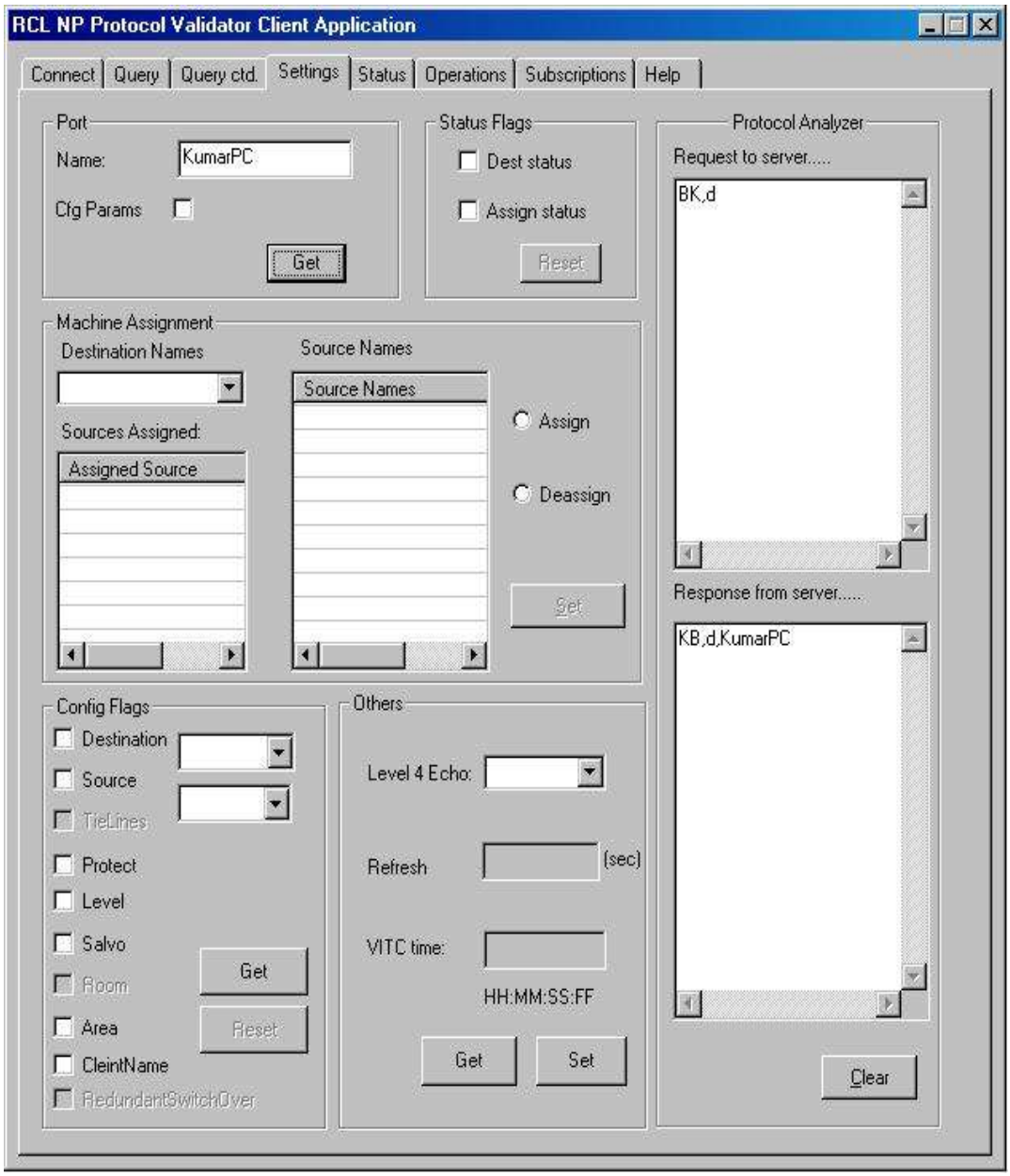
- GetRoomDetails: This button is used to get the specific information for a selected room. The specific information can be one of the following.

- Destinations that are part of the room.
- Tie lines that are part of the room.
- Rooms that are linked to the selected room.

The room name should be selected from the "Room Names" list box.

- Error Codes: When the user clicks this button, the list of error codes is displayed in the combo box.
- Error Definition: The user needs to specify whether he would like to see the error definition for the currently selected code in the combo, or whether he would like to see all error definitions. This he can do with the help of the two radio buttons provided i.e. All and Selected.
- Error Response Data: The results of the Error Definition query are displayed here. If the user clicked the 'Selected' radio button, the definition for that particular code is displayed. If he clicked the 'All' button, then the definitions for all the error codes are displayed here.

Settings Tab



Description of Controls

- Port Name and Cfg Params: If the user wants to see the Port Configuration Parameters in addition to the Port Name, the user needs to check the Config Params checkbox before clicking the Get button in the group box labeled ‘Port’.

- Source Names and destination Names: There are combo boxes for the user to select the source and destination for machine assignment or de-assignment. If the user tries to do an assignment or de-assignment when the combo boxes are empty, the GUI will prompt the user to query for the sources and destinations.

- Assign and Deassign: To perform an assign or a de-assign, the user can select a specific destination name from the “destination names” combo box and should select the source names “from source names” combo box. The user can now choose the

assign or de-assign operation by clicking on the appropriate radio button. Pressing on "Set" button will perform the chosen operation. The NP Protocol supports assigning (or de-assigning) one source to a destination at a time. The RCL Protocol supports assigning (or de-assigning) multiple sources to a destination using a single operation.

- Sources Assigned: This list box shows the list of source names assigned to a destination, which is selected in "Destination names" combo box. The list box is updated whenever assign or de-assign happens.

- Refresh Interval: When the user clicks the Get button, the Refresh Interval setting is displayed in the edit control. If he changes this value, the Set button gets enabled and the user can click it if he wishes to set a new refresh interval. [Note: If a non-zero refresh interval has been set, the user needs to send a message to the server before this time elapses, or else the NP/RCL connection with the server will be lost. So don't be surprised if you set it to 10 seconds and then don't do anything for 10 seconds, and then you find that you cannot communicate with the server anymore. This is a feature supported by NP/RCL.]This is supported only for the NP Protocol.

- Level 4 Echo: When the user clicks the Get button, the combo box will display whether it is ON or OFF. The user can change this setting and then click the Set button.

- Config Flags: When the user clicks the Get button, the edit control will display the Config flag value in Hex. The interpretation of this value will be displayed in the 'Response from server…' edit control.

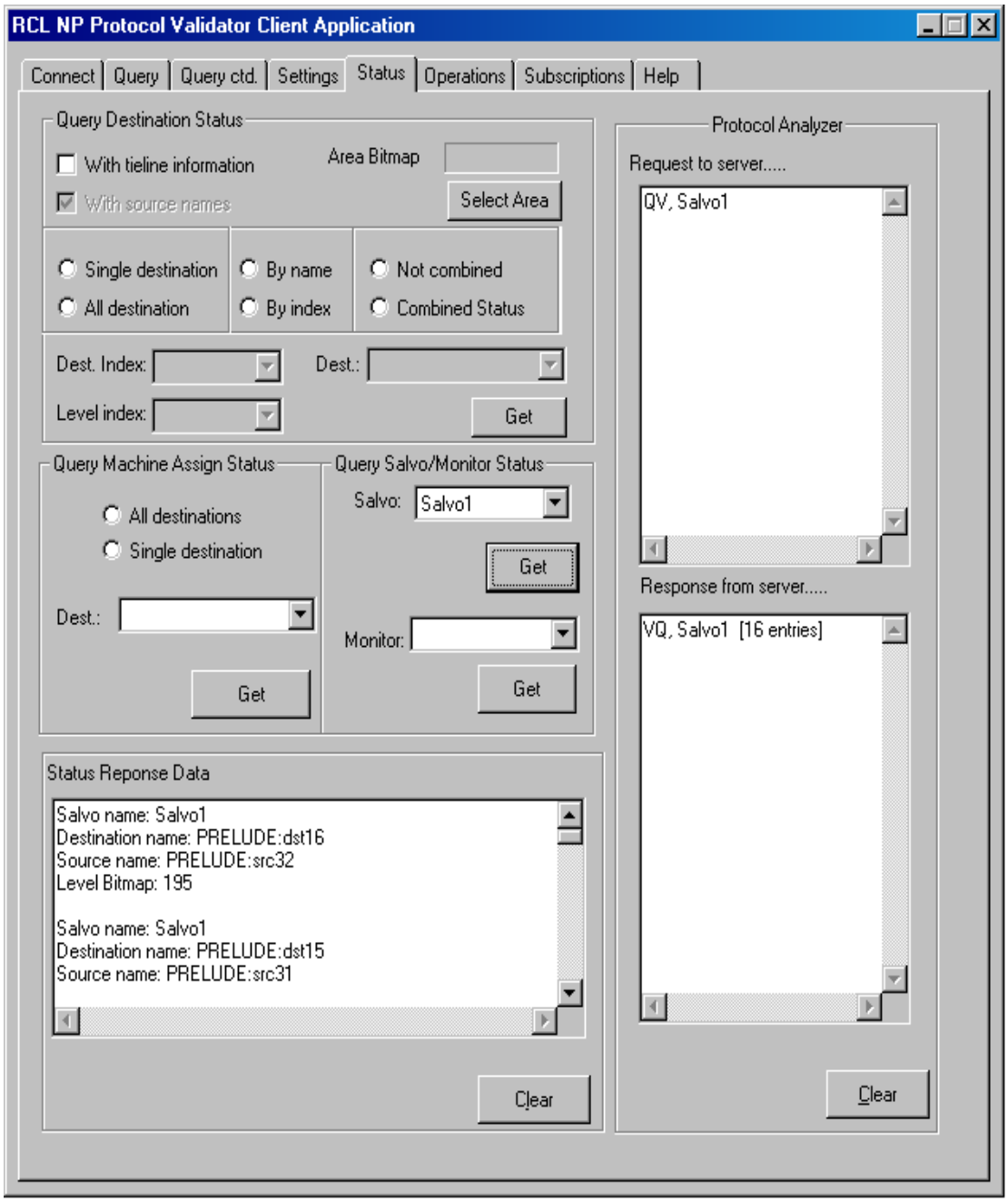
- Status Flags (Reset): Check boxes have been provided for the user to select which flags he wishes to reset. After checking the necessary flags, he has to click the Reset button.

- VITC time: Displays the current VITC time upon pressing the Get button.

- Dest status: Clears the flags associated with the QD command. After the check box is selected and the reset button is pressed, the next QD command will result in the server sending all destination statuses.

- Assign status: Clears the flags associated with the QA command. After the check box is selected, the next QA command will result in the server sending all assignment statuses

Status Tab



Description of Controls

This tab has several controls, as there are many different ways of querying the destination status.

- With tieline information: This check box needs to be checked, if the user wishes to obtain destination status with associated tie line information.
- With source names: This check box needs to be checked, if the user wishes to get the actual names of the sources taken to the specified destination(s). The source information will be given back even if there was no crosspoint to actually perform a take. If the user wants to know the true tally status of the destination he should leave

this unchecked. In this case, the server will send a NO_XPT status against the source if the take was not successful

- By name / Byindex: These radio buttons are intended for the user to specify whether he/she will be providing the name or index of the destination. They are not used if the Tieline Information check box is checked, since in such a case the destination should always be passed by name.

- Single destination/All destinations: The user must select the 'Single destination' radio button if he wants the status for a particular destination. If he wants the status for all destinations that have changed since his previous query, he needs to select the 'All destination' radio button.

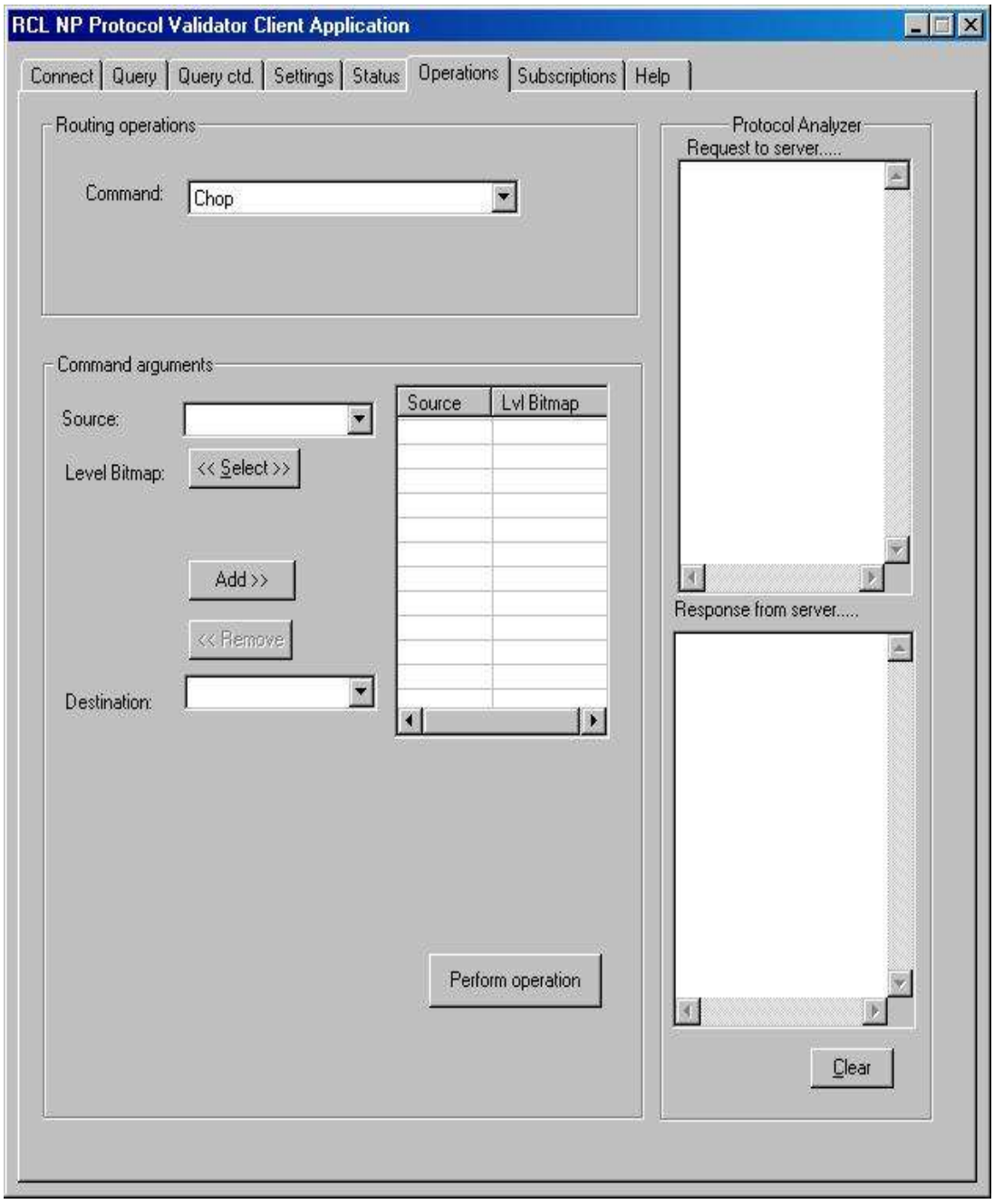
- Not Combined / Combined status: This is for the user to specify whether he wants the status of a destination on a particular level or whether he would like a combined level status.

- Dest. Index/Level Index/Dest.: These combos will be enabled/disabled accordingly depending on which parameters are needed. The user needs to make the necessary selections from these combo boxes. The user finally needs to click the View Status button to obtain the required status. The status information will be displayed in the 'Status Response Data' edit control at the bottom of the screen.

- Query Machine Assign Status: The user can get the assignment status for single destination or all destinations by selecting the appropriate radio button and clicking on the "Get" button of the same group. If the user selects the "Single destination" radio button the destination name should be select from "Dest" combo box. The assignment status is shown in the "Status Response Data" edit box.

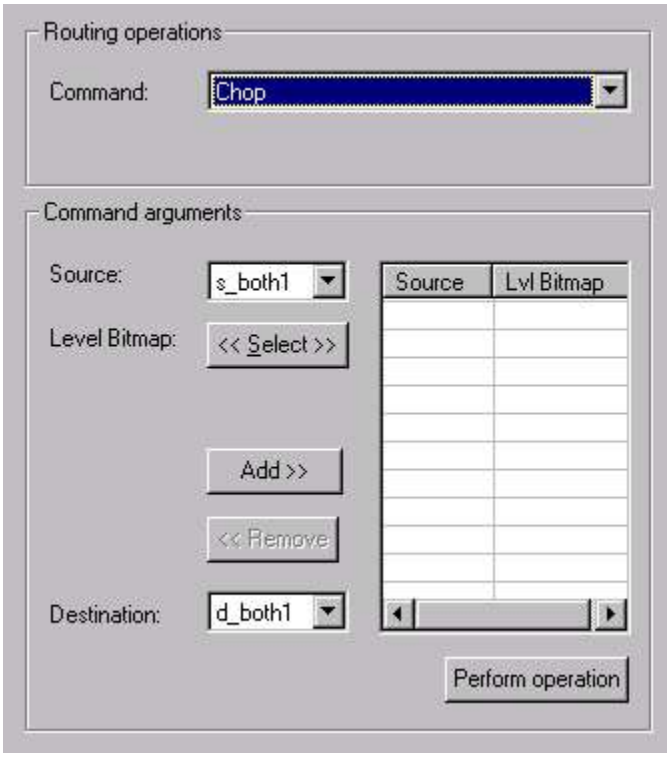
- Query Salvo/Monitor status: The user can query the salvo or monitor status by selecting salvo or monitor name form appropriate combo boxes and click on the "Get" button below the combo box, then the response shown in the "Status Response Data" edit box. If the user tries to get salvo or monitor status when the appropriate combo boxes are empty, the GUI will prompt a message to query for the salvo or monitors names.

Operations Tab



This is the screen that is displayed when operations tab is selected. The view of the tab changes as we choose the various ‘Command’ options from the combo box.

Chop



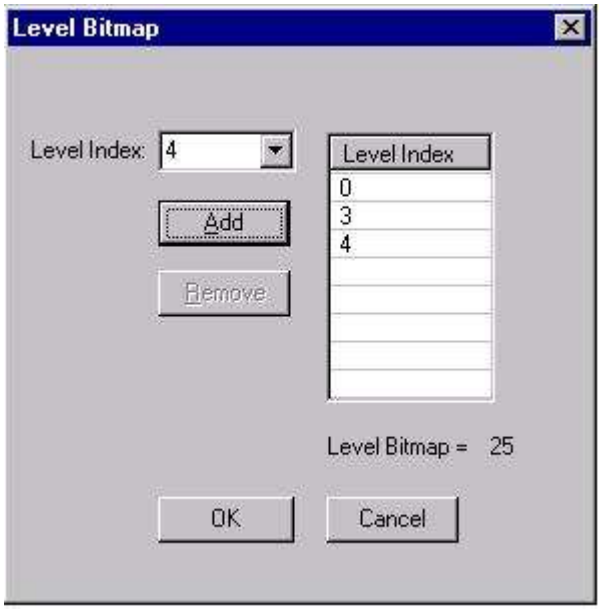
- Source: This contains the listing of all sources available for the operation.
- Level Bitmap (Select): This pops up the level bitmap dialog box where the user can choose the levels on which the operation needs to be performed.
- Add: Add the selected level(s) from the level bitmap dialog box for the corresponding source into (Source/Lvl Bitmap) list box.
- Remove: Removes the selected row from (Source/Lvl Bitmap) list box.
- Destination: This contains the listing of all destinations available for the operation.
- Perform Operation: Performs a chop.

Clear Tielines

- Destination: This contains the listing of all destinations available for the operation. Note: This is the only control, which, is enabled for this operation, and hence the operation is performed on destination(s).

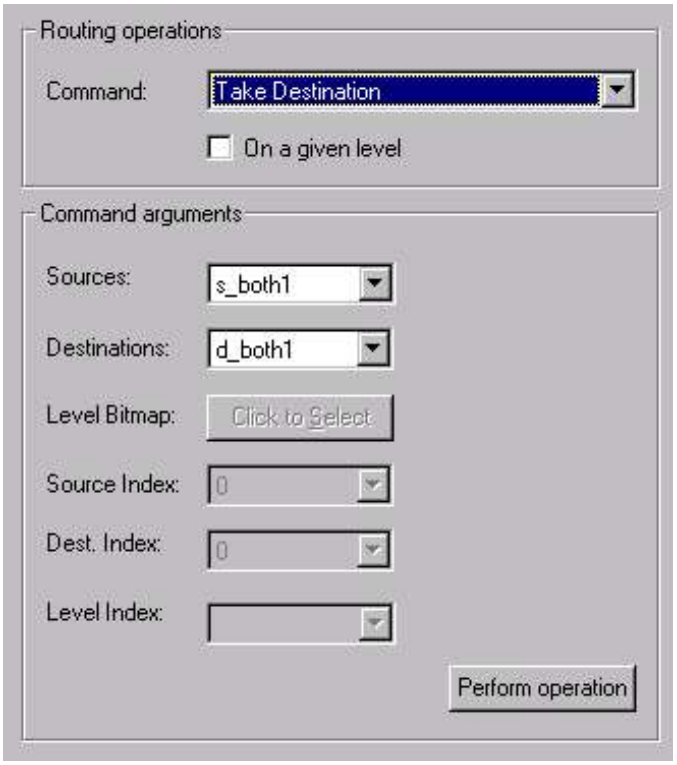
- Perform Operation: Performs a clear tieline operation.

Level Bitmap



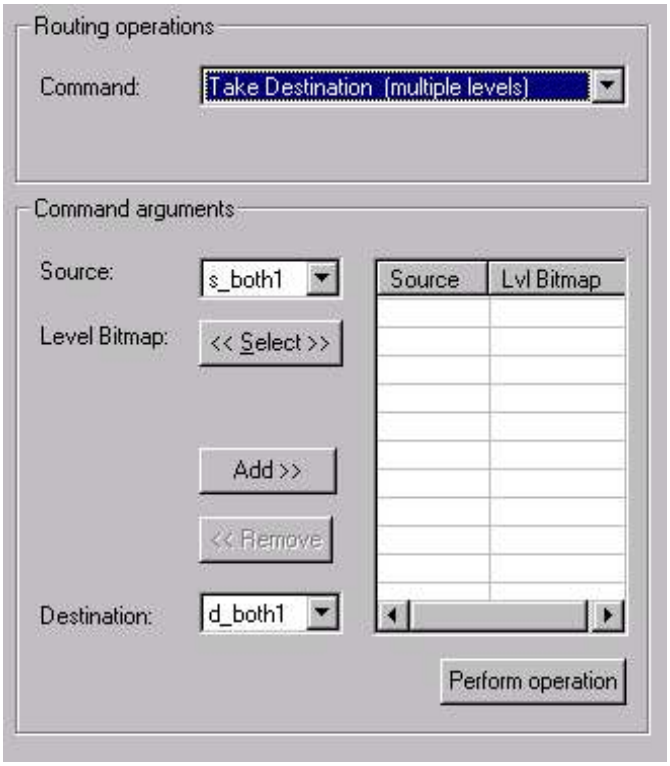
- Level Index: This contains the listing of all levels available for the operation
- Add: Add the selected level(s) from the 'level index' combo box into the 'level index' list box.
- Remove: Removes the selected row from 'level Index' list box.
- OK: Passes the level bitmap(s) chosen here to the application that uses it.
- Cancel: Cancels the operation.

Take Destination



- Source: This contains the listing of all sources available for the operation.
- Destination: This contains the listing of all destinations available for the operation.
- On a given level: This check box enables the level bitmap option, if the user wants to do a take on a particular level(s).
- Level Bitmap (Select): This pops up the level bitmap dialog box where the user can choose the levels on which the operation needs to be performed.
- Perform Operation: Performs a take.

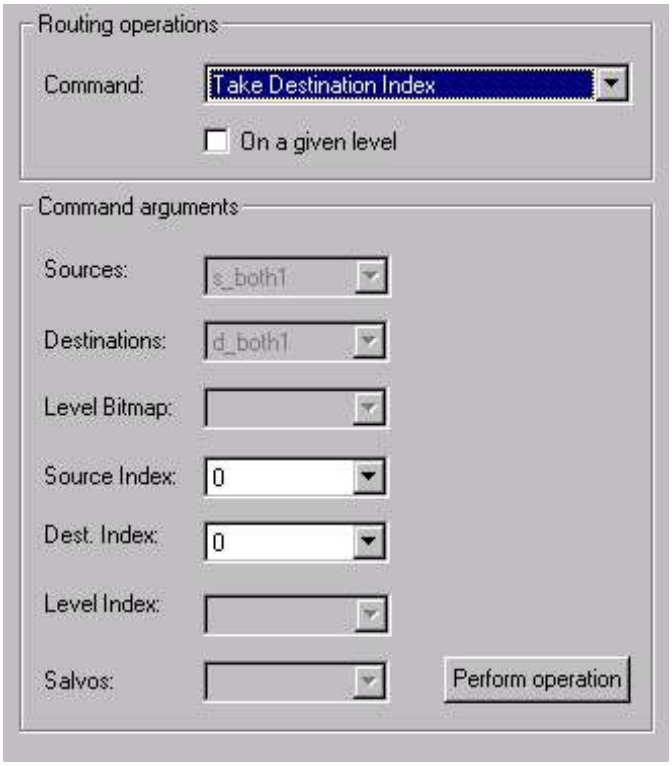
Take Destination (multiple levels)



Please refer to the control description given for the chop operation, it remains same for this operation too except for the following.

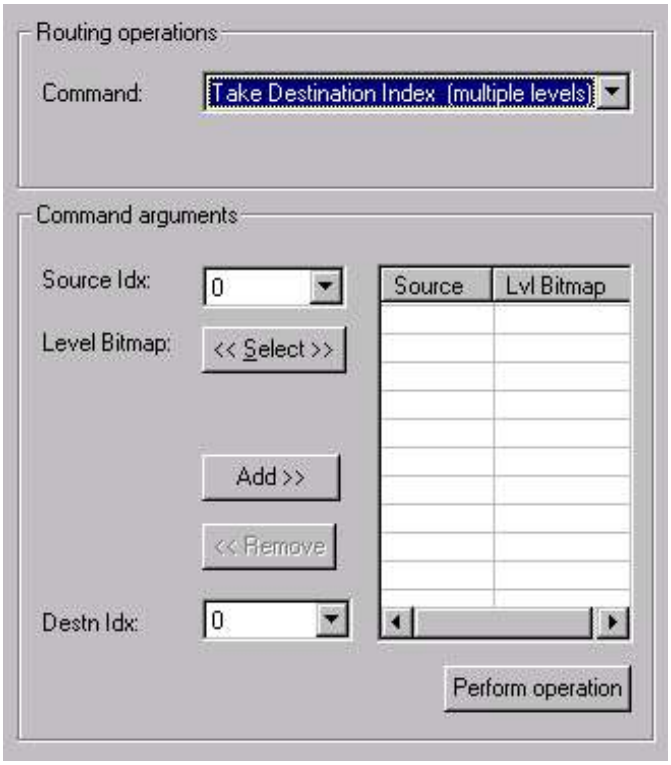
- Perform Operation: Performs a take on multiple levels.

Take Destination Index



- Source Index: This contains the listing of all source indexes available for the operation.
- Dest. Index: This contains the listing of all destination indexes available for the operation.
- On a given level: This check box enables the level bitmap option, if the user wants to do a take on a particular level(s).
- Level Bitmap (Select): This pops up the level bitmap dialog box where the user can choose the levels on which the operation needs to be performed.
- Perform Operation: Performs an indexed-based take.

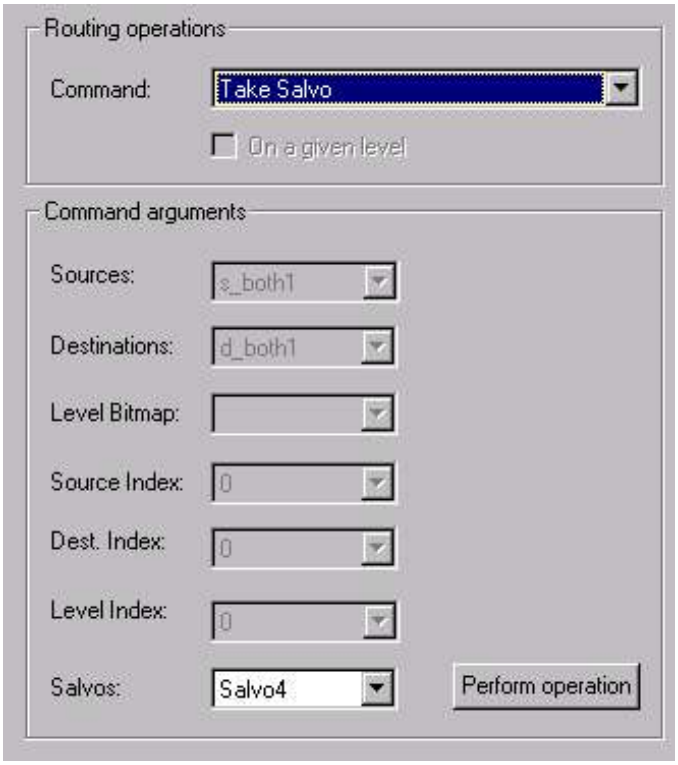
Take Destination Index (multiple levels)



Please refer to the control description given for the chop operation, it remains same for this operation too except for the following.

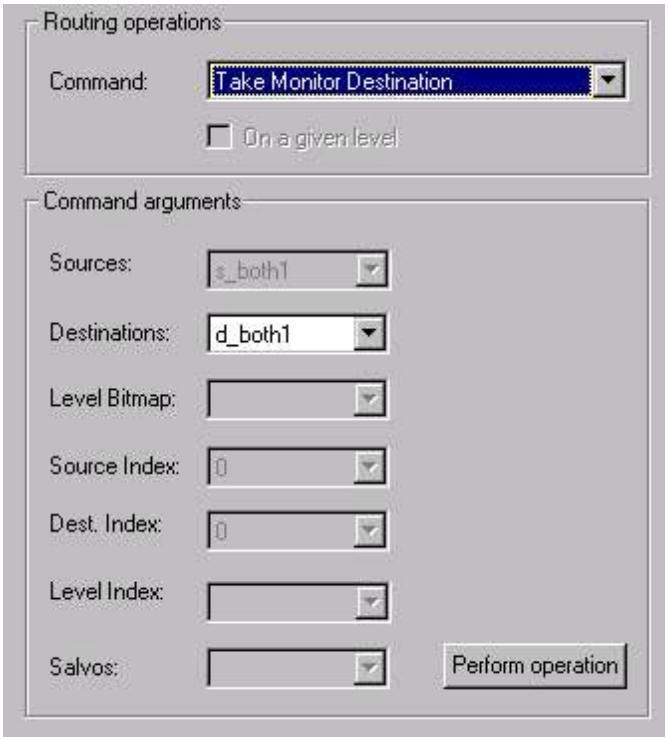
- Source Idx: This contains the listing of all source indexes available for the operation.
- Perform Operation: Performs an indexed-based take on multiple levels.

Take Salvo



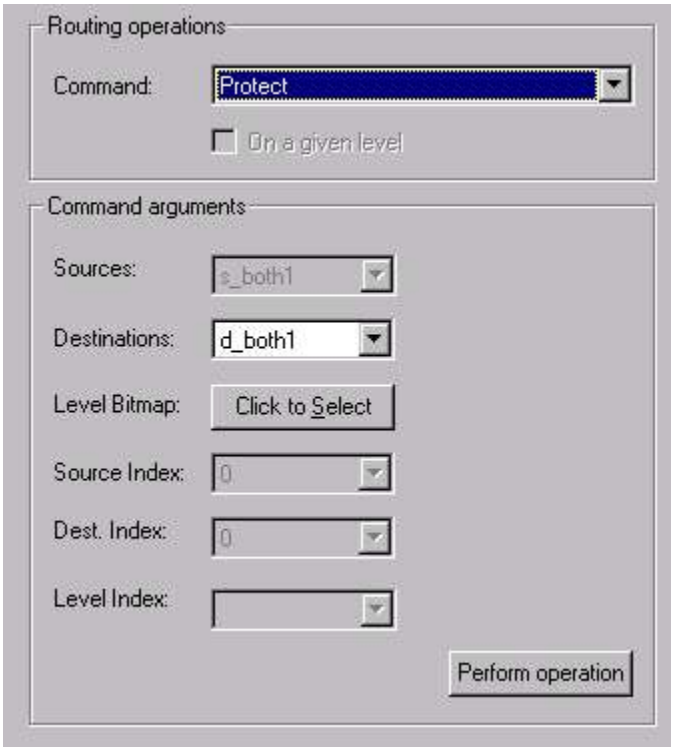
- Salvos: This contains the listing of all salvos available for the operation.
- Perform Operation: Performs a salvo operation.

Take Monitor Destination



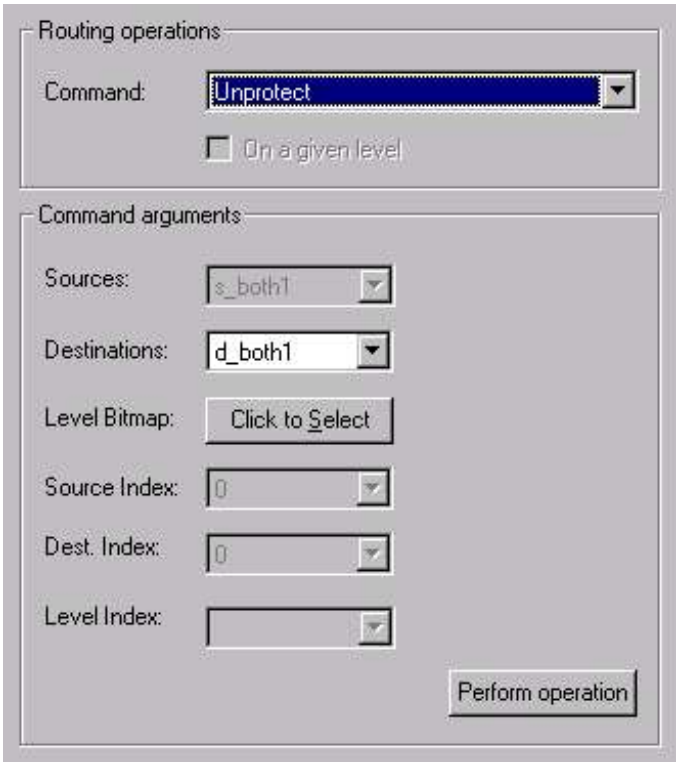
- Destinations: This contains the listing of all destinations available for the operation.
- Perform Operation: Performs a monitor operation.

Protect



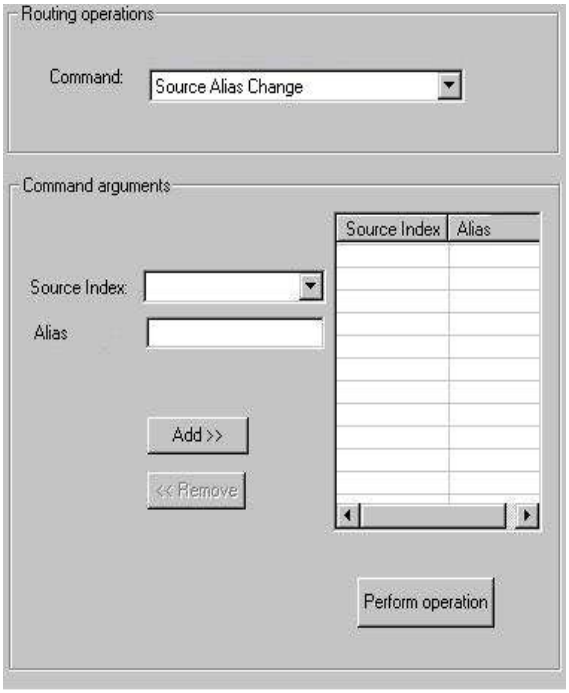
- Destinations: This contains the listing of all destinations available for the operation.
- Level Bitmap (Select): This pops up the level bitmap dialog box where the user can choose the levels on which the operation needs to be performed.
- Perform Operation: Performs a protect operation.

UnProtect



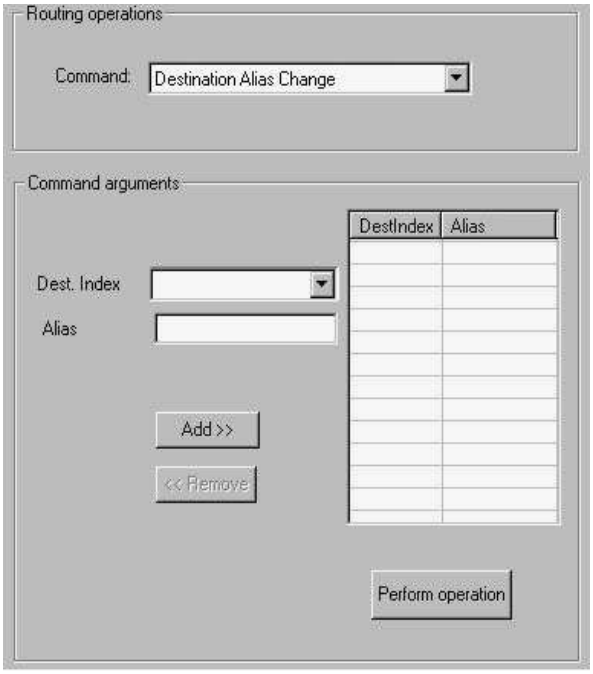
- Destinations: This contains the listing of all destinations available for the operation.
- Level Bitmap (Select): This pops up the level bitmap dialog box where the user can choose the levels on which the operation needs to be performed.
- Perform Operation: Performs an unprotect operation.

Source Alias Change



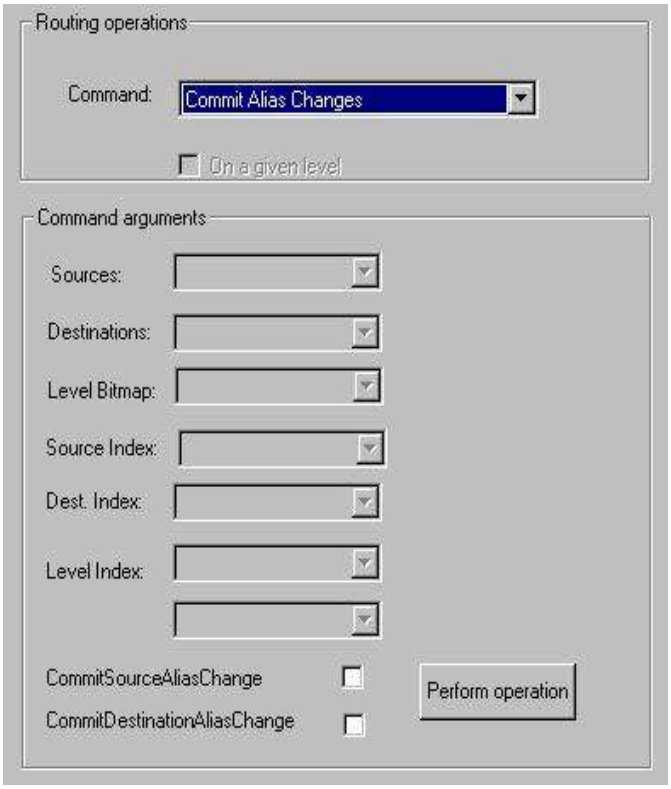
- Source Index: This contains the list of all source indices available for the operation.
- Alias : The user enters the fully qualified alias (AreaName:Alias) for the source index selected from the “SourceIdx” combo box.
- Add: Adds the source index and alias that are selected by user to Source Index/Alias list box.
- Remove: Removes the selected row from (Source Index/Alias) list box.
- Perform Operation: Performs source alias change operation.

Destination Alias Change



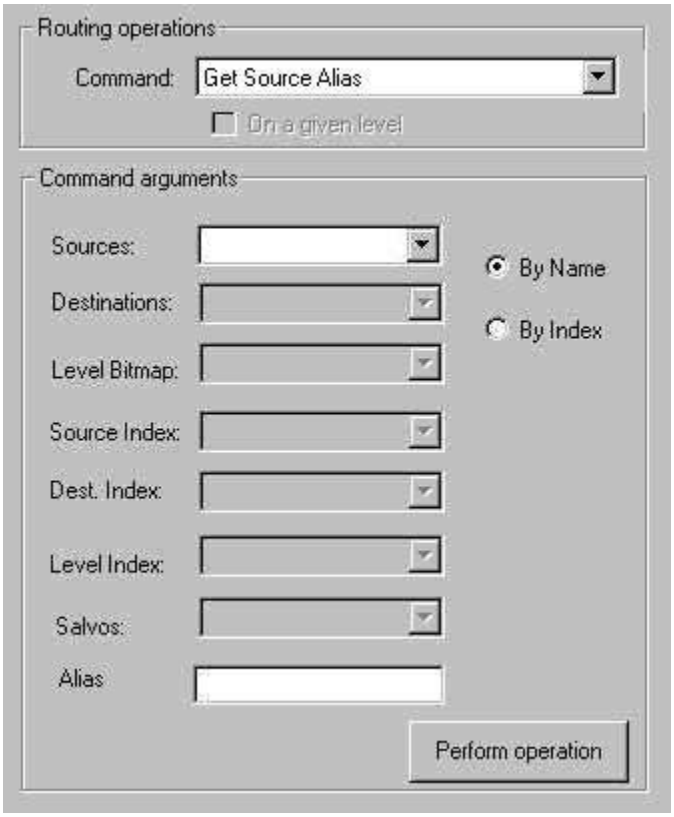
- Dest. Index: This contains the list of all destination indices available for the operation.
- Alias: The user enters the fully qualified alias (AreaName:Alias) for the destination index selected from the “Dest Index” combo box.
- Add: Adds the destination index and alias that are selected by user to DestIndex/Alias list box.
- Remove: Removes the selected row from (DestIndex/Alias) list box.
- Perform Operation: Performs destination alias change operation.

Commit Alias Changes



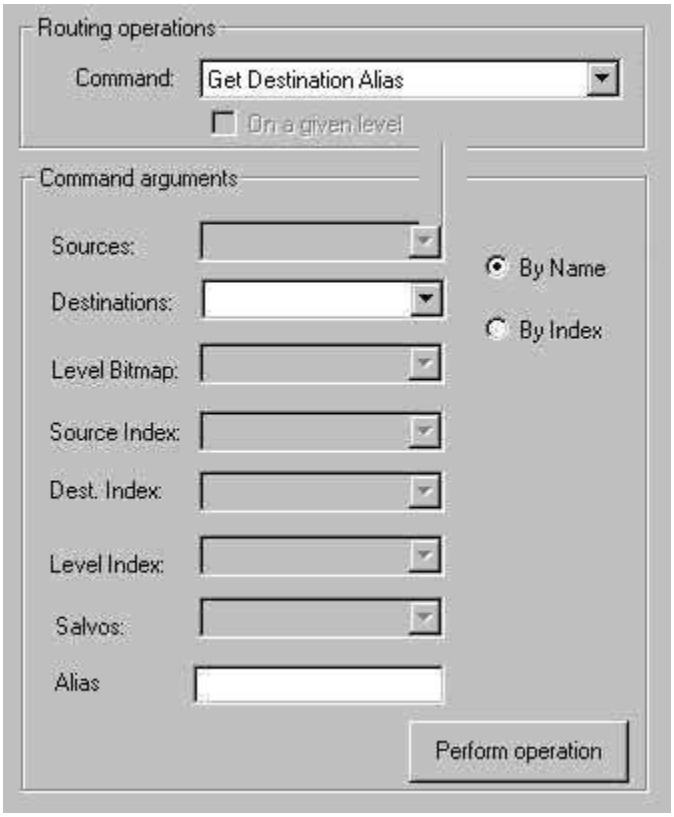
- CommitSourceAliasChange: The user can select the check box to commit the source alias changes.
- CommitDestinationAliasChange: The user can select the check box to commit the destination alias changes.
- Perform Operation: Performs the commit source/destination alias change operation.

Get Source Alias



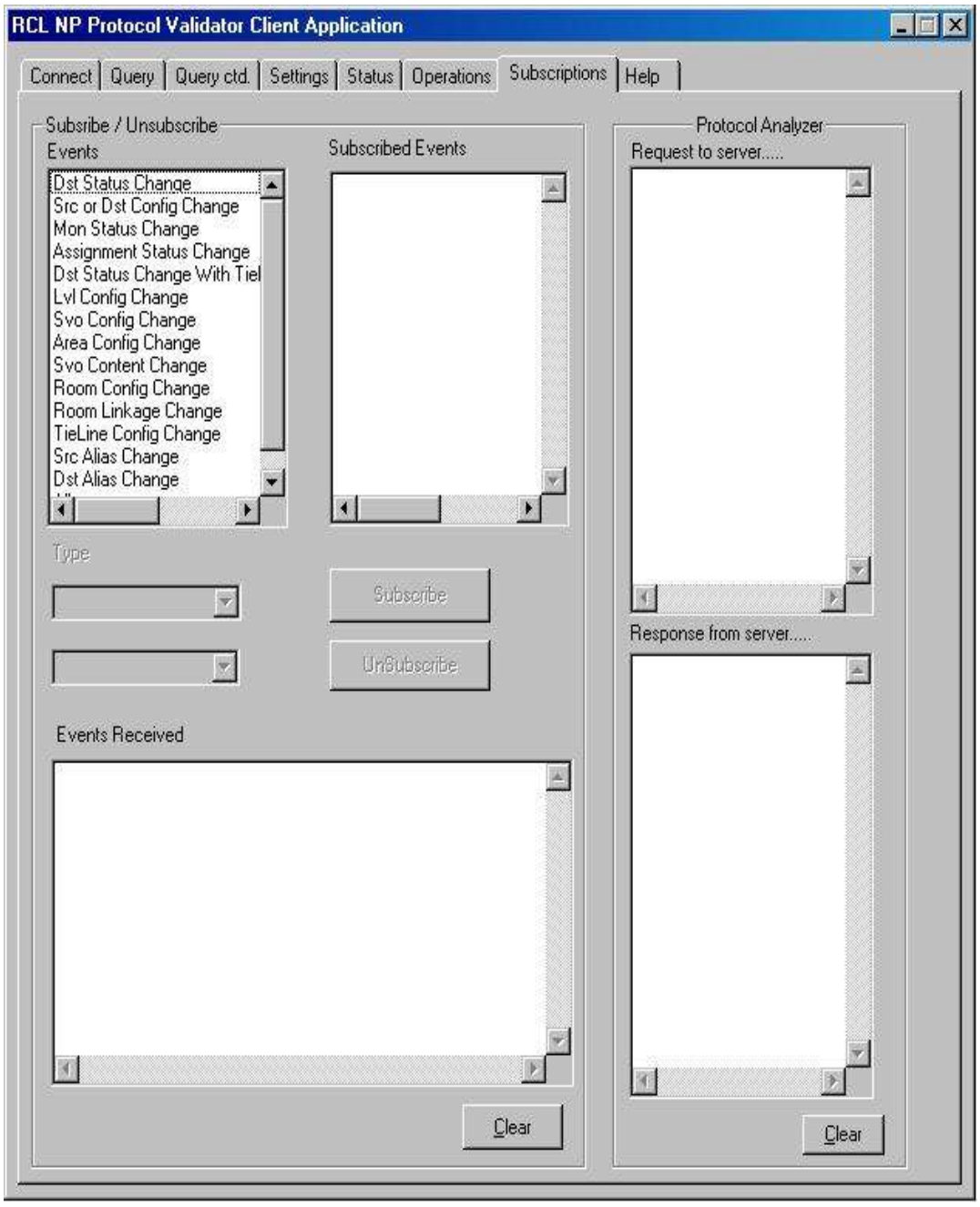
- Sources: This contains the list of all source names available for the operation.
- Alias: This field contains the alias for source name selected in the “Sources” combo box.
- Perform Operation: Performs the get source alias for given source name/index operation.

Get Destination Alias



- Destinations: This contains the list of all destination names available for the operation.
- Alias: This field contains the alias for destination name selected in the "Destinations" combo box.
- Perform Operation: Performs the get destination alias for given destination name/index operation.

Subscriptions Tab



Description of Controls

- Events: Contains listing of all subscription commands. Following are the commands that are listed.

- Dst Status Change: A RCL client can subscribe for the destination status of one or a group of destinations. Whenever the status of a destination changes, all the clients that have subscribed for that destination receive a notification.

- Src or Dst Config Change: A RCL client can subscribe for destination or Source Configuration modifications. Whenever the source or destination configuration

changes, all the clients that have subscribed for the configuration status change receive a notification.

- Lvl Config Change: A RCL client can subscribe for level configuration changes. Whenever the level configuration changes, all the clients that have subscribed for the level configuration status change receive a notification..

- Svo Config Change: A RCL client can subscribe for salvo configuration changes. Whenever the salvo configuration changes, all the clients that have subscribed for the salvo configuration status change receive a notification.

- Area Config Change: A RCL client can subscribe for area configuration changes. Whenever the area configuration changes, all the clients that have subscribed for the area configuration status change receive a notification.

- Svo Content Change: A RCL client can subscribe for salvo content Configuration changes. Whenever the salvo content changes, all the clients that have subscribed for the salvo content status change receive a notification.

- Mon Status Change: A RCL client can subscribe for the monitor status changes of one or a group of monitors. Whenever the status of monitor changes, all the clients that have subscribed for that monitor receives a notification.

- Assignment Status Change: A RCL client can subscribe for the destination assignment status change of one or a group of destinations. Whenever the assignment status of a destination changes, all the clients that have subscribed for that destination receive a notification

- Dst Status Change With TieLine: A RCL client can subscribe for the destination status change with tie line information of one or a group of destinations. Whenever the status of a destination changes, all the clients that have subscribed for that destination receive a notification.

- Room Config Change: A RCL client can subscribe for Room configuration changes. Whenever the room configuration changes (room is created/modified), all the clients that have subscribed for the room configuration change receive a notification.

- Room Linkage Change: A RCL client can subscribe for Room linkage changes. Whenever the room linkage changes, all the clients that have subscribed for the room linkage change receive a notification.

- Tie Line Config Change: A RCL client can subscribe for Tie Line configuration changes. Whenever the Tie Line configuration changes (tie line is created/modified), all the clients that have subscribed for the tie line configuration change receive a notification.

- Source Alias Change: A RCL client can subscribe for the source alias changes of specific area or for all areas. Whenever the source alias change happens, all the clients that have subscribed for that area receive a notification.

- Dst Alias Change: A RCL client can subscribe for the destination alias changes of specific area or for all areas. Whenever the destination alias change happens, all the clients that have subscribed for that area receive a notification.

- All: A RCL client can subscribe for all of the above-mentioned Events. If any changes occur in the status of any of the above events, a corresponding notification is sent to the client that subscribed for it

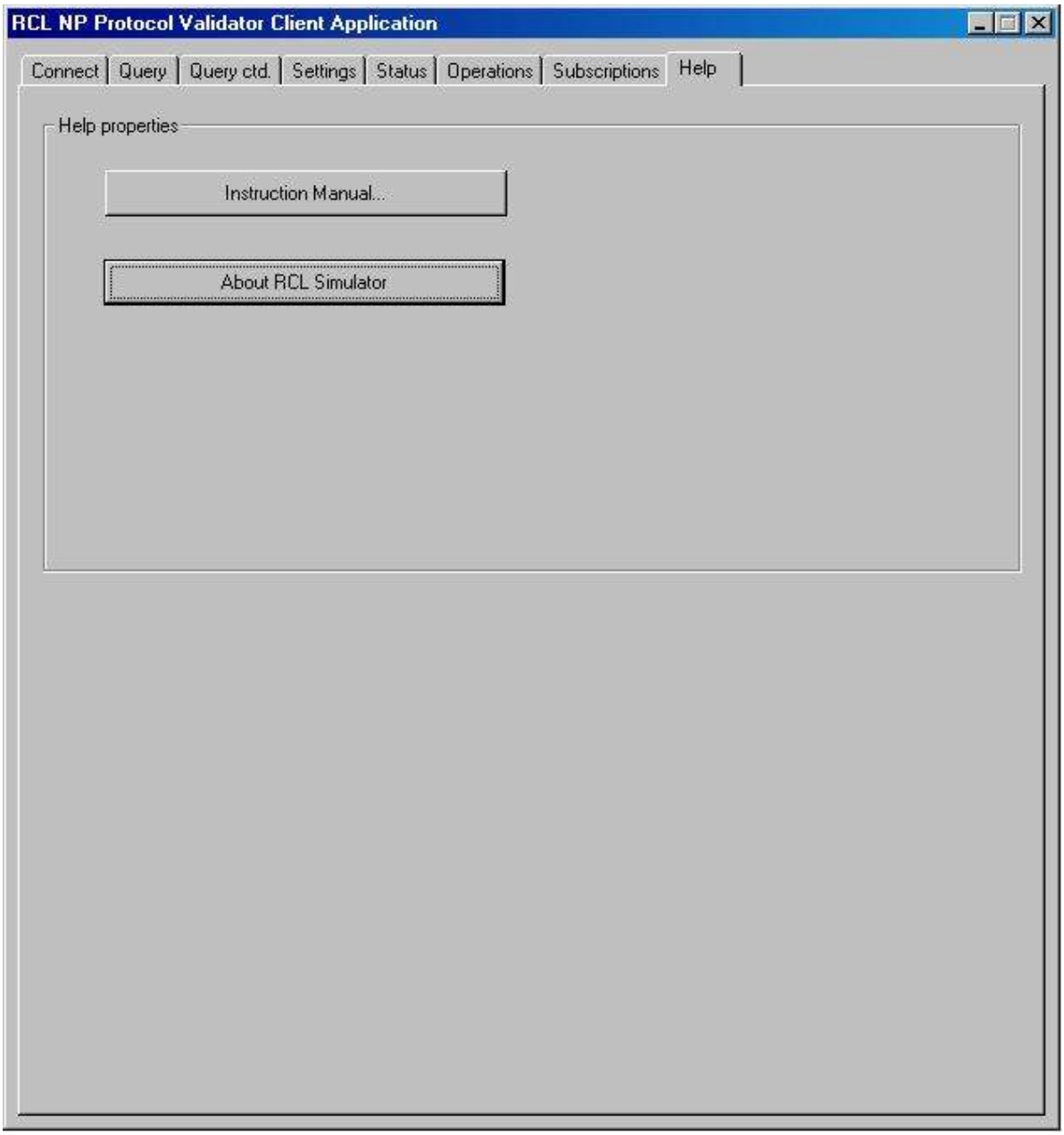
Subscribe requests for status/configuration change information. The specific information for which the client subscribes is decided based on the subscription type and parameters sent as part of the SB command.

This screen contain two buttons, Subscribe and Unsubscribe.

- Subscribe: When the subscribe button is pressed, it adds the selected events into the 'Subscribed Events' control from the 'Events' control. Subsequently the 'Request to Server' control is populated with the corresponding command.

- Unsubscribe: When the unsubscribe button is pressed, it removes the selected Events from the 'Subscribed Events' control and unsubscribe the same. Subsequently the 'Request to Server' control is populated with the corresponding command.

Help Tab



Description of Controls

- Instruction Manual: It opens the Instruction Manual RCLNPClientUserGuide.pdf file. The instruction manual is provided along with Encore installation CD .It should be placed in the same folder as the RCLNPClientGUI application.

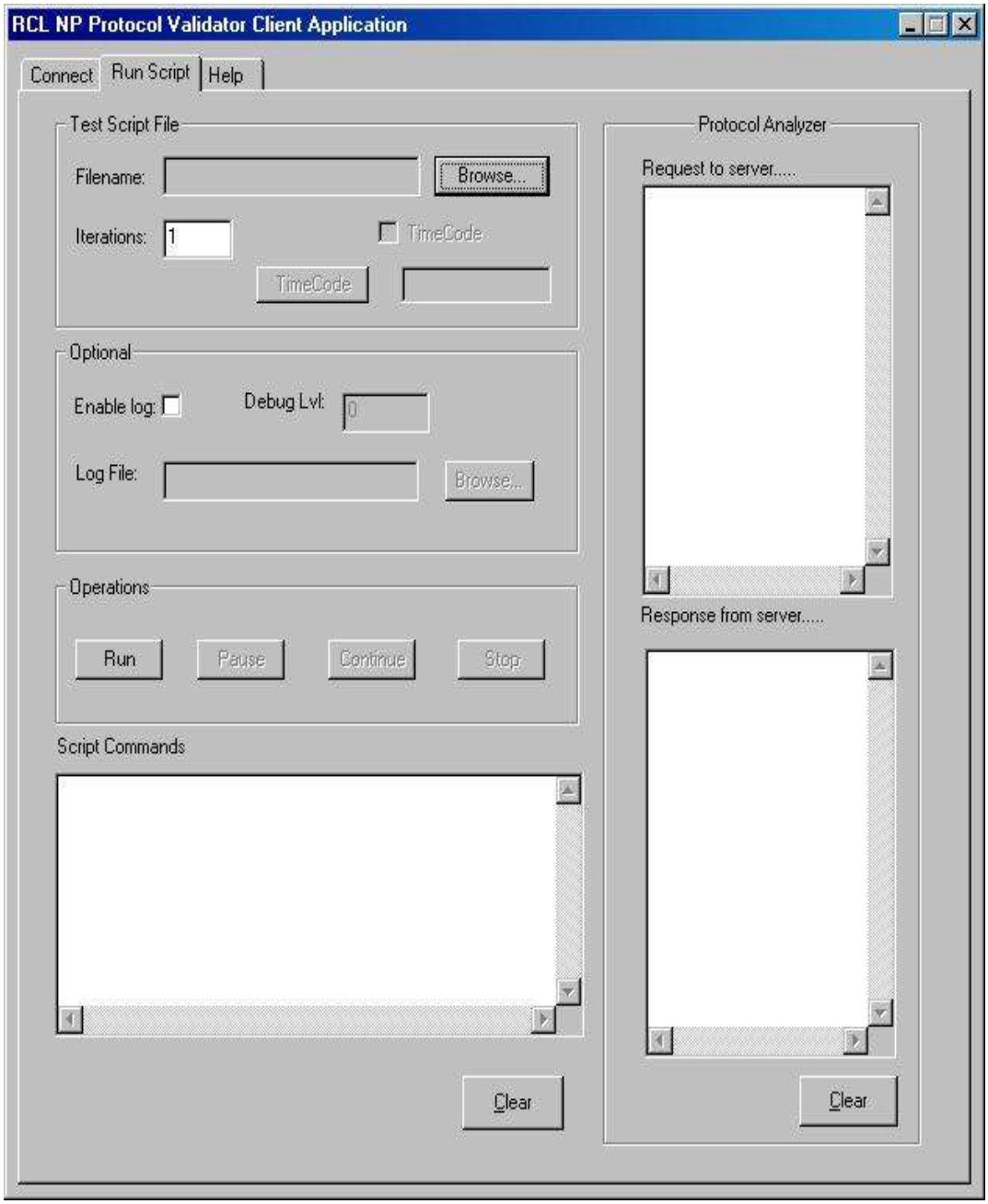
- About RCL Simulator: It pops up an 'About RCL Client Simulator' message box as shown below


About RCL Client Simulator
Version : V1.7.3b2
Date : Oct 16 14:25:00 2007
Copyright (c) Grass Valley Group Inc. 2007
OK

Test Script Mode

Overview

This is used in simulating an automation interface. The test script mode helps the user execute as many commands as he/she wishes as many number times as he/she wants the same set of commands to be repeated. This is achieved with the help of a script file that the user populates in a specified manner described in this section. The script file is loaded and an iteration count is specified for the number of times the script file has to be executed. The execution can be paused, continued or stopped. This gives the user total power to simulate his/her automation environment.



Description of Controls

- Filename: Accepts the filename of the script-file along with the path either directly from the keyboard or the user can navigate to the script-file using the browse button.

- Iterations: If the user wishes to run the script-file either once or repetitively, he can specify the number of times he/she would like to do that in the 'Iterations' edit box. The maximum number of iterations that can be supported is 10000.

- Enable Log: This check box can be selected if the user wants to log the Server response data into a file. Selecting the 'Enable Log' checkbox will result in a Browse button being enabled for specifying the name of the log file.

- Log file: Accepts the filename of the log-file along with the path either directly from the keyboard or the user can navigate to the log-file using the browse button.

- Debug Level: The value typed in this edit box will be used to determine the level of detail to which the Server responses are logged into the log-file. This value can range from 0 to 4. Specifying a value of 0 will result in an empty log file, while 4 will result in the maximum amount of logged data.

- Run: After all the above selections and values are keyed in, the user simply needs to click the 'Run' button to initiate the running of the script-file. In case of the timed operations a check is done to ensure that the script start time code is greater (occurs in future) than the current VITC time code. If the script, start time code is lesser than the current VITC time code the GUI pops an error message.

- Pause: If the execution of the script-file has to be halted for sometime, then the user can click the pause button. Once this is selected then the 'continue' button is the only option the user has.

- Continue: If the execution of the script-file has to be resumed from the paused state, then the user can click the continue button. Once this is selected then the options available for the user are 'pause' and 'stop'.

- Stop: If the execution of the script-file has to be terminated, then the user can click the stop button. Once this is selected the 'run' option is available to the user.

- Time Code (check-box): Please refer section Timed Operations

- Time Code (button): Please refer section Timed Operations.

- Script Commands: All the commands read from the Script File will be displayed in this window. Note: If lines in the Script File are preceded with a '#', these lines are considered as commented entries and will be ignored during execution.

How to write a script

The general format for the script commands except the timestamp commands.

```
t, Command Name, Command Parameters.
```

Alternately the script can also be created using the command (and sub parameter) enumeration as shown below.

```
t, Command Enumeration Type, Command Parameters.
```

Refer the Commands names supported by NP/RCL Protocols below for the list of Command Names (or command Enumeration Type), and refer the separate for the command parameters.

Table 1: Commands names supported by NP/RCL Protocols

Name	Command Enumeration Type	Sub Parameters
BK	QueryBackGroundParameters	Refer BK Command Parameters
QN	QueryNames	Refer QN Command Parameters
TD	TakeByName	
TI	TakeByIndex	
TA	TakeBreakawayByName	
TJ	TakeBreakawayByIndex	
TS	TakeSalvo	
TM	TakeMonitor	
PR	Protect	
UP	UnProtect	
QC	QueryCombinedDestinationStatus	
QD	QueryDestinationStaus	
QJ	QueryDestinaionStatusByIndex	
QK	QueryDestinationStatusWithTieLineInfoByIndex	
QI	QueryDestinationStatusOnSpecificLevelByIndex	
Qd	QueryDestinationStatusWithXptInfo	
Qi	QueryDestinaionStatusByIndexWithXptInfo	
Qj	QueryDstStatusOnSpecificLevelByIndexWithXptInfo	
QV	QuerySalvoStatus	
QP	QuerySalvoWithLockInfo	
QM	QueryMonitorStatus	
QA	QueryAssignmentStatus	
QR	QueryRoomDetails	Refer QR Command Parameters
QE	QueryErrorDefinition	

Name	Command Enumeration Type	Sub Parameters
AS	Assign	
DA	Deassign	
CA	AliasChange	Refer CA Command Parameters
CM	CommitAlias	Refer CM Command Parameters
SB	Subscription	Refer SB and UB Command Parameters
UB	UnSubscription	Refer SB and UB Command Parameters
GA	GetAliasName	Refer GA Command Parameters

Table 2: BK Command Parameters

Parameter	Parameter Enumeration Type
N	SystemName
R	SoftwareRevision
T	SoftwareVersion
t	PrtocolSoftwareVersion
F	ConfigFlags
f	ClearConfigFlags
D	ResetDestStatus
A	ResetAssignStatus
P	ConfigParams
E	Level4Echo
d	ClientName
I	RefreshRate

Table 3: QN Command Parameters

Parameter	Parameter Enumeration Type
S	SrcNames
IS	SrcIndices
D	DstNames
ID	DstIndices
A	AreaNames
IA	AreaIndices
N	MonitorNames
R	RoomNames
T	TieLineNames
L	LevelNames
V	SalvoNames
SA	SourceAliasNames
DA	DstAliasNames

Table 4: CA Command Parameters

Parameter	Parameter Enumeration Type
SA	SrcAliasChange
DA	DstAliasChange

Table 5: QR Command Parameters

Parameter	Parameter Enumeration Type
L	RoomLinks
D	RoomDestinations
T	RoomTieLines

Table 6: SB and UB Command Parameters

Parameter	Parameter Enumeration Type
DS	DestinationStatusChange
CN	ConfigurationChange
SV	SlavoContentChange
SL	SalvoConfigurationChange
LV	LevelConfigurationChange
AR	AreaChange
MN	MonitorStatusChange
AS	AssignmentStatusChange
RL	RoomLinkageChanges
RM	RoomConfigurationChanges
TL	TieLineConfigurationChanges
DK	DestinationStatusChangeWithTieLine

Table 7: CA Parameters

Parameter	Parameter Enumeration Type
SA	SourceAliasNameChange
DA	DestinationAliasNameChange

Table 8: GA Command Parameters

Parameter	Parameter Enumeration Type
SN	SourceName
SI	SourceIndex
DN	DestinationName
DI	DestinationIndex

Table 9: CM Command Parameters

Parameter	Parameter Enumeration Type
SA	SrcAliasCommit

DA	DstAliasCommit

Special script commands:

Wait

The command delays the execution of the script commands specified time in command.

The command format is:

```
t,Wait,TimeinMilliseconds
```

Loop

The command executes the set of commands for specific iterations.

The command format is:

```
Loop[, Iteration Count]
Script commands
EndLoop
```

If Iteration Count left blank then this loop will execute infinitely.

Sample Script Commands

The sample script commands are given below for a few queries.

The commands listed below use the indices for the source, destinations, areas and levels. All the indices are zero based hexadecimal values. For example, if the system has 32 sources the source index ranges from 0000X to 001fX.

- Query the source names in all the areas.

```
t,QN,S
  (OR)
t, QueryNames, SrcNames
```

- Query source names for the areas 1, 5 and 11 (note the bitmap formation)

```
t,QN,S, 0000000000000822
  (OR)
t,QueryNames,SrcNames, 0000000000000822
```

- Query the configuration flags

```
t,BK,F
  (OR)
t, QueryBackGroundParameters, ConfigFlags
```

- Set the client refresh rate as 100 seconds

```
t,BK,I,100
  (OR)
t, QueryBackGroundParameters, RefreshRate,100
```

- Take a source SRC10 on destination DST12 from area AreaX by name

  For RCL Protocol:

  ```
  t,TD,, AreaX: DST12,AreaX:SRC10
    (OR)
  t, TakeByName,, AreaX: DST12,AreaX:SRC10
  ```

  For NP Protocol (possible only if AreaX is configured as the active area for the client on Encore):

  ```
  t,TD, DST12, SRC10
    (OR)
  t, TakeByName, DST12, SRC10
  ```

- Take a source SRC10 on destination DST12 from area AreaX on levels 2 and 5 by name.

  For RCL Protocol:

  ```
  t,TD,, AreaX: DST12,AreaX:SRC10,00000024
    (OR)
  t, TakeByName,, AreaX: DST12,AreaX:SRC10,,00000024
  ```

  For NP Protocol (possible only if AreaX is configured as the active area for the client on Encore):

  ```
  t,TD, DST12, SRC10, 00000024
    (OR)
  t, TakeByName, DST12, SRC10, 00000024
  ```

- Take a source index 8 on destination index 5 from area index 2 by index

  For RCL Protocol:

  ```
  t,TI,,02:0005,02:0008
    (OR)
  t, TakeByIndex,, 02:0005,02:0008
  ```

  For NP Protocol (possible only if area 1 is configured as the active area for the client on Encore)

  ```
  t,TI, 0005,0008
    (OR)
  t, TakeByIndex,0005,0008
  ```

- Take a source index 31 on destination index 19 from area index 10 by index

  For RCL Protocol:

  ```
  t,TI,,0a:0013,0a:001f
    (OR)
   t, TakeByIndex,, 0a:0013,0a:001f
  ```

  For NP Protocol (possible only if area 11 is configured as the active area for the client on Encore)

  ```
  t,TI, 0013,001f
    (OR)
  t, TakeByIndex, 0013,001f
  ```

- Take a source SRC10 on level 1 and SRC12 on level 8 of destination DST20 from area AreaX by name.

  For RCL Protocol:

```
t,TA,, AreaX: DST20,02,AreaX:SRC10,00000002,
AreaX:SRC12,00000100
  (OR)
t, TakeBreakawayByName,, AreaX:
DST20,02,AreaX:SRC10,00000002, AreaX:SRC12,00000100
```

  For NP Protocol (possible only if AreaX is configured as the active area for the client on Encore)

```
t,TA, DST20,02, SRC10,00000002, SRC12,00000100
  (OR)
t, TakeBreakawayByName, DST20,02, SRC10,00000002,
SRC12,00000100
```

- Protect the Destination DST10 on level 1 (Note, level by level protect is NOT supported in Encore yet. Hence in Encore, it will result in all level protect)

```
t,PR, DST10,00000002
  (OR)
t, Protect, DST10,00000002
```

- Query the destination status for all areas (in case of NP protocol, this will result in the status response for the client's active area)

```
t ,QD
  (OR)
t, QueryDestinationStaus
```

- Query the destinations status for the areas 2 and 4 (only for RCL Protocol. Not available for NP)

```
t,QD,0000000000000014
  (OR)
t, QueryDestinationStaus, 0000000000000014
```

- Query the destination status for destination DST10 from the area AreaX (For RCL protocol only)

```
t,QD,AreaX:DST10
  (OR)
t, QueryDestinationStaus,AreaX:DST10
```

- Query the destination status for destination index 10 from the area 2 by index (For RCL protocol only)

```
t,QJ,2:000a
  (OR)
t, QueryDestinaionStatusByIndex,2:000a
```

- Query the destination status for destination index 10 from the area 3 for level 3 by index (For RCL protocol only)

```
t,QI,3:000a,03
  (OR)
t, QueryDestinationStatusOnSpecificLevelByIndex,3:000a,03
```

- Query the salvo “Salvo1” contents

```
t,QV, Salvo1
  (OR)
t, QuerySalvoStatus,Salvo1
```

- Query the destinations belongs to the room “RoomC”

```
t,QR,D,RoomC
  (OR)
t, QueryRoomDetails, RoomDestinations,RoomC
```

- Query the error definitions

```
t,QE
  (OR)
 t, QueryErrorDefinition
```

- Query the error definition for the error code 86 (note error code is hexadecimal value)

```
t,QE,8b
  (OR)
t, QueryErrorDefinition ,8b
```

- Assign SRC10 and SRC20 to destination DST10 from area AreaX(Note:-NP Protocol allows assigning one source at a time)

  For RCL Protocol:

```
t,AS,AreaX:DST10,02, AreaX: SRC10, AreaX:SRC20
  (OR)
t, Assign, AreaX:DST10,03, AreaX: SRC10, AreaX:SRC20
```

  For NP Protocol (AreaX should be configured as the active area for NP client in Encore):

```
t,AS,DST10,SRC10  and t,AS,DST10,SRC20
  (OR)
t,Assign, DST10,SRC10  and t,Assign,DST10,SRC20
```

- Change the aliases for source indices 11 and 31 from area index 5 as AliasSRC2 and AliasSRC3 respectively

```
t,CA,SA,02,05:000b,AreaX:AliasSRC2,05:001f,AreaX:AliasSRC3
```

```
  (OR)
t,AliasChange, SrcAliasChange,02, 05:000b,AreaX:AliasSRC2,
05:001f,AreaX:AliasSRC3
```

- Subscribe for all destinations status change

```
t,SB,DS
  (OR)
t, Subscription, DestinationStatusChange
```

- Subscribe for area index 10 destination status change

```
t,SB,DS,10
  (OR)
t, Subscription, DestinationStatusChange,10
```

- Subscribe for destination DST10 from area AreaX destination status change

```
t,SB,DS,AreaX:DST10
  (OR)
t, Subscription, DestinationStatusChange,AreaX:DST10
```

- Unsubscribe for all destinations status change

```
t,UB,DS
  (OR)
t, UnSubscription, DestinationStatusChange
```

- Unsubscirbe area index 10 for destination status change

```
t,UB,DS,10
  (OR)
t, UnSubscription,DestinationStatusChange,10
```

- Unsubscribe for destination DST10 from area AreaX destination status change

```
t,UB,DS,AreaX:DST10
   (OR)
t, UnSubscription, DestinationStatusChange,AreaX:DST10
```

- Subscribe for all the changes

```
t,SB
  (OR)
t, Subscription
```

- Unsubscribe for all the changes

```
t,UB
  (OR)
t, UnSubscription
```

- Query the assignment status of the destination DST10 from the area AreaX for 20 times with delay of 500 milliseconds.

```
Loop, 20
t, QA,AreaX:DST10
Wait, 500
EndLoop
```

The script is given below for few operations:

```
#Any command starts with "#" treated as comment
#Query for level names in system
  t,QN,L
#Query for source names in system
  t,QN,S
#Query for destination names in system
  t,QN,D
#Assign the SRC10 to DST10 from the AreaX
  t,AS,DST10,01,SRC10
#wait for the 500 milliseconds
  Wait,500
#Query the assignment status for DST10 from area AreaX
  t, QA, AreaX: DST10
#Take a Source SRC10 to DST10 from the AreaX
  t,TD,,AreaX:DST10,AreaX:SRC10
#wait for 500 milliseconds
  Wait,500
#Query the destination status for DST10 from AreaX
  t,QD,AreaX:DST10
```

Timed Operations

The RCL Client Simulator application provides facility for

- Issuing a command to be sent out to the server at a specified time. For each command, the Time of Firing is denoted as TF in the descriptions below.

- In case of RCL Protocol, commands such as Take (all flavors of take commands) and 'Take Salvo' commands can take a time stamp along with them. The server will then make sure that the commands are executed at the specified time, provided the conditions are met as specified in the Protocols manual. The scripts can be designed such that RCL Client simulator can embed a time stamp in to the command as per the protocol specification before sending the command to the server. For each command, the time stamp embedded is denoted as TE in the descriptions below.

To use one or both of the timed operations, the user will have to make sure that the following conditions are met.

1. Install a PCI bus based VITC (Vertical Interval Time Code) reader in the PC where the RCL Client Simulator application is running. The Application is tested for the card reading the time code present on the reference signal.(black reference)

2. Feed a valid VITC (supported types are NTSC, NTSC DF and PAL) to the card reader.

3. Enable timed operations on the UI and specify the start time at which the timed operations begin (denoted as TFB in the descriptions below). Please refer section Steps to enable timed operations and specifying start time for instructions.

The commands for timed operations will have a general structure as follows:

```
TFOn,t command,Offset,OtherParamsOfTheCommand
```

TFOn is the offset time for firing the command in the format hh:mm:ss:ff for this command. The TF is computed as:

$$TF = TFB + TFO_n.$$

TFOn can be specified as 00:00:00:00 to make the TF equal to TFB.

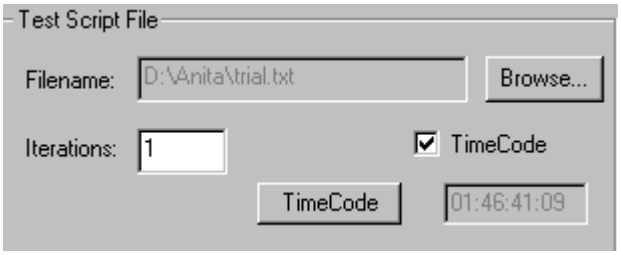
Offset is the offset in number of frames which is used in computing TE as follows:

```
TE = TF + Offset.
```

The Offset value should be specified ONLY for RCL Protocol and ONLY for the commands that support timed operation by the server (take commands and Take salvo). For these commands, if "embedding time stamp" is NOT desired, it can be skipped by providing back-to-back commas.

Steps to enable timed operations and specifying start time

1. Check the Time Code (check box). This check box is enabled ONLY if the application is able to find a PCI time code reader card on the system. Otherwise it remains grayed and no time code-related operations can be performed.



2. Click on the Time Code (button). Upon clicking this button a time code dialog box is popped.

3. Depending on the type of VITC connected to the VITC card reader in the PC, chose VITC type in the "Time Code Type" section. NTSC is chosen by default. Note that, for predictable results, the VITC fed to PC and the Encore SCB (or to any other NP/RCL server being tested) should be from the same source.

4. The user can retrieve the current VITC time to see if the card is working and also to use the current VITC as a reference. Clicking on "Get Time Code" button will make the application display the VITC time at that moment in the "Current VITC Time Code" section (this step is optional)

5. Enter the start time by typing the same in the text entry field against "Enter Time Code". It should in the form hh:mm:ss:ff.

6. Click OK to proceed further (clicking on cancel will throwaway all the inputs).

Sample scripts for time stamp commands

The below commands are faired from client based on the TF time.

```
TF= TFB + TFOn
```

- "Query the all areas destinations names" command at 30 frames from the start of the script time

```
00:00:00:30, t, QN, D
```

- Query the all areas destinations status by index at 10 minutes from the start of the script time.

```
00:00:10:00, t, QJ
```

- Protect the destination DST10 on level 4 from area AreaX at 1 hour 10 frames form the start of the script time.

```
01:00:00:10, t, PR, AreaX: DST10, 00000010
```

- Commit the source alias changes at 10 hours 20 minutes and 5 seconds from the start of the script time.

```
10:20:05:00, t, CM,SA
```

- Take a source SRC10 on destination DST10 from the area AreaX at 5minutes from the start of the script time.

  For NP Protocol

```
00:05:00:00, t, TD, DST10, SRC10
```

  For RCL Protocol (note the use of back to back commas to avoid embedding time stamp)

```
00:05:00:00, t, TD,, AreaX: DST10, AreaX: SRC10
```

- Take a source index 19 on destination index 17 from the area index 4 at 10 minutes and 30 seconds from the start of the script file.

  For NP Protocol

```
00:10:30:00, t, TI, 0011, 0013
```

  For RCL Protocol

```
00:10:30:00, t, TI, 04:0011, 04:0013
```

Sample scripts for time stamp takes

The below commands are faired from client based on the TF time and executed by Sever based on the TE.

```
TF= TFB + TFOn
```

```
TE= TF + offset
```

- Take a source SRC10 on destination DST5 from area AreaX exactly at 30 frames from the start of the script

```
00:00:00:00, t, TD, 30, AreaX: DST5, AreaX: SRC10
```

- Take a source SRC2 on destination DST10 from area AreaX exactly at 1 minute from the start of the script

```
00:00:59:24, t, TD, 6, AreaX: DST10, AreaX: SRC2
```

- Take a source 1 on destination 5 from area 1 exactly at 2 hours from the start of the script

```
01:59:59:10, t, TI, 20, 1:5, 1:1
```

- Take a source 5 on destination 6 from area 1 exactly at 10 hours 10 minute from the start of the script

```
10:09:59:05, t, TI, 25, 1:6, 1:5
```

- Take a source SRC15 on level index 0 and source SCR25 on level index 1 on destination DST16 from area AreaX at exactly 15 hours from the start of script

```
14:59:59:20,t,TA,10,AreaX:DST16,02,AreaX:SRC15,00000001,
AreaX:SRC25,00000002
```

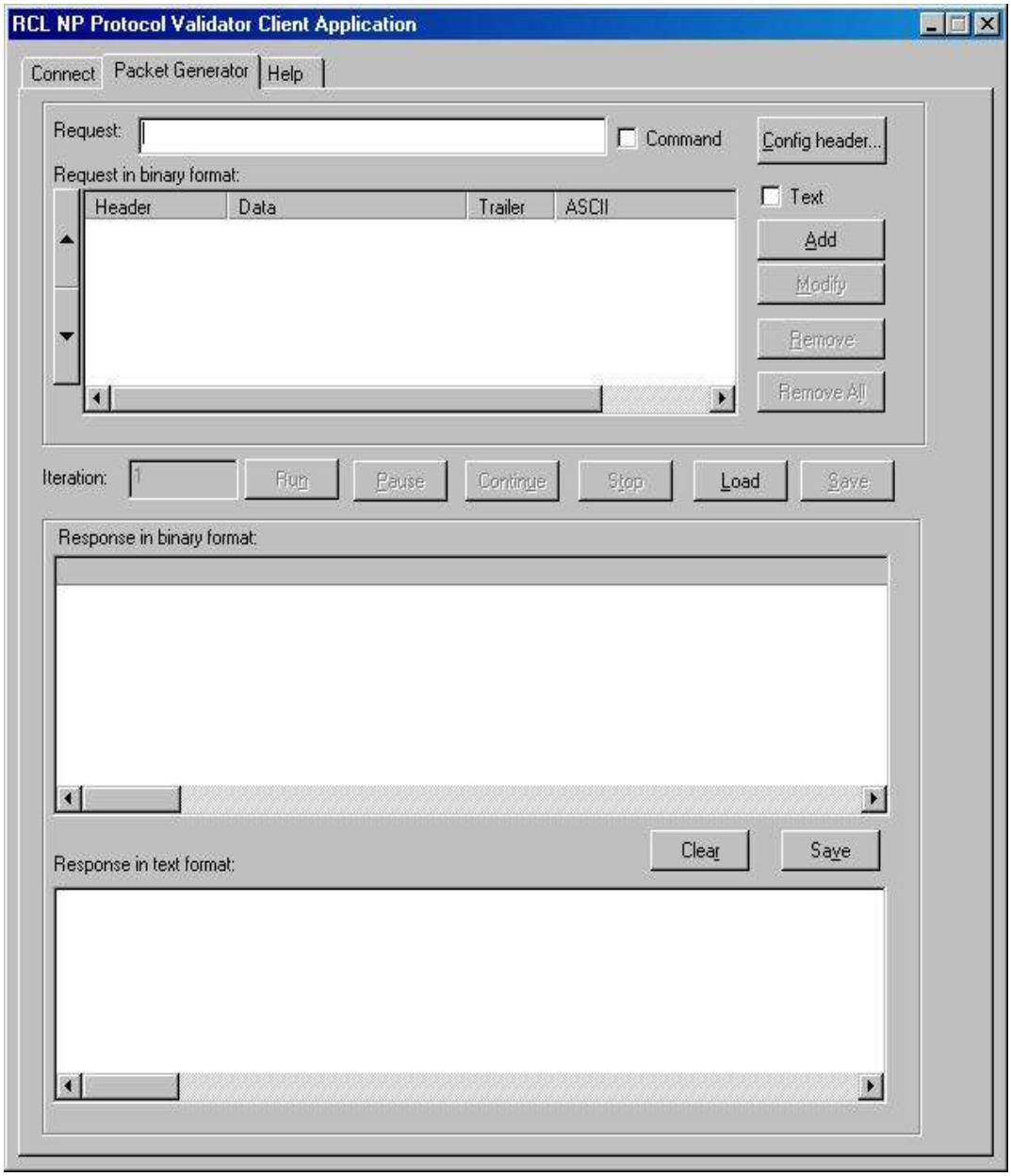
- Take a Salvo "Salvo1" exactly at 4 hours 30 minutes from the start of the script

```
04:29:59:02, t, TS, 28, Salvo1
```

Packet Generator

Overview

The Packet Generator is also a simulator of an automation interface. This is provided so that the users can enter the commands directly to be executed. Another alternative is to add a set of commands and store them in a file, which can be loaded later for execution. The commands can be executed as many number of times that the user wishes by specifying the iteration count. The user can also enter the header and trailer for the packet (command). The Packet Generator gives the user the flexibility of manipulating the commands as per his/her requirements. This is also an excellent tool to test the server corresponding to the respective protocols. The execution of the commands can be paused, continued or stopped. The display of the command is both in the textual as well as ASCII format.



Description of Controls

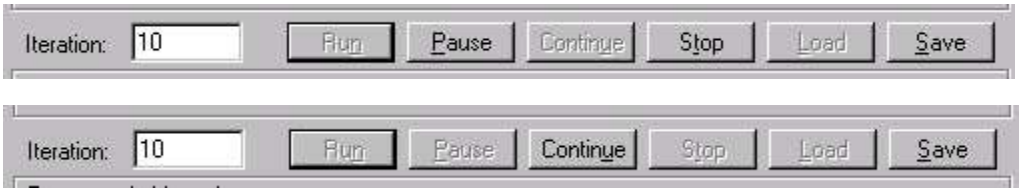
- Request: Accepts the command the user wants to execute.
- Text: Selection of this option replaces the non-printable characters like SOH, EOT and TAB with strings like <SOH>, <EOT> and <TAB> respectively. For instance, the QN,S command with the text option choosen, is displayed as follows:

  <SOH>R4G000... QN<TAB>S <EOT> 01 52 34 47 30 30

  With the text option not choosen, is displayed as follows:

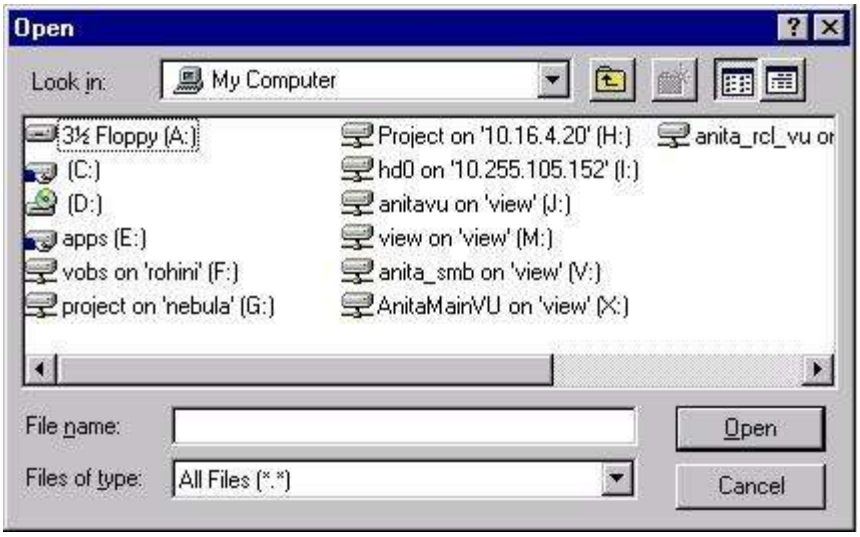
  R4G00030000... QN,S 01 52 34 47 30 30

- Iteration: If the user wishes to run the commands entered either once or repetitively, he can specify the no. of times he/she would like to do that in the 'Iteration' edit box. The default value is 1.
- Command: This is a check box that the user selects if he wants to enter a delay in the execution of server requests. If the user wants to introduce a delay of 5 seconds then he/she has to enter the following command into the 'request' edit box – 5000.
- Config Header…: Pops up the Header/Trailer Configuration dialog box.
- Add: Adds the commands entered into the 'request' edit control into the 'request in binary format' list control.
- Modify: Allows modification of the chosen command from the 'request in binary format' list control.
- Remove: Deletes the selected command(s) from the list control.
- Remove All: Deletes all the commands from the list control.
- Request in binary format: Contains the commands added by the user. The binary format of the commands is also displayed.
- Run: Executes the commands in the 'request in binary format' list control.
- Pause: Halts the execution of the commands till resumed again.
- Continue: Resumes the execution of the commands.
- Stop: Terminates the execution of the commands completely.



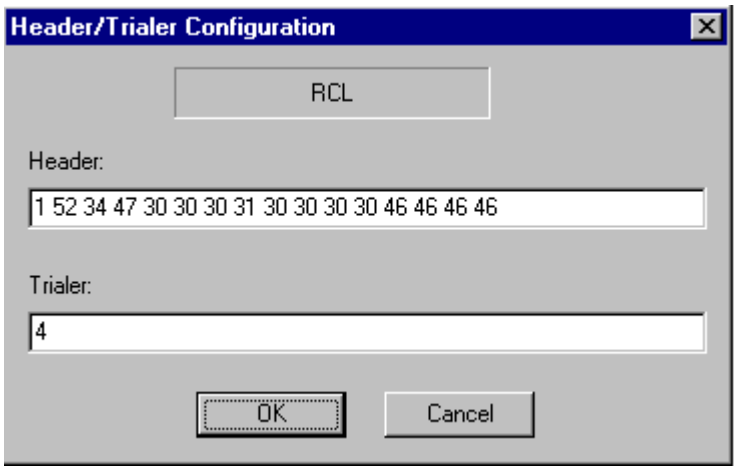
- Save: Saves the contents of the 'request in binary format' list control onto a file. The save dialog box is popped when the user clicks on this button. The user can choose the path where he wants to store the file.

- Load: Loads the file where the contents of 'request in binary format' list control was stored. The open dialog box is popped when the user clicks on this button. The user can then navigate to the stored file using the open dialog box.



- Response in binary format: Response from the server is populated in the binary format in this list control.
- Response in text format: Response from the server is populated in the text format this list control.
- Save: Saves the response from the server in the text format onto a file.
- Clear: Clears the contents in 'response in binary format' and 'response in text format' list controls.

Header/Trailer Configuration



Description of Controls

- RCL: Displays the protocol that is choosen .
- Header: Accepts the header that will be appended to the message of specified protocol.
- Trailer: Accepts the trailer that will be appended to the message of specified protocol.

Custom Protocol

The RCL client simulator supports the Custom protocol in Packet Generator mode. The Custom protocol allows the user to send custom packets by create our own header and trailer. The user can change the header and trailer configuration by using the “Header/Trailer configuration” dialog box.

The packet format is

```
Header, CommandBody, Trailer.
```

Packet Generator Example:

This section describes the sequence of steps to use the packet generator mode to send NP \ RCL commands to the server.

1. Choose the appropriate communication (Serial or TCP) and application (NP or RCL) protocols in the Connection Tab of the RCL Client Simulator.
2. Choose the Test Mode as "Packet Generator" mode by clicking on the Radio Option.
3. Click on Connect to connect to the server.
4. Go the Packet Generator tab.
5. Enter the command that you want to test in the "Request" text entry field. It is sufficient to enter just the application layer command as per the protocol specification. For example, enter *BK,F* to send the BK,F command. The RCL Simulator takes care of prefixing the SOH and appending checksum and EOT. It also takes care of replacing the commas with the delimiter (that is '\t'). Similarly, to query source names enter *QN,S* in the "Request" text entry field.

6. Now click on 'Add' button. The command that was typed is shown in the "Request in binary format" window with header, data, checksum and EOT.

7. Chose 1 as number of iterations. Do NOT check the "command" check box. Do check the "text" check box. Now click on "Run" button.

8. The server response in both binary and text format (they are almost identical) is shown in the appropriate list boxes.

9. These commands can be saved as a script file using the Save button and can be loaded back (using Load button) and executed.