

Fabian Server Command Reference

FabianServer32.exe

Written by: Chris Gil, Richard Kerry.

Contents

Contents	2
1 Introduction.....	3
2 Socket Connection.....	3
3 General Notes	3
3.1 Responses, or Not	3
4 Command Groups.....	4
5 Commands.....	4
5.1 GPI Client Commands	4
5.2 GPI Server Responses.....	7
5.3 InfoDriver Client Commands	7
5.4 InfoDriver Server Responses	9
5.5 Router Client Commands	9
5.6 Router Server Responses.....	12
5.7 Universal Server Response.....	12
5.8 Device Client Commands	12
5.9 Device Server Responses.....	13
5.10 Client Database Commands	13
5.11 Server Database Responses	16
5.12 System Commands.....	17
5.13 System Commands (User)	22
5.14 System Responses	27
5.15 Database Commands	27
5.16 Database Responses.....	36
5.17 Resource Commands	36
5.18 Resource Responses	39
6 Documents referenced.....	40
7 Notes	40
7.1 Warning	40
8 Version history.....	40
8.1 Driver version	40
8.2 Document version	41

1 Introduction

All commands consist of three letters followed by zero or more command dependent parameters, separated by spaces. Valid commands with correct syntax are not acknowledged unless the nature of the command returns some output. Unknown or incomplete commands result in an appropriate error message.

The following commands are provided to control a BNCS environment via the Fabian server. Router, GPI and InfoDriver control is supported.

An additional password is required before control system access is permitted.

2 Socket Connection

The Fabian server is a TCP/IP socket server that is on port 23 by default. To test the server any TCP client may be used.

Open a connection to the address of the server and the banner followed by the login prompt should appear.

It should be noted that the Fabian server is accessed using raw TCP and does not use any of the features of the Telnet Protocol.

3 General Notes

Numeric parameters are shown as <parameter%> as follows:

ARP 680 1 10

String parameters are shown as <parameter\$>. Some commands require that string values are enclosed in single quotes if spaces are required. Some commands don't require quotes but use all the text up to the end of the line.

AIW 682 'parameter' 24

Commands that cause a change of state on the system such as AGS, AIW, ARC will only succeed if

- The client has permission to modify the range on the device AND
- The output/slot/crosspoint on the device is not locked AND
- The device is actually present on the network

3.1 Responses, or Not

If echoing is on for a connected client, all commands will be echoed when received. This is in addition to any specific response to the command.

If a command is received which is not recognized, an "Unknown Command" response is returned.

Many commands generate an immediate response, which starts with the same three-letter name as the command itself.

Some commands generate an immediate response, with a different three-letter name.

Some commands (eg username, password) generate a response which doesn't quote the command itself or any other command. There is a response which indicates whether the command has worked.

Some commands (eg register, unregister, poll) don't produce a response. Their correct functioning may be deduced by the subsequent behaviour of the system. For example polling will not generate a specific response but is likely to cause a number of revertive messages to be generated.

4 Command Groups

Commands belong to the following groups

1. GPI Client Commands
2. GPI Server Responses
3. InfoDriver Client Commands
4. InfoDriver Server Responses
5. Router Client Commands
6. Router Server Responses
7. Universal Server Response
8. Device Client Commands
9. Device Server Responses
10. Client Database Commands
11. Server Database Responses
12. System Commands
13. System Commands (User)
14. System Responses
15. Database Commands
16. Database Responses
17. Resource Commands
18. Resource Responses

Note that in some cases there are a command and response with the same name.

5 Commands

5.1 GPI Client Commands

AGP	AppICore GPI Poll
------------	--------------------------

Purpose	Poll for the current state of a range of GPI inputs or outputs
Syntax	AGP <device%> <min%> <max%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the GPI driver • <min%> is the lowest index of the range • <min%> is the lowest index of the range
Notes	<p>The client will need to have previously registered for this range on the device using AGR.</p> <p>Access Control Applies - Read/Device.</p>
Example	AGP 801 1 4
Response	The server will poll its local system, and any reverts received will be converted into a series of AGR messages (ApplCore GPI Revertive)

AGR	ApplCore GPI Register
Purpose	Register for changes of state for a range of GPI inputs and outputs
Syntax	AGR <device%> <min%> <max%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the GPI driver • <min%> is the lowest index of the range • <min%> is the lowest index of the range
Notes	
Example	AGR 801 1 32
Response	None

AGS	ApplCore GPI Switch
Purpose	Change the state of a GPI output
Syntax	AGP <device%> <index%> <0 1>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the GPI driver • <index%> is the index of the output to switch • <0 1> is the state OFF or ON
Notes	<p>The state will change if the output is not locked and a response will be received if the index is within the range previously registered by the client.</p> <p>Access Control Applies – Write/Device/Address/Data.</p>
Example	AGP 801 16 1
Response	The server will normally respond with an AGR message indicating the new state

AGU	AppICore GPI Unregister
Purpose	Cancel registration for changes of state for a GPI device
Syntax	AGU <device%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the GPI driver
Notes	
Example	AGU 801
Response	None

AGF	AppICore GPI File
Purpose	Instructs the GPI driver <Driver%> to reload its 'Line Inversion' table from the drivers INI file.
Syntax	AGF <device%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the GPI driver
Notes	<p>(see Warning below)</p> <p>GF functionality is handled by the specific GPI driver.</p> <p>This command is unable to affect it if the driver doesn't support it.</p> <p>Access Control Applies – Write/Device</p>
Example	AGF 801
Response	

AGL	AppICore GPI Lock
Purpose	Locks (0) or unlocks (1) all indices between <Min%> and <Max%> on <Driver%>.
Syntax	AGL <device%> <Min%> <Max%> <0/1>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the GPI driver • <Min%> is the minimum index to be locked/unlocked • <Max%> is the maximum index to be locked/unlocked • <0/1> is 0 to lock, 1 to unlock
Notes	<p>GL functionality is handled by the specific GPI driver.</p> <p>This command is unable to affect it if the driver doesn't support it.</p> <p>Access Control Applies – Write/Device/Address.</p>

Example	AGL 709 25 30 0
Response	None

5.2 GPI Server Responses

AGR	ApplCore GPI Revertive
Purpose	Notification of a change of state of an input/output on a GPI device
Syntax	AGR <device%> <index%> <0 1>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the GPI driver • <index%> is the index of the input or output • <0 1> is the state OFF or ON
Notes	Access Control Applies – Read/Device/Address/Data.
Example	AGR 801 16 0

5.3 InfoDriver Client Commands

AIP	ApplCore Info Poll
Purpose	Poll for the current contents of a range of InfoDriver slots
Syntax	AIP <device%> <min%> <max%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the InfoDriver • <min%> is the lowest index of the range • <max%> is the highest index of the range
Notes	The client will need to have previously registered for this range on the device. Access Control Applies - Read/Device.
Example	AIP 682 1 16
Response	The server will respond with a series of AIR messages (ApplCore Info Revertive)

AIR	ApplCore Info Register
Purpose	Register for changes of state for a range of InfoDriver slots
Syntax	AIR <device%> <min%> <max%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the InfoDriver • <min%> is the lowest index of the range

	<ul style="list-style-type: none"> • <min%> is the lowest index of the range
Notes	
Example	AIR 682 1 800
Response	None

AIW	AppCore Info Write
Purpose	Change the contents of a InfoDriver slot
Syntax	AIW <device%> <content\$> <index%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the InfoDriver • <index%> is the index of the slot to change • <content\$> is the new contents
Notes	<p>If the slot changes the client will receive an AIR message if the client has previously registered for this range on the device.</p> <p>Access Control Applies – Write/Device/Address/Data.</p>
Example	AIW 682 'new contents for slot1' 1
Response	The server will normally respond with an AIR message containing the new contents

AIU	AppCore Info Unregister
Purpose	Cancel registration for changes of slot contents for a InfoDriver device
Syntax	AIU <device%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the InfoDriver
Notes	
Example	AIU 682
Response	None

AIF	AppCore Info File
Purpose	Requests InfoDriver <Driver%> to put string <Vars\$> in <Filename\$>. If <Switch%>=0 append otherwise create new file.
Syntax	AIF <device%> <Switch%> <Filename\$> <Var\$>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the InfoDriver

	<ul style="list-style-type: none"> Requests InfoDriver <Driver%> to put string <Vars\$> in <Filename\$>. If <Switch%>=0 append otherwise create new file.
Notes	Access Control Applies – Write/Device
Example	AIF 401 0 test401.txt 'message'
Response	None

AIL	AppCore Info Lock
Purpose	Locks (1) or unlocks (0) all indices between <Min%> and <Max%> on <Driver%>.
Syntax	AIL <device%> <Min%> <Max%> <0 or 1>
Where	<ul style="list-style-type: none"> <device%> is the device ID of the InfoDriver <Min%> is the minimum index to be locked/unlocked <Max%> is the maximum index to be locked/unlocked <0/1> is 1 to lock, 0 to unlock
Notes	Access Control Applies – Write/Device/Address.
Example	AIL 401 25 35 1
Response	None

5.4 InfoDriver Server Responses

AIR	AppCore Info Revertive
Purpose	Notification of a change of slot contents on an InfoDriver device
Syntax	AIR <device%> <index%> <content\$>
Where	<ul style="list-style-type: none"> <device%> is the device ID of the InfoDriver <index%> is the index of the slot <content\$> is the current slot contents
Notes	Access Control Applies – Read/Device/Address/Data.
Example	AIR 682 1 'latest content for slot 1'

5.5 Router Client Commands

ARC	AppCore Router Crosspoint
Purpose	Route a source to a destination on a Router
Syntax	ARC <device%> <source%> <dest%> { <mask\$> }

Where	<ul style="list-style-type: none"> • <device%> is the device ID of the Router • <source%> is the index of the new source • <dest%> is the index of the destination • <mask\$> is the optional mask
Notes	<p>If the crosspoint changes and the client has previously registered for this destination an ARR message will be received.</p> <p>Access Control Applies – Write/Device/Address/Data.</p>
Example	ARC 680 3 203 'FULL'
Response	The server will normally respond with an ARR message confirming the new source

ARP	ApplCore Router Poll
Purpose	Poll for the current source for a range of Router destinations
Syntax	ARP <device%> <min%> <max%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the Router • <min%> is the lowest index of the range • <min%> is the lowest index of the range
Notes	<p>The client will need to have previously registered for this range on the device.</p> <p>Access Control Applies - Read/Device.</p>
Example	ARP 680 1 10
Response	The server will respond with a series of ARR messages (ApplCore Router Revertive)

ARR	ApplCore Router Register
Purpose	Register for changes of state for a range of Router destinations
Syntax	ARR <device%> <min%> <max%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the Router • <min%> is the lowest index of the range • <min%> is the lowest index of the range
Notes	
Example	ARR 680 1 400
Response	None

ARU	ApplCore Router Unregister
Purpose	Cancel registration for changes of destinations on a Router device
Syntax	ARU <device%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the Router
Notes	
Example	ARU 680
Response	None

ARF	ApplCore Router File
Purpose	Tells <Driver%> to reload its INI file.
Syntax	ARF <device%>
Where	<ul style="list-style-type: none"> • Tells <device%> to reload its INI file.
Notes	Access Control Applies – Write/Device
Example	ARF 705
Response	None

ARL	ApplCore Router Lock
Purpose	If <Var%> is 1 then all destinations on <Driver%> between <Min%> and <Max%> are locked. A destination can not be changed when it is locked. Use <Var%> set to 0 to unlock a destination or range of destinations.
Syntax	ARL <device%> <Min%> <Max%> <Var%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the InfoDriver • <Min%> is the minimum index to be locked/unlocked • <Max%> is the maximum index to be locked/unlocked • <0/1> is 1 to lock, 0 to unlock
Notes	Access Control Applies – Write/Device/Address.
Example	ARL
Response	None

5.6 Router Server Responses

ARR	ApplCore Router Revertive
Purpose	Notification of a change of destination on a Router
Syntax	ARR <device%> <dest%> <source%>
Where	<ul style="list-style-type: none">• <device%> is the device ID of the InfoDriver• <dest%> is the index of the destination• <source%> is the current source
Notes	Access Control Applies – Read/Device/Address/Data.
Example	ARR 680 40 2

5.7 Universal Server Response

UR	Universal Revertive
Purpose	A packed binary format message combining multiple revertives and router name information.
Syntax	UR <binary data>
Where	<ul style="list-style-type: none">• <binary data> is a packed binary message format
Notes	Access Control Applies – Read/Device/Address/Data.
Example	

5.8 Device Client Commands

ADL	ApplCore Device List
Purpose	Return a list of device profiles of all devices known by this instance of CSI.
Syntax	ADL
Where	No parameters
Notes	Access Control Applies – Read or Write/Device.
Example	ADL
Response	Series of ADP responses – see below. One for each ini file visible to the CSI instance.

ADP	ApplCore Device Properties
Purpose	Return a device profiles of one device known by this instance of CSI.
Syntax	ADP <device%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the driver being queried
Notes	Access Control Applies – Read or Write/Device.
Example	ADP 401
Response	An ADP response – see below.

5.9 Device Server Responses

ADP	ApplCore Device Properties
Purpose	One device profile.
Syntax	ADP <device%> <version%> <type%> <sources%> <dests%> <ininame%> <dblist%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID of the driver • <version%> is the driver version – 1 or 2 (version 1 have config files such as ID_ddd.INI or GRD_ddd.INI, version 2 have DEV_ddd.INI config files) • <type%> is the driver type: <ul style="list-style-type: none"> ○ 0 – always this for version 2 drivers ○ 1 for Router (v1 only) ○ 2 for GPI driver (v1 only) ○ 3 for InfoDriver (v1 only) • <sources%> is the number of sources, ie the database 0 size • <dests%> is the number of destinations, ie the database 1 size • <ininame%> is the name of the .ini file it is using • <dblist%> is a comma-separated list of databases available (0 to 9 only, upper databases are not supported)
Notes	<p>This information comes from the local configuration files, via the local instance of CSI.</p> <p>Access Control Applies – Read or Write/Device.</p>
Example	ADP 943 2 0 0064 0064 DEV_943 0,1,9

5.10 Client Database Commands

ALR	ApplCore List Router
------------	-----------------------------

Purpose	Request the names of range of database names for a device
Syntax	ALR <device%> <switch%> <min%> <max%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID • <switch%> is the database number. This is typically as follows, though this is not enforced and may vary from system to system. <ul style="list-style-type: none"> ○ 0 = source name ○ 1 = destination name ○ 2 = long source name ○ 3 = long destination name ○ 4 – 9 = user defined databases • <min%> is the lowest index of the range • <max%> is the highest index of the range
Notes	Access Control Applies – Read/Device.
Example	ALR 680 0 1 10
Response	The server will respond with a series of ARN messages

ARI	AppCore Router Index
Purpose	Request the index of a router name
Syntax	ARI <device%> <switch%> <name\$>
Where	<ul style="list-style-type: none"> • <device%> is the device ID • <switch%> is the database number. This is typically as follows, though this is not enforced and may vary from system to system. <ul style="list-style-type: none"> ○ 0 = source name ○ 1 = destination name ○ 2 = long source name ○ 3 = long destination name ○ 4 – 9 = user defined databases • <name\$> is the database name to find
Notes	Access Control Applies – Read/Device.
Example	ARI 201 0 'source 1'
Response	The server will respond with an ARI message

ARM	AppCore Router Modify
Purpose	Change the name of a device database entry
Syntax	ARM <device%> <switch%> <name\$>

Where	<ul style="list-style-type: none"> • <device%> is the device ID • <switch%> is the database number. This is typically as follows, though this is not enforced and may vary from system to system. <ul style="list-style-type: none"> ○ 0 = source name ○ 1 = destination name ○ 2 = long source name ○ 3 = long destination name ○ 4 – 9 = user defined databases • <name\$> is the database name to find
Notes	Access Control Applies – Write/Device.
Example	ARM 201 0 'source 1'
Response	The server will respond with an ADC message if the client has registered for this index on the device

ARN	AppCore Router Name
Purpose	Request the name of a device database entry
Syntax	ARN <device%> <switch%> <index%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID • <switch%> is the database number. This is typically as follows, though this is not enforced and may vary from system to system. <ul style="list-style-type: none"> ○ 0 = source name ○ 1 = destination name ○ 2 = long source name ○ 3 = long destination name ○ 4 – 9 = user defined databases • <index%> is index of the database name to find
Notes	Access Control Applies – Read/Device.
Example	ARN 201 0 40
Response	The server will respond with an ARN message

ACD	AppCore Checksum Database
Purpose	Request a checksum value for a group of entries in a device database
Syntax	ACD <device%> <database%> <min%> <max%>
Where	<ul style="list-style-type: none"> • <device%> is the device number • <database%> is the database number • <min%> and <max%> are the minimum and maximum indices to be checksummed

Notes	<p>Generates a checksum by adding the ASCII values of all the names returned from the database items specified.</p> <p>Only the first 16 characters of each name will be considered.</p> <p>The checksum value is returned in hexadecimal.</p> <p>Access Control Applies – Read/Device.</p>
Example	ACD 401 1 10 100
Response	ACD 401 1 10 100 49DB

5.11 Server Database Responses

ADC	AppCore Database Change
Purpose	Notification of a change of a database name on a device
Syntax	ADC <device%> <switch%> <index%> <name\$>
Where	<ul style="list-style-type: none"> • <device%> is the device ID • <switch%> is the database number. This is typically as follows, though this is not enforced and may vary from system to system. <ul style="list-style-type: none"> ○ 0 = source name ○ 1 = destination name ○ 2 = long source name ○ 3 = long destination name ○ 4 – 9 = user defined databases • <index%> is the index of the database name • <name\$> is the current name
Notes	<p>This message will be sent if the client has registered for any index on this device on the server.</p> <p>Access Control Applies – Read/Device.</p>
Example	ADC 201 0 1 'source 1'

ARI	AppCore Router Index
Purpose	Response to a client ARI request containing the index of the requested name
Syntax	ARI <device%> <switch%> <index%>
Where	<ul style="list-style-type: none"> • <device%> is the device ID • <switch%> is the database number. This is typically as follows, though this is not enforced and may vary from system to system. <ul style="list-style-type: none"> ○ 0 = source name, ○ 1 = destination name,

	<ul style="list-style-type: none"> ○ 2 = long source name ○ 3 = long destination name, ○ 4 – 9 = user defined databases • <index%> is the index of the database name
Notes	<p>This message will only be sent as a response to a client request and must be interpreted in the context of the name specified in the request.</p> <p>Access Control Applies – Read/Device.</p>
Example	ARI 201 0 1

ARN	ApplCore Router Name
Purpose	Response to a client ARN request containing the a device database name
Syntax	ARN <device%> <switch%> <index%> <name\$>
Where	<ul style="list-style-type: none"> • <device%> is the device ID • <switch%> is the database number. This is typically as follows, though this is not enforced and may vary from system to system. <ul style="list-style-type: none"> ○ 0 = source name, ○ 1 = destination name, ○ 2 = long source name ○ 3 = long destination name, ○ 4 – 9 = user defined databases • <index%> is the index of the database name • <name\$> is the current name
Notes	Access Control Applies – Read/Device.
Example	ARN 201 0 1 'source 1'

5.12 System Commands

GSA	Get System Access
Purpose	Request the access configuration
Syntax	GSA <switch%>
Where	<ul style="list-style-type: none"> • <switch%> is the command index <ul style="list-style-type: none"> ○ 0, 1 = display currently active access settings <ul style="list-style-type: none"> ▪ 0 = don't show data filters ▪ 1 = show data filters, if any ○ 2,3 = display current user's access settings from the configuration file <ul style="list-style-type: none"> ▪ 2 = don't show data filters

	<ul style="list-style-type: none"> ▪ 3 = show data filters, if any <p><switch%> is optional. If absent the command behaves as if 0 was entered.</p>
Notes	
Example	GSA 2
Response	<p>You have access to the following devices...</p> <p>Write access</p> <p>Device 926 : 1,3,50</p> <p>Device 927 : 51,10-13</p> <p>Device 928 : 41-43</p> <p>Device 929 : 3,5,7-9,31-34,100</p> <p>Read access</p> <p>No devices listed</p> <p>Read/Write access (Read)</p> <p>No devices listed</p> <p>Read/Write access (Write)</p> <p>No devices listed</p>
Notes on Response	<p>As the example command is 2, the data is read from the configuration file, which may no longer be the same as is being used by the current user.</p> <p>The "Write access" section's contents are derived from the "DevWriteAccess_nn" lines.</p> <p>The "Read access" section's contents are derived from the "DevReadAccess_nn" lines.</p> <p>The latter two sections, "Read/Write access (Read)" and "Read/Write access (Write)", are derived from the "DevAccess_nn" lines. They show the affect of these lines on read and write control, taking into account the BncsAccess setting.</p> <p>GSA 0 would give a very similar listing but the latter two sections would not be shown as option 0 (and option 1) read the settings directly from the access control system which doesn't distinguish between these two sources of settings.</p>

GSF	Get System File
Purpose	Request the contents of a system file. Ie one in the directory configured as "DataPath".
Syntax	GSF <systemfilename\$> <localfilename\$>
Where	<ul style="list-style-type: none"> • <systemfilename\$> is the name of the file to be fetched • <localfilename\$> is the name of a local file to which the data might be written
Notes	The localfilename\$ parameter has no action witin Fabian Server, except that it is returned with the file data. It could be used by an automated client if the

	data were to be written to a local file.
Example	GSH FBSH_HLP.txt local_help.txt
Response	<pre> GSH 366 C:\BNCS\Core_System\logs\FabianServerData\FBSH_HLP.TXT LOCAL_HELP.TXT ----- General Help ----- This is a BNCS control server. You will require a login name and two passwords from the system administrator before being allowed access. BNCS is the Broadcast Network Control System and this server is a gateway into a control cluster. ----- <END> </pre> <p>The number, here 366, is the size in bytes of the file data that follows.</p>
Note	From version 4.9.24 this command is disallowed as it presents a security problem potentially allowing a user access to all files in the system.

GSH	Get System Help
Purpose	Request help data.
Syntax	GSH <code\$>
Where	<ul style="list-style-type: none"> • <code\$> is an optional short code indicating which help file is to be read. <ul style="list-style-type: none"> ○ 'com' for help on commands ○ 'gro' for help on groups ○ Anything else for general help information <p>These short codes are not case sensitive, and are detected as prefixes, so "Command" is equivalent to "com", and "GROUP" matches "gro".</p>
Notes	<p>The help data is read from a file in the directory configured as "DataPath".</p> <p>The files are :</p> <p>FBSH_CMD.TXT for help on commands</p> <p>FBSH_GRP.TXT for help on groups</p> <p>FBSH_HLP.TXT for general help</p>
Example	GSH Commands
Response	<pre> FABIAN V1.xx.xx - Commands ===== For a full list of commands please refer to the documentation <END> </pre>

GSI	Get System Information
Purpose	Request the banner page that the client sees when first connecting to the server
Syntax	GSI
Where	No parameters
Notes	
Example	GSI
Response	<pre>***** BNCS FABIAN Server - V4.9.21 - Apr 18 2016 14:53:11 Copyright (c) Atos IT Services 1995-2016 Main Server Up time = 20/04/2016 - 10:07:17 - Type 'HELP' for more information *****</pre>

GSN	Get System Name
Purpose	Request the server's system name. As configured as "ServerName".
Syntax	GSN
Where	No parameters
Notes	
Example	GSN
Response	GSN Main Server

GST	Get System Time
Purpose	Request the current date and time of the server
Syntax	GST
Where	No parameters
Notes	
Example	GST
Response	GST - System Time - 12/06/2002 - 16.08.39

GSU	Get System Users
------------	-------------------------

Purpose	Request a list of connected users/clients.
Syntax	GSU
Where	No parameters
Notes	The list contains the user name, IP address and remote port number of each connected user.
Example	GSU
Response	GSU 1 = username..... at 127.0.0.1;1316

RUT	Are You There
Purpose	Heartbeat message from server
Syntax	RUT
Where	No parameters
Notes	This message will be sent every 60 seconds by the server. The client should reply with an RUR message
Example	RUT
Response	RUR

SGM	Send Group Message
Purpose	Send a message to all members of a group
Syntax	SGM <group%> <message\$>
Where	<ul style="list-style-type: none"> • <group%> is the group number • <message\$> is the message to be sent
Notes	The message will be sent to all clients with the given group number other than the client that enters the command.
Example	SGM 2 All systems satisfactory
Response	SGM 2 All systems satisfactory This will not be seen by the client sending the command; it will be sent to all client sessions with the given group number.

SSN	Set System Name
Purpose	Sets the system name

Syntax	SSN <name\$> <sysmancode%>
Where	<ul style="list-style-type: none"> • <name\$> is the new system name • <sysmancode%> is the system management code
Notes	Requires a system management code matching the configuration.
Example	SSN 'Reserve Server' 42
Response	None

SST	Set System Time
Purpose	Sets the system time and date
Syntax	SST <date\$> <time\$> <sysmancode%>
Where	<ul style="list-style-type: none"> • <date\$> is the new system date • <time\$> is the new system time • <sysmancode%> is the system management code
Notes	Requires a system management code matching the configuration.
Example	SST 12/04/2016 11.17.00 42
Response	None

5.13 System Commands (User)

System commands relating to the current logged-in user.

GUE	Get User Echo
Purpose	Request the state of the user terminal echo
Syntax	GUE
Where	No parameters
Notes	If the flag is turned on this command will return 1, otherwise it returns 0. The user terminal echo flag determines whether the server echoes back a clients commands as they are typed in. Default is on (1)
Example	GUE
Response	GUE 1

GUF	Get User Fields
------------	------------------------

Purpose	Request the user's ordered list of scheduler database fields.
Syntax	GUF
Where	No parameters
Notes	
Example	GUF
Response	GUF 01,02,03,04,05,06,07,

GUG	Get User Groups
Purpose	Request a list of the user's group memberships.
Syntax	GUG
Where	No parameters
Notes	
Example	GUG
Response	GUG 03,05,07,

GUM	Get User Mode
Purpose	Request whether the user has packet-mode revertives enabled.
Syntax	GUM
Where	No parameters
Notes	
Example	GUM
Response	GUM 0

GUN	Get User Name
Purpose	Request the user name by which the client is logged on
Syntax	GUN
Where	No parameters

Notes	Returns the name by which the client is logged on
Example	GUN
Response	GUN user3

GUX	Get User eXtra
Purpose	Request whether the user has extra mode enabled.
Syntax	GUX
Where	No parameters
Notes	Extra mode causes normal-mode router reverts to have the source name added in addition to the source number.
Example	GUX
Response	GUX 0

SUC	Set User Control
Purpose	Send the control password as part of the login sequence
Syntax	SUC <control\$>
Where	<ul style="list-style-type: none"> • <control\$> is the control password for the client account (no single quotes required)
Notes	<p>This is the extra password required before access is allowed to the control functions provided by the server.</p> <p>This is not the same password used with the SUP command.</p> <p>No GPI, InfoDriver or Router functions are permitted without providing this password successfully.</p>
Example	SUC controlpass
Response	Control Access Granted (Read/Write[ALL])

SUE	Set User Echo
Purpose	Set the terminal echo state
Syntax	SUE <0 1>
Where	<ul style="list-style-type: none"> • <0 1> is the state of the echo flag OFF or ON
Notes	If the flag is 1 then any characters sent to the server will be echoed back to the

	<p>user. If set to 0 then there is no echo. Default is on (1).</p> <p>It is recommended that this command is issued after logging-on to avoid confusion of similar echoed client commands and server responses.</p>
Example	SUE 0
Response	None

SUF	Set User Fields
Purpose	Set the user's ordered list of scheduler database fields.
Syntax	SUF <fieldlist%>
Where	<ul style="list-style-type: none"> • <fieldlist%> is a comma-separated list of field numbers
Notes	
Example	SUF 1,2,3
Response	None

SUG	Set User Groups
Purpose	Set the user's group memberships.
Syntax	SUG <groupplist%>
Where	<ul style="list-style-type: none"> • <groupplist%> is a comma-separated list of group numbers
Notes	
Example	SUG 1,2,3
Response	None

SUM	Set User Mode
Purpose	Sets packet-mode revertives use for the user.
Syntax	SUM <0 1>
Where	<ul style="list-style-type: none"> • 0 to select normal-mode revertives • 1 to select packet-mode revertives
Notes	
Example	SUM

Response	None
----------	------

SUN	Set User Name
Purpose	Send the username as part of the login sequence This command starts a login sequence. Any data relating to a user previously logged-in to this port will be cleared.
Syntax	SUN <username\$>
Where	<username\$> is the user name for the client account (no single quotes required)
Notes	
Example	SUN username
Response	None

SUP	Set User Password
Purpose	Send the password as part of the login sequence. If this command is run at any time other than as part of a log-in sequence any data relating to a user previously logged-in to this port will be cleared.
Syntax	SUP <password\$>
Where	<ul style="list-style-type: none"> • <password\$> is the password for the client account (no single quotes required)
Notes	
Example	SUP password
Response	Ready . . .

SUQ	Set User Quit
Purpose	Informs the server that the client wishes to close the connection
Syntax	SUQ
Where	
Notes	This command enables the server to close the connection gracefully
Example	SUQ
Response	None

SUX	Set User eXtra
Purpose	Selects extra mode for the user.
Syntax	SUX <0 1>
Where	<ul style="list-style-type: none"> • 0 to disable extra mode • 1 to enable extra mode
Notes	Extra mode causes normal-mode router reverts to have the source name added in addition to the source number.
Example	SUX
Response	None

5.14 System Responses

RUR	Are You Reply
Purpose	Client response to the server heartbeat message RUT
Syntax	RUR
Where	
Notes	Failure to respond with RUR will result in the server closing the client connection, if the user has been configured to do so.
Example	RUR

5.15 Database Commands

It is believed that there is little or no active use of the Database commands, so the descriptions here are basic, intended to do little more than list the commands.

GCA	Get Current Amendment
Purpose	Returns the user's current amendment
Syntax	GCA
Where	No parameters
Notes	
Example	GCA

Response	GCA 0
----------	-------

GCB	Get Current Booking
Purpose	Returns the user's current booking
Syntax	GCB
Where	No parameters
Notes	
Example	GCB
Response	GCB 1

GCD	Get Current Day
Purpose	Returns the user's current selected day of the month
Syntax	GCD
Where	No parameters
Notes	
Example	GCD
Response	GCD 20

GCF	Get Current Field
Purpose	Returns the user's current selected field
Syntax	GCF
Where	No parameters
Notes	
Example	GCF
Response	GCF 0

GCM	Get Current Month
Purpose	Returns the user's current selected month

Syntax	GCM
Where	No parameters
Notes	
Example	GCM
Response	GCM 4

GCS	Get Current Status
Purpose	Returns the user's current booking's current status
Syntax	GCS
Where	No parameters
Notes	
Example	GCS
Response	GCS 0

GCY	Get Current Year
Purpose	Returns the user's current selected year
Syntax	GCY
Where	No parameters
Notes	
Example	GCY
Response	GCY 2016

GDE	Get Database Extended
Purpose	Returns data from a file in the user's currently selected date
Syntax	GDE <name\$>
Where	<ul style="list-style-type: none"> • <name\$> is the name of the file to be read
Notes	

Example	GDE test1.txt
Response	Contents of test1.txt

GDF	Get Database Field
Purpose	Returns the contents of the user's current selected field
Syntax	GDF
Where	No parameters
Notes	
Example	GDF
Response	GDF 123

GDI	Get Database Information
Purpose	Returns information on the currently selected date's database usage
Syntax	GDI
Where	No parameters
Notes	The response shows the number of bookings on the selected date, and the highest numbered amendment.
Example	GDI
Response	GDI For 20/04/2016: Bkgs=1, Highest Amendment=0

GDL	Get Database Lock
Purpose	Returns the locking status of the currently selected date
Syntax	GDL
Where	No parameters
Notes	
Example	GDL
Response	GDL 20/4/2016 1

GDO	Get Database Owner
Purpose	Returns the the name of the user which has locked the current date, or shows that it is not locked
Syntax	GDO
Where	No parameters
Notes	
Example	GDO
Response	GDO USER1 GDO 0 Not locked

GDR	Get Database Record
Purpose	Returns the current database record
Syntax	GDR
Where	No parameters
Notes	
Example	GDR
Response	GDR 123,2 ,0930,0945,1000,42,PARLIAMENTARY FEED NORTH

GDU	Get Database Update
Purpose	Returns the user's update mode setting
Syntax	GDU
Where	No parameters
Notes	This relates to the option to get all database updates immediately notified to other logged-on clients.
Example	GDU
Response	GDU 1

GNA	Get Next Amendment
Purpose	Returns the next amendment for the current booking
Syntax	GNA

Where	No parameters
Notes	
Example	GNA
Response	GNA

GNB	Get Next Booking
Purpose	Returns the next in-use booking number and amendment number
Syntax	GNB
Where	No parameters
Notes	
Example	GNB
Response	GNB 1001 0

GNF	Get Next Free
Purpose	Returns the next free booking number
Syntax	GNF
Where	No parameters
Notes	
Example	GNF
Response	GNF 2

SCA	Set Current Amendment
Purpose	Sets the current amendment number for the user's current booking
Syntax	SCA <amendmnet%>
Where	<ul style="list-style-type: none"> • <amendment%> is thh required amendment
Notes	
Example	SCA 123

Response	None
----------	------

SCB	Set Current Booking
Purpose	Sets the user's current booking
Syntax	SCB <number%>
Where	<ul style="list-style-type: none"> Number% is the required booking number
Notes	
Example	SCB 123
Response	None

SCD	Set Current Day
Purpose	Sets the user's current selected day of the month
Syntax	SCD <day%>
Where	<ul style="list-style-type: none"> <day%> is the required day of the month
Notes	
Example	SCD 20
Response	None

SCF	Set Current Field
Purpose	Sets the user's current selected field
Syntax	SCF <field%>
Where	<ul style="list-style-type: none"> <field%> is the required field number
Notes	
Example	SCF 5
Response	None

SCM	Set Current Month
Purpose	Sets the user's current selected month

Syntax	SCM <month%>
Where	<ul style="list-style-type: none"> • <month%> is the required month
Notes	
Example	SCM 4
Response	None

SCS	Set Current Status
Purpose	Writes the given status to the currently selected day, and sends a response to all members of the scheduler notify group
Syntax	SCS <status%>
Where	<ul style="list-style-type: none"> • <status%> is the required status
Notes	
Example	None
Response	SGM 1 20 4 2016

SCY	Set Current Year
Purpose	Sets the user's current selected year
Syntax	SCY <year%>
Where	<ul style="list-style-type: none"> • <year%> is the required year
Notes	
Example	SCY 2016
Response	None

SDE	Set Database Extended
Purpose	Writes data to a file in the user's currently selected date
Syntax	SDE <name\$> <data\$>
Where	<ul style="list-style-type: none"> • <name\$> is the name of the file to be read • <data\$> is the string to be written to the file Name\$ may not contain any spaces.

	Data\$ may contain spaces and doesn't need quotes.
Notes	
Example	<code>SDE test1.txt this is a test</code>
Response	None

SDF	Set Database Field
Purpose	Writes data to of the user's current selected field
Syntax	<code>SDF <data%/\$></code>
Where	<ul style="list-style-type: none"> • <data%/\$> is the data to be written. Some filelds take numeric data and some take strings. Quotes are not required.
Notes	
Example	<code>SDF 123</code>
Response	None

SDL	Set Database Lock
Purpose	Sets the locking status of the currently selected date
Syntax	<code>SDL <1/0></code>
Where	<ul style="list-style-type: none"> • 1 to lock • 0 to unlock Any non-zero value will lock
Notes	
Example	<code>SDL 1</code>
Response	None

SDR	Set Database Record
Purpose	Writes data to the current database record
Syntax	<code>SDR <data\$></code>
Where	<ul style="list-style-type: none"> • <data\$> is the data to be written Quotes are not necessary
Notes	

Example	SDR 123
Response	None

SDU	Set Database Update
Purpose	Sets the user's update mode setting
Syntax	SDU <1/0>
Where	<ul style="list-style-type: none"> • 1 to enable update notifications • 0 to disable update notifications
Notes	This relates to the option to get all database updates immediately notified to other logged-on clients.
Example	SDU 1
Response	None

5.16 Database Responses

DBC	Database Change Notification
Purpose	Notification of a database change
Syntax	DBC <day\$>/<month\$>/<year\$> <booking\$> (<amendment\$>)
Where	<ul style="list-style-type: none"> • <day\$>/<month\$>/<year\$> is the date where the change was made • <booking\$> is the booking changed • <amendment\$> is the amendment changed
Notes	<p>This is also sent to all connected clients which are marked as requiring database update notifications.</p> <p>These may be changed using the SDU command.</p>
Example	DBC 20/04/2016 123 (3)

5.17 Resource Commands

GRB	Get Resource Booking
Purpose	Reads any data for the user's currently selected booking
Syntax	GRB

Where	No parameters
Notes	The returned data will always include the booking number used, and will end with '/GRB'.
Example	GRB
Response	GRB 147 ANDREW___.____,1100,1200 ALFRED___.____,1115,1200 APPLCORE.____,1100,1230 /GRB GRB 123 NONE /GRB

GRC	Get Resource Commit
Purpose	Reads any data for the given resource between the given times
Syntax	GRC <name\$> <start-time%> <end-time%>
Where	<ul style="list-style-type: none"> • <name\$> is the name of the resource • <start-time%> is the start time for the query • <end-time%> is the end time for the query <p>Start and end time are optional. If absent, the start and end of the day are used (ie 0001 and 2359).</p>
Notes	
Example	GRC ANDREW 0800 1600 GRC DAVE
Response	GRC ANDREW___.____ 147,1100,1200 192,1415,1500 111,0830,0900 /GRC GRC DAVE___.____ NONE /GRC

GRL	Get Resource List
Purpose	Show all resources with one initial letter
Syntax	GRL <initial\$>

Where	<ul style="list-style-type: none"> • <initial\$> is the first letter of the group of resources to be listed
Notes	
Example	GRL A
Response	GRL ALFRED ANDREW APPLCORE ANDOVER /GRL

GRU	Get Resource Update
Purpose	Returns the user's resource update mode setting
Syntax	GRU
Where	No parameters
Notes	This relates to the option to get all resource updates immediately notified to other logged-on clients.
Example	GRU
Response	GRU 1

SRC	Set Resource Commit
Purpose	Adds or removes a timed use of a resource
Syntax	SRC <add/del> <name\$> <start-time%> <end-time%>
Where	<ul style="list-style-type: none"> • <add/del> select whether to add or remove the booking • <name\$> is the name of the resource • <start-time%> is the start time for the booking to be removed • <end-time%> is the end time for the booking to be removed
Notes	
Example	SRC
Response	SRC

SRR	Set Resource Remove
Purpose	Remove a resource name

Syntax	SRR <name\$>
Where	<ul style="list-style-type: none"> • <name\$> is a resource name Quotes are not necessary
Notes	
Example	SRR Andover
Response	RLC A

SRN	Set Resource New
Purpose	Add a resource name
Syntax	SRN <name\$>
Where	<ul style="list-style-type: none"> • <name\$> is a resource name Quotes are not necessary
Notes	
Example	SRN Andover
Response	RLC A

SRU	Set Resource Update
Purpose	Sets the user's update mode setting
Syntax	SRU <0/1>
Where	<ul style="list-style-type: none"> • 1 to enable update notifications • 0 to disable update notifications
Notes	This relates to the option to get all resource updates immediately notified to other logged-on clients.
Example	SRU
Response	None

5.18 Resource Responses

RLC	Resource List Change
Purpose	Informs clients of changes to the list of resource names
Syntax	RLC <initial\$>

Where	<ul style="list-style-type: none"> • <initial\$> is the first character of the changed name
Notes	The message is sent to all logged-on clients with Resource Update notification active
Example	RLC B

6 Documents referenced

See also the main FabianServer32 documentation.

7 Notes

Please report any errors found in this documentation to the Author or BNCS Support
collediacontrol.it-solutions.gb@atos.net

7.1 Warning

During the revision of this document it has been noted that some of the commands as implemented don't operate in conjunction with CSI in a way that appears to result in expected operation. Eg File commands - AGF, AIF, ARF.

8 Version history

8.1 Driver version

Version No	Date	Details	Name
1	14/06/2002	Document on which this is based.	Chris Gil
4.07.02	14/10/09	Version current at time of generation of this document.	Richard Kerry
4.09.01	31/3/2011	Adds Filtering. Adds connection port specification by User.	Richard Kerry

8.2 Document version

Version No	Date	Details	Name
1	Feb 09	Updated template	A Atkin
2	10 Nov. 09	First version using this template and first version for FabianServer32 (rather than v3fbserv).	Richard Kerry
2.1	Feb 2010	Some slight formatting changes	Andrew Prince
	31 March 2011	Adds GSA command	Richard Kerry
	03 March 2015	Amended styling to current conventions	Richard Kerry
	27/04/2016	Updated to ensure all commands and responses are listed. (Bug 2793)	Richard Kerry
	09/05/2016	Revised to fix undesired duplication of command grids within listing in section 4. Revised to note further commands that apply Access Control, added in Fabian Server version 4.9.22.	Richard Kerry
	30/06/2016	Revised following changes to detail of Access Control and registration usage for database related commands and responses.	Richard Kerry
	03/10/2016	Revised following changes to SUN and GSF commands. The SUN command will now log-out any logged-in user (but not close the TCP socket). This is to prevent any privileges available to the old user being made available to a new user for whom they may not be appropriate. (Bug 2829) The GSF command has been disallowed as it has been discovered that it can allow access to any file. (Bug 2830)	Richard Kerry
	10/10/2016	The SUP command, if used at any time other than as part of a log-in sequence, will log-out the user (but not close the TCP socket). (further work on Bug 2829)	Richard Kerry

Atos IT Services Limited
4 Triton Square
Regent's Place
London NW1 3HG, UK
<http://uk.atos.net/>

BNCS
4 Triton Square
Regent's Place
London NW1 3HG, UK
collediacontrol.it-solutions.gb@atos.net