Advanced Simulation Technology inc.
441-A Carlisle Drive
Herndon, Virginia 20170 U.S.A.
Tel. (703)471-2104 • Fax. (703)471-2108
www.asti-usa.com

ASTi
Telestra
Remote Control Interface
User Guide

Document: DOC-01-TELS-RC-1

Advanced Simulation Technology inc. 441-A Carlisle Drive, Herndon, Virginia, 20170 USA
Revision B (August 2000)

TABLE OF CONTENTS

Introduction

In order to facilitate easier integration into the simulator environment it is possible to perform system commands from a remote computer system, connected to the Telestra unit via a TCP/IP Ethernet connection. This greatly simplifies the inclusion of Telestra into the wider HLA based system control architecture, particularly due to interaction of RTI master exec software and the satellite applications that rely on this. Also, this removes the necessity of running the system with a keyboard and monitor permanently connected to the unit.

The Telestra system can be managed by remote control through a TCP/IP interface. Through this interface, a host computer can:

- Tell the Telestra federate to join or resign a federation
- Specify the federate name, federation name to join, and fed file
- Direct the Telestra system to shutdown or reboot

The host can also get information from the Telestra system:

- Joined/resigned status
- Current federate name, federation name, and fed file name
- RTI network activity information
- Lists of objects the federate is currently aware of

The host can only get this information if it is controlling the Telestra system. The commands and responses through this interface are all text based, human readable messages. A host emulator program is supplied with the Telestra system that can act as a console attached to the Telestra federate, either on the local box or a remote one. The connection is designed to be robust - if the connection is unexpectedly closed or broken, the socket will automatically re-open and begin listening for a new connection.

Also described in this document is a tool called hostemu (short for Host Emulator), which provides a method of experimenting with the Telestra remote control interface. There is also a method for printing the traffic on the remote control interface to the screen, which can be used in debugging the interface.

Setting Up the Connection

The remote control interface is set up in the configuration file for the Telestra federate (more information on the configuration file can be found in the document Telestra Quick Start Guide). For example, inserting “Control_Port = 45001” into the .cfg file will cause the Telestra federate to set the remote control interface to listen for TCP/IP connections on UDP port 45001. When “telestra” is run from the console, it takes remote control interface commands directly from the console it was run from if no remote control interface port was specified.

List of Commands and Responses

The Telestra federate expects all command lines to be terminated with a <cr> character (\n to you C programmers). If two command lines come at once, they will be executed in order. ***A command will not be executed until the final <cr> is received.***

Commands are not case sensitive. Some data for the commands may be (for example, the fed file name must be in the correct case). Where possible, data will not be case sensitive.

Telestra system replies that are of variable length (for example, when retrieving statistics or object lists) will end with the line “ENDLIST<cr>”. This is indication to the host computer that no more data is being sent.

If the interface receives a command it does not recognize, it will return “UNKNOWN COMMAND “ followed by the unrecognized command. If it receives a line of whitespace (followed by a <cr>), it will respond with a <cr>.

`HELP or ?`

Either command returns a list of all the supported remote control interface commands.

`JOIN <Federation Name>`

This command directs the Telestra federate to join the federation with the name specified. If no name is given, it will join the last federation name that was set. If the federate is already joined to a federation, it will resign and attempt to join again. It will do this even if the federation name is unchanged. If no federation name has been set (either through the remote control port or from the configuration file), then the JOIN command will fail and respond “JOIN <NONAMEGIVEN> FAIL”. The response to the JOIN command is either “JOIN FederationName OK”, “JOIN FederationName FAIL”, or “JOIN FederationName FAIL BAD_FED_FILE”. The last response occurs if the federate could not get handles for all of the class names and attribute names it requested from the RTI. The normal cause of this is that the .fed file does not have the ASTi SOM information incorporated into it.

`RESIGN`

This command tells the federate to resign. The response will be “RESIGN OK” or “RESIGN FAIL”. If the federate is not joined to a federation, no action is taken and the federate will respond “RESIGN OK”.

`SHUTDOWN`

This command will instruct the Telestra system to do an orderly system shutdown. If the federate is currently joined to a federation, it will resign before shutting down. The Telestra response to this command is “GOODBYE”, following by the connection being broken. The shutdown will not complete until some time after the “GOODBYE” response is received. It is a good practice to wait at least 60 seconds between receiving the “GOODBYE” and powering off the Telestra system. Otherwise, the file system may be corrupted.

`RESTART`

Acts exactly like SHUTDOWN, except it does an orderly reboot of the system.


```
NAME
NAME FEDFILE <NewFedFileName>
NAME FEDERATION <NewFederationName>
NAME FEDERATION_NUMBER <NewFederationNumber>
NAME FEDERATE <NewFederateName>
```

This command controls the setting of four federate options. <NewFedFileName> is the name of the .fed file the federation uses. <NewFederationName> is the name of the federation the Telestra federate will join. <NewFederationNumber> is the number of the federation as it is known to the DACS. (If you are only running a single federation, this should be set to 1.) <NewFederateName> is the name of the Telestra federate. The NAME command alone will simply return the values of these variables. The other forms will set the corresponding value, and return the values for all the variables.

The changes do not take effect until the federate joins the federation. If the values of the name variables are changed while the telestra federate is joined, the changes will not take effect until another JOIN command is issued.

```
STATUS
```

Returns either “JOINED <FederationName>” or “RESIGNED”. The federation name returned is the one currently joined, even if the NAME command has been issued to change the FEDERATION variable.

```
OBJECT
OBJECT TRANSMITTER
OBJECT RECEIVER
OBJECT ENTITY
OBJECT LOCAL
```

When called with no arguments, the OBJECT command returns a list of objects that the telestra federate is aware of. The telestra federate subscribes to Receiver, Transmitter, and entity objects. The name, RTI handle, and type of the object is returned. In addition, an object is designated as “local” if it is owned by the telestra federate in question (for transmitters and receivers), or if there is a local transmitter or receiver attached to it (in the case of entities).

A sample output looks like this:

```
Receiver   Viper3.UHF1.rx            handle: 65538 0x10002
Receiver   Viper3.UHF2.rx            handle: 65542 0x10006
Receiver   Viper3.VHF1.rx            handle: 65540 0x10004
Receiver   Viper3.VHF2.rx            handle: 65546 0x10010
Receiver local GeoCntr-6.Radio021.rx handle: 196630 0x30016
Receiver local GeoCntr-6.Radio022.rx handle: 196628 0x30014
Transmitter     Viper3.UHF1.tx       handle: 65537 0x10001
Transmitter     Viper3.UHF2.tx       handle: 65541 0x10005
Transmitter    Viper3.VHF1.tx        handle: 65539 0x10003
Transmitter    Viper3.VHF2.tx        handle: 65545 0x10009
Transmitter local GeoCntr-6.Radio021.tx handle: 196609 0x30001
Transmitter local GeoCntr-6.Radio022.tx handle: 196613 0x30005
ENDLIST
```


When issued with an argument, the OBJECT command will only return a subset of the objects - either transmitters, receivers, entities, or local objects.

Note the ASTi naming convention: EntityName.RadioName.tx, where EntityName is the entity the radio is attached to. RadioName is the name given to the radio object in the model, and .tx and .rx designate transmitters and receivers. This naming convention is used to make the output to debugging tools more “human readable”. Other radios that do not follow this naming convention will still be interoperable with the ASTi system.

The host entity name for a radio is also published as an attribute of the transmitters and receivers. See the ASTi SOM for more details.

`ACTIVITY`

This command returns activity counters from the Telestra system. A sample output would be:

```
lifeCounter = 68
Attributes Updates rx:10 tx:441 rx ignored:0
Interactions rx:7437 tx:4198
Number of Transmitters local:11 rti:5
Number Entities total:0 attached:0
```

The “lifeCounter” increments once a second while the entity is joined to a federation. It is a general indicator of federate health.

“Attribute Updates” reports the cumulative number of attribute updates sent and received by the federate.

“rx ignored” is the number received and ignored - this happens when the federate receives an update to an entity object that none of its radios are attached to. Note, attributes are object parameters such as power status, frequency, etc, and an attribute update occurs when one of these changes in some way.

“Interactions” is the number of interactions sent and received. Currently, this only includes audio packet interactions.

“Number of Transmitters” returns the number of transmitters owned by the telestra federate (local), and the number discovered from the RTI.

“Number Entities” returns the total number of discovered entities, and the number of entities that the telestra federate has radios attached to.

The telestra federate does not generate any type of entity objects.

Using the Host Emulator, hostemu

The Telestra system is shipped with a host emulation program. It is located in the /home/hlauser/bin directory, and can be run by typing:

```
hostemu
```

at the command prompt. The purpose of the host emulation is to experiment with the host interface and to verify that it is acting correctly. This applies to both the local Telestra unit, and to other remote Telestra units (if available).

If hostemu is run with no arguments, it will search the local box for sockets that may be remote control interface sockets for telestra systems. It will then attempt to attach to the first such socket it finds. If it is unable to find and connect to a socket, the hostemu program will print a usage message.

When attached, hostemu acts as a console. Text typed in is sent to the telestra remote control interface, and the replies are printed out on the console.

hostemu can be directed to connect to a specific socket on the local machine or to a remote Telestra unit over a network connection. To connect to port 45003 on the local machine, for example, you would enter

```
hostemu 45003
```

To connect to port 45003 on a remote network machine with an IP address of 192.42.172.151, you would type

```
hostemu 45003 192.42.172.151
```

or, alternatively,

```
hostemu 45003 HLAlab1.asti-usa.com
```

Either of the latter two commands would allow the local Telestra to act as a remote host for a second Telestra unit.

Debugging the Remote Control Interface

The telestra software can print the commands sent and received through the remote control interface. To do this, add the line

```
DEBUG_HOST = ON
```

to the configuration file. Commands to and from the telestra federate will be printed to the console:

```
From Host: ‘status\n’
To telestra : ‘status’
To Host: ‘RESIGNED\n’
```

“From Host” is the data that was received from the host computer. Note that the <cr> is explicitly indicated. A command will not be evaluated until a <cr> is received.

“To telestra” is the command as it is passed to the telestra system. These might be different if the host sends two commands at once. For example. if the host sends the string “status\nobject\n\n”, the debug output would be:

```
From Host: ‘status\nobject\n\n’
To telestra : ‘status’
To Host: ‘RESIGNED\n’
From Host: ‘’
To telestra : ‘object’
To Host: ‘ENDLIST\n’
From Host: ‘’
To telestra : ‘’
To Host: ‘\n’
```

The input software parses the input and passes the commands to the telestra one at a time.

When debugging is complete, you should comment out the DEBUG_HOST command from the configuration file.

The host emulation program, hostemu, can be used to see how the telestra software responds to various commands.

Using the Telestra System without a Monitor, Keyboard and Mouse

The Telestra system is configured to start automatically at system boot. The Telestra federate will start using /home/hlauser/default.cfg as its configuration file. If the Telestra federate is started using the Remote Control Interface, it will NOT be accessible from the console, so no keyboard input or screen output will be available from the federate. Note however that if a keyboard, mouse, and monitor are connected then the “hostemu” software can be used to provide local control.

Because the Telestra system is running Linux, it is possible to connect to the system via telnet. This allows another method of using it without a keyboard, monitor, or mouse.

Warning: If you do inadvertently start-up the Telestra unit, without a keyboard, mouse, and monitor connected Telestra will ONLY be accessible from a remote computer. It is not possible to connect the peripherals once Telestra has started. If you do not have a remote computer capable of connecting to the Telestra unit, your only option is to switch off the Telestra system. If this is done, do be aware that it is possible to cause file errors and possible disk corruption, due to not being able to shutdown the unit in a controlled fashion. If this occurs the only option may be to rebuild the disk. ASTi therefore recommends that during all testing of the remote interface operation a keyboard, mouse and monitor combination be connected to Telestra.

