Hardware Test eXecutive Daemon (HTXD) User Manual

Version 1.0

March 2016

Document Owner: Avinash Kuttan
avkuttan@in.ibm.com

IBM India Systems Lab

Table of Contents

1. Overview

HTX [Hardware Test eXecutive] is a test tool suite, which is used by various System p validation labs to verify the System p hardware design. It is used during processor bring up, hardware system integration, I/O Verification (IOV), Characterization and Manufacturing. The goal of HTX is to stress test the system by exercising all hardware components concurrently in order to uncover any hardware design flaws and hardware hardware or hardware-software interaction issues. HTX runs on AIX, Bare Metal Linux(BML) and distribution Linux. HTX offers a light weight HTX daemon (HTXD) which support command line interface and menu based user interactive interface.

HTXD is a light weight HTX daemon, which is based on client server architecture model. HTXD is suitable interface for automation and remote administration of test process. The important components of HTXD are:

i. Light weight HTX daemon (htxd)

ii. Command line client interface (htxcmdline),

iii. UI client (htxscreen)

htxd starts with system start up and keep listening for an incoming command from htxcmdline or htxscreen. HTXD also starts up with a new HTX installation.

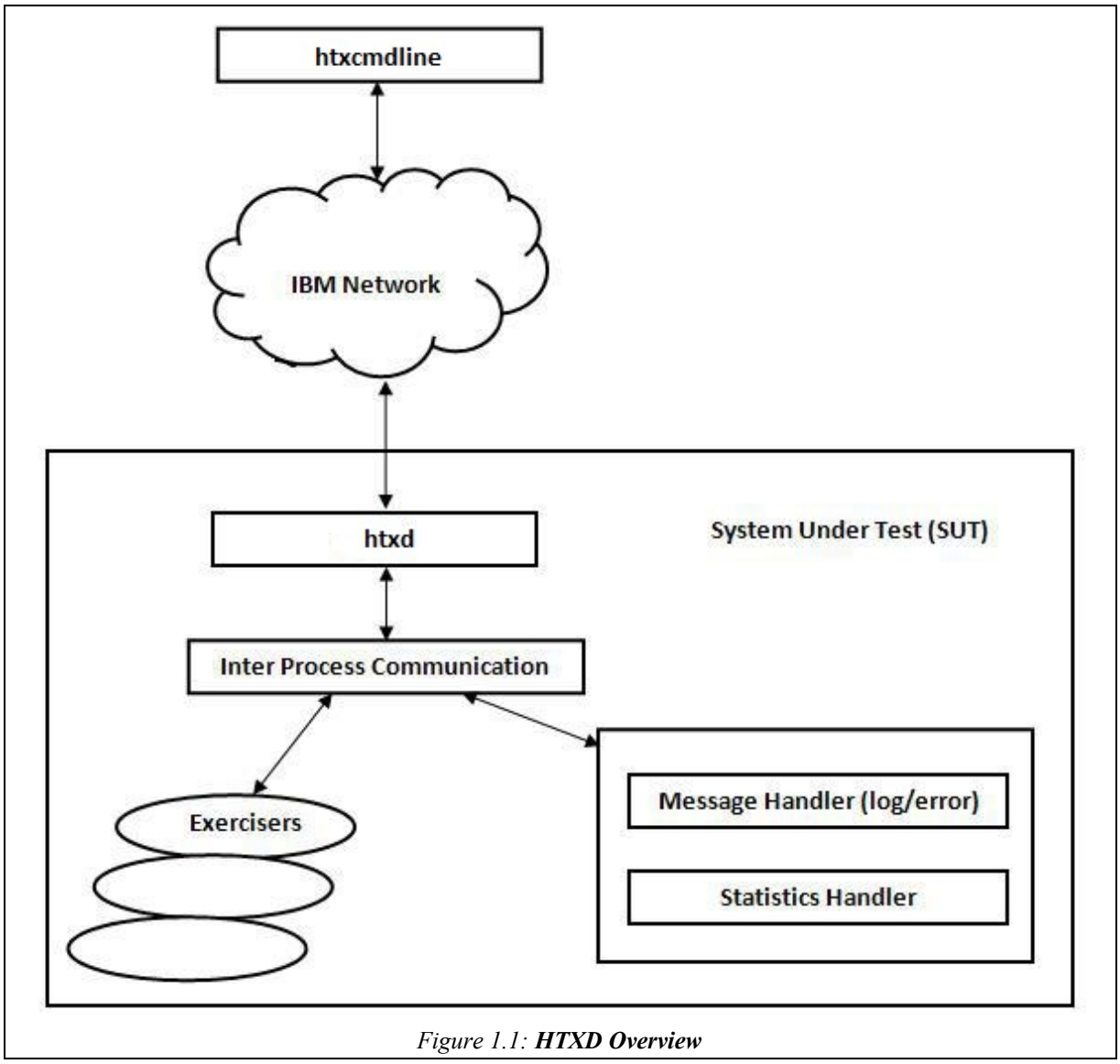

Figure 1.1: ***HTXD Overview***

HTXD daemon (htxd): 'htxd' is a light wieght htx daemon which runs as a back ground process. With system (SUT) start up 'htxd' starts up in idle mode and listens on port number 3492 to accept any incoming command. It monitors and controls the execution of all hardware exercisers. Before starting the daemon it has to be ensured that there is no instance of HTX currently running on SUT. Daemon maintains entire state of the run and is also responsible for maintaining message/error log infrastructure.

Command line client interface (htxcmdline): 'htxcmdline' is the client application, which can be run from any machine having network accessibility to SUT. User can send various commands to daemon using 'htxcmdline' to start/stop, monitor and control the test execution. Results received from daemon are displayed on standard output.

2. Obtaining HTXD

- HTXD components are bundled into HTX package.

3. HTX installation

HTXD is available with HTX package. Following are the steps to install HTX software on SUT.

- Log in to SUT machine as root user
- Download the HTX package as mentioned above
- Uninstall the HTX software, if older version is already present, by using following command:
- Install the downloaded HTX package, by using following command

AIX (LPP package)
Verify existing installation:
lslpp -L | grep htx
Uninstall:
installp -u htx
Install:
installp -qacd <HTX lpp package name> -FX all

Ubuntu (DEB package)
Verify existing installation:
dpkg --list "htx*"
Uninstall:
dpkg -r htxubuntu
Install:
dpkg -i <HTX deb package name>

Note: If OS/HTX is obtained through NIM, there are no extra steps required to install HTX.

4. Starting HTX daemon

HTX daemon (htxd) runs at the system under test. htxd is started by following methods

1. htxd starts with system start up, if the HTX package is installed
2. to start from start up script
   /usr/lpp/htx/etc/scripts/htxd_run
3. to start as service
   Linux: /etc/init.d/htx.d start
   AIX: /etc/rc.d/rc2.d/Shtxd start
4. with system reboot

5. with HTX package installation

   Note: HTXD does not create MDTs on the daemon start up. After the start up, HTXD is at idle state and it just listens for any incoming command.

5. Stopping HTX daemon

1. htxd runs on the system until shutdown the system
2. to stop from startup script

   /usr/lpp/htx/etc/scripts/htxd_shutdown
3. to stop as service

   Linux: /etc/init.d/htx.d stop

   AIX: /etc/rc.d/rc2.d/Shtxd stop
4. with system reboot

Note: htxd service command options:

Linux: /etc/init.d/htx.d [start | stop | restart | status]

AIX: /etc/rc.d/rc2.d/Shtxd [start | stop | restart | status]

6. HTXD command line interface (htxcmdline)

'htxcmdline' program (as part of HTX package) is a command line interface for HTXD to control and monitor hardware test execution. In case of any error while running htx command, appropriate error message will be displayed.

'htxcmdline' command options are described below:

Syntax:

'htxcmdline [-sut <host name/IP Address>] <OPTIONS>'

where,

-sut : <host name/IP Address> is IP address or host name of SUT (system under test) where daemon is running. Default is “localhost”

Options:

-createmdt
-listmdt
-refresh
-select [-mdt <mdt_name>]
-run [-mdt] <mdt_name>
-shutdown [-mdt] <mdt_name>
-getactecg
-query [all] [<device_name1> <device_name2> ...] [-mdt <mdt_name>]

-activate [all] [<device_name1> <device_name2> ...] [-mdt <mdt_name1>]
-suspend [all] [<device_name1> <device_name2> ...] [-mdt <mdt_name]
-terminate [all] [<device_name1> <device_name2> ...] [-mdt <mdt_name>]
-restart [all] [<device_name1> <device_name2> ...] [-mdt <mdt_name>]
-coe [all] [<device_name1> <device_name2> ...] [-mdt <mdt_name>]
-soe [all] [<device_name1> <device_name2> ...] [-mdt <mdt_name>]
-status [all] [<device_name1> <device_name2> ...] [-mdt <mdt_name]
-getstats [-mdt <mdt_name>]
-getecgsum [-mdt <mdt_name]
-geterrlog
-clrerrlog
-getvpd
-exersetupinfo -mdt <mdt_name>
-set_eeh [<EEH flag(1 0r 0)>]
-set_kdblevel [<kdb level flag (1 or 0)>]
-bootme [on | off | status]
-cmd <command to execute>

Note: With out any option or with any invalid option 'htxcmdline' command displays its usage. If path to ECG/MDT is not provided, default path for ECG/MDT file is /usr/lpp/htx/mdt

6.1. Option: createmdt

To create MDTs.

Syntax:

htxcmdline -sut <hostname> -createmdt

Example:

htxcmdline -sut localhost -createmdt

htxcmdline -createmdt

Details:

- When HTXD starts it does not create any MDT.
- createmdt option creates all the system MDTs
- Sample output for a successful execution is provided below.

```
#htxcmdline -sut localhost -createmdt

######################### Result Starts Here ###################################
mdts are created successfully.
######################### Result Ends Here #####################################
#
```

Figure 6.1: ***createmdt command output***

6.2. Option: listmdt

To list MDTs.

Syntax:

htxcmdline -sut <hostname> -listmdt

Example:

htxcmdline -sut localhost -listmdt

htxcmdline -listmdt

Details:

- listmdt option lists all the MDTs

6.3. Option: run

Starts test for specified ECG/MDT.

Syntax:

htxcmdline -sut <hostname> -run [-mdt <mdt_name>]

Example:

htxcmdline -sut localhost -run -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -run -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -run

Details:

- It starts test on devices present in specified ECG/MDT and sets ECG/MDT status as ACTIVE.
- Default mdt name is mdt.bu
- Sample output for a successful execution is provided below.

```
#htxcmdline -sut localhost -run -ecg /usr/lpp/htx/mdt/mdt.bu

######################### Result Starts Here ##################################
ECG (/usr/lpp/htx/mdt/mdt.bu) Activated.
######################### Result Ends Here ####################################
#
```

Figure 6.2: ***run command output***

6.4. Option: select

To select an MDT/ECG.

Syntax:

htxcmdline -sut <hostname> -select [-mdt <mdt_name>]

Example:

htxcmdline -sut localhost -select -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -select -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -select

Details:

- It just loads the specified MDT into the system, does not start run any exercisers
- Any device setting can be changed after the MDT is selected.
- Default mdt name is mdt.bu
- Sample output for a successful execution is provided below.

```
#htxcmdline -sut localhost -select -ecg /usr/lpp/htx/mdt/mdt.bu

######################### Result Starts Here ##################################
ECG (/usr/lpp/htx/mdt/mdt.bu) is selected
########################## Result Ends Here ###################################
#
```

Figure 6.3: ***select command output***

6.5. Option: shutdown

Shuts down test for specified ECG/MDT.

Syntax:

htxcmdline [-sut <hostname>] -shutdown [-mdt <mdt_name>]

where <mdt_name> is absolute ECG/MDT file name.

Example:

htxcmdline -sut localhost -shutdown -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -shutdown -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -shutdown

Details:

- Default MDT is currently running MDT.
- It shuts down test, and keeps the htxd on idle state.
- Sample output for a successful execution is provided below.

```
#htxcmdline -sut localhost -shutdown -ecg /usr/lpp/htx/mdt/mdt.bu

######################### Result Starts Here ##################################
Currently running ECG/MDT : No MDT/ECG is currently running
============================
ECG (/usr/lpp/htx/mdt/mdt.bu) shutdown successfully
########################## Result Ends Here ###################################
#
```

Figure 6.4: ***shutdown command output***

6.6. Option: getactecg

Displays currently running ECG/MDT name.

Syntax:

htxcmdline -sut <hostname> -getactecg

Example:

htxcmdline -getactecg

htxcmdline -sut localhost -getactecg

Details:

- Sample output for a successful execution is provided below.

```
#htxcmdline -sut localhost -getactecg

######################### Result Starts Here ##################################
Currently running ECG/MDT : /usr/lpp/htx/mdt/mdt.bu
============================
/usr/lpp/htx/mdt/mdt.bu

########################## Result Ends Here ###################################
```

Figure 6.5: ***getactecg command output***

6.7. Option: query

Displays test details of each device.

Syntax:

htxcmdline-sut <hostname> -query [all] <device_name1> [<device_name2> ...] [-mdt <mdt_name>]

htxcmdline -sut <hostname> -query all

Example:

htxcmdline -sut localhost -query -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -sut localhost -query mem -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -sut localhost -query fpu* -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -query -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -query

Note: '*' stands for device number.

Details:

- Sample output for a successful execution is provided below.

```
#htxcmdline -sut localhost -query -ecg /usr/lpp/htx/mdt/mdt.bu

######################## Result Starts Here ###############################
Currently running ECG/MDT : /usr/lpp/htx/mdt/mdt.bu
============================
-------------------------------------------------------------------------------
 Device   ST  ACTIVE COE Last Update  Count Stanza Last Error   Slot/  Num_errs
             SUSPEND SOE day Time                 Day  Time    Port
-------------------------------------------------------------------------------
mem       RN  ACTIVE SOE 277 01:59:29 324      5   NA   NA       NA            0
fpscr0    RN  ACTIVE SOE 277 01:59:29 297      1   NA   NA       NA            0
fpu0      RN  ACTIVE SOE 277 01:59:22 5        37  NA   NA       NA            0
fpu1      RN  ACTIVE SOE 277 01:59:27 6        30  NA   NA       NA            0
cpu0      RN  ACTIVE SOE 277 01:58:33 6        12  NA   NA       NA            0
cpu1      RN  ACTIVE SOE 277 01:58:43 3        12  NA   NA       NA            0
rng       RN  ACTIVE SOE 277 01:59:06 12       0   NA   NA       NA            0
######################## Result Ends Here #################################
#
```

Figure 6.6: ***query command output***

6.8. Option: activate

Activates device(s).

Syntax:

htxcmdline -sut <hostname> -activate <device_name1> [<device_name2> …] -mdt <mdt_name>

htxcmdline -sut <hostname> -activate all

Example:

htxcmdline -sut localhost -activate mem -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -activate mem -mdt /usr/lpp/htx/mdt/mdt.bu

Details:

- If no device name is specified with the ECG/MDT, the command will set the state of all the devices present in the ECG/MDT as ACTIVE.
- With option 'all', details of all the available devices are displayed.
- If the device status is stopped (ST) , the command changes it to running (RN).
- Sample output for a successful execution is provided below.

```
#htxcmdline -sut localhost -activate mem -ecg /usr/lpp/htx/mdt/mdt.bu

######################## Result Starts Here ###############################
Currently running ECG/MDT : /usr/lpp/htx/mdt/mdt.bu
============================
State   Dev     Adapt Desc   Device Desc         Slot Port
-------------------------------------------------------------
ACTIVE  mem     64bit        memory              NA
######################### Result Ends Here ################################
#
```

Figure 6.7: ***activate command output***

6.9. Option: suspend

To suspend device(s).

Syntax:

htxcmdline [-sut <hostname>] -suspend <device_name1> [<device_name2> …] [-mdt <mdt_name>]

htxcmdline -sut <hostname> -suspend all

Example:

htxcmdline -sut localhost -suspend mem -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -suspend mem -mdt /usr/lpp/htx/mdt/mdt.bu

Details:

- If the device is Running, the command change the state to Stopped (ST).
- If no device name is specified with the ECG/MDT, the command will set the state of all the devices present in the ECG/MDT as Suspend.
- With option 'all', state of all the available devices are set to suspend.
-
- Sample output for a successful execution is provided below.

```
# esrv -sut localhost -suspend mem -ecg /usr/lpp/htx/mdt/mdt.mem

######################## Result Starts Here ###############################
Currently running ECG/MDT : mdt.mem
============================

State    Dev      Adapt Desc    Device Desc      Slot    Port
-------------------------------------------------------------

Suspend mem       64bit        memory           NA

######################## Result Ends Here #################################
```

Figure 6.8: suspend command output

6.10. Option: terminate

Terminates device(s).

Syntax:

htxcmdline -sut <hostname> -terminate <device_name1> [<device_name2> …] -mdt [<mdt_name>]

htxcmdline -sut <hostname> -terminate all

Example:

htxcmdline -sut localhost -terminate mem -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -terminate mem -mdt /usr/lpp/htx/mdt/mdt.bu

Details:

- If no device name is specified with ECG/MDT, the command will set the status of all devices present in the specified ECG/MDT as Terminated (TM).
- With option 'all', status of all devices are set to Terminated (TM).
- Sample output for a successful execution is provided below.

```
#htxcmdline -sut localhost -terminate mem -ecg /usr/lpp/htx/mdt/mdt.bu

######################## Result Starts Here ###############################
Currently running ECG/MDT : /usr/lpp/htx/mdt/mdt.bu
============================
State    Dev      Adapt Desc    Device Desc          Slot Port
-------------------------------------------------------------
ACTIVE   mem      64bit         memory               NA
######################## Result Ends Here #################################
#
```

Figure 6.9: ***terminate command output***

6.11. Option: restart

Restarts terminated device(s).

Syntax:

htxcmdline -sut <hostname> -restart <device_name1> [<device_name2> …] -mdt <mdt_name>

htxcmdline -sut <hostname> -restart all

Example:

htxcmdline -sut localhost -restart mem -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -restart mem -mdt /usr/lpp/htx/mdt/mdt.bu

Details:

- If no device name is specified with ECG/MDT, command will set status of all devices present in the ECG/MDT as Running (RN).
- With option 'all', status of all devices of currently running ECG/MDT(s) are set to Running (RN).
- If device is Suspend state, command changes the device status to Stopped(ST).
- Sample output for a successful execution is provided below.\

```
#htxcmdline -sut localhost -restart mem -ecg /usr/lpp/htx/mdt/mdt.bu

######################### Result Starts Here ##################################
Currently running ECG/MDT : /usr/lpp/htx/mdt/mdt.bu
============================
State   Dev     Adapt Desc   Device Desc         Slot Port
------------------------------------------------------------
ACTIVE  mem     64bit        memory              NA
######################### Result Ends Here ###################################
```

Figure 6.10: ***restart command output***

6.12. Option: coe

Sets device mode as continue on error (COE)

Syntax:

htxcmdline -sut <hostname> -coe <device_name1> [<device_name2> …] -mdt <mdt_name>

htxcmdline -sut <hostname> -coe all

Example:

htxcmdline -sut localhost -coe mem -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -coe mem -mdt /usr/lpp/htx/mdt/mdt.bu

Details:

- If no device name is specified with ECG/MDT, command will set mode of all devices present in the ECG/MDT as COE.
- With option 'all', state of all the available devices are set to continue on error mode.
- Sample output for a successful execution is provided below.

```
#htxcmdline -sut localhost -coe mem -ecg /usr/lpp/htx/mdt/mdt.bu

######################### Result Starts Here ##################################
Currently running ECG/MDT : /usr/lpp/htx/mdt/mdt.bu
============================
State   Dev     Adapt Desc   Device Desc         Slot Port
-------------------------------------------------------------
COE     mem     64bit        memory              NA
######################### Result Ends Here ####################################
#
```

Figure 6.11: ***coe command output***

6.13. Option: soe

Sets device mode as stop on error (SOE)

Syntax:

htxcmdline -sut <hostname> -soe <device_name1> [<device_name2> …] -mdt <mdt_name>

htxcmdline -sut <hostname> -soe all

Example:

htxcmdline -sut localhost -soe mem -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -soe mem -mdt /usr/lpp/htx/mdt/mdt.bu

Details:

- If no device name is specified with ECG/MDT, command will set mode of all devices present in ECG/MDT to SOE.
- With option 'all', state of all available devices are set to SOE.
- Sample output for a successful execution is provided below.

```
#htxcmdline -sut localhost -soe mem -ecg /usr/lpp/htx/mdt/mdt.bu

######################## Result Starts Here ###################################
Currently running ECG/MDT : /usr/lpp/htx/mdt/mdt.bu
============================
State   Dev     Adapt Desc   Device Desc         Slot Port
-------------------------------------------------------------
SOE     mem     64bit        memory              NA
######################## Result Ends Here #####################################
#
```

Figure 6.12: ***soe command output***

6.14. Option: status

Displays current running status of device.

Syntax:

htxcmdline -sut <hostname> -status <device_name1> [<device_name2> …] -mdt <mdt_name>

htxcmdline -sut <hostname> -status all

Example:

htxcmdline -sut localhost -status mem -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -status mem -mdt /usr/lpp/htx/mdt/mdt.bu

Details:

- Sample output for a successful execution is provided below.

```
#htxcmdline -sut localhost -status mem -ecg /usr/lpp/htx/mdt/mdt.bu

######################## Result Starts Here ###################################
Currently running ECG/MDT : /usr/lpp/htx/mdt/mdt.bu
============================
-------------------------------------------------------------
             Last  Update    Cycle  Curr    Last  Error
ST  Device   Day   Time      Count  Stanza  Day   Time
-------------------------------------------------------------
RN  mem      277   19:10:29  20     5       NA    NA
######################## Result Ends Here #####################################
#
```

Figure 6.13: ***status command output***

6.15. Option: getstats

Displays statistics of all devices in the specified ECG/MDT(s).

Syntax:

htxcmdline -sut <hostname> -getstats -mdt <mdt_name>

Example:

htxcmdline -sut localhost -getstats -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -getstats -mdt /usr/lpp/htx/mdt/mdt.bu

Details:

- Command displays running statistics of all devices in ECG/MDT.
- Sample output for a successful execution is provided below.

```
#htxcmdline -sut localhost -getstats -ecg /usr/lpp/htx/mdt/mdt.bu

######################### Result Starts Here ################################
Currently running ECG/MDT : /usr/lpp/htx/mdt/mdt.bu
============================
mem:
  cycles                              =                58
  # good reads                        =                 0
  # bytes read                        =       183044407296
  # good writes                       =                 0
  # total instructions                =                 0
  # bytes written                     =        97174421504
  # good others                       =                 0
  # bad others                        =                 0
  # bad reads                         =                 0
  # bad writes                        =                 0
  # data transfer rate(bytes_wrtn/s)  =        285807104.00
  # data transfer rate(bytes_read/s)  =        538365952.00
  # instruction throughput(MIPS)      =           0.000000
```

Figure 6.14: ***getstats command output***

6.16. Option: getecglist

Displays list of all the available ECG/MDT.

Syntax:

htxcmdline -sut <hostname> -getecglist

Example:

htxcmdline -sut localhost -getecglist

htxcmdline -getecglist

Details:

- The command displays list of all available ECG/MDT files.
- The command also displays total number of available ECG/MDT.

6.17. Option: getecgsum

Displays test summary for a given ECG/MDT(s).

Syntax:

htxcmdline -sut <hostname> -getecgsum [-mdt <mdt_name>]

Example:

htxcmdline -sut localhost -getecgsum -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -sut localhost -getecgsum

htxcmdline -getecgsum

Details:

- If no ECG/MDT is specified with the command it displays summary of all the available ECG/MDT.

```
#htxcmdline -sut localhost -getecgsum -ecg /usr/lpp/htx/mdt/mdt.bu

######################### Result Starts Here ##################################
Currently running ECG/MDT : /usr/lpp/htx/mdt/mdt.bu
============================
/usr/lpp/htx/mdt/mdt.bu is in ACTIVE state
Total Devices       = 7
 Running            = 7
 Suspended          = 0
 Error              = 0
 Partialy Running   = 0
 Terminated         = 0
 Died               = 0
 Hung               = 0
 Completed          = 0
 Inactive           = 0


######################### Result Ends Here ####################################
#_
```

Figure 6.15: ***getecgsum command command***

6.18. Option: geterrlog

Displays HTX error log.

Syntax:

htxcmdline -sut <hostname> -geterrlog

Example:

htxcmdline -sut localhost -geterrlog

htxcmdline -geterrlog

Details:

- HTXD monitors execution of all the hardware exercisers, in case of any error, it logs the error in the error log file /tmp/htxerr. The command 'geterrlog' displays the error log details on the standard output.
- If there is no error, the error log file will be empty and the command displays corresponding output.

6.19. Option: clrerrlog

Clears HTX error log.

Syntax:

htxcmdline -sut <hostname> -clrerrlog

Example:

htxcmdline -sut localhost -clrerrlog

htxcmdline -clrerrlog

Details:

- The command delete all the content of the error log file /tmp/htxerr. After execution of the command, the file remains as an empty file.
- It is highly recommended that user should use this command to clear error log, and do not delete or edit the error log file directly.

6.20. Option: getvpd

Displays status details of system resources.

Syntax:

htxcmdline -sut <hostname> -getvpd

Example:

htxcmdline -sut localhost -getvpd

htxcmdline -getvpd

Details:

- This command displays system VPD details on the standard output.

6.21. Option: exersetupinfo

Displays setup status of exercisers in specified ECG/MDT.

Syntax:

htxcmdline -sut <hostname> -exersetupinfo -mdt <mdt_name>

Example:

htxcmdline -sut localhost -exersetupinfo -mdt /usr/lpp/htx/mdt/mdt.bu

htxcmdline -exersetupinfo -mdt /usr/lpp/htx/mdt/mdt.bu

Details:

- Set up of all exercisers in specified ECG/MDT is successful, command displays value '1', if any of the exercisers in the specified ECG/MDT is down, command displays value '0'
- Sample output for a successful execution is provided below

```
#htxcmdline -sut localhost -exersetupinfo -ecg /usr/lpp/htx/mdt/mdt.bu

######################### Result Starts Here ##################################
Currently running ECG/MDT : /usr/lpp/htx/mdt/mdt.bu
============================
setup_of_all_exers_done = 1
########################## Result Ends Here ###################################
#
```

Figure 6.16: ***exersetupinfo command output***

6.22. Option: bootme

Displays setup status of exercisers in specified ECG/MDT.

Syntax:

htxcmdline -sut <hostname> -bootme [on | off | status]

Example:

htxcmdline -sut localhost -bootme on

htxcmdline -bootme on

Details:

- bootme option is to enable or disable bootme testing
- bootme status shows the current bootme status

6.23. Option: refresh

To refresh HTXD configuration.

Syntax:

htxcmdline -sut <hostname> -refresh

Example:

htxcmdline -sut localhost -refresh

htxcmdline - refresh

Details:

- In case of any change in .htx_profile HTXD should be refreshed to accept the change
- No MDT should be running while refreshing HTXD.
- Sample output for a successful execution is provided below

```
#htxcmdline -sut localhost -refresh

######################### Result Starts Here ###################################
Currently running ECG/MDT : No MDT/ECG is currently running
============================
system refresh is completed successfully
########################## Result Ends Here ####################################
#
```

Figure 6.17: ***refresh command output***

6.24. Option: set_eeh

To set or unset EEH flag.

Syntax:

htxcmdline -sut <hostname> -set_eeh [<EEH flag value (1 0r 0)>]

Example:

htxcmdline -sut localhost -set_eeh 1

htxcmdline -set_eeh 1

Details:

- set EEH flag with the provided argument value.
- Valid EEH flag values are 1 or 0
- Sample output for a successful execution is provided below

```
#htxcmdline -sut localhost -set_eeh 1

######################### Result Starts Here ###################################
set eeh flag successfully
########################## Result Ends Here ####################################
#
```

Figure 6.18: ***set_eeh command output***

6.25. Option: set_kdblevel

To set or unset KDB level flag.

Syntax:

htxcmdline -sut <hostname> -set_kdblevel [<kdb level flag (1 or 0)>]

Example:

htxcmdline -sut localhost - set_kdblevel 1

htxcmdline - set_kdblevel 1

Details:

- set KDB level flag with the provided argument value.
- Valid KDB level flag values are 1 or 0
- Sample output for a successful execution is provided below

```
#htxcmdline -sut localhost -set_kdblevel 1

######################### Result Starts Here ###################################
set kdb level flag successfully
########################## Result Ends Here ####################################
#
```

Figure 6.19: ***set_kdblevel command output***

6.26. Option: cmd

Executes shell command on SUT and displays output on standard output.

Syntax:

htxcmdline -sut <hostname> -cmd "<shell command>"

Example:

htxcmdline -sut localhost -cmd "ls -ltr /usr/lpp/htx/bin"

htxcmdline -cmd "ls -ltr /usr/lpp/htx/bin"

Note:

User should ensure following points while running '-cmd' option:

- ✓ If the 'shell command' is not complete and expect more input from user, that may cause the daemon for long waiting to respond. In this case, user has to explicitly kill the waiting command to make the daemon active.
- ✓ Do not run 'htxcmdline' command as the 'shell command' with '-cmd' option
- ✓ Do not mix shell commands with 'htxcmdline' commands

7. Recommended steps to run HTXD

- Log on to SUT as root user.
- Before start testing with HTXD, ensure that following conditions are met on SUT,
  - ✓ compatible HTX package is installed
  - ✓ '/tmp' directory is having at least 256MB of free space
  - ✓ swap space is enabled as per the OS norms
  - ✓ media drives contain appropriate test media
  - ✓ no other instance of HTX/STX is active on SUT

  Note: Below mentioned 'htxcmdline' commands can be executed from any machine having network accessibility to SUT or from SUT itself.
- htxd does not generate any mdts automatically with its start up. So following command is required to dynamically generate mdts.

  'htxcmdline -sut <SUT hostname> -createmdt'
- View the list of available MDTs

  'htxcmdline -sut <SUT hostname> -listmdt'
- Now user can start HTX run with specific MDT

  'htxcmdline -sut <SUT hostname> -run -mdt <mdt_name>'
- To query the device list

  'htxcmdline -sut <SUT hostname> -query -mdt <mdt_name>'
- To check status of current run

  'htxcmdline -sut <SUT hostname> -status -mdt <mdt_name>'
- To change SOE/COE/ACTIVATE/SUSPEND state

  'htxcmdline -sut <hostname> -coe <device name> -mdt<mdt_name>'

  'htxcmdline -sut <hostname> -soe <device name> -mdt<mdt_name>'

  'htxcmdline -sut <hostname> -activate <device name> -mdt <mdt_name>'

  'htxcmdline -sut <hostname> -suspend <device name> -mdt <mdt_name>'
- Check for errors

  'htxcmdline -sut <SUT hostname> -geterrlog'
- Stop the test

  'htxcmdline -sut <SUT hostname> -shutdown -mdt <mdt_name>'

  Note: after stopping the MDT by shutdown command, the daemon still running and listen for any incoming command.

- Miscellaneous settings:
- EEH flag setting. default is 1.

  'htxcmdline -sut <SUT hostname> -set_eeh <newvalue>'

- Trap to KDB setting. Default kdb level flag is 0, and trap to KDB by HTX exercisers on miscompare is disabled.

  'htxcmdline -sut <SUT hostname> -set_kdblevel <new value>'

8. Files generated by HTXD

Following are the important files generated by HTXD on SUT.

8.1. /tmp/htxd.start.stop.time

This file captures HTXD/ECG/MDT start/stop time.

Tip: User can refer this file to understand the run time history of HTXD.

8.2. /tmp/htxd_trace

This file is recorded with all logs from HTX daemon for current run.

Tip: 'tail' command can be used to follow the log update.

Sample usage: 'tail -50f /tmp/htxd_trace'

8.3. /tmp/htxmsg

HTX runs a message handling task (hxsmsg), which records all the information/error messages sent by the hardware exercisers and HTX daemon.

8.4. /tmp/htxerr

HTX runs a message handling task (hxsmsg), which records all the error messages sent by the hardware exercisers and HTX daemon. Any update in this file would mean failure of the test.

8.5. /tmp/htxstats

HTX runs a statistics handling task (hxstats) which generates the statistics of hardware exercisers and store it in /tmp/htxstats file after the execution of 'getstats' command option.

For each device, it records following information in this file.

cycles {number of rule file passes completed}

good reads

bytes read

good writes

total instructions

bytes written

good others

bad others

bad reads

bad writes

data transfer rate(bytes_wrtn/s)

data transfer rate(bytes_read/s)

instruction throughput(MIPS)

Note: An empty /tmp/htxstats file will be created after starting an ECG/MDT.

9. Device status

Device status captures the running status. Following diagram depicts the device status transitions.

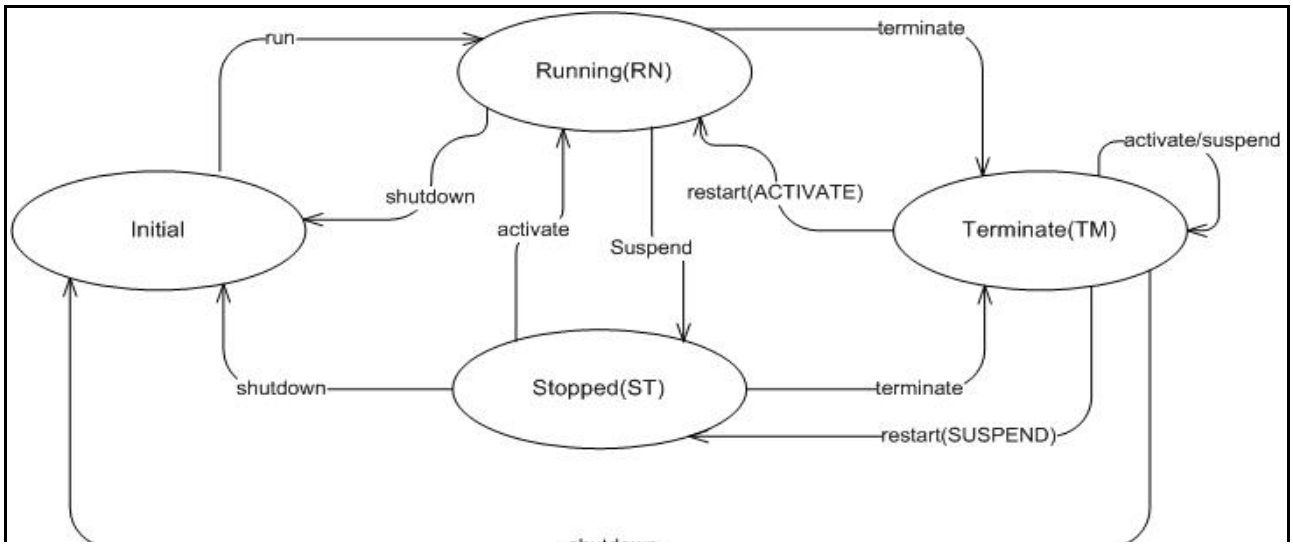

Figure 9.1: device status diagram

(a) Initial status: After starting HTX daemon and before running any ECG/MDT, device will be in the initial status (indicated by __).

(b) RN (Running) : After starting test for specified ECG/MDT, status of all the ACTIVE devices present in it is displayed as RN.

(c) ST (Stopped): After starting test for specified ECG/MDT, status of all the SUSPENDED devices present in it is displayed as ST.

(d) TM (Terminated): After starting test for specified ECG/MDT, if any device present in it is terminated using 'terminate' option of 'htxmdline', its status changes to TM.

(e) ER (Error): If any running device encounters any error, then its status changes to ER.

(f) HG (Hung): If there is no update from any running devices for 2 hrs., then its status

changes to HG.

(g) DD (Dead): If any running device terminates by itself, then its status changes to DD.

(h) CP (completed): If any running device completes specified number of cycles (specified as MAX_CYCLE in running ECG/MDT), then its status changes to CP.

(i) DT (Detached): If any running device terminates due to Dynamic Reconfiguration(DR), then its status changes to DT.

Note: After shutting down the test for specified ECG/MDT using 'shutdown' option of 'htxcmdline', status of all the devices present in it will be changed to initial status.

10. States of ECG/MDT

Following are different states of an ECG/MDT.

- UNLOADED: After starting HTX daemon, initial state of all the ECG/MDT will be UNLOADED.
- ACTIVE: State of specified ECG/MDT changes to ACTIVE once test is started for it.
- INACTIVE: Its an intermediate state of ECG/MDT when it is not running but resources are allocated with HTX daemon.
- PARTIALLY RUNNING: If any running device encounters error and continue to run, the state of ECG/MDT changes to PARTIALLY RUNNING.

11. Error log format

Device id:

Timestamp:

err=

sev=

Exerciser Name:

Serial No:

Part No:

Location:

FRU Number:

Device:

Error Text:

---

12. Trouble shooting tips

- If any of the 'htxcmdline' command returns connection refused error, check the status of HTX daemon on SUT using following command
  'netstat -na | grep 3492'

- If the daemon is running properly, check the network connectivity between the client and SUT.

13. Reporting defects

- In case of any issues found, the defect can be opened against corresponding component in CMVC/CQ. Following are the components under HTXD.
  htxd : HTX daemon component

- Following detail should be provided from the SUT while reporting a defect.
  - machine access details and steps to access the machine
  - output of 'ver' command
  - contents of /tmp/htxerr, /tmp/htxd.start.stop.time
  - contents of the exerciser specific log files
  - Steps to re-create the issue.

14. Limitations

15. List of Abbreviations

BML	Bare Metal Linux
CMVC	Configuration Management and Version Control
COE	Continue On Error
CQ	Clear Quest
DR	Dynamic Reconfiguration
MDT	Machine Device Table
ECG	Exerciser Control Group
HMC	Hardware Management Console
HTX	Hardware Test eXecutive
IP	Internet Protocol
LPAR	Logical PARtition
MDT	Master Device Table
NIM	Network Installation Manager
SOE	Stop On Error
STX	System Test eXecutive
SUT	System Under Test
TCP	Transmission Control Protocol
UI	User Interface
URL	Uniform Resource Locator
VPD	Vital Product Data

Table 1: Abbreviation List

16. Contacts

- Preeti Dhir/India/IBM@IBMIN, preedhir@in.ibm.com
- Avinash Kuttan/India/IBM@IBMIN, avkuttan@in.ibm.com

17. Document Change History

Version	Date	Name	Change Description
1.0	05/Oct/2015	Avinash Kuttan	Create

Table 2: Change History

END OF DOCUMENT