MediaCmd Command Line Interface

(c) copyright 1998~2011 Drastic Technologies Ltd.

All Rights Reserved.

www.drastictech.com

Table of Contents

Overview

The MediaCmd command line utility allows commands, or groups of commands, to be sent quickly and efficiently to a local or network remote DDR. The basic usage is as follows:

mediacmd - Send MediaCMDs to a Drastic DDR.
Usage: mediacmd [-m <mediacmd in quotes>] [-f <file or mediacmds>] [-a <address
>] [-p <port>] [-c <channel>]

    -m <mediacmd in quotes> A single MediaCMD string.
    -f <file or mediacmds>   A series of MediaCMD strings.
    -a <address>   Network address of the target DDR (127.0.0.1).
    -p <port>   Network port of the target DDR (1234).
    -c <channel>   Channel in the DDR (0).
    -r <return>   What element to return from the app.
    -q   Quiet output.

By default, the mediacmd program will connect to the local DDR, the one the mediacmd is running on, at ip address 127.0.0.1 (localhost), port 1234 (the default) and the first channel (0). To target a different DDR on the network, set the -a/-p/-c arguments:

mediacmd -a 192.168.1.10 -p 1234 -c 0

Any parameters not set, automatically revert to the defaults. The -r parameter specifies which return should be returned from the mediacmd and, if the -q lag is not set, then which will be printed out. The -q flags specifies that mediacmd should not print out anything other then errors. The -m parameter is how commands are sent to the DDR. The -f allows a list of commands to be sent instead of just one. These will described below.

Sending commands

The format of the commands are the same on the command line, or on a line within a file containing a sequence of commands. Individual commands are supplied has a parameter to the -m command.

-m "play&start=1:00:00&end=2:00:00&loop"

When sending on the command line, the whole command should be contained with double quotes (")
so they are interpreted as a single command. For a series of commands, a file may be passed in with the -f argument.

-f C:\testcommands.mcmd

The command file's individual lines are of the same format as the -m commands. A file might look something like this (NOTE: Lines beginning with a # are comments):

testcommands.mcmd:

_____

Make sure we are in clip mode (not conform)

setvalue&cmdalt=clipmode&position=1

Load a file into the clip bin from the disk

pause&clipid=clip1&filename=V:\TestFile.mov

Load the other file we are going to need

pause&clipid=clip2&filename=V:\AnotherFile.mov

Set our time code source to LTC

setvalue&cmdalt=tcsource&position=2

Load the first clip (clip1) at one hour to the output

pause&clipid=clip1&position=01:00:00;00

Play the file

play

Wait for 10 seconds (assume file is 12 seconds long)

sleep 10000

Setup the next file to play when the current one ends (seamlessly)

play&clipid=clip2&start=00:00:01:00&end=00:01:01:00&deferred

MediaCmd: main commands

The first parameter of the -m must be be one of the following commands:

- Stop – Full stop/all stop/e to e

- Pause – Pause on current frame, seek or load

- Play – Play, either at normal speed or shuttle speeds. May also load and seek.

- Record – Record to the disk or tape

- RecStop – Prepare for a record

- Eject – Eject the current tape or media

- Transfer – Transfer to/from an internal channel and an external channel

- Insert – Insert media into the clip bin or time code space

- Blank – Remove media from the clip bin or time code space

- Delete – Delete media from the storage and blank it

- Trim – Alter a clip or time code space edit

- ChanSelect – Change the currently selected channels

- GetState – Get the current channel state

- SetState – Set the current channel state

- GetValue – Get a setup value

- ValueSupported – See if a setup value is supported

- SetValue – Change a setup value

- Error – Report an error

- Terminate – Kill the current operation

- Abort – Abort the current operation

MediaCmd: command modifiers

With these commands a number of modifiers are available. Each modifier must be separated by an ampersand (&) on the command line.

- channel=%d – specify the channel this command should be sent to
- position=%s – set the position element for a command
    - 1:00:00:00 – go to one hour
    - +5 – go forward from the current location 5 frames
    - -5:00 – go backward from the current location 5 seconds
    - 1800 – go to one minutes (specified as 1800 frames, not drop frame time code0
- start=%s – set the start element (see position for format)
- end=%s – set the end element (see position for format)
- speed=%d – set the speed element for a command
    - 65520 – normal forward play (100%)
    - -65520 – reverse play
    - 32760 – half play speed (50%)
    - -655200 – 10 times reverse speed
    - 0 - pause (no play)
- timems – millisecond time for the command
- cmdalt – set the cmdalt element of the mediacmd
- videochannels – which video channels to use (bitwise)
- audiochannels – which audio channels to use (bitwise)
- infochannels – which information channels to use (bitwise)
- clipid – 8 character clip identifier
- filename – filename for the command
- string – sting to be used in the command
- There are a number of flags that may be used, just like the elements above
    - Deferred – wait for previous command to complete before new this command
    - OverrideDeferred – override a previous deferred command
    - Loop – Loop whole clip, or a start/end subset
    - AllIDs – Command should affect all available clip ids
    - NoClipFiles – Ignore clip space clips
    - NoTCSpaces – Ignore conform space files

- IsShuttle – The command should be interpreted as a shuttle, even for normal play
- UsingCurrent – Use the current start/end/position
- UseFrameCount – Use the absolute frame count, not the time code values
- Fields – Use fields, if not a progressive signal formats
- Ripple – When removing a file, ripple the following files back
- Trigger – Wait for a trigger
- Preview – Doing a preview, not a full play
- Test – Don't do the command, just see if it exists
- NoReturn – Don't return any information from the command

MediaCmd: examples

Basic Commands

play

– Normal play at normal speed

play&speed=32760
– Play at 50% forward speed

play&speed=-65520
– Play at 100% reverse play speed

play&start=1:00&end=4:00&loop
– Play from one second to four seconds in a loop

stop
– Stop (e to e pass through) the channel

pause
– Pause the channel

pause&position=1:00:00
– Seek to one minute

pause&clipid=::test

- Load an existing clip in the system

pause&clipid=NewClip&filename=\\server\share\test.mxf

- Lad a new file as a new clip id (local)

record&clipid=newrec&end=5:00
– Record a new file name 'newrec' which will be five seconds long

Dealing with Picon Images

Server Mode, clip: Kroatien, file: KroatienMovie.mov

SetValue&cmdalt=1000000&clipid=Kroatien&position=200
– Make a new picon from frame 200 of the clip Kroatien
– result name: KroatienMovie.picon.jpg

GetValue&cmdalt=1000000&clipid=Kroatien&position=ffffffff
– Return the actual file name of the picon file (char elem 9)
– result name: Kroatien.picon.jpg

GetValue&cmdalt=1000000&clipid=Kroatien&position=4294967295
– Return the size of the picon file in the Position elements
– result: dwPosition = 7900

GetValue&cmdalt=1000000&clipid=Kroatien&position=1
– Return the actual bytes of data for the JPEG picon frame in arbID
– result: Not available in HTTP, have to use C/C++

SetValue&cmdalt=1000000&filename=V:\Media\KroatienMovie.mov&position=100
– Make a new picon from frame 100 without associating it with the clip
– result name: KroatienMovie.picon.jpg
(not normally used, conflicts with vtr tape mode picon)

VTR Tape Mode, Time line 00:00:01:00 Kroatien.mov?

SetValue&cmdalt=1000000&filename=V:\Media\Kroatien.mov&position=1000
– Make a new picon from the frame at position 1000, default for file
– result name: Kroatien.picon.jpg

GetValue&cmdalt=1000000&filename=V:\Media\Kroatien.mov&position=ffffffff
– Return the actual file name of the picon file (char elem 9)
– result name: Kroatien.picon.jpg

GetValue&cmdalt=1000000&filename=V:\Media\Kroatien.mov&position=4294967295
– Return the size of the picon file in the Position elements
– result: dwPosition = 7900

GetValue&cmdalt=1000000&filename=V:\Media\Kroatien.mov&position=1
– Return the actual bytes of data for the JPEG picon frame in arbID
– result: Not available in HTTP, have to use C/C++