

Packet Ship *Streamline* Media Server

Application Note **AN-SL-901** v.1.2.1

Analysing MPEG2 Transport Streams
with ps-analyse-mpeg2ts

Introduction

This application note describes how to use the `ps-analyse-mpeg2ts` tool to find the streaming rate profile and PIDs in use in an MPEG-2 Transport Stream asset. You can use the rate information to configure assets in the Packet Ship Streamline video server, or just to check the encoding.

Installing the analyser

The MPEG-2 Transport Stream analyser is supplied in a Debian or Red Hat package `ps-analyse-mpeg2ts`. This needs to be installed on the content production machine. It can be installed on the same machine as the video server itself, but beware of overloading the disk when live streams are running.

Debian:

```
# dpkg -i ps-analyse-mpeg2ts_1.2.1-1_i386.deb
```

CentOS / Red Hat:

```
# rpm -i ps-analyse-mpeg2ts_1.2.1-1.i386.rpm
```

Using the analyser

The analyser is installed in `/usr/bin/ps-analyse-mpeg2ts`, and in its simplest usage simply takes a command - “rate” or “pids” - and the filename of the TS file to be analysed.

Finding the rate profile

```
$ ps-analyse-mpeg2ts rate foo.ts
#Packet Ship MPEG2 Transport Stream analyser version 1.2.1
#Analysing file foo.ts for rate
Duration: 19.587      seconds
Average rate:      4442054      bits/sec

#Rates measured between every PCR:
Min rate: 3203534    at      0.070
Max rate: 5203703    at      1.123
Spread:   2000168
```

Finding the PID values

```
$ ps-analyse-mpeg2ts pids test.ts
#Packet Ship MPEG2 Transport Stream analyser version 1.2.1
#Analysing file test.ts for pids
Type      SID      PID (hex)
=====
PAT                0
PMT                66 (0x42)
Audio         0      68 (0x44)
Video         0      69 (0x45)
```

General options

The analyser takes the following general options *before* the command:

<i>Option</i>	<i>Effect</i>
-q or --quiet	Runs quieter

e.g.

```
$ ps-analyse-mpeg2ts -q rate foo.ts
Duration: 19.587      seconds
Average rate:      4442054      bits/sec
Min rate: 3203534   at          0.070
Max rate: 5203703   at          1.123
Spread:   2000168
```

Note the comment lines and blank lines have been removed – this makes it easier to automatically parse the output.

Rate options

The analyser takes the following rate-specific options *after* the “rate” command:

<i>Option</i>	<i>Effect</i>
-a or --average	Show only the average rate in bits/sec
-g or --graph	Output rate figures for producing a graph
-t or --time <seconds>	Analyse rate over the given period (by default it analyses between PCR values)

e.g.

```
$ ps-analyse-mpeg2ts -q rate -a foo.ts
4442054
```

This number can be used to directly set the rate value for an asset configuration in the video server.

```
$ ps-analyse-mpeg2ts -q rate -g -t 1 foo.ts
0.000    5035941
1.000    4513446
2.000    4646910
3.000    5038854
4.000    4596636
5.000    4219907
6.000    4268596
7.000    4568291
8.000    4928155
9.000    4652062
10.000   4227977
11.000   4546641
12.000   4184630
13.000   4455050
14.000   4383334
15.000   4177643
16.000   4481267
17.000   4516321
18.000   4139449
```

This gives the rate in one-second samples, and can be used to directly generate a rate profile graph (see below).

PID options

The analyser takes the following PID-specific options *after* the “pids” command:

<i>Option</i>	<i>Effect</i>
-v or --video	Extract only video PIDs, decimal number output only
-a or --audio	Extract only audio PIDs, decimal number output only
-t or --time <seconds>	Analyse PID values over the given initial period (default 1 second)

e.g.

```
# ps-analyse-mpeg2ts -q pid -v foo.ts
69
```

Plotting rate profile graphs

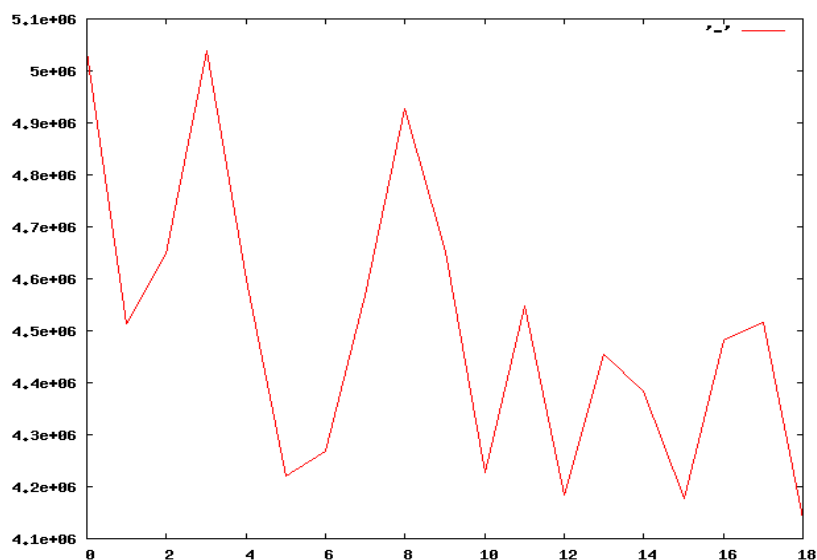
The output of “rate -g” can be used as the input to create a graph of the rate over time. You can save the output to a file and import it into OpenOffice Calc or Excel, or more simply, in Linux, pipe it straight into 'gnuplot':

```
$ (echo "plot '-' with lines"; ps-analyse-mpeg2ts -q rate -g -t 1 foo.ts) |
gnuplot -persist
```

This shows the graph as a window; to create it as a PNG, use:

```
$ (echo "set terminal png"; echo "plot '-' with lines"; ps-analyse-mpeg2ts -q
rate -g -t 1 foo.ts) | gnuplot > rate.png
```

This produces the following graph:



There are of course many `gnuplot` options that can be used to improve on this! See www.gnuplot.info for details.