



# Packet Ship *Streamline* Media Server

## Application Note

AN-SL-603

v.3.1.0

Indexing MPEG2 Transport Streams  
for 'trick mode' fast-forward and rewind

## Introduction

This application note describes how to create Packet Ship Index (PSI, PSI2) files from MPEG-2 Transport Streams, to enable visual 'trick mode' fast-forward and rewind, and other frame-accurate optimisations, with the Packet Ship Streamline media server. The indexer supports both MPEG-2 Video and H.264 video, but both must be wrapped in an MPEG-2 Transport Stream.

---

## Installing the indexer

The MPEG-2 Transport Stream indexer is supplied in a Debian or Red Hat package `ps-index-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-index-mpeg2ts_3.1.0-1_i386.deb
```

CentOS / Red Hat:

```
# rpm -i ps-index-mpeg2ts_3.1.0-1.i386.rpm
```

(or similar for 64-bit packages)

---

## Using the indexer

The indexer is installed in `/usr/bin/ps-index-mpeg2ts`, and in its simplest usage simply takes the filename of the TS file to be indexed:

```
$ ps-index-mpeg2ts foo.ts
Indexing foo.ts into foo.ts.psi2
Output index format: PSI2
PMT on PID: 66
Detected H.264 video stream on PID: 121
Creating index of 358 blocks
Duration: 416.856 seconds
Done
```

By default, it simply creates the index file with “`.psi2`” appended in the same directory. This is also where the `ps-streamd` daemon expects to look for index files, so it should not usually be necessary to change this. If you do need to, you can supply the output file as a second argument.

The PSI2 format is for use with the Packet Ship Streamline video server version 3.1 (Antigua) onwards. The indexer can also create the old PSI format for previous versions with the `-f` option (see below). The PSI2 format is considerably smaller than the PSI for the same file because it references the I-frame data rather than including it.

## Content type

Note that unlike the previous version of the indexer, it is no longer necessary to include a `-h` flag to specify H.264 – this is now automatically recognised from the input.

## Options

The indexer takes the following options:

<i>Option</i>	<i>Effect</i>
-? or --help or no parameters	Shows help
-q or --quiet	Runs quieter
-v or --verbose	Runs more verbosely
-f or --format <psi psi2>	Sets the output index format (default PSI2)
-t or --tables	Include PAT/PMT/CAT data (default)
-n or --no-tables	Do not include PAT/PMT/CAT data
-d or --dump	Outputs the selected frames as a pure transport stream (without the index) to standard output, for capture and checking.
-s or --seek-only	Generate only offsets for seek, no trick mode frames
-S or --seek-pcr-only	As above, but using only PCR value s(not frames)
-c or --ca-pid <n>	Capture CA data from given decimal PID
-If or --index-flags <n>	PSI2 only: Set index flags (default 0, no flags currently defined)
-Ib or --index-blocksize <n>	PSI2 only: Set index block size (default 256)
-In or --index-links <n>	PSI only: sets the number of backward and forward index links (default 16)
-Ip or --index-precision <n>	PSI only: sets the precision of the offset index in bytes (default 65536)

There are also -e and -E options for encryption which are documented separately as part of the Keyline DRM product.

## Encrypted streams

The indexer is designed to work seamlessly with streams encrypted with Packet Ship's Keyline DRM. It is also able to index encrypted streams from other CA vendors, but the issues are complex – please contact Packet Ship support for more information if you require this.

The '-c' option allows the capture of an additional PID to be passed through into the output, for example a Conditional Access (CA) PID carrying key data.

## Offset-only indexes

The “-s” or “--seek-only” option produces an offset-only index file. This contains the same index and timing information for alignment of seeks and determination of average rates as a full file, but doesn't contain the actual I-frame data for trick mode. This is useful if you want to provide accurate seeking etc. but don't need actual trick mode support.

## Performance

Indexing is a fairly processor-intensive task, and you should expect a typical 3GHz processor with a single standard disk to take around 5 minutes for a two-hour video – around 25 times faster than real time.