

What's new in GigA+ 0.2 (compared to builds <= 0.1-7.x)



Overview

GigA+ 0.2 contains a number of improvements in functionality. Apart from bug fixes (contributing to overall stability), the changes are about ease of configuration, monitoring and third-party product compatibility. This document presents the changes in some detail, for the technical angle one should consult the man pages supplied with the installation.

DVR: time-shifted playback

Streams/channels can be viewed from a point in the past as if they were live. For instance, on a channel that stores 1 day of live video, one could watch the stream 4 hours before the actual time by simply specifying a negative shift in the playlist URL.

Example:

```
$ vlc http://acme.tv:4046/hls-m3u/tvmonde24/playlist.m3u8?utime=-14400
```

DVR: third-party request support

New request formats from third-party applications have been supported, both embedding DVR-related parameters in the playlist filename (instead of supplying those parameters via an HTTP parameter list).

Playlist: playlist_dvr_range-{unix-time}-{dur_sec}.m3u8

Example:

```
$ vlc
http://acme.tv:4046/hls-m3u/tvmonde24/playlist_dvr_range-1499069128-3600.m3u8
```

Playlist: index-{unix-time}-{dur_sec|now}.m3u8

Example:

```
$ vlc http://acme.tv:4046/hls-m3u/tvmonde24/index-1499069128-3600.m3u8
```

All-in-one control script: gigaplus

Control over launching, checking status and stopping components used to be split between two scripts: `gxa-requests.sh` and `gxa-channels.sh`. Getting a host key for the license was also a separate call (to one of the modules). Now there's `gigaplus` - one script for all.

```
$ ./gigaplus
Usage: gigaplus {command} [channels|requests|all] [{channel}]
Commands:
  key                = generate a license key;
  setup [cores]      = set up common environment/core policy;
  create             = create new channel;
  enable|disable     = enable or disable (suspend) channel;
  start              = start component(s)/channel(s);
  stop               = stop component(s)/channel(s);
  status             = show status of components/channels;
Options:
  --nopin            = no CPU affinity to be set;
  --no-upkeep        = disable upkeep (crash/restart & health monitor);
  --alert            = generate alert if channel is offline and/or oversized.
  --debug mask       = list app(s) in debug mode (lb:dwg:pm:ws:ng).
  -t                 = trace mode: see commands & trace info;

GigA+ control script.
$
```

Getting a license key:

```
$ gigapplus key  
System key: [4083b942140de80b8f2bbac0ee7be479e59a11c7dc8f429d] (0x410a)
```

For more examples of the script's usage, please refer to [GigA+ Setup Manual](#):

Online setup manual (PDF)

There is now a user-friendly [GigA+ Setup Manual](#) document that describes the (very) basic steps of setting up GigA+ for the initial testing. It steps one from the point of obtaining a test license (`gigapplus key`) to creating the first HLS channel (`gigapplus create`). The [website](#) links to the document (see the **Jump To** section).

Setup & configuration UI (n-curses)

Both environment and channel setup can now be performed via ncurses-based wizards, walking one through a series of dialogs. Knowing configs is still very much an obligation, as is reading man pages... (sigh). The changes affect `gigapplus.env` and *channel specs*, but **not** `gigapplus.conf` (not yet).

Example: set up the environment (`gigapplus.env`)

```
$ sudo gigapplus setup
```

Crash management: upkeep monitor

A lot of work has gone into making sure that abnormal `exit(3)`'s (*no core*) and crashes (*core dumped*) of binary modules are accounted for. An instance of the `upkeep` script is now attached to every instance of `gxws`, `gxng`, `gxpm`, etc. If a module `exit(3)`'s or crashes, then the binary, relevant logs and the core (if any) are copied to a special subdirectory created under `/opt/gxa/core`. Then a text notification with the stack backtrace (if there was a crash) and the bottom 50 lines of the log is emailed to the system administrator.

Free utilities:

A `utils` subdirectory (under your distro's downloads from the website) with free utilities is to be provided. For now, it would have only `ncl` (network client/probe) tool normally supplied with [GigaTools](#).