

---

## uh\_01\_a processes connected to aleph sluggishness?

- **Article Type:** General
  - **Product:** Aleph
  - **Product Version:** 18.01
- 

### Description:

Prod's opac was not running Sat AM (Apr 26) when our libraries opened.

We saw that the PC server, while running, was extremely sluggish, and that several processes (various services) had been running in excess of 500 minutes each. We tried to restart the WWW and PC servers. That didn't help so we ran aleph\_shutdown then aleph\_startup. aleph\_startup was running extremely slowly, so we eventually physically rebooted the server hardware. (The hardware hadn't been rebooted in over 12 months.)

Aleph started up quickly following the reboot and has been running smoothly since. Just prior to the reboot, server monitors showed that the load on the application server was as high as we've ever seen it, while there was virtually no load on the database server.

Investigation this week showed that 6 processes called uh\_01\_a were running early Sat when Aleph was so sluggish. From the services documentation, it appears that uh\_01 is basically a house-keeping routine, but we don't really understand what it does and why it would have been running redundant simultaneous processes.

Does uh\_01 have anything to do with the ARC ETLs we run several nights/week? Or is it just coincidental that we saw uh\_01\_a start shortly after the Fri night ETL? Does uh\_01 routinely run after particular other services and we've just never noticed it? Or is it something we should consciously be running on a regular basis?

### Resolution:

As noted in KB 3993: "The ./util/uh\_01\_a.cbl program generates the 'setenv base\_demo\_lib'." KB 16384-3714 also touches on its function.

The uh\_01\_a is called by only one \$aleph\_proc proc: def\_lib\_env:

```
unsetenv base_demo_lib  
cobrun uh_01_a
```

def\_lib\_env, in turn, is executed only by the \$data\_root/prof\_library proc:

```
source $aleph_proc/def_lib_env
```

prof\_library is, of course, referenced by a large number of routines.

It seems that uh\_01\_a is executed as part of prof\_library to get the base\_demo\_lib for the library. It is a very simple function which should take only a second or two. I don't know why it would be hanging like this.

Since this is the first time that it has happened -- at \*any\* site, that we know of --, I suggest that we monitor the situation and wait and see if it happens again, and what similarities there are in the circumstances (ETL, no-ETL, etc.).

- 
- **Article last edited:** 10/8/2013