

# Changes in the L2 Infrastructure

Pedro A. Movilla Fernández

May 13<sup>th</sup>, 2008

# Currently



- L2 control nodes: `b0felix` (default) `192.168.0.1`  
`b0l2cnt100` `192.168.0.20`  
`b0l2cnt101` `192.168.0.21`
  - L2 decision nodes: `l2quad01` (default) `192.168.0.10`  
`l2quad02` `192.168.0.11`
  - L2 build node: `killerinstinct` `192.168.0.3`
- private network, see  
`/etc/hosts`  
`/etc/dhcpd.conf`
- L2 build scripts: `b0home:~cdfprod/TriggerDB/current/`  
`...mounted on all L2 control nodes`
  - L2 build libraries: `b0felix:/cdf/code-L2/build-l2cal/`  
`...local hard disk`

# L2 Nodes Update



- L2 control nodes: ~~b0felix~~ 192.168.0.1  
b0l2cntl00 (default) 192.168.0.20  
b0l2cntl01 192.168.0.21
- L2 decision nodes: l2quad01 (default) 192.168.0.10  
l2quad02 192.168.0.11
- L2 build node: killerinstinct 192.168.0.3

1. b0l2cntl00 replaces b0felix  
b0l2cntl01 serves as spare  
b0felix will stay connected for a while

planned for this week

2. l2quad01/02 will be replaced by  
new AMD Dual QuadCores (similar two old ones)

2 new PC's ordered,  
replacement in a few weeks

3. l2quad01 is going to replace killerinstinct

# ProcMon Update



- ProcMon "l2con" check in:  
`$FER_DIR/src/procAction`  
`$FER_DIR/src/procActionHandler`
- Previous version:  
Hardwired check of control process on `b0felix`, decision node was ignored
- New version (**already running**):
  - Check of **both control and decision processes** on all 5 nodes
  - Check of **intercommunication setup** using new process lock files  
`/tmp/.cntl_lock` (L2 control)  
`/tmp/.dcsn_lock` (L2 decision)
  - New script scans private network via `ssh-agent` as user "ace"
  - Have the flexibility to switch between the default and hot spares w/o changing hardwired ProcMon settings

# ProcMon Update (2)



## ProcMon L2 checks:

1. Exactly one L2 control process (`TEDCpp`) is overall running
2. At least one L2 decision process (`l2node.exe`) is overall running, max. one per node
3. At least one L2 decision process is setup to connect to one responding L2 control node
4. Connected nodes must provide valid responses to ProcMon script (includes validation of ip addresses provided by the process lock files)

As long as the above requirements are met:

- More than one decision node processes listening to the same L2 control process are allowed.
- One or more control and decision nodes may be disconnected.
- L2 decision processes ignored by L2 control are allowed, passive L2 control processes are always prohibited.

# ProcMon Update (3)



## 1) Fake alarms may be triggered

- by running `xemacs TEDCpp.cc` or alike on one of the three L2 decision nodes - will eliminate this stupid feature a.s.a.p.
- when switching to the spare node during ProcMon check
- due to time out of `ssh ace@192.168.0.10` if node is busy (didn't happen as yet as far as I know)
- due to delay in updating the L2 control lock file - currently happens via RTserver during CONFIG state

**Plan:** Will force update during PARTITION state.

## 2) Real problem not seen by ProcMon (and aces?):

- Crash/respawn of L2 decision process during data taking
  - L2 decision may be idle and waits for L2 control start directive
  - Requires to start a new run during data taking
- Triggers TO alarms etc. but reason often unclear

**Plan:** Will extend diagnosis in ProcMon so that aces can react properly.

Run a L2-specific slim ProcMon version (on b0gateway?) detached from RTserver for more frequent L2 checks (every minute instead every 10 min)

# How To Start/Stop L2 Control



```
1. ssh l2@b012cnt100
2. sudo xemacs /etc/inittab
   for starting(stopping) uncomment(comment) the last line:
   p2:2345:respawn:/cdf/code-L2/tld/current/l2pulsar_ted/src/runTEDCppTest.sh
3. sudo /sbin/telinit Q
```

- Running two control processes on the same or another node will confuse RTserver and trigger a ProcMon alarm within 10 minutes
- IP address(es) of L2 decision node(s) to communicate with is passed to L2 control via RunControl via smartsockets (use cardEditor to change DB entry)
- /tmp/.cntl\_lock correctly updated during CONFIG (not before – will change to PARTITION). Switching to another L2 decision node before CONFIG will trigger a false ProcMon alarm.
- Log files are in `/cdf/code-L2/tld/current/log/*`

# How To Start/Stop L2 Decision



```
1. ssh b012cnt100
2. ssh root@192.168.0.10, enter password
3. xemacs /root/run/official/log-run-l2node.tcsh
   and edit the variable ted_ip on top of the script
       set ted_ip = 192.168.0.20    for b012cnt100
       set ted_ip = 192.168.0.21    for b012cnt101
       set ted_ip = 192.168.0.1     for b0felix
4. xemacs /etc/inittab
   for starting(stopping) uncomment(comment) the last line:
   p2:3:respawn:/root/run/official/log-run-l2node.tcsh
5. /sbin/telinit Q
```


- IP address of active control node needs to be passed **manually** as above since the decision node is not connected to the RTserver
- `/tmp/.dcsn_lock` will usually correctly reflect the communication setup.
- Log files are in `/logs/l2node/*`
- When changing the active L2 control node:

```
1. Edit the variable ted_ip manually like above
2. Restart the process via killall -INT l2node.exe
```





# L2 Build Update

- L2 build scripts: `b0home:~cdfprod/TriggerDB/current/`  
...mounted on all L2 control nodes
- L2 build libraries: `b0felix:/cdf/code-L2/build-l2cal/`  
(currently) ...local hard disk
- **Plan:** Move target location to `b0home:~cdfprod/TriggerDB/triggerfiles/l2share/`  

- Updated build version already exists and old libs already copied manually to there.
- Visible from all L2 control nodes
- We want to switch to the new L2 build in the moment we start using the new L2 control nodes

# L2 Build Update (2)



- `b0home` : `~cdfprod/TriggerDB/current_UPDATE/`  
    `BuildL2.pl` (master script invoked by TriggerGui)  
    `BuildL2Filar.pl` `RemoteBuildL2Filar.sh`
- **Actions in the old version:**
  1. login to `b0felix` as user "build"
  2. from there submit compile tarball to `killerinstinct` as user "build"
  3. copy build back from `killerinstinct`
  4. copy libs to `/cdf/code-L2/build-l2cal`
  5. duplicate libs using tagset number
- **Actions in the new version:**
  1. login to one available L2 control node as user "build"
    - checked in the order `b0l2cnt100`, `b0l2cnt101`, `b0felix`
    - not necessarily the active node
  2. & 3. like in old version
  4. scp libs and sources (tar.bz2) to  
    `~cdfprod/TriggerDB/triggerfiles/l2share/` as user "cdfprod"
  5. clean up temporary files on control node
  6. create soft link instead of duplicate

# L2 Build Update (3)



- At the beginning of a run, the L2 decision process connects to the rsync server running on the L2 control node to fetch build libraries.
- L2-CAL path in control node's `/etc/rsyncd.conf`
  - old version (b0felix):  
`/cdf/code-L2/build-l2cal`
  - new version (b0felix, b0l2cntl00/01):  
`/cdf/onln/code/cdfprod/TriggerDB/triggerfiles/l2share`
- We usually want the rsync server be the ACTIVE L2 control node.

# Tests



- Updated ProcMon ran reliably for weeks, except for two or three fake alarms.
- Ran L2 control process on new nodes b0l2cntl00/01 and checked all kinds of L2 control/decision configurations, including parasitic runs (used also for ProcMon test).
- Ran L2 control process on b0l2cntl00/01 during COSMICS runs and in PHYSICS end of store runs w/o problems.
- Tested functionality of updated build scripts.
- Tested L2 decision node's accessibility to build libs at new location using the rsync server on b0l2cntl00/01 and b0felix.

... everything seems to be fine