



**BRTT**

June 2006

# Antelope 4.8

- What's New?
  - Linux upgrade to 2.6.11 (SuSE 9.3)
  - Mac upgrade to Tiger, also works on iMacs
  - miniSEED enhancements – B1001, opaque, Steim I
  - **cdorb2db** / **db2msd** – attempt to deal with CD LIFO problems
  - **orbserver64** / **diskserver** – really large ORBs
  - **rt2orb** – finally released RT130 acquisition program
  - **pfe** – new GUI for navigating/inspecting complex pf objects
  - Many **q3302orb** changes
  - **dbbuild** enhancements, new **orbxfer2** program, some **dbpick** changes
- Near term development

# Enhancements to miniSEED

- **libmsd** now supports Steim 1 compression, opaque blockettes and B1001's
- Currently B1001's can be produced and populated with the time stamp microsecond details in all Antelope programs (e.g. **orb2db**) by setting a parameter in **trdefaults.pf**
- Currently, other B1001 attributes, such as timing quality, are not automatically populated (except with fixed values)
- The current **libmsd** provides the basic infrastructure necessary for the arbitrary process of assigning sometime severely aliased timing quality information, and other waveform status values such as 'calibration in progress', to miniSEED data

# **cdorb2db / db2msd** – attempt to deal with CD LIFO problems

- **cdorb2db** is a special replacement for **orb2db** which will ingest CD-1.0 ORB packets and populate a waveform database
- **cdorb2db** continuously maintains complete waveform day volumes that are initialized with “gap” sample values and places the CD-1.0 packets into their appropriate places within each day volume
- **cdorb2db** assumes that timing in these CD-1.0 packets is perfect – i.e. each sample value is precisely aligned with the nominal sample time
- Waveform formats are hardwired to “s4” (big-endian) or “i4” (little-endian)
- Normal SEED formatting with Steim compression can be accomplished with post-processing using **db2msd**

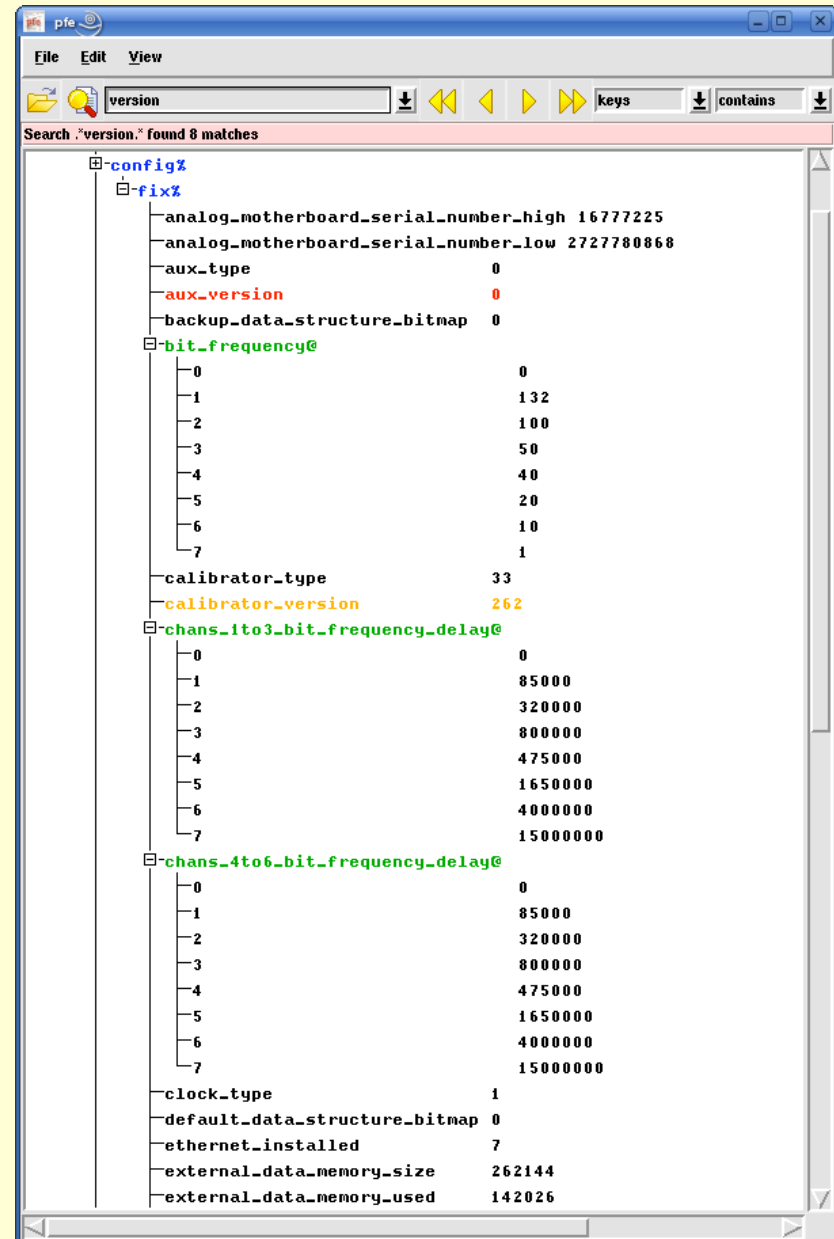
# orbserver64 / diskserver – really large ORBs

- **orbserver64** is a special replacement for **orbserver** which will allow unlimited size ORB buffers
- **orbserver64** is only available for the Solaris/SPARC version of Antelope
- Any Antelope client can connect to **orbserver64** and the interactions with **orbserver64** are the same as with **orbserver**
- **diskserver** provides large ORB capability in a 32-bit environment across all Antelope versions
- **diskserver** uses a specially organized directory containing **forb (5)** format files as the ORB buffer and provides read-only access to clients
- the ORB buffer files used by **diskserver** are written by **orb2disk**, a program that transfers ORB packets from a regular ORB, maintained by **orbserver**, to the set of **forb (5)** files (with size limitations) that are used by **diskserver**

# **rt2orb** – finally released RT130 acquisition program

- **rt2orb** will connect to one or more Reftek RT130 dataloggers using the UDP “Reftek Transfer Protocol” and write out waveform data and RT130 log information and status into output ORBs
- **rt2orb** is similar in its design and setup to **q3302orb**
- However, there are many missing, incomplete and stubbed-out functions in **rt2orb** including status waveforms, repackaging of output waveform data, automatic RT130 configuration, command and control functions and general status monitoring capability.
- The RT130 dataloggers must be configured to use the “c0” compressed waveform data format only when using **rt2orb**.
- There are a host of issues relating to the Reftek protocols that the user needs to be aware of when using **rt2orb**. These issues are discussed in the **BUGS AND CAVEATS** section of the **rt2orb(1)** man page. Please read this carefully if you intend to use **rt2orb**.

pfe – new  
GUI for  
navigating  
/ inspecting  
complex pf  
objects



# many `q330orb` changes

- PB-14 token parsing and reporting
- Support for base96 encoding
- Support for sensor calibration
- Support for on-demand generation of Q330 “user” messages
- Support for automatic generation of Q330 “user” messages
- Quanterra protocol acknowledgment “tuning” to decrease communication bandwidth
- Increased caching across link cycles to prevent data gaps (also involves a firmware modification in the Q330)
- Extensive redesign of log messages
- Additional status variables: **thruput**, **gp1**, **bfr**, **np24**, **ni24**
- New **q330proxy** program – SLIP proxy for Q330s
- New **q330snoop** script – decoding and printing raw Q330 UDP packet contents used in conjunction with **snoop** or **tcpdump**



# other changes

- **dbbuild**, **orbxfer2** – talk to Dan
- **Dbpick**
  - Support for interactive general timing marks
  - New **typein** commands in support of general timing marks
  - New **stamax** **typein** command
  - Hopefully will now work on any 16-bit display

