Antelope 4.6

Antelope 4.6 – What's New and Different?

- Administrative: architectures, operating systems, licensing, etc.
- Infrastructure changes
- New programs and scripts
- New toolbox modules
- Changes to existing modules
- Back-compatibility issues

Administrative

Before We Get Started…

- ***Antelope*** core software
  - Scrupulously developed, tested, maintained and supported by BRTT
  - Included on ***Antelope*** CDs
  - Both closed and open source
- ***Antelope*** contributed software
  - Developed and maintained (maybe) by original authors (could be BRTT, but generally not)
  - BRTT assumes no responsibility for effectiveness or support
  - Built by BRTT (but not tested) and included on ***Antelope*** CDs
  - Conforms to a minimum set of requirements
    - will build with normal ***Antelope*** toolbox
    - documented
    - complete (no third party software)
  - Open source
  - Users have access to source CVS repository through Antelope User Group web site at http://www.indiana.edu/~aug
- Software Junkyard
  - Generally designed to work in an ***Antelope*** environment – HOWEVER:
  - Absolutely no implied support or maintenance of any kind
  - No software requirements whatsoever – anything goes (may be incomplete, ineffective, undocumented, etc.)
  - **NOT** on ***Antelope*** CDs
  - Open source
  - Users will have access through Antelope User Group web site
  - New concept - not yet implemented

Architectures, OSs and Installing

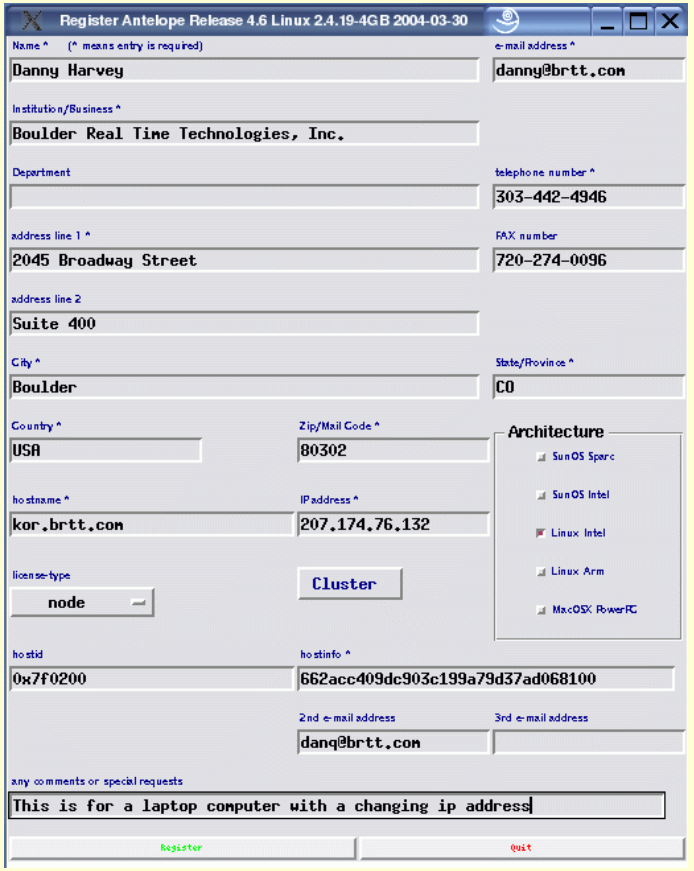
- ***Antelope*** 4.6 was built on the following:
  - SUN / SPARC / Solaris 8 (like 4.5)
  - INTEL / X86 / Linux 2.4.19 (like 4.5) using SuSE 8.1
  - Special stripped-down "volpi" version
  - We dropped the X86 / Solaris version
  - There are no ARM/X-SCALE or Mac versions that are currently being released
- All versions are on the same CD (like 4.5)
- Installation like 4.5 (mostly)

Installation and Administrative: Installing and Registering

- Prior to installation:
  - You will need roughly 800 Mbytes of space
  - You will need your e-mail running properly (in order to send the license request)
  - You will need to be connected to the internet (in order to obtain software patches)
  - You will need a graphical X-windows interface (in order to run the installation scripts)
  - Install as root or some other user?
    - Probably root for multiple user access
  - ***Antelope*** MUST install at `/opt/antelope`
    - Either 1) space at `/opt/antelope`, or 2) a symbolic link at `/opt/antelope` to some other place, or 3) auto/remote mounting at `/opt/antelope` from some other place
    - Probably need root permission for creating `/opt/antelope`
  - Look at `README` on the root directory of the ***Antelope*** distribution CD for more information

Registering and Licensing

- Major changes to registration and licensing
  - previous licenses were "fragile" – new licenses should be much more robust
  - registration form changes to support user requested architectures and node clusters



Installation and Administrative : Licenses

- **How does licensing work?**
  - ***Antelope*** software modules automatically read the **`$ANTELOPE/data/pf/license.pf`** file to determine the license type and to read the license keys.
  - Based upon the license type, the ***Antelope*** software module then looks at the system time, queries the host computer and/or its internet connections and computes another set of license keys.
  - If the computed keys match the keys in the **`license.pf`** file, then the license is OK and the module will run properly, otherwise an error exit will occur.
  - A set of different license keys can be listed in the **`license.pf`** file, in which case the license checking software will check the computed keys against all of the keys in the **`license.pf`** file. If any of the keys match, then the module will run properly.
  - The current state of licensing for a particular ***Antelope*** installation can be checked by running **`check_license [-v[v]]`**
  - New licenses **MUST** be requested by running **`register_antelope`**

Installation and Administrative : Licenses

- **License types**
  - **Subnet licenses**
    - These licenses will enable multiple host computers that have ip-addresses within a single class-C subnet (first three numbers of numerical ip-address).
    - BRTT will not give out subnet licenses for "private" networks (e.g. 192.168.xxx.xxx) or for any subnets that are not name served and do not belong to qualified users. Usually this means that we will not provide subnet licenses for host computers that get their ip-addresses via DHCP or private non-name served addresses behind a firewall.
  - **IP licenses**
    - These licenses will enable a single ip-address. Any host computer that is configured at the enabled address should be able to run the software.
    - Same restrictions as subnet licenses.
  - **Node licenses**
    - These licenses will enable a single host computer regardless of its internet connections.
    - This is accomplished through the use of a "host fingerprint" – a hashed version of a complex hardware-based identification profile that is intended to uniquely identify a particular computer system.
- **Other license restrictions**
  - Independent of the license types, ***Antelope*** licenses can be time-limited, architecture/OS limited, and/or capability limited.

Installation and Administrative : Licenses

- The current state of licensing for a particular ***Antelope*** installation can be checked by running **`check_license -vv`**
- The node fingerprint, **`hostinfo`**, and the list of currently configured ip-addresses is determined.
- Each license key is compared against the node fingerprint and the ip-addresses to determine if any are OK

Infrastructure Changes

Antelope Distribution

- New versions of perl (5.8) and tcl/tk (8.4)
- Both perl and tcl/tk are located in "versioned" sub-directories (`/opt/antelope/perl5.8` and `/opt/antelope/tcltk8.4`)
- This means that both 4.5 and 4.6 versions of ***Antelope*** can be installed on the same filesystem and both versions can be used without having to uninstall 4.5.

Antelope Parameter Files

- New macro substitution-type capabilities in 4.6:
  - **`&file(`*`path`*`)`**, **`&filename(`*`path`*`)`**, **`&datafile(`*`path`*`)`**, **`&datafilename(`*`path`*`)`** - substitute relative file paths or file contents
  - **`&env(`*`name`*`)`** - substitute value of ***`name`*** environment variable
  - **`&name`**, **`&{`*`name`*`}`** - substitute value of defined ***`name`*** parameter

```
% cat example.pf
sta             AAK
chan            BHZ
time            1057768615.00
date            &exec(epoch +%Y-%j &time)
dfile                   &{sta}_&{chan}_&date
% pfecho -i example
date
        2003-190
dfile
        AAK_BHZ_2003-190
```

ORB Changes

- If an `@` character is appended to an ORB hostname specification, then the ORB open will never return with an error, but retry forever:

```
% orb2orb igpprt.ucsd.edu:anza@ localhost
```

- If an `@` character precedes an ORB match or reject string specification, such as `@spec`, then the *spec* part is used as a file name of a file that contains a list of match or reject strings:

```
% cat my_select_strings
AZ_CRY/BBA/HS
CI_SVD_.HZ/.*
% orbstat -d all -m @my_select_strings igpprt.ucsd.edu:anza
AZ_CRY_HHE
AZ_CRY_HHN
AZ_CRY_HHZ
AZ_CRY_HLE
AZ_CRY_HLN
AZ_CRY_HLZ
CI_SVD_BHZ
CI_SVD_HHZ
CI_SVD_LHZ
```

- The **valid_ip_addresses** table in the **orbserver.pf** file may now contain entries of the form:

```
207.174.76.0      255.255.255.0      /db/.*|/log/.*
```

  which allows only **/db/.*** (database row) and **/log/.*** (log message) type packets to be read from hosts on the 207.174.76.0 class-C subnet.

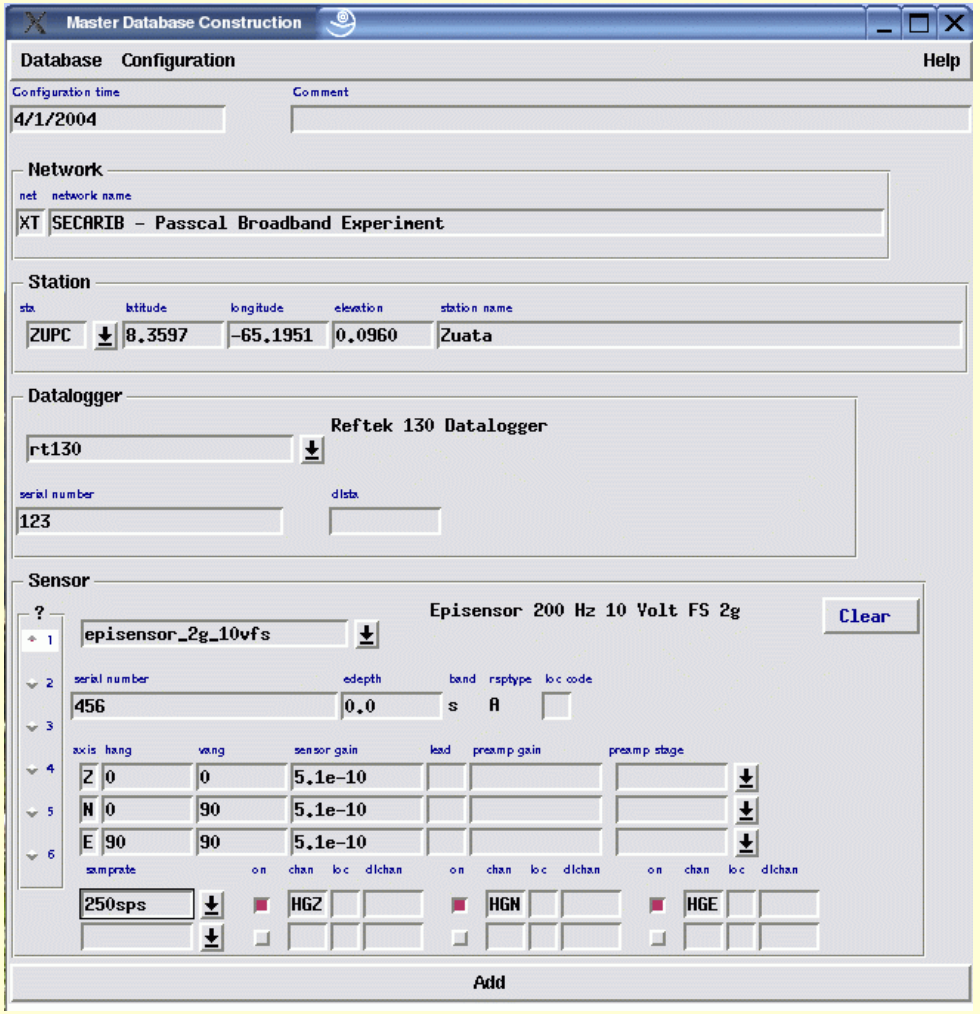
- A new **orbserver.pf** parameter, **reject_message**, can be specified as a string that will be displayed at the client end whenever an ORB connection is rejected due to improper permissions

New Programs and Scripts

dbbuild

A new program (perl script) for building the "metadata" part of a Datascope database (*site*, *sitechan*, *sensor*, *instrument*, *calibration*, *stage* tables plus external instrument response files)

- Can operate in either interactive or batch mode.
- Can run from a master configuration file
- Uses user-configurable single-stage response files plus a set of parameter files that describe "standard" dataloggers, pre-amps and sensors



BRTT

June 2004

dbbuild

- Included are 180 single-stage response files in **$ANTELOPE/data/responses**
- Included are 23 sensor parameter files in **$ANTELOPE/data/instruments/sensors**:

Guralp CMG-3	Guralp CMG-3 NSN	Guralp CMG-3T	Guralp CMG-40T
KMI Episensor	KMI FBA-23	KMI Ranger SS1	Geotech KS54000
Oyo Hall/Sears HS10	Mark Products L22	Mark Products L28	Mark Products L4
Streckeisen STS-1	Streckeisen STS-2	Terratech	Nanometrics Trillium

- Included are 20 datalogger parameter files in **$ANTELOPE/data/instruments/dataloggers**:

Guralp DM24	KMI Etna	KMI K2-Mt. Whitney	KMI Makalu
Quanterra Q330	Quanterra Q730	Quanterra Q4120	
Reftek RT130	Reftek RT72A-07	Reftek RT72A-08	Nanometrics Trident

- Files in the **responses**, **sensors** and **dataloggers** sub-directories are well documented ASCII files that can be changed and created by our users to produce new instrument components

qtunbale

A perl script for extracting mini-SEED data from a Quanterra PB/14 Packet Baler:

- Uses the native http-web interface for extracting mini-SEED files in user specified time ranges from the Baler
- Will repackage mini-SEED files on-the-fly into user specified time volumes and user specified file names
- Will rename the SEED network-station-channel-location codes on-the-fly based on user specified translations (e.g. change serial number type station codes into normal alphabetic codes)
- Will automatically pipe the resulting mini-SEED files into a Datascope database extractor or into SEED packets for real-time ORB transfers
- **WILL NOT** control the Baler in regards to 1) power cycling, 2) ip-address assignments, 3) Baler filesystem maintenance

orbwfrms

A program for computing running RMS averages of waveform data through a set of user specified filters:

- Written in C to provide a complete source code example of robust, high-performance real-time continuous waveform processing
- Source code, parameter file template, man page and Makefile can be found in `/opt/antelope/4.6/example/c/rtproc`
- As part of this project, we have introduced a number of new tools in the ***Antelope*** toolbox

dlcmd

A program for sending commands to all data acquisition modules and dataloggers:

- Eventual replacement for **`k2cmd`**, **`qtcmd`**, etc.
- Can "broadcast" commands to any number of data acquisition modules and dataloggers
- Currently functional with **`qt2orb`**, **`k22orb`**, and **`q3302orb`**
- In the 4.6 release only supports **`pingt`** command (and **`getconfig`** command for Q330 dataloggers only)
- Will be supported in the future by new ***Antelope*** tools that will offer standard interfaces for user-developed data acquisition modules

dbgme

A program for computing ground motion estimates :

- Operates on data contained in a Datascope database and outputs database referenced binary numerical X-Y grids of estimated peak ground acceleration, velocity and instrumental Mercali intensity
- Requires pre-computations of measured peak acceleration and velocity along with earthquake location and magnitude
- Designed as a basic "substrate" for managing user written "delegate" subroutines for making the various corrections due to geological effects, for modeling the ground motion in areas away from the observations and for making the empirically-based motion-to-intensity extrapolations.
- Relies on a new set of computational geometry tools in the ***Antelope*** toolbox for final interpolation onto regular Cartesian grids
- **DOES NOT INCLUDE ANY GRAPHICAL OR WEB INTERFACES FOR DISPLAYING THE GRIDS**

Various New ORB-based Programs

- **orbxchange** – Multi-threaded version of **orb2orb** that will:
  - Maintain simultaneous and independent connections to any number of **orbserver**s running on any hosts
  - Automatic switchover to alternate **orbserver**s whenever the primary connections go down
  - Checks to insure that packets are not duplicated
- **rtorbcmd** – Client/server for sending and executing remote commands
  - Includes a rudimentary authentication capability to prevent spoofing
  - Will cause the remote execution of ANY command on the remote UNIX system
- **orbtail** – ORB browser for displaying error and other logs that are contained in an ORB
- **orbstaseg** – Detection-based real-time waveform segmenter ***for a single station***
  - Segments so that the output segmented waveforms have no time overlaps
  - Output segmented waveforms can be re-packaged both channel-wise and time-wise
- **orb2genc** – De-multiplexes and reformats any waveform packets into GENC format. Also will fill in the *calib*, *calper* and *segtype* packet fields

New Toolbox Modules

New ORB-based Tools

- **`pktchannelpipe(3)`** – MT-safe tool that provides a time-ordered queue for buffering waveform data channels in real time as they appear on an ORB:
  - Unstuffs packets automatically and provides the programmer with channel-based queues instead of packet-based queues
  - Programmer can specify tables of channel select and reject expressions that automatically sift the incoming data
- **`pktchannelcalib(3)`** – MT-safe tool that provides a means for efficient retrieval of site and instrument response information from a database to be used in ORB waveform packet processing:
  - Caches database views to prevent new view generation for each waveform packet
  - Refreshes cache only when 1) a new channel is seen in the data stream or 2) one or more of the database tables have been modified
- **`orbreapthr(3)`** – Threaded version of **`orbreap(3)`**
  - Interfaces with other threads through the use of the new **`pmtfifo(3)`** POSIX MT-safe FIFO tool.
  - Provides "peek" and timeout capabilities for reaping ORB packets.
- **`orbputthr(3)`** – Threaded version of **`orbput(3)`**
  - Interfaces with other threads through the use of the new **`pmtfifo(3)`** POSIX MT-safe FIFO tool.
  - Allows many threads to write packets through a single ORB connection.

Other New Tools

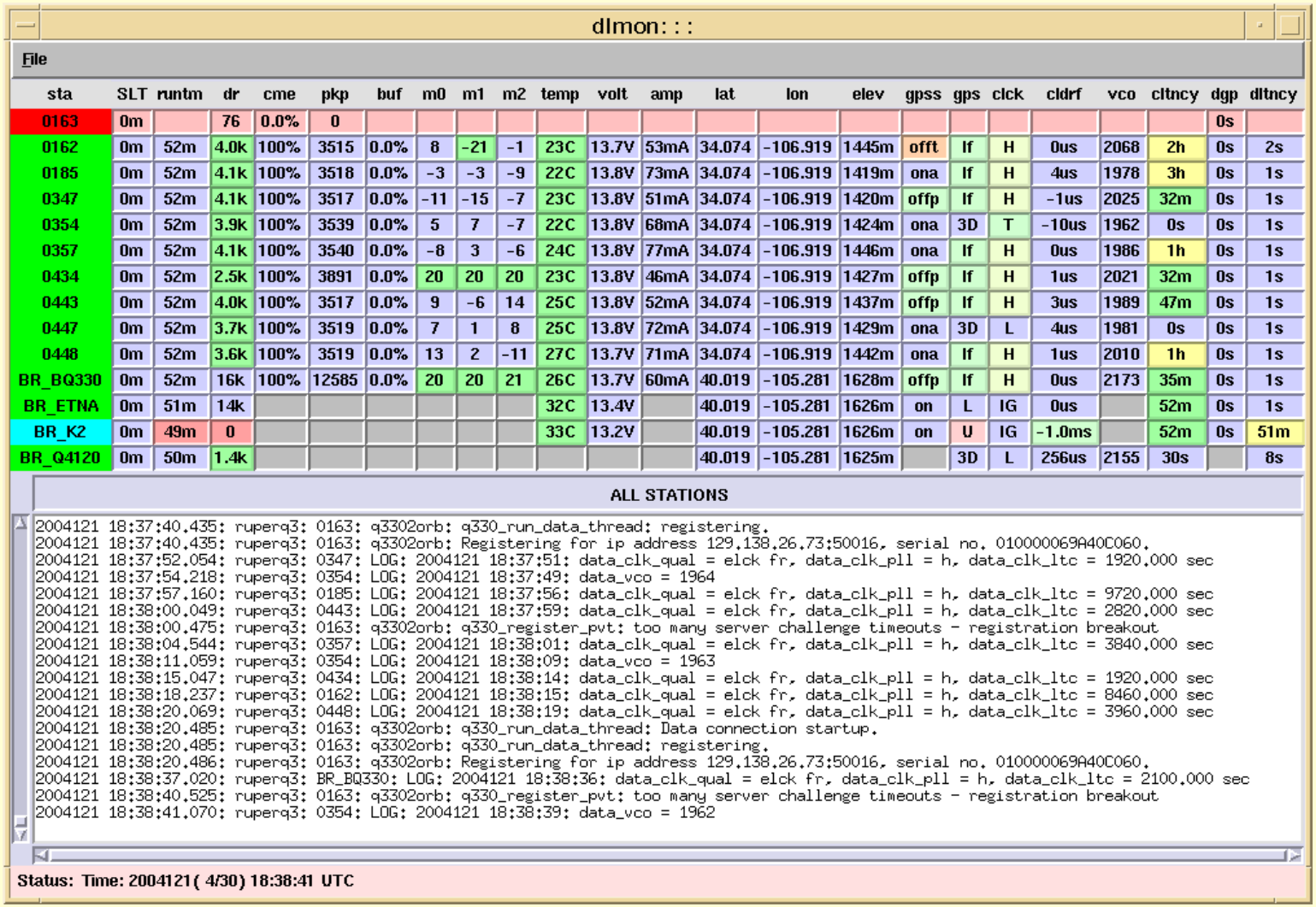
- `pmtfifo(3)` – POSIX-based MT-safe tool that provides a First-In-First-Out (FIFO) queue of pointers:
  - Allows synchronization of many different threads by "pushing" and "popping" pointers on to and off of pointer queues
  - Provides both blocking and unblocked access
  - Provides "peeking" and timeout capabilities
  - Blocking interfaces can be dynamically changed to unblocked interfaces
  - Blocking calls can be temporarily released from another thread
  - Pointers that have been popped off of the queue can be put back with an "unpop" call
- `parse_param(3)` – A high level interface to the various `pfget(3)` calls:
  - Provides the programmer with control of error processing

Changes to Existing Modules

dlmon

- Supports any number of multiple targets (`qt2orb`, `k22orb`, and `q3302orb`)
- Uses a generalized naming standard for status, log, command and response ORB packets
- Provides command-line select expressions and ORB start positions
- Provides a set of templates for defining particular manufacturer status variables and a command line option for selecting a particular template
- Supports Quanterra and Kinemetrics dataloggers and replaces the depricated `k2mon` and `qtmon` scripts
- Provides expanded control capability for Kinemetrics dataloggers
- Sorts the dataloggers in the status display so that the problems (red) are always at the top and otherwise alphabetically

dlmon



BRTT

June 2004

q3302orb

- Many changes included in patches to the 4.5 version:
  - Many bug fixes on original version
  - Changes to support systematized generation of status waveform packets
  - Arbitrary time-channel packaging of output waveform data
- Integrate with new **`dlmon`** and **`dlcmd`**:
  - Support **`dlcmd`** **`getconfig`** command that will return a (nearly) complete configuration state in parameter file form
- Streamlined open/close and register/deregister
- Streamlined packet acknowledgements
- Automatically detect datalogger reconfiguration and cycle (close/deregister/reregister/open) any running data connections
- Convert to new **`pktchannelcalib`** tool

k22orb

- Integrate with new **`dlmon`** and **`dlcmd`**
- New command for erasing the Altus disk and running monitor commands
- Can specify "any" for **`serial`** and **`model`** parameters for point-to-point wired connections
- Convert to new **`pktchannelcalib`** tool
- Convert to new **`orbreapthr`** tool
- Support for continuous streaming data rates other than 100 sps (continuous sample rate is now specified in the parameter file)
- Streamlined initial startup for continuous connections
- User can specify the Altus streaming timeout parameter for continuous connections (this allows the user to put the datalogger into "mode 1" so that the data stream will never timeout)

- **qt2orb** – Integrate with new **dlmon** and **dlcmd**
- **orbwfmeas/dbwfmeas** – Add new **wfmgme** measurements (peak vector velocity, peak vector accelration, Wood-Anderson displacement) to support **dbgme**
- **orbmag/dbml** – Implement waveform clip limits to prevent clipped channels from being used to compute magnitudes
- **orbtrigger** – Implement a new number-of-stations breakpoint table that is used to send out preliminary pick lists to **orbassoc**
- **orbassoc** – Use a new **phase_sifter** parameter that is a UNIX regular expression matched against the input pick list phase codes for each grid to restrict the candidate pick list to certain detections

Back-Compatibility Issues

- **WE CHANGED THE SCHEMA DEFINITION OF THE *calibration* TABLE!**
  - You will have to convert all *calibration* tables to the new schema definition using the **`convert_calibration`** script
  - Changes were necessary to support **`dbbuild`**
- The 4.6 **`dlmon`** will probably not work with the prior version data acquisition programs – also, the prior versions of **`qtmon`** and **`k2mon`** will probably not work with the 4.6 data acquisition programs