Developments in Antelope Python, Web support, and various tools

Dr. Kent Lindquist

Lindquist Consulting, Inc.

European Quanterra And Antelope User's Group Meeting

LINDQUIST
CONSULTING
INCORPORATED

May 15, 2008
Vienna, Austria

Short Comments

- Python interface to Antelope
- Observatory standard practice
- Source-code handling
- Web Interfaces …

LINDQUIST
CONSULTING
INCORPORATED

Antelope Python Interface

- Yet another scripting language interface
- Why?
  - More toolboxes!
    - Scientific Computing
    - Matplotlib, Matlab plotting emulator
  - Towards web-based waveform plotting

Acknowlegments

- PASSCAL authors who shared their beginnings
- Bob Busby and IRIS
- Frank Vernon, Rob Newman, Alex Clemesha (Array Network Facility)
- Ole Nielsen, Duncan Gray, Nariman Habili, Phil Cummins (Geoscience Australia)

What is Python / Why another script interface to Antelope?

- Lots of people use Python
- Scientific, Math, CompSci and Graphics capabilities
- Object-oriented programming accessibility
  - Complementary to C++
  - Antelope object support in Python should surpass that in PHP, Perl
- Nice features, e.g. unit testing
- It’s fun

Approach

- Pyrex ?
- SWIG (Simplified Wrapper and Interface Generator) ?
- Ctypes ?
- “The hard way”
  - Properly balance C / Script interface boundary
  - Exact features of Antelope API (C)
  - “Look and Feel” of Python (Python wrappers)

Python Database pointers

- Full-fledged Python ‘Type’ and Object
- Addressable attribute fields
  - db.record
  - Familiar to Matlab and C coders
- Implements sequence abstraction, subclasses Python lists
  - db[3]
  - Familiar to Perl coders
- Implements Python dictionary access
  - db[‘record’]
  - Familiar to Python coders

Python Method calls

- Procedural (class) methods
  - dbsubset(db, expr)
- OO (instance) methods
  - db.dbsubset(expr)
- Basic calls with wrapped lists
  - Datscope.dblookup(db,”,’origin’,”,”)
- Enhanced calls with Python idioms
  - dblookup(db, table=‘origin’)

Switching to Python

- Design class hierarchies
- Code indentation has syntactic meaning
- Quit putting semicolons everywhere

Current status

- Basic version funded
- Fledgling -- development in progress
- Can do waveform-extraction from demo database and plot
- Currently developing against SAGE and Mac-Fink python
  - Easy access to matplotlib
- Target Summer 2008

Observatory Standard Practice

- Regularization of rtexec coding practices
- Lindquist Consulting, Inc. Technical report on rtexec.pf guidelines
- Cloning of Antelope contributed-code repository style source-code trees
- SDLC
  - Design / Test / Deploy
  - Write programs as generalized tools
  - Establish a coherent toolkit

Code handling

- Localmake
- Build_sourcetree
- Source-code packaging utilities

localmake

- Handling external code linkages
  - Localmake.pf
  - Open Motif, PHP, Python, Matlab
- Special $ANTELOPEMAKE.local extension to make rules
- Part of contributed-code compile
- Compromise between self-contained Antelope and free-for-all GNU approaches
- Morphed into an application packager (dbxcor)

Build_sourcetree

- The problem:
  - Lots of locally grown code not appropriate for Antelope contrib but necessary for local observatory
  - Need automated build methods
  - Need Antelope linkages
- Solutions:
  - Lindquist Consulting, Inc. 2005 technical report: Antelope-dependent source-code trees
  - Built one of these (make rules, directory structure, environment variables, user environment, etc) one too many times
  - Build_sourcetree(1) perl script
  - E.g. /opt/anf, /opt/yoyodyne

Packaging mechanism

- Companion to build_sourcetree
- Distribute compiled results as
  - Tarballs
  - Solaris packages
- Similarities to localmake(1)
- Work in progress, not yet released

Dbrtfm

- Tool to progressively read the full Antelope man page set
- Took about 9 months
- Database-driven
- Options for
  - Short page
  - Long page
  - Specific page

Web Support Technologies

- Web applications:
  - Added value
    - Intuitive User interfaces
    - Removes many hardware/kernel dependencies
    - Broad distribution of results
    - Bringing monitoring results to target audiences
  - Not a panacea…
    - Some tasks still best as GUIs
  - Nevertheless, extremely powerful
    - Why are you running a monitoring system?
    - Bring important results to important people!

Web Development in Antelope

- Introduction to Dr. Rob Newman
- Kent Lindquist / Rob Newman Team approach
  - Antelope infrastructure to create and prepare knowledge products
  - Interface design to bring products close to the user
  - Active technologies to deliver sophisticated products in a dynamic, interactive, and clear manner
- Many, many component technologies
  - XML, Django, AJAX, JSON, SOAP, …
  - Distinct area of expertise
- Useful applications: Dr. Newman…

Thank You

Feedback / Questions
Welcome

LINDQUIST
CONSULTING
INCORPORATED