The Athena Startup Kit (ASK)

Computing in HENP 2003
La Jolla, March 24-28

Wim Lavrijsen, LBNL

Goals

- Make using the Atlas sw framework easier:
  - Automate end-user tasks, to save time
  - Integrate framework, releases, and release tools, to improve perceived coherency
  - Encapsulate volatile wisdom (from web-pages and peoples' heads), to make it more accessible
- Address the needs of all users, from newbies to developers and librarians

Why ASK?

- Currently, in Atlas, under heavy development:
  - The software which the end-user runs
  - The framework which runs that software
  - The tools which manage that framework
  - The configuration of those tools
- Inescapably, a *state of flux* results, and:
  - Software usage becomes low-level
  - Users run into and "fix" the same problems

ASK' Task

- To use Atlas software you have to:
  - Configure your tools *properly*
  - Execute any needed commands in the *correct order*
  - Supply the necessary resources and input at the *right time*, from the *right place*

Getting things right, though *hard for an end-user*, is easy for an automated system => ASK

Use of Python

- Framework scripting uses Python
- Effective "glue," since it's a scripting language
- Tools are accessed on a background shell
  - Provides a closed environment
  - Same work model as end-user
- Python interpreter is used for ASK CLI
- See also GANGA presentation

Step 1: Clean-room

- Python's *os* module
  - *os.environ* dictionary (C's *getenv* and *putenv*)
  - *os.system* calls (C's *system*)
- Run a shell in the background ('/bin/sh')
  - Keep it alive in order to maintain state
  - Communicate through *pipes* (Python's *popen2*)
  - Encapsulate in a Python API

Step 2: Smaller Granularity

- Encapsulate access to tools
  - Locate tools and resources
  - Verify configuration (and repair as needed)
- Build tasks from ordered, low-level commands
- Construct recipes from these tasks
  - Provides input and localization for commands
- Add user interfaces (GUI / CLI) on recipes

Step 3: Be Ready

- Implement workarounds for know problems
  - if release == 'x.y.z': ...
- Try things that *might* work
  - Fall back on alternative tools, configuration, defaults
  - Computer can try things much faster than user can
- Do something that *makes* it work
  - Explicitly fix the workspace (non-temporarily)
- Allow for expert access at all levels, at any time

Top High-Level Demo

[lxplus] mkdir work

[lxplus] cd work

[lxplus] ask

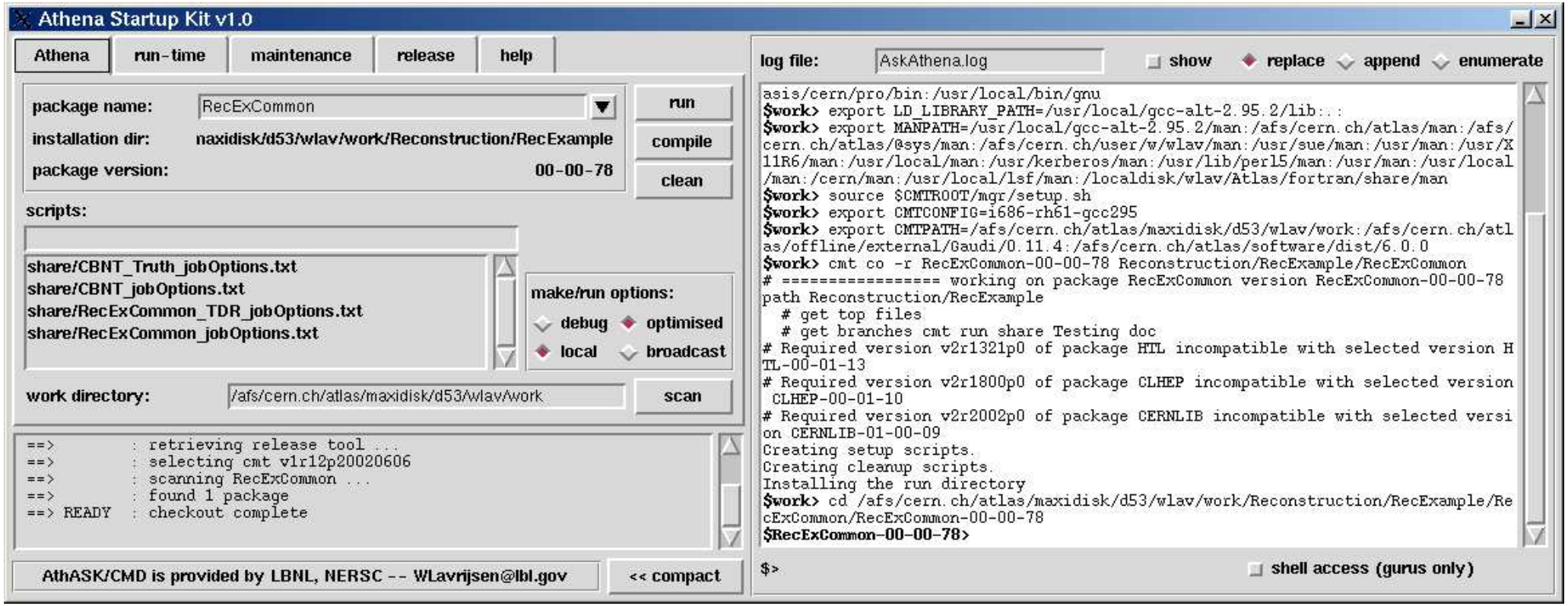
==> : welcome to Athena Startup Kit v1.0

==> NOTE : enter "help()" to receive help

>>> run('AtlfastOptions.txt')

[runs Atlas fast simulation from latest release]

GUI Demo


Athena Startup Kit v1.0
Athena
run-time
maintenance
release
help
package name:
RecExCommon
installation dir:
naxidisk/d53/wlav/work/Reconstruction/RecExample
package version:
00-00-78
run
compile
clean
scripts:
share/CBNT_Truth_jobOptions.txt
share/CBNT_jobOptions.txt
share/RecExCommon_TDR_jobOptions.txt
share/RecExCommon_jobOptions.txt
make/run options:
debug
optimised
local
broadcast
work directory:
/afs/cern.ch/atlas/maxidisk/d53/wlav/work
scan
==> : retrieving release tool ...
==> : selecting cmt v1r12p20020606
==> : scanning RecExCommon ...
==> : found 1 package
==> READY : checkout complete
AthASK/CMD is provided by LBNL, NERSC -- WLavrijsen@lbl.gov
<< compact
log file:
AskAthena.log
show
replace
append
enumerate
asis/cern/pro/bin:/usr/local/bin/gnu
$work> export LD_LIBRARY_PATH=/usr/local/gcc-alt-2.95.2/lib::
$work> export MANPATH=/usr/local/gcc-alt-2.95.2/man:/afs/cern.ch/atlas/man:/afs/cern.ch/atlas/@sys/man:/afs/cern.ch/user/w/wlav/man:/usr/sue/man:/usr/man:/usr/X11R6/man:/usr/local/man:/usr/kerberos/man:/usr/lib/perl5/man:/usr/man:/usr/local/man:/cern/man:/usr/local/lsf/man:/localdisk/wlav/Atlas/fortran/share/man
$work> source $CMTROOT/mgr/setup.sh
$work> export CMTCONFIG=i686-rh61-gcc295
$work> export CMTPATH=/afs/cern.ch/atlas/maxidisk/d53/wlav/work:/afs/cern.ch/atlas/offline/external/Gaudi/0.11.4:/afs/cern.ch/atlas/software/dist/6.0.0
$work> cmt co -r RecExCommon-00-00-78 Reconstruction/RecExample/RecExCommon
================ working on package RecExCommon version RecExCommon-00-00-78
path Reconstruction/RecExample
get top files
get branches cmt run share Testing doc
Required version v2r1321p0 of package HTL incompatible with selected version HTL-00-01-13
Required version v2r1800p0 of package CLHEP incompatible with selected version CLHEP-00-01-10
Required version v2r2002p0 of package CERNLIB incompatible with selected version CERNLIB-01-00-09
Creating setup scripts.
Creating cleanup scripts.
Installing the run directory
$work> cd /afs/cern.ch/atlas/maxidisk/d53/wlav/work/Reconstruction/RecExample/RecExCommon/RecExCommon-00-00-78
$RecExCommon-00-00-78>
$>
shell access (gurus only)

GaudiPython Demo

```
>>> create( 'MyPackage', '4.2.0' )
MyPackage> MyPackage.algorithms = [ 'MyAlg' ]
MyPackage> update()
MyPackage> make()
MyPackage> run( 'MyPackage.py' )
athena> theApp.run( 3 )
```

[runs MyAlg on the first three events]

Summary

- ASK makes running the Atlas framework easier
  - New users can immediately run software
  - Shields end-user from changes
  - Is able to workaround / fix problems
  - Provides skeleton analysis packages
- Python is an effective "glue" language
  - Access to underlying system well implemented

Further Resources

- Online documentation:
  - http://cern.ch/wlav/athena/athask/index.html
- Code access:
  - Atlas CVS: /offline/Control/AthASK
  - /afs/cern.ch/user/w/wlav/public/[dev][pro]