

SCons Discussion Core Week October, 2008

Minutes

Attendees: Joanne Bogart, Toby Burnett, Emmanuel Cephas, Jim Chiang, Richard Dubois, Navid Golpayegani, Heather Kelly, Bryson Lee, Leon Rochester, Tracy Usher

Joanne provided a nice run through of GoGui.

Under Build Options one can choose something other than the default compiler.

Highly recommended that users have GLAST_EXT set up as we do now. GoGui may be fine without it, but the SCons underpinnings will probably not be.

First time running SCons, users will specify a BASE PATH, and perform a container checkout. Joanne notes that she has had trouble fully testing a container checkout on windows - however that may be due to her intermittent windows connection issue, rather than a true problem with GoGui and checking out the code.

Each package included in the package tree includes its tag name, obtained from the SConscript file which includes: # Version <packageName>-xx-xx-xx. If one checks out another copy of the same package, the user will name it in the form: <packageName>-xx-xx-xx. The "original" version of that package will be just named <packageName>. GoGui provides package information so that one can always check the tag associated with any particular subpackage.

The Container level SConstruct file can be viewed. In the lower left panel, the subdirectories of the checkout package, or the selected subpackage are available for viewing.

The executables may be started from the top level exe directory or from the bin directory which includes a variant subdirectory. That subdirectory contains all the wrapper scripts that setup the environment for running and application. These wrapper scripts are generated by the builds, via an SCons-add-on from Navid called generate-script. Unfortunately there is a problem with the scripts on Windows, that needs to be worked out. There is also a script that only sets up the environment but doesn't run any applications called _setup.py. So if one is interested in merely setting up their environment for python, for example, one could use this _setup.py script.

It is possible to run in gdb.

For the builds, the initialization step is long..about 20 seconds on linux.. longer on windows.

Action Items for GoGui

1. Add a configure option to use emacs as the editor for files like release.notes.

Status

New Release Manager:

- Running RHEL4 builds
- RHEL5 will be activated when the RHEL5 boxes are reconfigured by SLAC Computing
- 64 bit builds could be activated at any time
- Windows builds were enabled yesterday - some wrinkles to fix
- Mac builds will begin once Qt is compiled

New Release Manager Web pages a week or more away [see minutes from this morning: <https://confluence.slac.stanford.edu/display/SAS/Release+Manager+Web+Front+End+Discussion+October+2008>]. Current version is behind SLAC firewall and is available here:

<http://glast-tomcat03.slac.stanford.edu:8080/releasemanager/releaseManager-NEW.jsp>

Current version of SCons in use is 1.0.1

Science Tools

Builds using SCons are enabled and are being run through the new RM. The SConscript files do require maintenance. The hope is that once the new RM web pages are up and available, developers will be more likely to step in and tweak the SConscript files as necessary.

Joanne has asked that the ST externals.scons be updated to use ROOT 5.18c-gl1.

GlastRelease

Emmanuel has worked about SCons search for OBF libraries by temporarily setting an rpath (runtime path) option, so that builds will actually progress. This allows him to move forward as we work to work through a reorganization of OBF.

<https://confluence.slac.stanford.edu/display/SAS/GlastRelease-scons+build+status>

OBF

Three libraries have duplicate names.

Reorganization of OBF to conform to <externalName>/compiler/lib

In anticipation of our move to SCons and the reorganization of the externals to follow the previously mentioned convention, our CMT externals will be updated to follow the structure.

CHS

We use CHS/eventFile in the IdfReader package. At the very least CHS/eventFile must be updated to use SCons. After talking with Bryson, it is clear all of CHS will need to be updated in order to meet their requirement of running their regression testing through the ReleaseManager...and some of their existing scripts will have to be rewritten to work in a non-CMT world.

Bryson requests an checkout package-level example so that he may start porting the CHS checkout package to the SCons world. From Bryson:

We (FOS) use the CHS checkout package to build relocatable RPM's that we install in SLAC AFS space for our various software "platforms". We use glastpack.pl to extract a suitable source tree from offline CVS, and then build with CMT. We rely on the RM HEAD and release build processing to perform the regression tests on the event-decoding software portions of CHS. However, to my knowledge there are no customers of the RM-created CHS build directory hierarchies in NFS.

CHS Update 2008-10-24 by Bryson

I've got the CHS package mostly building under SCons. What's missing are three modules from the top level of CVS that are used, I think, by ConfigSystem and nothing else, and don't yet have SConscript files:

- LATC_vrfy
- MOOT
- fswDecipher

Of these, LATC_vrfy looks moderately complicated to convert, since it reaches out heavily into FSW's production CMX instance to find things like the register descriptions.

The core functionality of using DFI from the "fsw" external library to decode event packets into .evt files is working in that it both builds and the resulting system passes the regression test from the CHS/regression package.

Along the way I had to make some modifications to the copy of the Offline SCons "framework" that is committed beneath CHS-scons. Some of these might be useful for converting other packages or extlibs that CHS doesn't use:

1. external.scons supports lists of directories for the "include-path", "lib-path", and "bin-path" elements of an external library definition. In places where these elements are used to further define scalar quantities (like the "ld-path" element) the first entry of the input list is used.
2. external.scons now supports a "linkflags" element in extlib definitions that can be used to add extlib-specific entries to the LINKFLAGS variable of construction environments. Specifically, the "fsw" extlib uses this feature to pass the "rpath-link" argument to ld so that executables can link successfully against shareables picked up from FSW.
3. Fixed the processing done by the "includes" argument to the "registerObjects" Tool so that package public-header exports to the central include directory preserve any subdirectory hierarchy. See rdbModel and xmlBase as examples.
4. Added a new "wrapper_env" argument and associated processing to registerObjects that works with new code in generateScript to allow packages to "publish" environment variables to the wrapper scripts generated for the executables they produce. The "wrapper_env" argument expects a list of ("exe name", {"var": "value", ...}) tuples as input. See CHS/regression for an example of how this is used to ensure that FMX_C_FDB and LD_PRELOAD are set correctly for the event-decoding regression test.
5. Modified generateScript so that if scons is invoked with --with-GLAST-EXT (as it always must be, AFAICT), the user-supplied external-library path is exported as the value of the GLAST_EXT environment variable in the generated wrapper scripts.

Other Items

Tag Collector

Installer

Java installer with command line option

JobOptions file location transition

New Tagging Convention will be unleashed once tagging utilities are provided through GoGui and the tag collector

Note branch tags will have the form: xx-xx-xx-grx

Setting up the environment without CMT

Toby asks how will this work for Python?

Providing updated Documentation for the Workbook

Initial model will be to create pages (perhaps Confluence) that Chuck can point to. Ultimately the pages will be imported into the workbook when they have reached sufficient maturity. Further updates will be handled through the CVS checkout/update mechanism used by SciTools users. For example here is the GoGui user doc:

<http://www.slac.stanford.edu/exp/glast/ground/software/notes/GoGui/GoGui-use.shtml>

Windows Discussion

<http://confluence.slac.stanford.edu/display/SAS/SCons-Windows+Status+October+22%2C+2008>

GoGui Demo for Joe the Plumber

GoGui is available on central linux for RHEL4. How hard will it be to get Qt for RHEL3 to make an installation available there too?

Outstanding issues Joanne highlighted on Monday:

1. Tagging Feature

Joanne would like to understand how most users tag. She (and I) primarily use rtag, but there are instances when the tag command is required. Joanne has located and understands the existing MRvcmt code. She should be able to translate it over to GoGui.

2. Windows

First step is to understand what SCons provides. Joanne is stuck trying to get SCons to create a VS project. It is expected that this will not provide all the required developer features we desire. The anticipated course of action will be to follow the model used for CMT, where we have separate code that builds the Visual Studio project files for us. Toby has some python code for this purpose and Tracy has previously worked with that code to force SCons to produce more useful project files.

3. Override directories

Need to modify the way we use SCons to make override directories possible. Some of Joanne's existing GoGui code assumes that this ability exists. The plan is to support one over-riding directory. Leon states he can live with that. That override directory may contain any number of packages. The fix will come in the SConstruct files, which while not pretty, will be invisible to most users.

<http://confluence.slac.stanford.edu/display/SAS/GoGui+Status+and+Laundry+List>

Where To From Here?

Demonstrating CMT and SCons equivalence.