

GoGui Status and Laundry List

Status

Latest release version [available for download](#) is 1.4.0 (Linux, Windows), or 0.9.6 (Mac).

The most comprehensive and up-to-date documentation may be found [here](#) .

Display and navigation

All primary components of [main window](#) (image from old GoGui version) are in place and active:

- Central area for display of output and per-package files of interest, such as release.notes
- Menu bar with working menus
- Tool bar with working tools
- Status area
- Package hierarchy for base release
- [Packages in supersede directory](#) (image from old GoGui version)
- File hierarchy within selected package

[Annotated image](#) from GoGui 1.3.0. The latest version of GoGui (1.4.0) looks similar; the only difference is that the newer version does not have a CVS tag button.

Configure options

- [Select base](#) root directory
- Select supersede directory
- [CVS-related set-up](#):
 - path to cvs command
 - value for CVSROOT
 - value for CVS_RSH
- SCons-related set-up:
 - [path to SCons](#) command
 - [select external libraries](#) root directory [Added warning in case selected path != GLAST_EXT environment variable 10/16]
 - [build options](#): compiler, debug or not, optimized or not
 - extra SCons options: [menu item](#) under Options brings up [dialog](#) allowing user to add arbitrary SCons options to the command

Functions

- Select a package
- Display per-package information
 - Release notes
 - SConscript
 - <package>Lib.py
 - ChangeLog
 - version information
 - command output
- Search displayed information for string
- Repository functions
 - cvs update (3 flavors)
 - cvs commit [but maybe cannot handle containers like celestialSources properly yet]
 - [check out "regular" package](#)
 - [check out container](#)
 - ~~cvs rtag~~ Now that tag strings include the package name it is no longer necessary to support tag as well as rtag. See also the SCons-aware [command line tagger](#), which uses the same underlying utilities as the GoGui tag function. OBSOLETE. All tagging should be done from a SLAC Linux node
- Build functions
 - [Build a package](#)
 - Clean a package
 - Build or clean arbitrary target(s) with arbitrary options by using input field at the bottom of the GoGui window. Only is active if full container (rather than package) is selected in the navigation pane.
- Remove package
- [Exclude a package](#) (SCons will use next version in line if there is one)
- [Run a program](#) (as of Sept 29 using setup file generated by SCons)
- Open terminal window. GoGui will look for a file called _setup.sh (setup.bat on Windows) and will run it first. This means your environment in that window is suitable for running programs with no additional setting of environment variables, etc.
- File-specific functions selected by right-click in package hierarchy
 - run a program
 - debug a program (on Linux with gdb only)
 - [cvs diff](#) of file
 - [browse](#) a file
 - edit a file (preferred editor may be set from the Options menu)
 - for solution (*.sln) files on Windows with VS 2008, option to bring up Visual Studio

- GoGui debug output goes to [msg tab](#) by default. On Linux only, one can instead send debug messages to console by invoking as **GoGui --ggDebug**. When debugging GoGui invoked this way, messages go to debugger.

Performance





Execution of the first command SCons is given is especially slow. Subsequent commands for a code base as large as ScienceTools (to say nothing of GlastRelease) are still intolerably slow with default behavior (reread everything to regenerate all dependencies), especially on Windows. GoGui now makes use of the SCons facility to run subsequent commands without re-checking the dependency tree. SCons must be re-initialized if the base directory (the one containing the SConstruct file) or external library have changed. GoGui handles this automatically. It is also possible for the user to [explicitly ask for SCons to re-init](#) if s/he knows the dependency tree has changed. As of Sept. 10 SCons commands run asynchronously. They can be aborted.

As of GoGui release 1.2.0, the explicit init step has been abolished. Newer versions of SCons as well as the hardware platforms now at most people's disposal are quite a bit faster. The init step (now done implicitly is no longer "intolerably slow", and there is something to be gained in simplicity and robustness by always including it.

A welcome bonus of using a supersede directory is that SCons init (implicit for GoGui 1.2.0 and beyond; explicit for older versions) is typically much faster. It only checks the packages in the supersede directory; the base installation acts like an external library.

To do




Display and navigation

-  Visual representation of variants is misleading. Should disable builds of variants that our implementation of SCons will not in fact build and indicate visually which variant is the buildable one. - FIXED as of Nov.6.
-  Several operations (e.g. checking out a package, excluding a package...) invalidate the set up of the currently running SCons process (if there is one). This needs to be clearer. [Show state in SCons icon as of 0.9.2]
-  Window does not always update spontaneously after a package is excluded (or un-excluded) to display the new state properly. [Fixed as of 0.9.3]
- **Fix scrolling behavior.** Minor unpleasantness in base browser display: if keys are used to scroll a page at a time the highlighted package can change if the actual selected package scrolls out of view.
-  Add visual indication of whether SCons process is running or not. [Done as of 0.9.2]
- **Indicate initialization errors better.** When the SCons initialization process skips a package (usually because of an error in its SConscript file) indicate graphically somehow in the file hierarchy. Messages in the global output tab can be difficult to spot.

Configure options

- **select preferred debugging environment**
- (maybe) **select preferred cvs diff format** [No ability to configure yet, but added -u option to improve output in 0.9.2]

More functions or enhancements

- Additional file-specific functions selected by right-click in package hierarchy
 -  run Doxygen [Done as of version 0.9.6]
 - **Do a better job of determining which files are executable, which are editable, and which are both (e.g. scripts)**
-  **capture qDebug() output in another panel of central tabbed widget.** [Done as of 0.8.8]
- **implement dry-run builds and cleans** (print what they would do without actually doing it). Can do this by using target field at the bottom of the GoGui window and including SCons --dry-run option.
-  special handling for container packages (celestialSources, irfs) for various cvs operations: checkout, update, commit, tag [tag is done as of Nov. 3].

Bugs

- If the specification of scons command doesn't resolve to an actual file GoGui crashes. ~~At the very least should get rid of default and force user to supply something since this default often causes the crash.~~ [Done in 0.9.2]