SCons-generated Studio Files

Status

- Global solution files all.sln (includes projects for absolutely all libraries and executables) and allGleam.sln (projects for all libraries, Gleam executable, and test_Gleam executable) exist and do almost everything they're supposed to.
- All libraries are built successfully except for RootDisplay, which also fails when built directly by SCons
- public headers, job options, etc. will be installed by Visual Studio just as they are by the SCons build

There is just one item left to attend to before alpha-testing. It's a tricky one, though, as well as essential. Project files for the Gleam executable and for Gaudi test programs all fail.

Most likely some special handling will also be needed in order to get Visual Studio to build programs using GuiSvc but I don't consider that a show-stopper.

Recent history

As of the beginning of this week (May 14th) 20+ projects were failing to build out of about 90 total. Many were derivative failures: a library wouldn't build because it depended on another library which was failing. Independent patches were made to the following packages:

- gcrSelectRootData - bug in SConscript Fixed in tag gcrSelectRootData-02-02-08
- G4Propagator - eliminate unnecessary #include of inaccessible header. Fixed in tag G4Propagator-03-01-01
- celestialSources/GRBobs - Root macros in xml directory (file extension .C) were misinterpreted by VS. It tried to compile them with C compiler. Fix so far (not committed because still under discussion) is to not install these files. They then don't become part of the VS project.
- GCRCalib source file was accessing a header in TkrRecon, TkrRecon/src/Filter/ITkrFilterTool.h which was not public, hence not accessible with standard include path. I moved the header to TkrRecon/TkrRecon/Filter/ITkrFilterTool.h, made it public, and changed all references to it to be consistent with the new location. Not committed since it could be that GCRCalib shouldn't be doing things which require this file.
- TkrRecon was failing to compile because various headers under its src directory couldn't be found. Made a patch to one of the files in the SConsFiles package to add **packageName**/src to the include path.
- GlastClassify library was not getting built because the order of project builds wasn't quite right. (Was ok in GlastClassify.sln but not the global solution files.) The static library in the same package, ClassificationTree.lib, needs to be built first. There was a problem in the Gleam solution file - I still don't know exactly how it came about - which propagated to the global files. In fact this library is not necessary to build the Gleam executable, so I took out the dependence on it, a small change to Gleam/SConscript (not yet committed). This fixed the problem in the global files. There is still something a bit odd about GlastClassify.sln itself, but it's usable.