

Checkout3 Windows Digel

Comments on the Windows distribution of Science Tools

This page is a summary of some problems related to installing and running science tools under Windows. For some of these, work on fixes may already be in progress. Except for possibly the lack of support for FTOOLS on Windows, these are not insurmountable problems.

- Three of the science tools (**gtobssim**, **gtorbsim**, **gtburstfit**) will not run as delivered unless the user has the VC development environment installed. The problem is that the C runtime library MSVCR71D.dll (D for Debug) cannot be found. The installer in fact installs it, but the wrapper scripts generated by the RM do not point to it. Apparently this is a fairly deep problem with how the tools are built; the reason that many of the tools want to have access to MSVCR71D.dll is only because cppunit, a testing program, wants it. For the 3 above-named tools, the problem is somewhat deeper in that the RM does not seem to be able to automatically determine that the tools are going to think that they want the MSVCR71D.dll. Work arounds come to mind, like having the RM always put the DLL in the Path variable.
- We are currently **distributing only debug builds** in VC8. My understanding is that there's no fundamental reason for not making optimized builds, but the difficulty is in finding a way for the RM to get the information it needs from build logs on Windows via Isf.
- The RM still **installs all of the test applications** and assorted applications that are not intended for users. This is not a Windows-specific issue, of course.
- The **wrapper scripts always start a new command window** when a tool is started. This is inconvenient for all of the tools, not just for the tools like ModelEditor that do not need a command window at all. The new windows do not go away after the tool completes running, and all of the various variables like Path and PFILES that the wrapper scripts modify just keep getting longer if a tool is run from one of these new windows (which in turn opens another Window). Also, I imagine that any kind of scripting together of the wrapper scripts would be problematic.
- The **wrapper scripts do not work if the tools are installed in a directory path that has a space in its name**. This is an especially attractive nuisance because so many directories in Windows have spaces (like 'Program Files') and the default installation directory suggested by the Java installer is most likely within 'My Documents'. The fix is not difficult.
- **FTOOLS in general are not available under Windows**. Although the very useful fv tool does have a Windows version, the utilities for manipulating FITS files are quickly sorely missed. I do not think there's any momentum or motivation for HEASARC to support FTOOLS under Windows.
- Connecting ModelEditor with ds9 is in principle possible, but I have not managed it yet. This is not a problem with the RM, and maybe not even Windows specific, but enough additional work has to be done (installing a Python DS9 library and installing and compiling XPA) that we really cannot realistically expect users to do them. To the extent possible and reasonable, we should **integrate the installation of the science tools and their interfaces to Python and ds9**.