Making rhel7 builds with RM

Subtasks

Db updates

These are well described in Tom's notes.

Jenkins

Need to get together with one or both of Tony and Max for part of this. See more helpful notes from Tom.

RM programs must run on rhel7

Such as checkoutBuild, compileBuild, triggerBuild, etc. For Linux OSes, they're kept under ~glastrm/grits-cpp. The sym link bin is translated according to the value of @sys on that platform. For rhel7 values are currently (and likely to remain) amd64_rhel70 and amd64_rhel60.

Externals

RM programs depend at a minimum on Qt and MySQL. Similar to the executables, they're stored under a directory with a name of the form /nfs/farm/g/glast/u35/grits-externals/**some-dir** where some-dir is in the platform's sysname list. For rhel7 this will be amd64_rhel70

I downloaded source for Qt open source version 4.8.6, unpacked and configured like this:

./configure -static -qt-sql-mysql --prefix=/nfs/farm/g/glast/u35/grits-externals/amd64-rhel70 -nomake examples -nomake demos -nomake translations -nomake tools

Tom's notes say to install the Spirit miniboost package (it's include files only), but it is not installed as such for rhel6. I can find no evidence in source code, Makefiles or Qt .pro files that it's used.

They also mention stx-exparser.   That **is** used, but apparently only by workflow, not by releaseManager.   And its README explains that it depends on the Spirit miniboost package! But only at build time. RM itself doesn't need it. On rhel6 all the boost files needed can be found under /usr/include/boost.   That's not currrently the case for rhel7.

Possible short-cut

It may be that we can get by with the rhel6 binaries of RM programs and externals to begin with. (The rhel5 build of GoGui, another Qt-based program, is perfectly usable on rhel6 and comes up with no complaints on rhel7.) Ultimately we should have a native rhel7 build, but if the rhel6 binaries are usable the db updates and Jenkins interface could be tested much more quickly.