DARBS SETUP for Windows

1. Download Required Files
2. CMake v3.14 - Windows win64-x64 Installer - https://cmake.org/download/
3. MinGW v16.0 - https://nuwen.net/mingw.html (Boost.Signals was deprecated in version 1.54.0 and removed from Boost C++ Libraries beginning from 1.69)
4. Boost v1.67
5. dlfcn-win32 library (to stop this error "fatal error: dlfcn.h: No such file or directory #include <dlfcn.h>" during ksclient compilation on windows)
6. DARBS files
7. Install CMake
8. Run installation file with admin rights
9. Select "Add System PATH"
10. run> cmake --version to verify installation
11. Install MinGW
12. Extract MinGW to a preferred location
13. Add ..\MinGW\bin to system PATH
14. run> gcc--version to verify installation
15. NB - Boost.Signals may have to be built except for (MinGW v16.0)
16. Optional - Building Boost
17. Extract Boost to a preferred location
18. run> bootstrap.bat gcc
19. run> b2 toolset=gcc variant=release link=static --with-signals --with-thread --with-<other required libraries>
20. Compile DARBS (..\trunk)
21. Open command prompt and navigate to ..\trunk\build folder
22. run> cmake -G "MinGW Makefiles" ../
23. Optioinal - You may need to edit CMakelists.txt in ..\trunk if you encounter errors such as "Unable to find the requested Boost libraries" and add the following:
    1. if (WIN32)

set(BOOST\_ROOT "e.g. C:/local/boost\_1\_67\_0")

set(BOOST\_LIBRARYDIR "e.g. C:/local/boost\_1\_67\_0/stage/lib")

set(BOOST\_INCLUDEDIR "e.g. C:/local/boost\_1\_67\_0/include")endif(WIN32)

* 1. if(CMAKE\_CXX\_COMPILER\_ID STREQUAL "GNU")

set(CMAKE\_CXX\_COMPILER\_ARCHITECTURE\_ID "x64")

endif() #--Sometimes, cmake may skip libraries expressed in this form "libboost\_signals-mgw82-mt-d-x64-1\_67.a"

1. Optional - You may need to edit CMakelists.txt in these folders \server, \terminal, \ks

*Add this line: target\_link\_libraries(<target> ws2\_32 mswsock) #--to avoid errors like "undefined reference to `\_\_imp\_WSAStartup'" etc*

1. Compile dlfcn-win32 library (#--"fatal error: dlfcn.h: No such file or directory #include <dlfcn.h>" during ksclient compilation on windows)
   1. Extract dlfcn-win32 to a preferred location
   2. Build using MinGW. Open command prompt and navigate to ..\dlfcn-win32 folder
   3. run> cmake -G "MinGW Makefiles"
   4. run> make
   5. Change the name of dlfcn-win32-config.cmake to e.g. dlfcn-win32-config2.cmake and save.
   6. Edit contents of dlfcn-win32-config.cmake and save.

*Comment the macros leaving only the following:*

set(dlfcn-win32\_INCLUDEDIR "<dlfcn-win32 root folder>")

if(NOT TARGET dlfcn-win32::dl)

include(<dlfcn-win32 root folder>/dlfcn-win32-targets.cmake)

endif()

set(dlfcn-win32\_LIBRARIES dlfcn-win32::dl)

set(dlfcn-win32\_INCLUDE\_DIRS ${dlfcn-win32\_INCLUDEDIR})

* 1. Edit CMakelists.txt in \ks folder. Add this lines:

set (dlfcn-win32\_DIR "<dlfcn-win32 root folder>")

include\_directories(" <dlfcn-win32 root folder>")

if (WIN32)

find\_package(dlfcn-win32 REQUIRED)

set(CMAKE\_DL\_LIBS dlfcn-win32::dl)

endif(WIN32)

target\_link\_libraries(ksclient ${CMAKE\_DL\_LIBS})

* 1. run> make
  2. copy libdl.dll from <dlfcn-win32 root folder> to ..\MinGW\bin

1. Compile DARBS (..\xml\branches)
2. Do same as item 5.
3. If you encounter error " error: no match for 'operator=' (operand types are 'std::\_\_cxx11::string' {aka 'std::\_\_cxx11::basic\_string<char>'} and 'boost::filesystem::path') name = path.filename();"
4. Edit file: ..\ks\src\AbstractConditional.cpp
5. Change line 56: name = path.filename() to name = path.string()
6. To run examples, run using compiled exes in ..\xml\branches