Beginners Guide for using CMake to build RODOS

2012-07-04

Introduction

To use cmake to build RODOS, there are several questions to be answered first:

- *What platform do you want to build for?* Look in the *cmake* subdirectory of RODOS if there is a file with the name of your platform. Remember that name.
- *What operating system are you building on?* On Windows you have to use the *CMake*-GUI, on Linux you can also use *ccmake* or simply just *cmake*.
- *Do you want the tutorials of RODOS to be build?* There is an option to build some tutorials that you can use to test whether RODOS works properly on your machine.
- *Do you have a project directory for your own RODOS project?* It's possible to use *CMake* to build your own RODOS project. The only thing you have to do is creating a *CMakeLists.txt* file specifying the structure and place of your project. All settings used for building the RODOS library (compiler, compiler flags, etc.) are available in your *CMakeLists.txt* file. Even the RODOS library is available to be linked against.

Now you should be ready to start building RODOS.

Starting CMake

Here the different ways to use different *CMake* user interfaces to build RODOS is described. After building, the RODOS library will be in a subfolder of the *libs* folder in the RODOS directory while all built applications (tutorials and maybe your own applications) will be in the *bin* subfolder of your build directory.

CMake-GUI

1. Start the GUI.
2. Enter the directory RODOS is located in.
3. Enter the directory you want to build in. *Note*: This should not be the RODOS directory itself but a directory dedicated to build in.
4. Click *Configure*.
5. Set the options you want (explained below).
6. Click generate.
7. Navigate to the build directory and use the generated files to build RODOS.

ccmake

1. Navigate to your build directory on your command line. *Note*: This should not be the RODOS directory itself but a directory dedicated to build in.
2. Call *ccmake* with the first command line argument being the relative or absolute path to the directory RODOS lies in.
3. Press `[c]` to configure RODOS. The options should show.
4. Set the options you want (explained below).
5. Press `[c]` again to configure RODOS with the new options.
6. Press `[g]` to generate your Makefile.
7. Use the generated files to build RODOS by calling *make*.

cmake

1. Navigate to your build directory on your command line. *Note*: This should not be the RODOS directory itself but a directory dedicated to build in.
2. Call *cmake* on your command line while specifying the options you want to be set (options are explained below). Specify the options with the following scheme: `-D<OptionName>=<OptionValue>`
3. Use the generated files to build RODOS by calling *make*.

Options

- `TOOLCHAIN_FILE`: This option specifies the file that sets up the build environment for building RODOS for your platform. It's the file with the name of your platform in the *cmake* subdirectory of RODOS. *Important:* Only enter the filename but not the extension (*.cmake*).
- `BUILD_TUTORIALS`: If you choose to set this option, some tutorials of RODOS will be built. If you specify this on the command line use *true*, *false*, *on* or *off* as the value.
- `BUILD_TESTS`: If set to true, this option will make *CMake* to build the tests for RODOS (normally this should not be what you want except you are changing things in the RODOS core and want to test if RODOS behaves as expected). If you specify this on the command line use *true*, *false*, *on* or *off* as the value.
- `OWN_PROJECT`: If you want an own project to be built with *CMake* you can specify the directory the *CMakeLists.txt* file lies in with this option as an absolute or relative path. It also may be a relative path from the RODOS root directory.