Building Stroika

# Common

Stroika is a C++ class library. For the most part, it’s built make. However, internally, some of these make rules use perl etc scripts.

# Quick Start

## For the very impatient

wget --tries=10 --no-check-certificate --output-document=STROIKA\_DOWNLOAD.tar.gz <https://github.com/SophistSolutions/Stroika/archive/master.tar.gz>

tar xf STROIKA\_DOWNLOAD.tar.gz

cd Stroika-master

make all run-tests

## For the more patient (hints about what to try next)

* wget --tries=10 --no-check-certificate --output-document=STROIKA\_DOWNLOAD.tar.gz <https://github.com/SophistSolutions/Stroika/archive/v2.0a18.tar.gz>

*or*

wget --tries=10 --no-check-certificate --output-document=STROIKA\_DOWNLOAD.tar.gz <https://github.com/SophistSolutions/Stroika/archive/master.tar.gz>

* tar xf STROIKA\_DOWNLOAD.tar.gz
* cd Stroika-2.0a18 (or whatever extracted)
* make help

*Not needed, but gives some idea of make options*

* make check-tools

*Not needed, but tells you if you are missing anything critical*

* make default-configuration

Not needed, but it’s a springboard for setting up the configuration you want.

* + Review/edit ConfigurationFiles/DefaultConfiguration.xml
  + Or try something like
    - make default-configuration DEFAULT\_CONFIGURATION\_ARGS="--enable-assertions --enable-trace2file --compiler-driver 'g++-4.8'"

*or*

* + - make default-configuration DEFAULT\_CONFIGURATION\_ARGS="--help"
* make all

*Builds everything (takes perhaps 15 minutes? – depends a lot on OS/computer speed).*

* make run-tests

*Runs regression tests (optionally on remote machines, or with VALGRIND)*

# Required Tools

## Required for ALL platforms

* git
  + not needed to build, but
* make
* patch
* perl
* wget

## For Windows

* Visual Studio.net 2013 (or later)
* Cygwin

Including

* + dos2unix
  + unix2dos

## For UNIX

* Compiler
  + gcc 4.7 or later OR
    - Stroika is currently tested with gcc 4.7- gcc 4.8
  + llvm (clang++) 3.4 or later

Optional Components

* curl
  + If present, Stroika can be configured to include it and take advantage of it
* openssl
  + You can use the statically linked copy in ThirdPartyProducts, or the os-intalled .so files.

# Build Process

On any platform, building Stroika, and all is demo applications and regression tests is as simple as cd’ing to the top-level directory, and typing make

## Special Targets

* make

Make with no arguments runs ‘make help’

* make help

Prints the names and details of the special targets

* make all

builds the stroika library, tests, demos, etc.

* make run-tests

Builds Stroika, and all the regression tests, and runs the regression tests

* make project-files

Builds project files which can be used for things like visual studio (not needed)

* make check-tools

Checks if the tools needed to build Stroika are installed and in your path. This is done automatically, and generally not needed explicitly.

## Configuration

Building Stroika requires building special configuration files. But using the above mechanism automatically builds them for you. To customize your Stroika configuration, you can manually run

* Configurations can be generated via
  + make default-configuration
    - This generates ConfigurationFiles/DefaultConfiguration.xml
* Configurations can be applied via
  + make apply-configurations
    - This generates makefiles (as appropriate for your platform/Configuration.xml file), and C++ #include files

## Using Visual Studio.net

Visual Studio.net project and solution files are available for the Stroika demos, top-level project files, and regression tests. Once you have built your configuration files (see above), you can use the project files to build, test, extend and develop Stroika.

## Using QtCreator (on unix)

Run Library/Projects/QtCreator/CreateQtCreatorSymbolicLinks.sh to create project files at the top level of your Stroika directory. Then you can open that .creator file in qtCreator, and build and debug Stroika-based applications.