Building Stroika

# Common

Stroika is a C++ class library. For the most part, it’s built using a thin layer of perl scripts to build configuration, and then invoke platform specific build tools.

This is NOT necessarily well thought out, nor well done. I considered using cmake, or ant as a portable way to organize builds. I still might. But for now – its perl scripts + platform specific mechanisms.

# Required Tools

{this section also needs a rewrite}

## For Windows

* Visual Studio.net 2012 (or later)
* Cygwin

Including

* + curl
  + dos2unix
  + git
  + make
  + patch
  + perl
  + unzip
  + wget

## For UNIX

* Compiler
  + gcc 4.6 or later OR
    - Stroika is currently tested with gcc 4.6- gcc 4.8
  + llvm 3.4 or later
    - (still doesn’t work, but close)
* perl

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
* Configurations can be applied via
  + make apply-configurations

These are generally not needed (done automatically).

But these internally run the below scripts, which can be run

* GenerateConfiguration.pl
  + Generates ConfigurationFiles/DefaultConfiguration.xml
* ApplyConfiguration.pl
  + 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.