# Notes:

Think about ease of changeabilty.

Prioritise decisions.

Cross platform development.

Multi-threading.

Keep as much as possible out of project/solution file: decoupling from Visual Studio to aloow easr of using different tools.

For consistency file referenced from MurkyDev.

Motivation for compiling with both MS vc++ and Intel: Different diagnostics/static analysers/debuggers. Verify potential compiler bugs for sanity.

# Setup Procedure

Vs: new win32 app, empty project.

Make sure reference all versions (32/64) when altering project settings.

Add x64 to configurations.

Use Intel compiler/rebuild

Note: From **project/solution** file to development root “../../../”

Get both opengl32.libs. There are two:64 bit, 32bit. Add to additional libs.

## Add includes

(visual studio) project/murkyframework properties/configuration properties/c++/additional include dir…

../../../murkyFramework/include

../../../external

../../../external/glew/include

## Add libraries

(visual studio) project/murkyframework properties/configuration properties/linker/general/additional…

(visual studio) project/murkyframework properties/configuration properties/linker/input/additional…

Opengl32.lib

## Add to pre-processor

GLEW\_STATIC

To remove Boost/Visual Studio paranosis: \_SCL\_SECURE\_NO\_WARNINGS

## Add new project

Create/copy directory structure.

Rename dev c++\new project\build\Visual Studio\\*.vcxproj , \*.vcxproj.filters

Change \*.vcxproj: <RootNamespace> …blah… , <ProjectName> …blah…

## Add new gfx

# Directory Structure

<dev c++ >

-<murkyFramework>

- -<src>

- -<build>

- - -<VS 2013> .vcproj .sln

-<proj1>

…

# Conventions

Classes start with caps apart from simple like u32, vec3.

use namespaces and classes.

Namespaces start with lower cse

Spaces not tabs.

<https://google-styleguide.googlecode.com/svn/trunk/cppguide.html>

# Maths

Vector rotation, performed (intuitively)left to right.

Ie M is an orthonormal base, muliplying R by v.x.

v.M = v’

# Tools

Visual Studio with Visual Assist, Intel Studio, //Pvs Studio,

Notepad++.

# Visual studio solution/project

<http://msdn.microsoft.com/en-us/library/b142f8e7.aspxSolution> :collection of projects

One solution per spp.

Header files referenced from root directory. dev c++

Set solution project dependencies for each project!

$(SolutionDir)\build Solution files

$(ProjectDir)\build Individual project files .vcxproj

Each project self-contained.

Good Projects to examine/reference:

<http://stackoverflow.com/questions/1398445/directory-structure-for-a-c-library>

Threading

|  |  |
| --- | --- |
| Thread 0  Graphics | Thread 1 |
| Gfx start pushing  Gfx finish pushing | Input.  Control |