

Coding Guidelines

Change History

When	Who	Why
Jan 27, 2000	Gitm	Initial version

Introduction

This document describes the development guidelines that should be used through the project. This includes a description of the CVS hierarchy for the source code, source code formatting style and code documentation procedures.

Source Code Formatting Style

C++

For C++, the following coding style guidelines should be used. The /tankwar/template directory contains a set of template source files that can be used for new source files.

Variable Naming

Variable naming should use the 'polish' notation. The prefix applied to the variable name will identify the data type that the variable contains. Some examples are shown in Table 1.

Prefix	Data type
sz	LPSTR, LPCSTR
i	int
ch	char
b	BYTE
dw	DWORD
l	long
u	ULONG
f	BOOL, bool
str	BSTR

Table 1 - Data Type Prefixes

In addition, pointer types should be prefixed with the character 'p', for example a pointer to an int would have the prefix *pi*.

Arrays should be treated as pointer types, for example – an array of integers would also use the prefix *pi*.

Prefixes for user defined types should be chosen appropriately and used consistently throughout the project. As the project progresses Table 1 will be updated accordingly.

Code Structure

All blocks should be indented by two characters, TAB characters should not be used. A simple statement is shown below.

From this example you can see that for flow control statements (such as switch, if, while etc) the opening brace should be placed on the same line as the statement. The code block following the statement should then be indented two characters.

```

/** DLL Entry Point

    Main DLL entry point. Provides the default processing – nothing
    special needed here.

*/

BOOL APIENTRY DllMain(HANDLE hModule, DWORD ul_reason_for_call, LPVOID
lpReserved)
{
    switch (ul_reason_for_call) {
        case DLL_PROCESS_ATTACH:
        case DLL_THREAD_ATTACH:
        case DLL_THREAD_DETACH:
        case DLL_PROCESS_DETACH:
            break;
    }
    return TRUE;
}

```

Code Documentation

The project uses the doxygen program to generate documentation from the source files. This requires some additional information to be placed in the comment blocks.

As shown in the source code example above, every function should be preceded by a specially formatted comment block (`/**` instead of `/*`). Doxygen will insert the text from this block into the generated HTML documentation.

Each sub-project directory has an appropriate doxygen configuration file placed in it to control formatting of the generated documentation.

End-User Documentation

The end-user documentation will be generated as a HTML help file. To build the .CHM help file you will need the Microsoft HTML Help Compiler (available from the Microsoft site).

The source files for the end-user documentation can be found in the `/tankwar/docs` directory of the CVS repository.

The Build Process

The build process will be run every Saturday night (Australian local time – GMT +10:00) and the results posted to the developers mailing list the next day. Please ensure that you can build successfully before committing any changes to the repository.

The build process will generate the following items

1. Debug and Release versions of all tools.
2. Debug and Release versions of the game and dedicated server.
3. Documentation (in HTML and CHM format) for all source modules.
4. End user documentation (in HTML and CHM format).
5. Game data files.

As we approach Alpha stage for the project a self extracting setup will also be built with the Release executables and end user documentation.

A zip of the build results will be made available on the TankWar web site the following day if the build was successful.

To be able to do a build on your own machine you will need to have the following tools installed (more information on these tools can be found in the *Additional Tools Required* section) :

HTML Help Workshop
DOXYGEN
Microsoft Windows Installer for Visual Studio

In addition, you will have to have you system configured so that the Visual Studio tools can be run from the command line (look for the file VCVARS.BAT and make the appropriate changes to your AUTOEXEC.BAT).

Additional Tools Required

This section provides information on the additional tools that will be required to work on the project.

WinCVS

The WinCVS application is a freeware graphical CVS client for use on the windows platform. The application is available from <http://www.wincvs.org>. To use WinCVS as a developer against the TankWar CVS archive you will need to configure it for SSH connections. Instructions on this are given in the SSH section below.

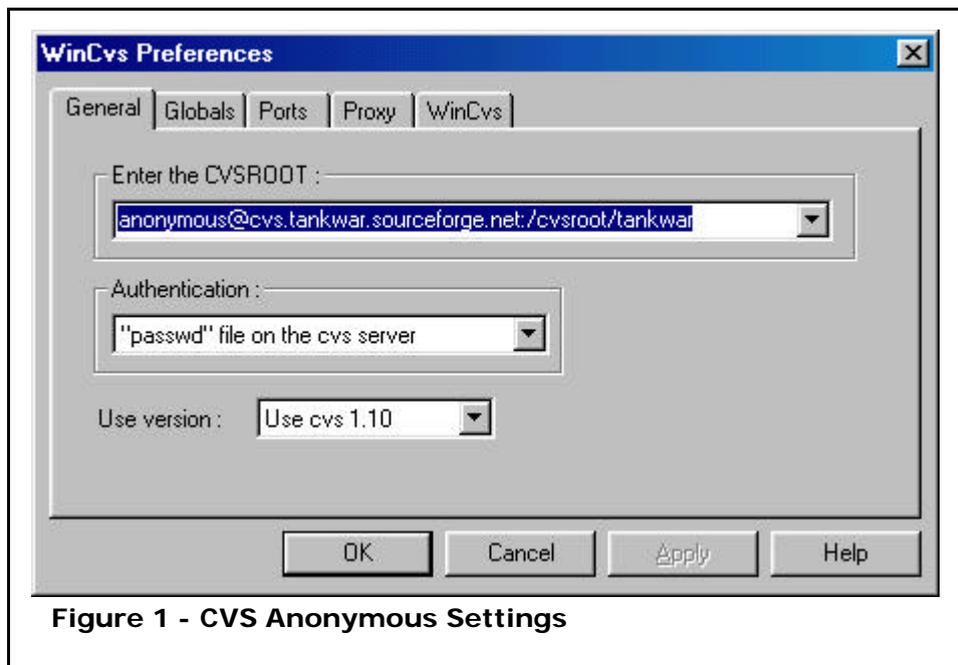


Figure 1 - CVS Anonymous Settings

If you simply want to look at the source code for the project you can connection anonymously to the CVS server using the preferences show in Figure 2.

SSH

To access the TankWar shell account you need to have a version of SSH installed correctly. There are a number of locations where SSH can be downloaded from – do a search on AltaVista for 'ssh wincvs' which will point you to the appropriate locations.

Doxygen

The Doxygen utility is used to automate the process of documenting source code. Binary versions of this utility are available for most flavours of Unix and the Windows platform. For more information, and to download, visit the authors website at <http://www.stack.nl/~dimitri/doxygen/index.html>.

CVS Directory Structure

The directory structure maintained in the CVS repository (show in Figure 2) will only contain the source files (including game graphics and maps) that are required. The binaries, or anything constructed from these sources will **not** be kept in the CVS repository.

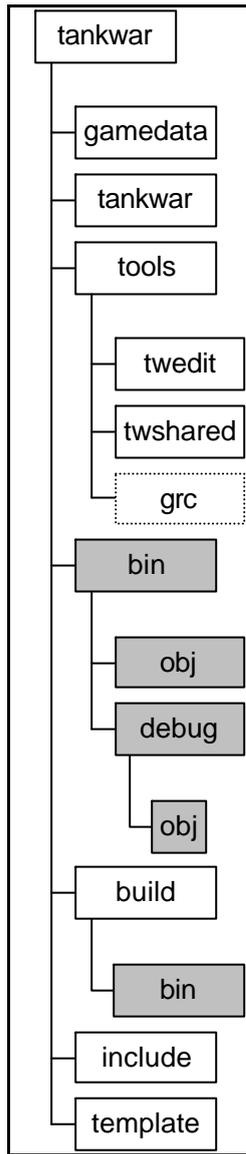


Figure 2 - Directory Structure

The directories marked in light grey are not stored in CVS – these directories contain binaries for tools or binaries built by the build process.

The contents of the directories are described in Table 2. Please be careful **not** to check in the greyed directories. It would be useful to configure a filter in WinCVS to exclude all binary (*.EXE, *.OBJ, *.PCH) files during check-in and check-out operations.

Directory	Contents
/tankwar	The root tankwar source directory.
/tankwar/gamedata	Contains the game images and map
/tankwar/tankwar	Source for the TankWar game exec
/tankwar/tools	Source for miscellaneous tools
/tankwar/tools/twedit	The TankWar game/map editor sou
/tankwar/tools/twshared	Source for the shared library.
/tankwar/tools/grc	Source for the game resource comp
/tankwar/bin	Non-debug executables
/tankwar/bin/obj	Object files generated by the build
/tankwar/bin/debug	Debug executables.
/tankwar/bin/debug/obj	Object files generated by the build
/tankwar/build	Build tools (Makefile etc)
/tankwar/build/bin	Additional build tools (doxygen etc)
/tankwar/include	Shared include files.
/tankwar/template	Template source files.

Table 2 - Directory Structure