

CPLAT Reference

Installing CPLAT

Copyright © 2005 Ken Stahlman.
All Rights Reserved.
Web site: http://ksoft.net/cp_home.htm
Revised - 9/21/05

Table of Contents

Introduction	3
Folder Structure.....	3
Compilers	4
CodeWarrior	4
CW IDE Global Source Trees	4
XCode	5
Visual.NET	5
Whats Installed	6
Naming Conventions.....	6

INTRODUCTION:

When installing CPLAT, you need to insure that the following is done in order for the projects to be able to find the files. All projects are set up to find the files in relation to the CPLAT global source tree folder.

- 1) Create a folder to hold the CPLAT files, in the screen shot below it is Release.
- 2) Set up CodeWarrior global source trees.

FOLDER STRUCTURE:

When installing CPLAT, you need to create a folder, and then create or place in it the following directories which will contains the files needed for CPLAT development. The screen shot below shows a typical installation.

CPLAT (*Release*)

Builds

Documenntation_Generated

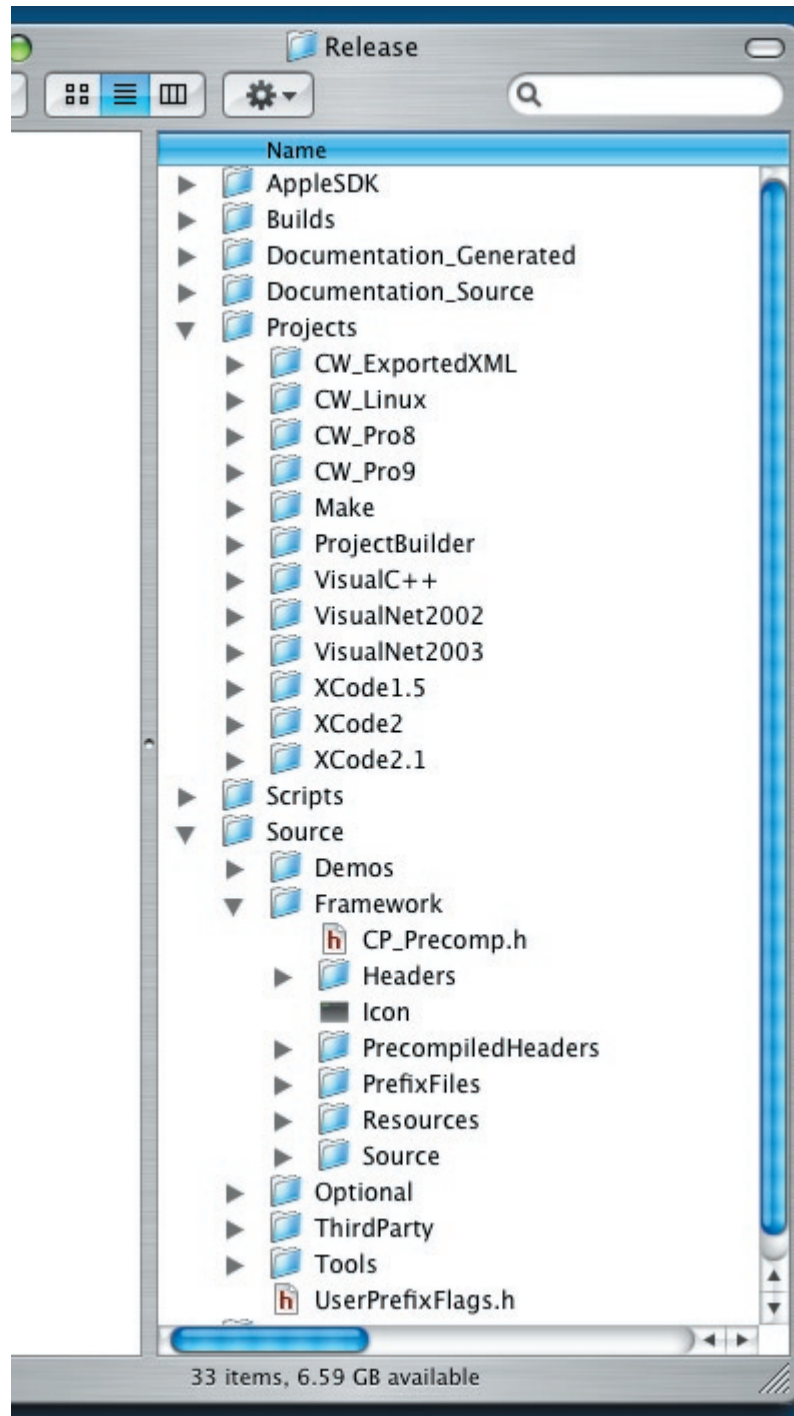
Documentation_Source

Projects

Scripts

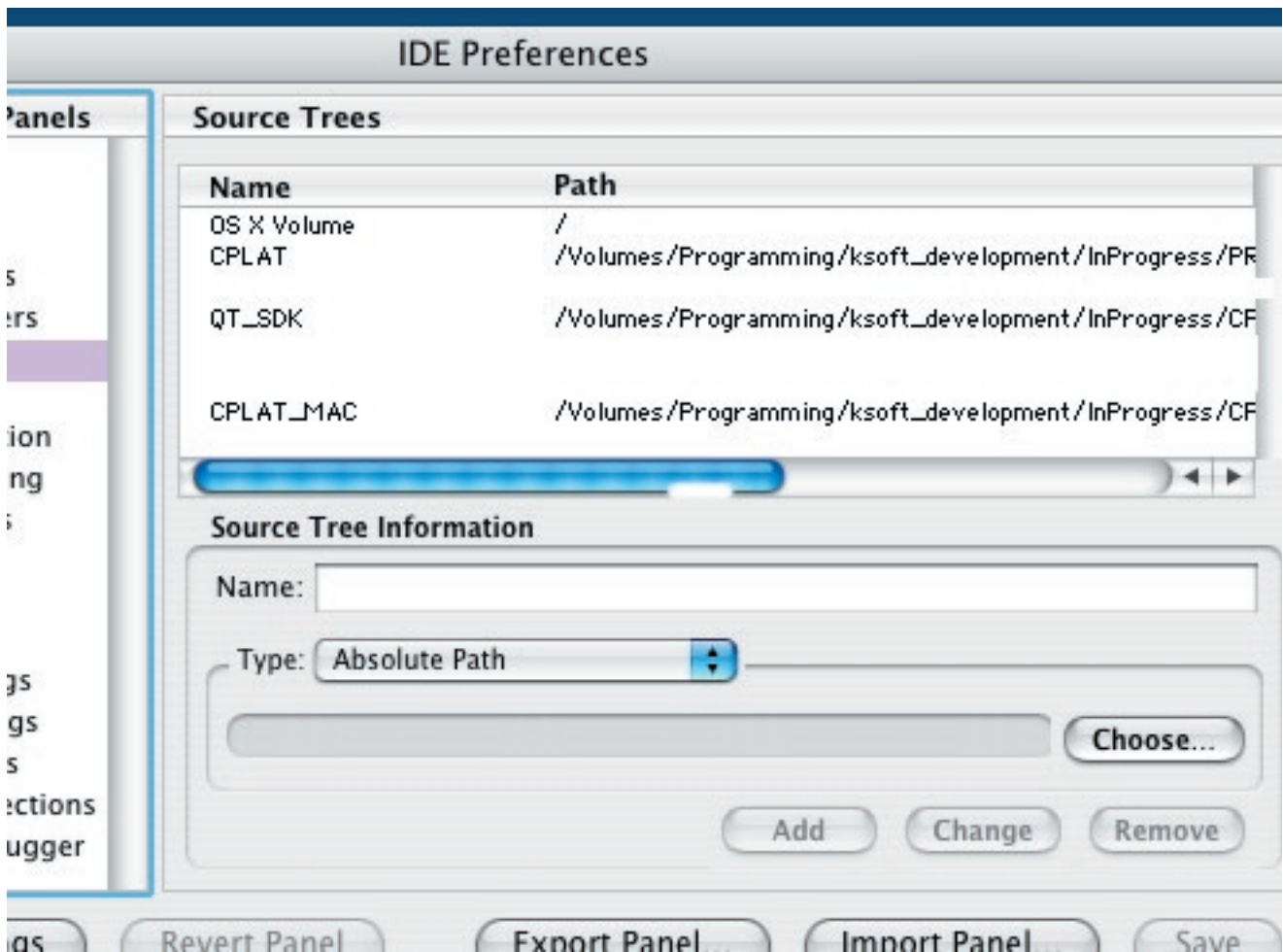
Source

NOTE: The folders shown above in italics, *Documenntation_Generated*, *Documentation_Source* and *Scripts* are all optional. The only folders needed are: Builds, Projects, Source.



COMPILERS:

CODEWARRIOR:



In the Metrowerks CodeWarrior IDE you need to create some global source trees as shown below:

CPLAT

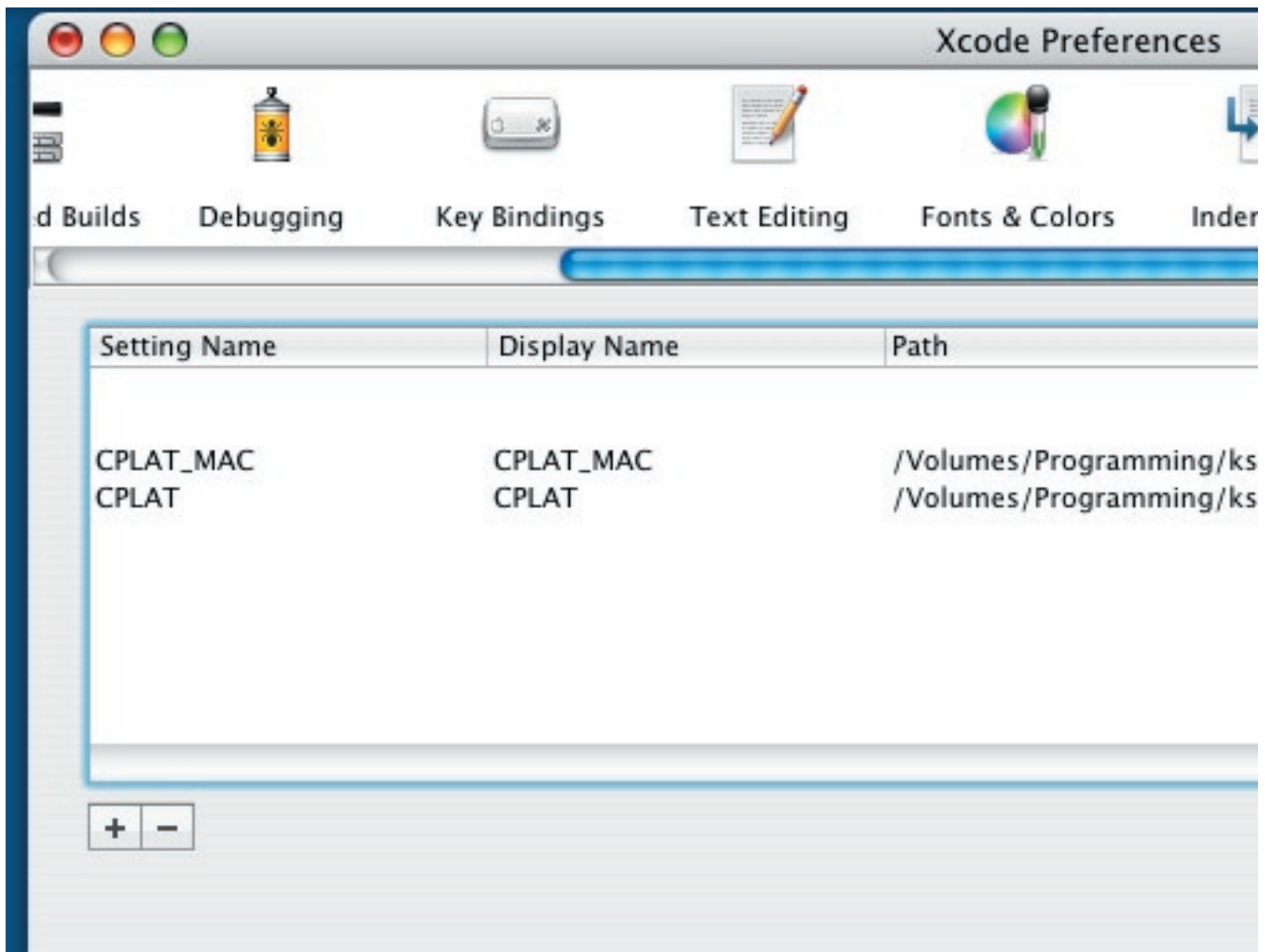
This is the main source tree that points to the outer folder of your CPLAT installation. All projects are set up to be relative to this path, and assume the subfolders are set up and named as shown above in IDE Preferences screen shot.

QT_SDK

The QT_SDK points to the QuickTime SDK, and this folder is only needed to do QuickTime stuff under Windows.

XCode:

In XCode, like Codewarrior, you need to setup a global source tree.



VisualNET:

Both Visual.NET versions should build and run without any special setup needed.

WHATS INSTALLED:

Below is a short description of the contents of each of the folders installed in your CPLAT setup.

Framework

This is where all of the CPLAT framework files are located. Inside the framework folder you will find the following subfolders:

These folders contain the following items:

Headers - this is the folder that contains all of the CPLAT framework headers, and is broken down into Core, Defines, Documens, Files_Stream, Mac_only, Linux_only, UI and Windows_only to help organize a little by catagory of useage. You will find the source files organized in the same sub folder structure.

PreCompiledHeaders - this folder contains all of the pre-compiled headers used in CPLAT. Any time you change the version of the CodeWarrior IDE, be sure to rebuild the files in here.

Resource - fhis folder contains all of the common resources used in the framework. There is a sub folder for MacOS, Windows and Linux development.

Source - this contains all of the source files to CPLAT, and is organized in 3 sub folders, the same as the headers folder.

Third Party - this folder contains files and folders of some freely available third party libraries used by CPLAT such as:

expat - this is the XML parser used by CPLAT.

MacOS - contains MacOS only libraries.

Apple - FileStuff - FSCopyObject source for performing files copies under OSX.

Windows - files used only under Windows. CPLAT only.

UserPrefixFlags.h - this file is a common include file used by all of the precompiled headers to bring in common defines that a user can tailor a CPLAT application with. This is the file you will edit to customize how you want CPLAT to act. Some of the flags that can be set are: enable Unicode support, use QuickTime/QTML under Windows, enable pane drag and drop, enable XML streaming support.

NAMING CONVENTIONS:

Throughout CPLAT, I've tried to use a consistant naming convention as follows:

nd -- non debug versions

d -- debug versions

M -- when used in a class name, these classes are designed as a mix-in class

TARGET_OS_MAC -- used for MacOS only stuff

TARGET_OS_WIN32 -- used for Windows only stuff

TARGET_OS_LINUX -- used for Linux only stuff

