

目录

前言	1
1. 获取 Opencpn 源码	1
2. 指南	1
2.1 操作系统	1
2.2 VC++ 2010 Express	1
2.3 Cmake	1
2.4 Gettext	1
2.5 WxWidgets	1
2.6 构建 OpenCPN 源代码	2
2.7 Visual Studio 编译解决方案	3
2.8 通过 VisualStudio 运行调试版	3
3. 后语	3

前言

本文章档着重于 windows 下 opencpn 二次开发环境的快速部署。

1. 获取 Opencpn 源码

通过以下命令得到最新测试版

```
"git clone -b beta31_stable https://github.com/OpenCPN/OpenCPN.git"
```

也可直接通过链接  [Download ZIP](#) 获取源码压缩包。

2. 指南

2.1 操作系统

在开始编译之前, 如果使用 XP 系统, 则需要更新 Service Pack 3。

2.2 VC++ 2010 Express

访问 <http://www.microsoft.com/visualstudio/en-us/products/2010-editions/visua...> 下载

运行安装程序, 连续单击完成安装向导, OpenCPN 不需要 Silverlight 和 SQL Server, 所以无需安装这两项, 这样可以节省安装时间和内存。

Service Pack 1

我们有必要进行一切更新。好像如果你没有安装 Visual Studio 2010 Service Pack 1, 你就无法将 WX projects 转换成 VS 2010 格式。打开 Help 菜单, 选择 Check for Updates。我们有必要重复进行这项操作。

2.3 Cmake

访问 <http://cmake.org/cmake/resources/software.html> 下载

在安装的同时, 我们建议安装程序将 Cmake 添加到系统 path 变量中。

2.4 Gettext

通过 <http://www.poedit.net/> 安装 Poedit——其 gettext 是可用版本中最好的。

将 Gettext 添加到你的 path 中——如果你安装了 Poedit, 右键我的电脑 (**My Computer**), 选择属性 (**Properties...**) 在高级页 (**Advanced**) 上点击环境变量 (Environment Variables) 按钮, 将 *C:\Program Files\Poedit\bin* 添加到 *PATH* 系统变量中。

2.5 WxWidgets

http://www.wxwidgets.org/downloads/#latest_stable 下载 wxMSW-2.8.12-Setup.exe 并进行安装。

必要修改

打开你安装和解压 wxWidgets 的文件夹, 编辑文件 include\wx\msw\setup.h, 进行如下设置:

```
#define wxUSE_GRAPHICS_CONTEXT 1
#define wxUSE_GLCANVAS 1
```

编译 wxWidgets :通过命令行 (速度更快, 强烈建议采用)

启动 Visual Studio 2010 Command Prompt, 打开: cd C:\\${WXDIR}\build\msw

通过以下命令使用 makefile 进行编译:

```
nmake -f makefile.vc BUILD=release MONOLITHIC=0 SHARED=1 UNICODE=1 USE_GDIPLUS=1 USE_OPENGL=1
```

如果你想完全调试 OpenCPN, 创建 wxWidgets debug dlls

```
nmake -f makefile.vc BUILD=debug MONOLITHIC=0 SHARED=1 UNICODE=1 USE_GDIPLUS=1 USE_OPENGL=1
```

将解决方案转换成 VC++ 2010 格式

打开: C:\\${WXDIR}\build\msw\wx_dll.dsw, 对于出现的所有项目转化的问题, 回答 Yes。

保存解决方案 (Ctrl-S), 你可以忽视警告。

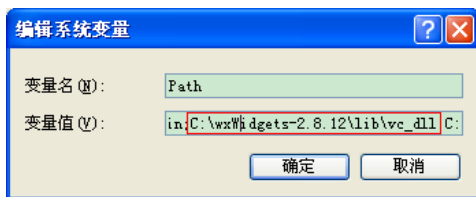
在 project **core** 的属性中, 为 **DLL Unicode Debug** 和 **DLL Unicode Release** 配置, 将 **GDIPlus.lib** 添加到

Configuration Properties -> Linker -> Input -> Additional Dependencies.

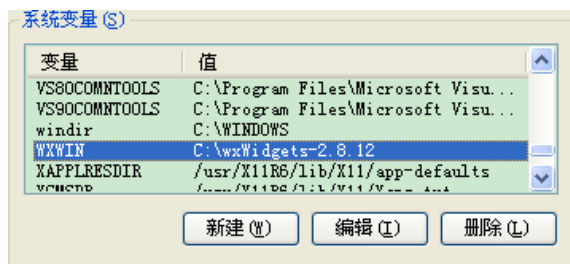
Odbc 无法完成编译——可以忽略, 因为 OpenCPN 不会使用

使 wxWidgets 操作更简单 (建议使用)

你应该把 `C:\${WXDIR}\lib\vc_dll` 添加到 `path` 环境变量中。完成添加后,你应该重启和运行该程序(`cmd.exe`, `cmake-gui`, `VisualStudio` 等等) 确保它们“看到”已经变化的环境变量。如果你无法确定, 重启 Windows, 一切都会完成设置。如果你没有这样做, 日后运行调试版本会遇到问题。



如果 Cmake 没有找到 wxWidgets 安装, 试着创建另一个叫做 `WXWIN` 的环境变量, 值为 `C:\${WXDIR}` (比如 `C:\wxWidgets-2.8.12`)——它会帮助 Cmake 找到各种库。



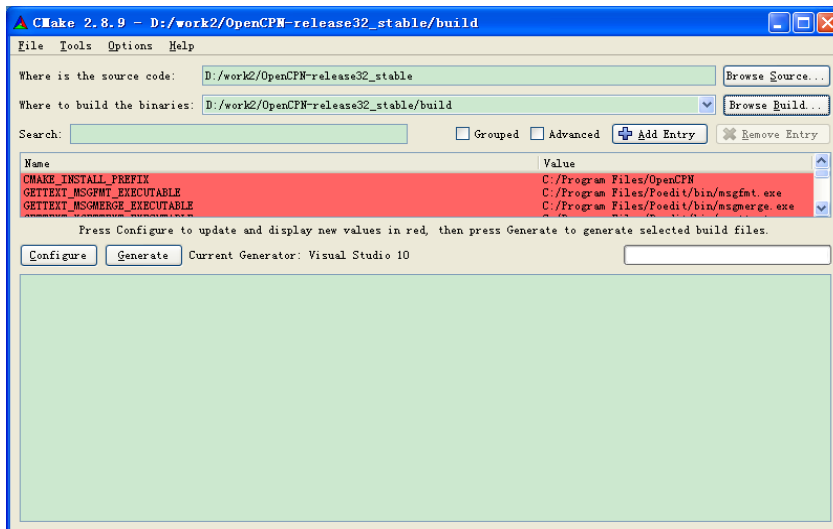
最后, 再次提醒你不要忘了重启并运行参与构建的程序。

2.6 构建 OpenCPN 源代码

在顶层代码目录下创建一个叫做 `build` 的目录

从开始→程序→Cmake 2.8 (Start menu → Programs → Cmake 2.8) 运行“CMake (cmake-gui)”

填写源代码, 创建目录



点击 `Configure` 按钮

如果让你选择生成器, 选择“Visual Studio 10”

如果提示信息为红色并且生成 (`Generate`) 按钮无法使用, 请在此点击 `Configure` 按钮

忽略 `GTK2_GTK_INCLUDE_DIR-NOTFOUND` 和 `wxwidgets_wxrc_EXECUTABLE_NOTFOUND`

点击生成 (`Generate`) 按钮

解决方案和项目文件需要建在 `build` 目录下

2.7 Visual Studio 编译解决方案

打开 Cmake 产生的解决方案 (build/OpenCPN.sln)

编译这个解决方案或单个项目

在编译任何插件之前要先编译 OpenCPN 项目 (在配置安装中进行修复)

如果你想进行调试, 别忘了将 OpenCPN 选为启动项目 (start-up project)。

2.8 通过 VisualStudio 运行调试版

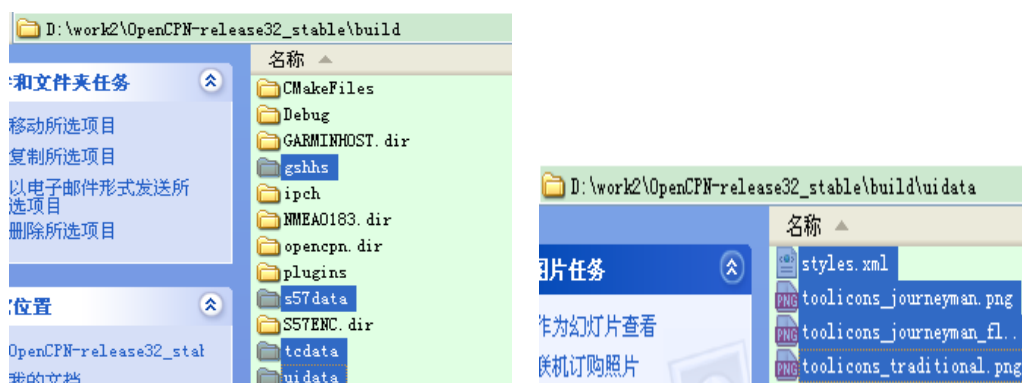
为了能够通过 Inside VisualStudio 运行 OpenCPN, 你需要将以下文件夹从源代码树的 data 子文件夹中复制到构建 (build) 文件夹中:

gshhs, s57data, tcdata.

你需要在构建 (build) 文件夹中创建一个 **uidata** 子文件夹, 然后将以下文件从 src\bitmaps 复制到该文件夹中:

styles.xml, toolicons_journeyman_flat.png, toolicons_journeyman.png, toolicons_traditional.png

如下图所示:



3. 后语

更多资讯请访问 http://www.opencpn.cn/Developer_Manual.pdf,

http://opencpn.org/ocpn/compiling_source_windows

如有疑问可发邮 cpnsupport@rochern.com,

也可在 <http://www.landho.cn/forum-198-1.html> 发帖讨论。