

## **MacZoop 2.5 for MPW Users**

MacZoop, the small yet very capable Macintosh application framework written by Graham Cox, is developed with Metrowerk's Codewarrior C/C++ environment. Since Codewarrior projects can not be automatically translated into MPW make files, those of us using MPW will need to expend a little extra effort to use MacZoop. Thankfully, Graham has worked closely with MacZoop's users to ensure maximum compatibility, making MacZoop generally very easy to use, even under MPW.

At the time of this writing, MacZoop 2.5f4 was latest version available. 2.5f4 is intended to be the last development version of MacZoop before it is finalized. MacZoop 2.5 is also the first version to utilize Graham's new Views object hierarchy, and I think that you'll be pleasantly surprised.

Everything in the Minimal and Demo projects has been tested with MPW 3.6 and Universal Headers 3.4. All of it compiles, and nearly all of it works. I've also gone through the Tutorial, so I've tested ZPrinter, and that works.

Files not in these projects—Undo, Threads, Networking, and others—may or may not need modification to work with MPW. I also have not attempted compiling for Carbon.

The one MacZoop class that doesn't work correctly is ZExpParser, which does compile and run. However, in the Demo Project, ZGraphWindow will always display "Oops, syntax error in expression." This is because ZExpParser is not working correctly with MPW just yet. This is only a serious error if your project requires this expression parser.

### ***Before Compiling any Project...***

The MacZoop.proj.rsrc file needs to have a SIZE resource added to it. In the MacZoop manual, this is done from within CodeWarrior; for MPW, this has to be done within ResEdit or by adding a resource text file to the project and Rezing it. The options should be:

- ID "-1"
- acceptSuspendResumeEvents
- canBackground
- doesActivateOnFGSwitch
- is32bitCompatible
- isHighLevelEventAware
- localAndRemoteHLEvents
- isStationeryAware
- Min Size: 512000 (500kB)
- Size: 1048576 (1MB)

### ***The Minimal Project***

The Minimal Project shows you the basic capabilities of MacZoop, and allows you to check that the core MacZoop source files are compiling correctly. Before trying to compile with the supplied make file, you must first change the MacZoopDir variable, defined near the top of the make file. It is currently pointing to the root MacZoop 2.5 directory on my computer ('Macintosh HD:Applications (Mac OS 9):Programming:MacZoop:MacZoop 2.5'). Since you probably haven't set up the same folder names, you'll need to modify this so that MPW will find all the necessary files.

### ***The Demo Project***

The Demo Project shows off some MacZoop's more advanced functionality, and will probably get you hooked on MacZoop, just as it has the rest of us. Again, you'll need to change the MacZoopDir variable in the makefile to point to your MacZoop installation.

### ***Tutorial Project***

For the Views section of the tutorial, you will need to add "More Classes:Basic Views" to the "includes" list. (see Common Errors, below.)

It helps to remember to add your new source files to the make file. (see common errors, below.) These include ZHelloWindow.cpp, ZHelloView.cpp, ZViews.cpp, and ZPrinter.cpp, as you work your way through the tutorial.

### ***Building Your Own MacZoop Makefiles***

To build a MacZoop project in MPW, follow the normal procedure for creating an MPW project.

- Select the Application option.
- Select the PowerPC option
- MacZoop can be built as either a Carbon or Classic project.
- Add all .cpp files in the "Required Classes" folder, and whatever other sources are necessary for your project, and add MacZoop.proj.rsrc or Carbon.proj.rsrc, as appropriate, from the {MacZoopDir}Projects:Common directory.
- Click on the "Additional Libraries" button and select
  - MacOS libraries
  - QuickTime
  - Navigation Services
- Click on the "Include Paths" button, and add the following search paths:
  - i "{MacZoopDir}Headers" ∅

```

-i ":" ∂
-i "{MacZoopDir}Required Classes:Carbon Only" ∂
-i "{MacZoopDir}Required Classes" ∂
-i "{MacZoopDir}More Classes:Streaming" ∂
-i "{MacZoopDir}More Classes:Support" ∂
-i "{MacZoopDir}More Classes:Undo" ∂
-i "{MacZoopDir}More Classes:Views:Standard UI Views:Dialogs" ∂
-i "{MacZoopDir}More Classes:Views:Standard UI Views:"

```

This is enough to get the minimal project (required classes) compiling. For projects that include other MacZoop files, you may need to include additional search paths.

In the above, I've assigned {MacZoopDir} to the root MacZoop directory at the top of my makefile (in my case, this is 'Macintosh HD:Applications (Mac OS 9):Programming:MacZoop:MacZoop 2.5'). This makes it easier to copy the same makefile for different projects, and to write in this Read Me document.

The “∂” character can be created by pressing option-d.

- Click on the “General Options” button and
- Turn on exceptions.
- Use Type 'APPL' and Creator 'ZOOB' for the supplied projects and tutorial.
- In addition, you'll need to add a few options to the “C Options” and “C++ Options” by clicking on the “PowerPC Options” button:
- In C Options and C++ Options:
  - w off turn warnings off, to simplify your life
  - alloca for support of alloca() in the full project, ZWASTEWindow, and others.
- In C++ Options only:
  - bool on for bool support

### ***Common Errors***

Errors you run into in building MacZoop projects will generally be one of three kinds, each one common to all MPW projects, and each one easily solved:

1. Unable to open input file "xxxx.h". In this case, you need to add the header's folder to the Include search paths (-i "{MacZoopDir}some path"). This is a fairly common error to encounter in MacZoop, since Graham Cox has been authoring it using CodeWarrior, which includes all of the standard MacOS headers in one pre-compiled header. For the supplied Projects and the Tutorial we've fixed this, so you shouldn't run into it at first. However, when

you strike off on your own and start working with other source files, then you'll be more likely to encounter this error.

2. No prototype for some function, some constant is undefined, or MrC(pp) expects a comma between a variable type and the declared variable. MPW tells you which file is creating the problem. In this case, the source file is missing the necessary header. Use your editor to find which file declares the (function/data type/constant), and add it to the offending source file (I use Alpha, which can do a multiple file text search in some directory or directory hierarchy; MPW's editor and BBEdit also have this capability).
3. Link errors, such as "Reference to unresolved symbol ".ZPrinter::ZPrinter(ZWindow\*)". In this case, PPCLink is reporting that a function is called but was not compiled in the project. You either have not included a necessary source file in the make file (under "Source Files," "Object Files," or "Necessary Dependencies"), or you need to add a library to the makefile's list of libraries. You can use your editor to find which file (source code or header) contains the missing functions and add the appropriate file to your project.

The most important thing to remember is: enjoy yourself! MacZoop makes authoring Macintosh programs far easier.

*Tom Hopper*  
*February, 2001*