

some WavesWorld IB palettes (2.3)

266379_PixelRule.tiff ↪

send mail to wave@media.mit.edu if you have questions.

38470_CheckMark.tiff ↪ This software has been tested on 3.2 and 3.3 running on black, white, and beige (and a teeny bit on SPARC) hardware. It requires the **shipped, final** version of **3.2** (or greater) on all hardware platforms. Earlier versions than that are **not** supported.

968699_CheckMark.tiff ↪ Note that if you are running **3.2** on white or beige hardware, you may not be able to use the WW3DKit palette, although you can use the BuildaDudell app instead. Basically, if you are running in 24 bit mode on white or beige and are running 3.2, you'll crash the WindowServer. This is NeXT's bug, and they've **fixed it in 3.3DevPR1**; hopefully it will stay fixed in 3.3 golden...

983608_CheckMark.tiff ↪ Note you must use **gnutar** to untar this software, as it has very long pathnames that tar will lose on.

812229_PixelRule.tiff ↪

Hi folks. Welcome to the 2.3 release of some of the IB palettes I've written for my PhD system, WavesWorld.

I wrote the core version of this some of this code two summers ago (1993), and in addition to trying to finish my PhD (which uses this software), I've fixed scads of bugs and actually fleshed out most of the stuff I designed then. It's still not done yet (what is?), but it's still

pretty, urm, *interesting*, and I think people will find this code instructive, and potentially useful. [382329_PixelRule.tiff ↗](#)

This version still has very sparse documentation (although I do have a pretty reasonable start on a tutorial for the 3D stuff). The more people I think are interested and beginning to use the palettes, the more documentation I can be coerced to write, so send me feedback about what you don't understand. There are plenty of examples, though, and given the nature of the product (a set of IB palettes which work really well in IB's test interface mode), I think people should be able to dive right in.

[628403_PixelRule.tiff ↗](#)

If you do want to dive right in, and you grabbed the "Complete" release, I've already compiled (NIHS) the tcl palette and a simple app to show off the WW3DWell, so here's a quickie get-your-feet-wet-fast intro:

[285255_PixelDottedRulePadded.tiff ↗](#)

.dir.tiff ↗ WavesWorld

- 1** Drag **BuildaDudell.app** into ~/Apps, select the ~/Apps directory and do a cmd-u (update viewers). This makes it so certain icons show up right in the workspace, like *.mdl, *.eve, *.cam, etc.
- 2** Drag the files WWTCLKit.palette and WW3DPalette.palette from the **Executables/** directory into **/LocalDeveloper/Palettes**. If that directory doesn't exist, please create it. These are compiled MAB, so you should be okay whether you're running on black, white, HP or SPARC hardware.

3 Start up IB

5 If you have loaded in the **PAThumbwheel** palette, go into IB's Preferences, unload it, quit IB, and start again. I use objects from this palette and have built them into WWTCLKit, so you can't have both in at the same time. I fixed a few bugs in the objects, so I didn't want to (a) tell you to load in a palette you might not have to use mine and (b) have you using buggy objects. After unloading, quit IB and start it again (since it doesn't unload the objects).

6 Load in **WWTCLKit.palette** using **Tools->Load Palette...**

955780_PixelDottedRulePadded.tiff ↵

473109_.dir.tiff ↵ WWTCLKit IB palette

- 1 Double click on **Examples/TclCalculator/English.lproj/Calculator.nib**
- 2 Select the WWTCLInterp instance in the File's Window.
- 3 Go into test interface mode and change the temperature values, noting how they keep in synch.
- 4 Double click on the "controls" cell on the WWTCLInterp's IB inspector to open up that nib file and inspect the objects in the file.

182414_.dir.tiff ↩ WW3DKit using BuildaDudell.app

- 1 Read **Documentation/WW3DKit/A_IntroWW3DKit.rtf** and give it a go. The source files that accompany it are in that directory. All of the examples can be done using the already compiled **BuildaDudell.app**, or if you're not on a real 24 bit machine, or you're running 3.3, you can try stuff from inside of **IB**'s test interface mode. Using BuildaDudell.app can be easier than uses IB's test interface mode for both the tcl stuff and the 3D stuff, because it's more "NeXT-like", i.e. you can double click on stuff and launch it, you can have multiple documents open, etc.