

some WavesWorld IB palettes: tcl & 3D

266379_PixelRule.tiff ↗

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

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 started writing this code two summers ago, and have continued to hack away at it these past two years. I'm still releasing it as matriculateware. Basically, for each palette of the two you find useful, I think \$15 (each) is fair. If you find yourself not just using the palette but also find the source code useful, I think that's worth another \$15 (again, each). Any money goes to keeping me eating until I graduate, and would be greatly appreciated. Y'all can send checks to:

Michael B. Johnson
29 Radcliffe Rd.
Sommerville, MA 02145

If you pay, I'll answer your questions and send you mail when I write new stuff related to these palettes, and keep you apprised of updates and bug fixes.

815996_PixelRule.tiff ↗

Legal Notes

The 1.0 version of this code was designed and written last summer out in CA. Various bits of debugging and mild extending have gone on since then, but the bulk of the code (and all of the design) was done when I was not employed by the MIT Media Lab. Therefore, the code in the 1.0 release is **copyright 1993, Michael B. Johnson**. Feel free to use this code in any software you write which is for personal use that does not directly contribute to you making money.

If you use this code to make money (i.e. consulting with it) or you want to embed some of these objects (or large chunks of them) into a commercial app, you need to get written permission from me to do so. I'm sure we can come up with reasonable terms that both parties can

agree with; I just need to get through this last year of my PhD and then I can get a real job, I'm not expecting to make significant oodles of cash.

In the 2.0 release, the tcl palette has stayed largely unchanged, while the 3D palette has had a fair amount of restructuring, although the bulk of the new code is just the implementation of last summer's design. If you want to use this stuff for anything vaguely commercial, you'll have to talk to the MIT Media Lab, as the 2.0 release of this software is **copyright Michael B. Johnson & the MIT Media Lab**.

I'm bundling source with this because I think there are a lot of interesting things people can learn by looking at this code, and because I find the notion of software which doesn't come with source and doesn't have a support staff associated with it is, well, kinda pathetic. Please don't make me regret this by ripping me off. Please pay the shareware fee if you find this stuff useful, and negotiate a deal in good faith with me and MIT if you want to use the code in a commercial way.

To reiterate: original 1.x release is copyright 1993 Michael B. Johnson. The 2.0 release is copyright Michael B. Johnson and the MIT Media Lab.

Also, the WavesWorld logo: 581289_.dir.tiff ↵, and it's various derivations are copyright Michael B. Johnson (designed by Elizabeth Glenewinkel), and should not be used without permission. We worked very hard at coming up with icons for WavesWorld (a system which is **much bigger** than just these two palettes) and have tried to come up with a unique, consistent look. Your respecting of the imagery bundled with this package will help us achieve that.

Thanks.

382329_PixelRule.tiff ↵

This version has (of course!) very sparse documentation. Part of the problem is the general purpose nature of these objects. The more people I think are interested and beginning to use the palettes, the more documentation I can be coerced to write, so send my 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 ↖

These palettes go reasonably far in addressing some problems I've always had with IB and the fact that I wanted to use IB more and the linker less. Also, the 3D palette finally addresses many of the problems I've had with animation systems. It begins (well, just grazes, really) some modeling issues and hands all rendering over to the greatest thing to happen to Computer Graphics since Ivan Sutherland decided to go to MIT; the RenderMan interface.

There are two palettes included in this release, which allow access to the two software kits I've written: **WWTCLKit** and **WW3DKit**.

955780_PixelDottedRulePadded.tiff ↖

679742_.dir.tiff ↖ **WWTCLKit IB palette**

This is my latest take on how to integrate **tcl** (pronounced "tickle"), the embeddable tool command language from Berkeley, into the NeXTSTEP development environment. For more info on tcl, look in `./Source/tcl/doc`, read the newsgroup `comp.lang.tcl`, and ftp to `harbor.ecn.purdue.edu` and see all the cool extensions people have written for and with tcl.

There are many different ways to integrate tcl into NeXTSTEP; this is just one way. This palette really isn't intended to be all that useful by itself, although take a look at the TclCalculator example, which is a pretty cool example. I really designed it to be used with the other palette included here: the WW3DKit. Currently, the objects in the WW3DKit only run on top of NeXTSTEP, but they are designed to port to other environments fairly painlessly. Since I started running NeXTSTEP on an HP/735, this part of things has gotten much less interesting, but when I graduate in a few months, it will suddenly become much more interesting...

891827_PixelDottedRulePadded.tiff ↖

715065_.dir.tiff ↖ **WW3DKit IB palette**

The WW3DKit's IB palette only has one visible object on it: the WW3DWell. A WW3DWell is a UI element that looks like a color well, except that it's got some 3D text in it. As you might expect from its name, you can drop a .rib into the WW3DWell and see it. Also, like you might expect, you can click the edge of the WW3DWell and get a control panel (like the NXColorPanel that pops up when you click the edge of a NXColorWell).

The idea of the WW3DWell is that you want to integrate 3D into your app, but you don't really want to learn the 3DKit. Eventually, perhaps, but for now, you just want to be able to integrate 3D into an app with a minimum of fuss. There's much, much more power available in the WW3DPalette than that, but that was the easy home run part.

The WW3DPalette includes a full blown 3D modeling language called **eve**, which is based on tcl (with full RenderMan binding and a few other commands). I've included some simple example showing some the the canonical RenderMan quadrics, as well as some neat, more complex examples of a robot and some eyeballs.

168179_PixelRule.tiff ↗

Installing

Unfortunately, there doesn't seem to be a really standard place to install palette libraries and their headers. In previous versions I had decided to follow the lead of NeXT's mini-example TTools, but for this release (and after taking a look at 3.2), I've decided to organize things a little more sanely (I think):

```
palettes go in /LocalDeveloper/Palettes
libraries go in /LocalDeveloper/Libraries
headers go in /LocalDeveloper/Headers/WavesWorld
doc go in /LocalLibrary/Documentation/WavesWorld
```

The source should probably go in /LocalDeveloper/Source, to mirror NeXT's new layout for the gnu source, but I don't install them for

you.

If you have strong religious convictions about other places they should go, make sure you hack the various examples' **Makefile.postamble** files to reflect your changes.

The Examples should go under **/LocalDeveloper/Examples**, and it's probably best if you kept them in their own subdirectory.

Also note that any of the Examples which actually can be compiled (i.e. **TclCalculator**) have links in their **English.Iproj** directory to various nibs in the two palettes. If you move things around, those links might not stay valid. This is easy to see, as when you bring up the PB.project file and look at Interfaces those will be gray, not black. You can either copy the nibs into the project or make new, valid links. I like the links because then changes are automatically reflected in the examples, as opposed to having to **remember** to copy them into all the right places when I change something...

694720_PixelDottedRulePadded.tiff ~

- 1 In a Terminal window, cd into **WavesWorld/Source/Libraries/tcl** and type **make**. This will compile **libtcl.a** in MAB form (although you might want to change it's notion of MAB, depending on what machine you're on), **mv** it to **/LocalDeveloper/Libraries**, and **cp tcl.h** to **/LocalDeveloper/Headers/WavesWorld**. Important note: I actually had to modify the base tcl distribution, so you really better use mine until tcl7.5 comes out.
- 2 Start up ProjectBuilder and open the project from **Source/IBPalettes/WwtCLKit**. go over to the "Build" section and select "install". Build. If you want to build the palette MAB, make sure you have selected that in PB's Preferences **before** you open the project. This will build the palette, it's accompanying library, and the precompiled headers for the palette, and install everything in the appropriate places.
- 3 Start up Project Builder and open the project from **Source/IBPalettes/WW3DKit**. go over to the "Build" section and

select "install". Build. If you want to build the palette MAB, make sure you have selected that in PB's Preferences **before** you open the project. This will build the palette, it's accompanying library, and the precompiled headers for the palette, and install everything in the appropriate places.

- 4 So that you get the right icons for things, I've included a little app (**Source/Applications/BuildaDudell**) that is kind of the minimal useful app built out of the two kits. It also claims ownership for a bunch of extensions (.mdl, .mdlTemplate, .eve, .wwcam, .cam, etc.) and make it so that the right icons show up. If you install it (start up PB, click install, then open up your ~/Apps directory and click cmd-U), you should see the right icons when you start looking around in the various example directories.

95765_PixelRule.tiff ↗

Acknowledgements

717630_CheckMark.tiff ↗ Thanks to Dr. John Ousterhout and the fine folks on comp.lang.tcl for making tcl a frighteningly useful tool. Also, the most interesting ideas in the WWTCLKit palette are inspired directly by Tk, the X windows toolkit based on tcl, also written by Dr. O.

847914_CheckMark.tiff ↗ Thanks to Thomas Burkholder and Jeff Martin of NeXT, for their TTSwitchView & TTTimer and PAMThumbWheel objects, respectively. Awesomely useful code...

159103_CheckMark.tiff ↗ Thanks to the folks at Pixar that put up with me this summer where I thrashed a lot of these ideas out, especially Dana Batali, Dan McCoy, Tony Apodaca, and Ed Catmull.

522916_CheckMark.tiff ↗ Thanks to Elizabeth Glenewinkel, for a critical graphic design eye and the WavesWorld logo: .dir.tiff ↗ and help on all the other WavesWorld icons (many not included in this palette distribution) and UI.

369141_CheckMark.tiff ↗ Thanks to my advisor David Zeltzer here at MIT for supporting me all these years.

253862_CheckMark.tiff ↪ Thanks to Ian Wilkinson at Canon UK for WW3DText, BugWave, and for banging on WavesWorld.

470622_CheckMark.tiff ↪ Thanks to Mike Hawley here at the Media Lab for early palette help.

608697_CheckMark.tiff ↪ Thanks to Bruce Blumberg and Andrew Stone for all their NeXTSTEP hackery help over the years.

383473_CheckMark.tiff ↪ Much thanks go to Larry Gritz, for his awesome renderer and shaders, and for being a fellow RenderMan hacker out here in the wilds of academia.

340562_CheckMark.tiff ↪ A special tip-o-the-keyboard to Charlie, whose prepayment of the shareware fee for these palettes (both 1.0 and now 2.0!) caused me to get off my lazy ass and package them up and make them generally available. You can thank him for the fact that I actually got this out to you. Money talks, and I can be bought...