

According to Hoyle...

Cross-Platform Software Development from a Macintosh Perspective: Runtime Revolution

by Jonathan Hoyle

jhoyle@maccompanion.com

Last September, we began our series on cross-platform software development for the Macintosh http://www.maccompanion.com/archives/september2005/Columns/According_to_Hoyle_1.htm. Over these past several months we have examined many varied cross-platform frameworks and development environments:

- **Qt:** <http://www.maccompanion.com/archives/october2005/Columns/AccordingtoHoyle.htm>
- **wxWidgets:** <http://www.maccompanion.com/archives/november2005/Columns/AccordingtoHoyle.htm>
- **CPLAT:** <http://www.maccompanion.com/archives/december2005/Columns/AccordingtoHoyle.htm>
- **REALbasic:** <http://www.maccompanion.com/archives/january2006/Columns/AccordingtoHoyle.htm>

This month we examine *Runtime Revolution*, the easy to use cross-platform application development environment <http://www.runrev.com>.

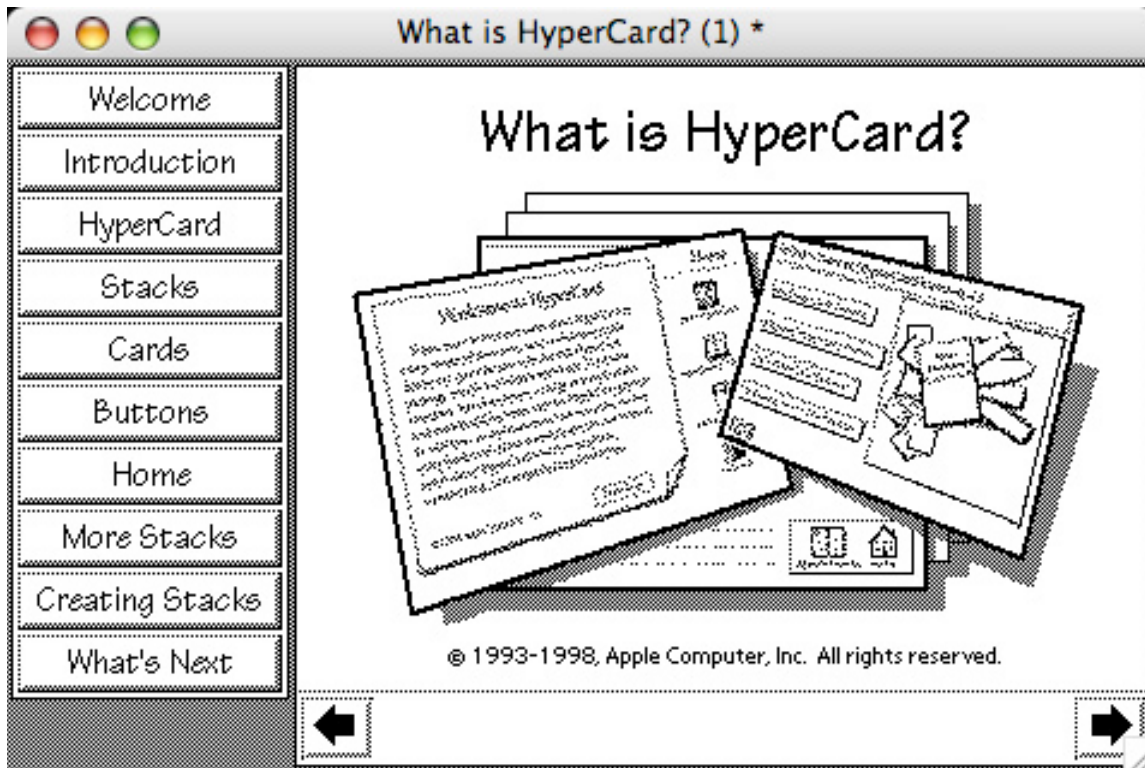
Runtime Revolution History & Overview

Runtime Revolution traces its pedigree back to *HyperCard*, Apple's killer programming environment, simple enough to be used by children and professionals alike. First released in 1987, HyperCard allowed users to mix text, graphics and sound, and put them together in index card like arrangements. Scrollbars, buttons and icons allowed you to link to other cards or to return home. Created by Bill Atkinson (author of *MacPaint*), HyperCard was programmed with a simple language called *HyperTalk*. The user could create HyperCard documents, called *stacks*, which could be run on any Macintosh since HyperCard was bundled free with the OS. It created a huge following, with stacks of every kind being created by fascinated users. The Worldwide Web format is very similar to, and likely influenced by, HyperCard.

Unfortunately, HyperCard is an example of one of Apple's biggest missed opportunities. Apple stopped bundling the full version with the Mac operating system in 1992, giving away only the player. Those wishing to create stacks had to purchase it from Apple for \$99. Apple also showed no signs of interest or support, despite its huge following. The last major upgrade was version 2.0 in 1990. Updates thereafter came fewer and farther between, and grudgingly at that. The final version was version 2.4 released in 1998.

Of course, a user base of that size is not easily ignored, and other solutions became available. *SuperCard* was the first HyperCard clone, adding more advanced color capabilities. There were others, including *HyperStudio*, the open source initiative *FreeCard*, and the outrageously expensive *MetaCard* (\$995). Now along comes *Runtime Revolution*.

First of all, Runtime Revolution is not just another HyperCard clone. RR is a powerful development environment for creating and scripting cross-platform applications. Unlike HyperCard, stacks created by RR are compiled into true applications. You can, however, use RR to open and run old HyperCard stacks (see below). Best of all, the applications you create can be targeted for Mac OS X, Classic Mac OS, along with various flavors of Windows, Unix and Linux. Though priced at a rather high \$299, the software you create is royalty-free.

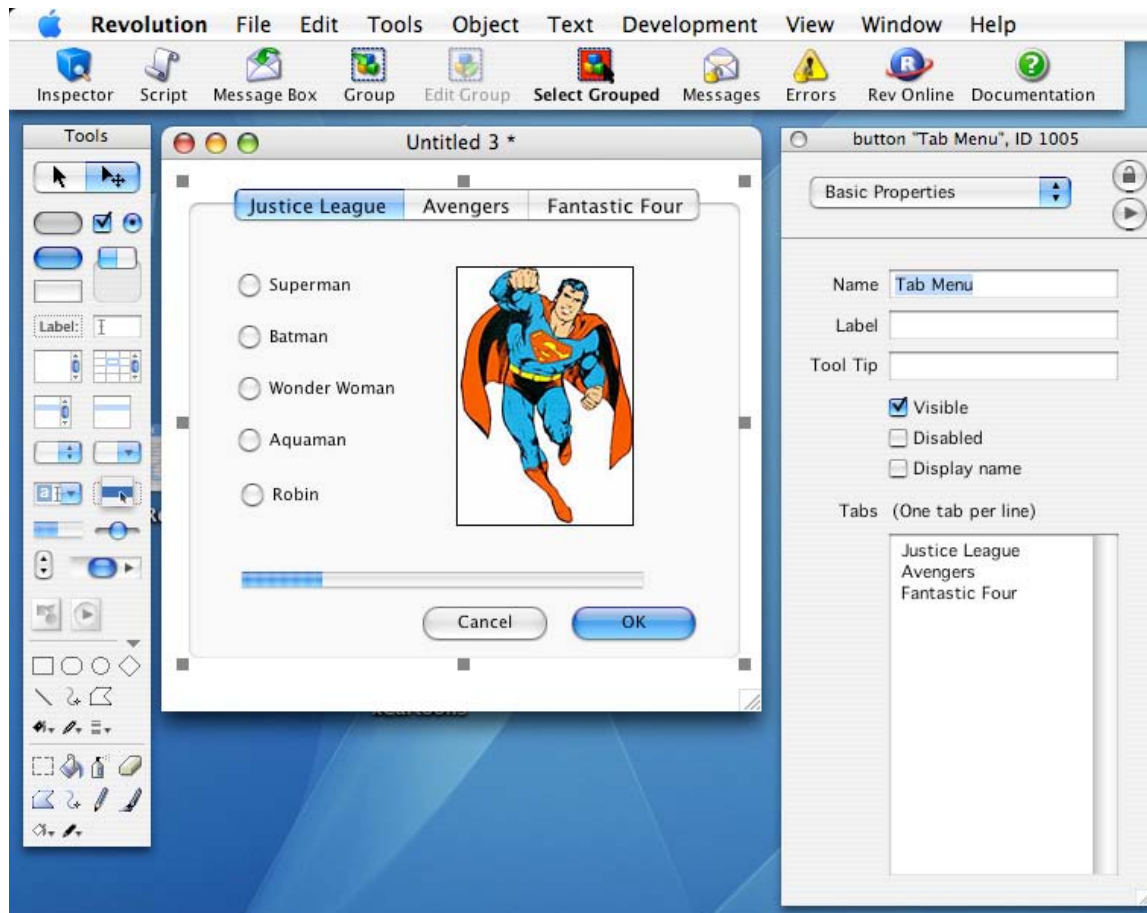


Above is a screenshot of Runtime Revolution opening the basic “What is HyperCard” stack that was installed with HyperCard 2.4.1. Note that this is an Aqua-based Mac OS X window which faithfully represents the original HyperCard look and feel.

Creating Applications with Runtime Revolution

The Runtime Revolution environment appears to be a cross between REALbasic and HyperCard. You begin by creating a mainstack, which you link to other cards. Like REALbasic, you can grab widgets from a palette to drag onto your window, double-click on it to set behavior, etc. Right-clicking on a widget allows you to edit the associated script (written in Revolution's *Transcript* language, an extension of HyperTalk). Script authors will find it easier to build applications with RR than with virtually anything else out there.

Revolution is a much more powerful programming environment than any of its HyperCard-like predecessors. For example, SQL database support is available, with a point and click *Database Query Builder*, which is used to wrap SQL commands. Yet, you can still include older HyperCard functionality, like XCMD's and XFCN's on the Mac OS version.



To test its ease of use, I decided to create a project without reading any of the documentation other than the online help. I began with the project I built for REALbasic last month as my model. As you can see from the screenshot above, it was very intuitive to drag controls and images onto the screen. I did not see an immediate way to create a group box around the radio buttons (as there is in REALbasic), but I imagine that there is a way to do it.

I didn't get much further than building the screens before needing to sit down and learn the *Transcript* scripting language. *HyperTalk* programmers of old will find very little trouble getting up to speed with it, as would modern *AppleScript* programmers. Being a 4GL (4th generation language), Transcript is substantially the same as *HyperTalk*, and quite different than the *Basic* programming language. I found keeping the documentation nearby essential. Even more useful was a very old *HyperTalk* book I happened to still have around.

Runtime Revolution vs. REALbasic

Runtime Revolution is not as powerful as REALbasic, but it is written to be as intuitive and as user friendly as possible. Those who have not been trained in a formal programming language will, without question, find that Runtime Revolution is the best, fastest way to create cross-platform applications. On the other hand, those versed in the *Basic* programming language will likely find REALbasic preferable. At RR's price point of \$299, you can purchase *REALbasic Professional*, making the choice essentially a toss-up.

What about REALbasic's \$99 Standard (non-Pro) edition? Runtime Revolution has an answer to that as well: for that same price, they offer *DreamCard*, a "light" version of RR. DreamCard stacks are not applications in the true sense but rather documents that require the DreamCard player (which is free to download and distribute). In this way, DreamCard is more like *HyperCard* than *REALbasic*. DreamCard uses the same Transcript language and is cross-platform (via the free players available), but lacks certain power features such as SSL and databasing capabilities.

Conclusion

Runtime Revolution is an excellent environment for application development, particularly for students and hobbyists. Script writers will find themselves with the ability to generate cross-platform applications with relative ease. The \$299 price is a bit high compared with similar IDE's, but if you are willing to forgo the actual creation of an application, you can purchase *DreamCard* for a third that price and distribute stacks instead.

Regardless of which direction you choose, creating cross-platform applications comes with ease with Runtime Revolution's line of products. Non-programmers and script writers will not find a better environment with which to work.

Coming Up Next Month: AMPC (Axiomatic Multi-Platform C)!