
Subject: Java or Cocoa w/ IDL: any experiences?

Posted by [Matt Haffner](#) on Tue, 07 Aug 2007 22:00:51 GMT

[View Forum Message](#) <> [Reply to Message](#)

For about ten years, I've based most of the tools for my project using IDL as the workhorse for math/plotting and TCL/Tk as the GUI front-end. I designed a small TCL command set using the RPC method of accessing IDL for this connection. It's worked well for a while (and been quite portable), but changes in TCL/Tk landscape (including abandonment of a few add-on toolsets we use) over the last few years have left me wanting to move on to something else for the GUI front-end. If I stick with TCL/Tk, I'll need a major overhaul anyway...

Most of our group is standardized on OS X now, so I have a few options I've been pondering. I haven't found a lot of information googling around for some of these combinations and was wondering if anyone had experience trying some of these combos out:

* Cocoa/ObjC + callable IDL: I'd assume any IDL graphics are still relatgated to XWindows here? Is this even feasible, especially with event processing? Right now, with TCL/Tk, I use a standard IDL direct graphics window for plotting and user data interaction (clicks, etc.).

* Cocoa/ObjC + RPC IDL: This would probalby be close to what I'm dealing with now, but I'm curious if there are any pitfalls right off the bat.

* Java + IDL: This would obviously be the most portable, but the temptation to try out the Cocoa UI and services is hard to resist. How is this working in practice? I'm mostly interested in Java apps at first, but ION looks interesting to explore for public access to our data down the road.

Thanks for any comments or tips.

mh
