Subject: Re: Other IDL / Mac advantages
Posted by Wolf Schweitzer on Sat, 27 Oct 2001 21:55:01 GMT
View Forum Message <> Reply to Message

Just got back from holiday, interesting note Logan Lindquist sent. His E-mail confirms that, yes, many people think all Mac users are after is some kind of "artsy" thing and really can't, can't, can't let go of that notion. By posting some copied text that says that IDL supports Apple Script on Macs (thanks, thanks), he apparently thinks Mac users are also so stupid they can't read that themselves. Too funny. But then, it escapes many people that Swatch, the Swiss fashion watch maker company, now also sell a full sized "chrono watch" model that weighs only 60 gr. (Aluminium casing); most just talk about the shee shee stuff of that company and buy the 150 gr. Steel model. Shlepp :-)

Functionality may hide beneath good design :-) But anyone saying that IDL on Mac OS 9 looks "artsy" possibly shows themselves off as ignorants, since it may have escaped them that the IDL widgets don't even follow the preMacOSX GUI in any strict way. Instead, IDL on Mac has some general IDL appearance :-) Personally, I would not be the least bit interested in the way any IDL knob or window looks on the widget interface as long as I can get and enter information and run and develop my software.

IDL is brilliant since it saves an enormous time developing solutions based on the way the language is implemented: Applescript is to other tools in interapplication scripting like IDL to other ways of producing analytical results and graphical displays from matrix data: it is possible to do it with other tools, but I they just suck.

Of course, Logan is right, one could possibly try all kinds of systems, OSes or languages any time on any platform, and they all compare pretty well on a "to do" basis. But maybe (?) if you are good and happy doing it with one, it's a pretty good thing to continue doing it with that one if all you want to do is continuing to doing that - which is precisely what I had in mind with my little setup here.

As I am in fact using Apple Script as a specific interface language on a Mac OS, and since OS X has certainly a better memory management than OS 9 does, better stability, the option of fiddling with the console, and is in fact the very Mac OS that is taking over from existing OS 9 - admittedly kind of slow paced, but taking over -, I would very much like IDL for Macintosh to support Applescript under OS X.

Theoretically, it is *possible* for me to sit down and change to any other type of system any time such as trying to find how to create data tunnels to database and other software using Visual Basic or Visual C++ after buying all of these Microsoft products, sure..., and I know that I

could have done that earlier ... however, since I maybe wrongly thought this was a discussion involving me as a CUSTOMER of a PRODUCT (i.e., IDL), and I WANT to continue using the product the way I used it since I evaluated what I wanted to use it for and found it fastest and best to do it that way - which, as far as I know, is my right do to -, I do have a REQUEST in that very role, which, in my eyes, is fair enough to state since I pay my dues.

In other words, I do not want to have to rewrite my code from scratch. Since I am operating under research conditions (software can be interactive, interpreted, open, etc.), IDL / Applescript is the ideal combination. In order to prevent any interference to that from happening, I already did this:

1. I asked Research Systems by E-mail BEFORE I STARTED whether they would continue IDL for the Mac or drop it, in October 2000 (two thousand).

2. They answered right away that they planned to support it until IDL 6.0 and so on.

3. I spent time on working on a project, using IDL, and will continue to do so. I invested in a G4, now I want to see the IDL taking care of Altivec so one sees that the person who implemented it had the manual on the table.

3. Now I am interested to see functional products. Rsinc promised, now I want to see results.

Do I care how any IDL widgets "look" ? No, I didn't, I don't and I won't. I am interested in the numerical and graphical results of any IDL procedures and functions, and having them interactively shoved through to/ from other applications. That's all. As long as the specialists can make IDL interact with OS X over Applescript using X-Windows (about which I would have no idea to understand how that would be established), ...very fine :-) :-) !!!!!

Wolf

>> An important advantage for IDL on Macintosh is its ability to do
>> parameter-passing with Applescript. In order to understand the
>> usefulness of that you need to know what other applications also do
>> Applescript on a Mac OS.
>
>
> This is a true statement, as you can see...

>
>
>
> *****
>
> IDL 5.5 Functional Summary
>
> Development & Programming Tools
>
> Macintosh AppleScript support
>
> *****
>
>
>
> This is also supported under a windows enviroment.
>
>
>
> ******
>
> IDL 5.5 Functional Summary
>
> Development & Programming Tools
>
> Callable Windows DLL
>
> ActiveX control (dual interface)
>
> ******
>
>
>
>> As Applescript would not sell without Mac OS and we are all happy it's
>> also part of Mac OS X, I think that IDL would need to be shipped with
>> Mac OS X - it is just an essential ingredient for the scientific
>> Macintosh community.
>
> I do not know the details of porting stuff that has been written for
> Linux over to OS X but I wouldn't image that they would be much
> different, since OS X is based on a Linux kernel. IDL already supports
> Linux on Alpha's and x86. So the real question is if they already have a
> compilation that is somewhat similar, and there is enough support to
> figure out the details of porting the x86 Linux or the Alpha Linux over
> to OS X, why not start an open sourced development of such?
>
>
>

> I just went and reviewed what the VP of RSI said about this issue.
>
>
>
> "> > Now for the good news. In subsequent discussions with Apple they
> have made
>>> us aware of a commercial X-Windows library for OS X. We are in the
>> process
>>> of evaluating it for use in a native Unix/X-Windows implementation
> of IDL
>>> and ENVI for the Mac OS X platform. This would solve many technical
>> issues
>>> for us and allow us to continue to support the Macintosh platform both
>>> natively and profitably, as it would leverage off our other Unix/X
>>> platforms. The only thing this does not accomplish is providing IDL
> with a
>>> new Aqua UI and widget set."
>
>
>
> So it looks like the big gripe that many of you have is unfounded,
> because RSI can't afford to pay a graphic artist/computer scientist to
> redesign the GUI of IDL OR they don't want to change the look because of
> IDL is a professional product. Thus the redesign would make the
> interface less professional looking. At least they are considering
> porting to OS X. It all then comes down to a usability issue. OS X users
> would have to get used to a slightly different interface. I suggest stop
> complaining and wait to see they decide to support it.
>
>
>
> Their business people are looking at the financial information related
> to how many Mac users buy or renew licenses each year. The decision so
> far seems to be a preliminary one. Business people will change their
> mind if it is deemed profitable for the company.
>
>
>
> About the pricing. If everyone would remember back to economics, the
> quantity/demand curves and the price/cost curves will give us some
> useful tool to analyze their decisions.
>
>
>
> Since it's software the quantity/demand curve operate a bit differently.
> Easy to produce once the code exists. The amount of demand gives us some
> idea of why it costs so much. I agree that they should reconsider their
> pricing structure for educational software[increase demand - lower

> prices], but I also think they are doing a good job of targeting the
> specific group of students that is mostly likely to use the language in
> the business environment. When compared to mathematical programs such as
> MatLab or Mathmatica, I think that those are better targeted towards
> Math majors. I haven't used either extensively. The trick is to get
> future users to become familiar with the language. You do not do this by
> limiting the number of copies that an institution can buy. These future
> users will hopefully equal future dollars spent once they graduate and
> get a job. I am an example! It would be helpful if someone who actually
> bought an educational version to contribute to the price range we are
> talking about.
>
>
>
> That's all I have to say for now. It's lunch time and I'm hungry,
>
>
>
> Logan Lindquist
>
