
Subject: Re: Linux Question

Posted by [Rick Towler](#) on Fri, 18 Feb 2005 17:54:03 GMT

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

Wow, this thread has covered a lot of ground....

Paul Van Delst wrote:

> David wrote:

>

>> Jonathan Greenberg writes:

>>

>>

>>> David: you are the expert on IDL, one thing I haven't figured out is

>>> what does IDL have that Matlab/Octave doesn't?

>>

>> In a word, objects. :-)

>

> I disagree mightily with that, but anyway....

>

> Hasn't matlab been object based from the get go?

As of release 7 it has a mature, fully implemented object API. More so than IDL. Most everything is an object. This probably raises the bar a bit for entry into the object club though. IDL's object API is really easy to pick up. I was a master after a few days of David's tutelage. ;)

> For example, the standard out-of-the-box "plot" in matlab

> has the functionality that I lay awake at

> night dreaming about for IDL plots. You can zoom, change the line

> properties, print, etc. You can probably create legends too (but I'm not sure). This is how a regular old "plot" should work in IDL. I guess

> iTools are the attempt to do this in IDL.

Exactly! And if you take much of what has been said in this thread about "kids these days" and how they don't know anything maybe some of you can see why iTools was born?

The first time I threw up a plot window in MATLAB I was amazed... So is just about everyone else who has used products like these. Given a choice between this and IDL's direct graphics it is a no-brainer.

Sure, iTools have many limitations but I suspect they will be addressed in future releases.

> - Everything in matlab seems to be a string, 4-byte integer, or 8-byte

> real. What's the go with that?

As of Release 7 you have all of your standard types: char, int8-int48, uint8-uint64, single, double, complex. And like I said, they're all objects :)

Other high points?

A great GUI builder where all of the GUI code is hidden, you just deal with a .m file containing the callbacks (Although IDL does have more widgets I am so sick of coding GUIs by hand.) *TIGHT* Java integration:

```
import java.util.GregorianCalendar;
import java.util.TimeZone;
startCal = GregorianCalendar(1601, 0, 1);
gmt = TimeZone.getTimeZone('GMT');
startCal.SetTimeZone(gmt)
```

Holy cow. Is that Java? MATLAB? Who cares, it works? The MATLAB compiler is pretty cool too. The user community website is far more friendly than RSI's. It has a built in C compiler to build MATLAB's version of .dlms!

MATLAB has some major shortcomings for an IDL user: It doesn't pass by reference (well, not the way you want it too) and it doesn't have KEYWORDS (well I don't consider result=myFunc(argIn, 'keyword1', data1, 'keyword2, 'data2') keywords.) The licensing is a PITA. MATLAB is just as quirky as IDL, maybe more so if you are used to IDL's quirks and consider them "normal". And try posting on comp.soft-sys.matlab... With hundreds of posts a day it just isn't as friendly as c.l.idl-pvwave (And there is *no one* posting about their personal adventures or their tennis game!)

So at the very least RSI has some stiff competition and they will need to deliver a quality product and nurture their user base. Michael started an offshoot regarding (more or less) the latter point. As for feature requests, I have found RSI to be very receptive. There is a lot more they could do on the nurturing front though (they still owe me some posters and t-shirts!) The user contrib site is a great idea but the interface sucks and the "newsgroup" on the RSI website is just a mean trick. Those poor users who post there...

-Rick