Subject: Re: IDL versus MATLAB : could you help me ?????
Posted by Mark Hadfield on Sun, 02 Dec 2001 21:28:34 GMT
View Forum Message <> Reply to Message

From: "Roy E Hansen" <ro-edgah@online.no>

Thanks for the thorough comparison Roy.

> MatLab strong points / IDL weak points:
>
> 4) MatLab have functionality for variable number of output parameters.

I'll just elaborate on that one for those who don't know both packages.

Matlab has the ability to pass more than one variable from a function via the return value. Thus the standard function griddata can be called with either of the following syntaxes (from the griddata documentation):

```
ZI = griddata(x,y,z,XI,YI)
[XI,YI,ZI] = griddata(x,y,z,xi,yi)
```

In the former a single array (sorry, matrix) is returned; in the latter three ,matrices are returned, ZI containing z-grid data in both cases, and XI and YI containing ancillary x- and y-grid data.

This is cool.

However there is a deeper reason why Matlab needs this facility and IDL can get by without it: Matlab passes function arguments by value only, whereas IDL passes them by reference (with important exceptions I won't go into here). In other words, IDL can pass information out through function arguments and Matlab can't. So if there were a griddata in IDL (there isn't but it would be easy to write) you would invoke it something like this (remembering that IDL is not case-sensitive):

```
zgrid = griddata(x,y,z,xi,yi,XGRID=xgrid,YGRID=ygrid)
```

Here zgrid receives the return value of the function and xgrid and ygrid optionally receive the ancillary data.

There are pros and cons to the "reference-vs-value" issue. Passing by reference should be significantly more efficient in some cases because it reduces unnecessary copying of data. It is my impression that (for this reason & others) IDL is a much better tool when you want to deal with really large arrays. On the other hand IDL users are occasionally bitten by routines unexpectedly changing arguments (or unexpectedly failing to change arguments). When writing IDL routines you should always be aware that your arguments belong to the caller and not to you.

Mark Hadfield
m.hadfield@niwa.cri.nz http://katipo.niwa.cri.nz/~hadfield
National Institute for Water and Atmospheric Research

--