Subject: Re: Can DLLs Multi-task?
Posted by Rick Towler on Thu, 03 Oct 2002 18:15:45 GMT
View Forum Message <> Reply to Message

"David Fanning" <david@dfanning.com> wrote >

> Here is a question from an IDL user who has no newsgroup
> access.

> ;**
> I am developing an IDL program which have to call some external C
> routines (actually some DLL, because I work on Windows environment).
> I am wondering if it is possible in some ways to call more than one DLL
> at the same time, I.e. in multitasking, so telling to IDL not to wait
> for the DLL return.

I don't think so... I don't think you can tell IDL to return immediately after the call to your external routine since AFAIK (and I don't know much) IDL doesn't really expose any mechanism for callbacks from external routines. And how would you get your data back to IDL?

But...

> For example I need to launch a DLL to continuously monitor some
> temperature sensors, but in the meanwhile I have to run another DLL to
> read an image from a CCD device. I would need to have the temperature
> variables continuously updated by the first DLL so that I can read and
> display their values while getting the image from the CCD device.
> Is all that possible with IDL?
> (I have IDL 5.5)

IDL does expose one callback mechanism that can be used by external routines, WIDGET_STUB and it's associated functions.

I am no astrophysicist (really, the closest I come is Sky & Telescope) so please pardon my ignorance. I am guessing you have this CCD device that is rather slow in acquiring an image. It is temperature sensitive and you need to monitor the temperature of the CCD while it is acquiring the image so you can apply some corrections?

Off the top of my head you could issue a call to acquire from your CCD device. That call would set a one shot timer and then return to IDL. That timer in your external routine would fire and call its callback which would call your CCD acquisition routine. While this is going on, a timer loop in IDL would poll your temp sensors. When your CCD acquisition routine returns (in your external timer event callback), you make a call to

IDL_WidgetStubIssueEvent() and return the memory address of your CCD data.

The trick, which I haven't been able to figure out, is how to get at that data which is located at the address you would return with your call to IDL_WidgetStubIssueEvent(). I have played around with this before, but didn't know what to do with the address I returned to IDL. My guess is that you make another call to your external routine where you pass the address and it returns your data? But like I said, I didn't get it to work.

Anyone care to elaborate on that?

-Rick
