Subject: Re: IDL Help on Mac OS
Posted by JDS on Thu, 25 Jun 2009 16:00:31 GMT
View Forum Message <> Reply to Message

On Jun 24, 2:05 pm, "Kenneth P. Bowman" <k-bow...@null.edu> wrote:
> In article <MPG.24abfa0d38deee0798a...@news.giganews.com>,
> David Fanning <n...@dfanning.com> wrote:
>
>> Kenneth P. Bowman writes:
>
>>> The whole thing is a continuous irritation. Has anyone found
>>> another way to use Help that is not so intrusive?
>
>> OK, scratch the Mac from my list of potential new
>> computers. :-(
>
>> Cheers,
>
>> David
>
> What's so jarring about the IDL Help is that it is so un-Mac-like. Mac applications
> generally coexist peacefully and all operate in pretty much the same
> way. No other Mac application I know of operates like this.
>
> I wouldn't mind having an IDL Help application in the Dock if it actually did
> something, but this weird arrangement where there is an application with no
> real user interface, while the actual help window is in the browser, is very
> unintuitive to long-time Mac users.
>
> Mainly I want to be able to hide the IDL Help 99% of the time and not have it
> in my face every time I switch to my web browser.

Full agreement. An additional irritation is it spawns a new browser window/tab for each and every help invocation. I have IDLWAVE working with it (out soon I hope), and find it a fair bit less compelling than the old Assistant. One advantage I do note is the ability to use bookmarks and full text searching in the browser. If you use Spaces, I find that if you keep one browser window in the space where IDL runs, it will target that one preferentially. Then just remember to close the accumulating tabs occasionally, and the irritation factor goes down.

My understanding of how this thing actually works is:

1. Locally spawn a JAVA-based local web server bundled with IDL/ Eclipse. Wait 10s for it to start.
2. Decompress the HTML files which are no longer bundled as stand-alone, but in some opaque format.

3. Send a command to the browser to query the resource intensive and completely unnecessary local web server, so that it can read a simple HTML file recovered in step #2, and pass it off to the browser.

Compare this to the old method:

1. Use a browser (or Assistant, just a glorified browser) to look at the local relevant HTML file.
2. There is no step 2.

That said, this is Eclipse's help system, not IDL's, so they may feel their hands are tied. How does it work on other OS's?

JD
