Subject: Re: 6.3 reactions?
Posted by Richard French on Tue, 09 May 2006 04:23:57 GMT
View Forum Message <> Reply to Message

I hope that someone who really understands all of this IDL Bridge stuff and who doesn't use objects for every programming task will provide a simple description for civilians of just what this capability can do for the everyday user, and what it can't do. I'm having trouble figuring out whether this is a cool feature that I'd be crazy not to employ, an easy way to try to do parallel processing, or something that would require me to reprogram everything I do to obtain the leverage from multiple processors. So, all of you Ernest Hemingways out there, how about some simple, declarative sentences with an example or two of how this feature works and why it is a Good Thing. Many of us will be grateful!

Dick French

> On Mon, 08 May 2006 07:29:54 -0700, codepod wrote:
>
>> JD, the implementation is simpler than what you outline. The object is
>> only present on the "parent" process, so the actual object instance
>> data is not shared between the processes. The child process has no
>> indication it's being run as a worker: it just receives commands and
>> executes them. IDL variables are transferred between the processes by
>> the bridge using shared memory. The parent process can Get or Set
>> variables in the child process. Overall it's a very simple interface
>> that brings a powerful new dimension to IDL applications.
>
> Thanks for the info. Does Get/Set actually copy variables, or
> just provide references in shared memory (which of course would require
> IDL setting up the existing heap for sharing)? That would allow you to
> Get/Set self in an object, and have it operated on independently in both
> sides.
>
> JD
>