Subject: Re: IDL to C translator
Posted by Liam Gumley on Wed, 08 Apr 1998 07:00:00 GMT
View Forum Message <> Reply to Message

Ebeth Jones wrote:
>
> Liam Gumley wrote:
>
>>
>>> 2) Not everybody has IDL and you may want to share an application.
>>> BTW: Matlab has a matlab -> C translator. I suspect for the above reasons.
>>
>> Surely if your application is brilliantly implemented in IDL, then
>> anyone that wants to use it will fall over themselves in the rush to buy
>> an IDL license.
>
> I feel like a trout on a hook, but here goes....
>
> no matter how brilliantly your application is implemented in
> IDL, if you need to implement that code in a real system, i.e.
> a fielded military system, chances are the hardware that is
> going in the fielded system will not even entertain the thought
> of hosting IDL for your brilliant application, and your brilliant
> application in IDL will not run at real time (not near real time,
> real real time) no matter how much you tweak it. Unless you
> can enlighten me further....

Well surely you would never code a military real-time application in IDL in the first place. A *prototype*, certainly. But the idea behind prototypes is that once you're done with them, you throw them away. In fact, you might even throw away two prototypes, e.g.

(1) Prototype V1 (IDL) demonstrates the user interface, to get feedback from users (no functions)
(2) Prototype V2 (IDL) demonstrates basic functionality
(3) Realtime V1 (C++) is the first production version

Code re-use in this process might happen from steps (1) to (2), but it probably would not happen from step (2) to step (3). The key information that gets refined in this process is the application *design*. It is perfectly appropriate to code the application at step (3) in whatever language is appropriate. I'm just saying that you're probably wasting your time if you try and 'translate' the code from step (2) for use in step (3).

Cheers,
Liam.