

---

Subject: Re: IDL support for international characters (unicode)  
Posted by [George N. White III](#) on Sat, 10 Mar 2007 16:08:47 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Thu, 1 Mar 2007, Mirko wrote:

> On Feb 27, 8:24 am, "Lasse Clausen" <l...@lbnc.de> wrote:

> [...]

> Having a program with a greek character alpha displayed instead of  
> five characters is a matter of easthetics and convenience. We have  
> been programming without greek characters for close to 50 years now.  
> We can keep doing it that way. But in the past 50 years we have  
> switched from machine to assembly, to procedural, and now to object  
> ways of doing programming. This is one example of change of  
> programming practice.

There are many examples where implementations had mistakes that would have been easier to recognize if the notation in the code and the documentation were more similar. One approach that has been used ("literate programming") relies on a source file that uses typesetting markup and can be processed to produce either a document or conventional program sources. Some such systems allow use of greek symbols in the program source, and replace them with conventional names when the sources are extracted. This has been done for C and Fortran (the project, fweb, like many legacy Fortran codes, is, no longer being maintained).

> Some programs are transcriptions of mathematical formulae. It would  
> be much easier to write them and debug them if the program notation  
> followed the mathematical notation as close as possible.  
>  
> And, also imagine doing math using multi-letter names for variables in  
> the equations. I would not find it very appealing.

I have used literate programming tools with IDL, but only in a form where the source fragments in the documentation are assembled into the actual source files when the document is formatted. See:

<<http://www.cse.ohio-state.edu/~gurari/tpf/html/LitProg.html>>

for a simple system based on TeX. This seems to be a good compromise between simplicity (so it doesn't require a big maintenance effort) and functionality.

--

George N. White III <aa056@chebucto.ns.ca>

---