Subject: Re: Idltags Version 2.2

Posted by kump on Wed, 13 Jul 1994 15:57:39 GMT

View Forum Message <> Reply to Message

Could someone explain to me what a "tags file" is? And while you're at
it, why would someone need it and what are Idltree and Shadow are used for?

Ken Kump

Biomedical Image Processing Laboratory
Department of Biomedical Engineering
Case Western Reserve University
Cleveland, Ohio 44106, USA

E-mail: kump@morph.ebme.cwru.edu

Subject: Re: Idltags Version 2.2

Posted by shapiro on Thu, 14 Jul 1994 20:55:26 GMT

View Forum Message <> Reply to Message

kump@morph.ebme.cwru.edu (Kenneth S. Kump) writes:

>
> Could someone explain to me what a "tags file" is? And while you're at
> it, why would someone need it and what are Idltree and Shadow are used for?
>
> ---
>
> Ken Kump
>
> Biomedical Image Processing Laboratory
> Department of Biomedical Engineering
> Case Western Reserve University
> Cleveland, Ohio 44106, USA
>
> E-mail: kump@morph.ebme.cwru.edu

A tags file is a file that contains the names of all the procedures and
functions in your code. Along with the names are stored the name of the
file where the procedure (when I say procedure I am referring to both
procedures and functions) can be found and an ex(1) command which can
located the first line of that procedure.

If you use vi(1) to edit your source code and you want to edit a
procedure you just type :ta <procedure name>. Vi(1) will open the file
and place the cursor at the first line of your procedure. If you have

the cursor over the first letter of a procedure name then ^] (ctrl-]) will open that file and place the cursor at the first line of the procedure. To return to where you started, after editing another file you can use ^~ (ctrl-~).

Idltree takes a tags file and for each procedure generates a list of all the dependent procedures. A dependent procedure is one that is called from with another procedure. In the following example the procedure one depends on both procedure two and three.

```
pro three,a
  print,'three = ',a
end ; three

function two,a
  return,a+1
end ; two

pro one
  three,two(2)
end ; one
```

Shadow reads the list output from idltree, spanws an IDL process and `compiles' (executes a .run) for all the dependent procedures of a specified program. Shadow then saves this `compiled' version using the save command. Shadow can also output a file full of @<filename> directives.

These three programs are not that useful until you start to write IDL applications with 60+ files and 5,000+ lines. Many of the problems that shadow was written to solve can be solved by placing one procedure in each file. This, however, obscures the relationships between groups of procedures.

-Andrew T. Shapiro
CSES/CIRES University of Colorado
shapiro@cses.colorado.edu Campus Box 216
(303) 492-5539 Boulder, CO 80309-0216, USA