Subject: Call Tree Generator
Posted by Calvin King on Wed, 11 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Hello IDL Experts,

Is there a program that looks at an IDL procedure/function and generates the call tree? I inherited a large number of IDL programs for a project. I am going through the programs one at a time starting at the top-on-down to make sure I have all of the necessary programs before compilation and use. Just to make sure I am explaining myself clearly, the top level program A calls programs B and C; B calls D, E, and F; and C calls G, H, I, and J, and so on... Is there an automated way to make sure I have all of the programs: A, B, C, D, E, F, G, H, I and so on?

Thanks, in advance, for any pointers!

Calvin King
Email: cking@sandia.gov

Subject: Re: Call Tree Generator
Posted by davidf on Wed, 11 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Mirko Vukovic (mvukovic@taz.telusa.com) writes:

> Also, how about resolving all of an object's calls? I did not quite
> understand the documentation.

Uh, no. Not object definitions. At least not in IDL 5.2. I haven't checked the latest release. I got a sneak preview of IDL 5.3 earlier this week. Wow! There are going to be some *really* nice things there. I'm looking forward to the soon-to-be-released beta version. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Call Tree Generator
Posted by davidf on Thu, 12 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Mirko Vukovic (mvukovic@taz.telusa.com) writes:

> Lets try it this way (I think this is how the French got their bomb).
>
> Do you think it is wise for me to work on a plotting object that can
> print itself -- a user friendly and smart plot,oplot?
> (there was a discussion regarding that a couple of days ago.)

Let me put it this way. There is *always* room for an
enterprising entrepreneur to write a nice piece of code that,
for whatever reason (let's say it's political), a software
company can't write. In other words, I don't think you are
going to see nice direct graphics objects from RSI anytime
soon. :-)

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Call Tree Generator
Posted by Mirko Vukovic on Thu, 12 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

In article <MPG.121c66121385d41c98988a@news.frii.com>,
davidf@dfanning.com (David Fanning) wrote:
> Bill Biagiot (wbiagiot@suffolk.lib.ny.us) writes with
> respect to IDL 5.3:
>
>> Can you give a simple list of some of the better features?
>
> Uh, not yet. But as soon as I can, I would be happy to.
> I did see a couple of things in the IDLDE that are going
> to make my life MUCH easier. :-)
>
> Cheers,
>
> David

>

Lets try it this way (I think this is how the French got their bomb).

Do you think it is wise for me to work on a plotting object that can
print itself -- a user friendly and smart plot,oplot?
(there was a discussion regarding that a couple of days ago.)

:-)

Mirko

Sent via Deja.com http://www.deja.com/
Share what you know. Learn what you don't.

Subject: Re: Call Tree Generator
Posted by edward.s.meinel on Thu, 12 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Mirko Vukovic <mvukovic@taz.telusa.com> wrote:
> In article <MPG.121bc33671cb20f1989887@news.frii.com>,
> davidf@dfanning.com (David Fanning) wrote:
>> Calvin King (cking@sandia.gov) writes:
>>
>>> Is there a program that looks at an IDL procedure/function and
> generates
>>> the call tree?
>> Start a fresh IDL session. Compile your top-level program.
>> Now type Resolve_All.
> Does it catch really everything?

No. For example, it will not catch event handlers that are not in the
same text file as the procedure that uses it.

Consider DF's box event handler. One option is to rewrite it
specifically for each widget that needs a box-drawing event and include
it in each procedure file. In this case, RESOLVE_ALL will compile it.
Another option is to write a general box event handler. This will allow
you to use it in multiple widgets, but RESOLVE_ALL will miss it because
it apparently doesn't resolve event handler assignments within
calls to WIDGET_CONTROL. Does 5.3 fix that?

Ed Meinel

Sent via Deja.com http://www.deja.com/

Share what you know. Learn what you don't.

Subject: Re: Call Tree Generator
Posted by davidf on Thu, 12 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

Bill Biagiot (wbiagiot@suffolk.lib.ny.us) writes with
respect to IDL 5.3:

> Can you give a simple list of some of the better features?

Uh, not yet. But as soon as I can, I would be happy to.
I did see a couple of things in the IDLDE that are going
to make my life MUCH easier. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Call Tree Generator
Posted by wbiagiot on Thu, 12 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

> I got a sneak preview of
> IDL 5.3 earlier this week. Wow! There are going to be some
> *really* nice things there. I'm looking forward to the
> soon-to-be-released beta version. :-)
>
> Cheers,
>
> David
>

David,

Can you give a simple list of some of the better features?

Thanks,

Bill B.

Sent via Deja.com http://www.deja.com/
Share what you know. Learn what you don't.

Subject: Re: Call Tree Generator
Posted by Mirko Vukovic on Thu, 12 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

In article <MPG.121bc33671cb20f1989887@news.frii.com>,
davidf@dfanning.com (David Fanning) wrote:
> Calvin King (cking@sandia.gov) writes:
>
>> Is there a program that looks at an IDL procedure/function and
generates
>> the call tree? I inherited a large number of IDL programs for a
project.
>> I am going through the programs one at a time starting at the
>> top-on-down to make sure I have all of the necessary programs before
>> compilation and use. Just to make sure I am explaining myself
clearly,
>> the top level program A calls programs B and C; B calls D, E, and F;
and
>> C calls G, H, I, and J, and so on... Is there an automated way to
make
>> sure I have all of the programs: A, B, C, D, E, F, G, H, I and so
on?
>>
>> Thanks, in advance, for any pointers!
>
> Start a fresh IDL session. Compile your top-level program.
> Now type Resolve_All. If you get error messages, you don't
> have all the programs. :-)
>
> You will know which ones you need: the ones Resolve_All
> can't find.
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting
> Phone: 970-221-0438 E-Mail: davidf@dfanning.com

> Coyote's Guide to IDL Programming: http://www.dfanning.com/
> Toll-Free IDL Book Orders: 1-888-461-0155
>
Does it catch really everything?

Also, how about resolving all of an object's calls? I did not quite
understand the documentation.

thanx,

mirko

Subject: Re: Call Tree Generator
Posted by Mirko Vukovic on Fri, 13 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

In article <MPG.121d0e8a710d246a98988e@news.frii.com>,
davidf@dfanning.com (David Fanning) wrote:
> Mirko Vukovic (mvukovic@taz.telusa.com) writes:
>
>> Lets try it this way (I think this is how the French got their
bomb).
>>
>> Do you think it is wise for me to work on a plotting object that can
>> print itself -- a user friendly and smart plot,oplot?
>> (there was a discussion regarding that a couple of days ago.)
>
> Let me put it this way. There is *always* room for an
> enterprising entrepreneur to write a nice piece of code that,
> for whatever reason (let's say it's political), a software
> company can't write. In other words, I don't think you are
> going to see nice direct graphics objects from RSI anytime
> soon. :-)
>
> Cheers,
>
> David
> --

Silly me. A long time ago, I though capitalism was about high
quality products. But as one friend pointed out to me, it
is about profits and egos mainly, with quality products a b(u)yproduct.

Mirko

Sent via Deja.com http://www.deja.com/
Share what you know. Learn what you don't.

Subject: Re: Call Tree Generator
Posted by m218003 on Mon, 16 Aug 1999 07:00:00 GMT
View Forum Message <> Reply to Message

In article <MPG.121bc33671cb20f1989887@news.frii.com>,
davidf@dfanning.com (David Fanning) writes:
> Calvin King (cking@sandia.gov) writes:
>
>> Is there a program that looks at an IDL procedure/function and generates
>> the call tree? I inherited a large number of IDL programs for a project.
>> I am going through the programs one at a time starting at the
>> top-on-down to make sure I have all of the necessary programs before
>> compilation and use. Just to make sure I am explaining myself clearly,
>> the top level program A calls programs B and C; B calls D, E, and F; and
>> C calls G, H, I, and J, and so on... Is there an automated way to make
>> sure I have all of the programs: A, B, C, D, E, F, G, H, I and so on?
>>
>> Thanks, in advance, for any pointers!
>
> Start a fresh IDL session. Compile your top-level program.
> Now type Resolve_All. If you get error messages, you don't
> have all the programs. :-)
>
> You will know which ones you need: the ones Resolve_All
> can't find.
>
A good method, but not perfect. It doesn't find any routine called with
EXECUTE, CALL_PROCEDURE, or CALL_FUNCTION. But it should be possible to
write a program that does the following:
* resolve_all
* get all compiled routines with -- was it HELP,... ?
there is also some output of the filenames for these routines
* loop through all files and look for EXECUTE, CALL_... statements
and analyze their argument. If the argument is a simple
string, you can assume there should be a file named <string>.pro
if the argument is some sort of variable, it's harder. Maybe
it's enough to simply print out a list of all EXECUTE and
CALL_... statements. Of course you can do that with grep
(one of the many many reasons why I will never switch from Unix
to Windows - although I am told it exists for Windows which is
to say for DOS)

just $0.02, or 0.0212 Euro ;-)

Martin

> Cheers,
>
> David
>
