Subject: no backwards compatibility in IDL 5.6
Posted by Alexander Rauscher on Thu, 27 Feb 2003 12:52:15 GMT
View Forum Message <> Reply to Message

sorry for posting the same thing twice under different subjects, but i
think this is important...

many of idl programs have to be adapted due do the non existing
backwards compatibility of atan (and probably many other functions) .
one wouldn't expect a change in such a fundamental function. so now
atan(z, /phase) gives the same result as atan(z) in older versions did,
where z is (re,im)... this is worse than stupid. this is dangerous.

or does anybody know away to circumvent the new "features" of IDL 5.6?

alex

Subject: Re: no backwards compatibility in IDL 5.6
Posted by JD Smith on Fri, 28 Feb 2003 20:02:15 GMT
View Forum Message <> Reply to Message

On Fri, 28 Feb 2003 12:42:42 -0700, David Fanning wrote:

> Pavel Romashkin (pavel_romashkin@hotmail.com) writes:
>
>> Why is EXECUTE used in this program? Why can't the value just be
>> returned from each CASE? Execute will slow it down and as far as I can
>> tell, does nothing special. There is no code that follows the CASE to
>> prevent you from returning at any point. Will it not compile in 5.4
>> with the extra keyword? I thought keyword mismatches are runtime
>> errors. Am I missing something?
>
> Alright, here is why I am using EXECUTE. If I change the code to this:
>
>   returnValue = 0.0
>   version = Float(!VERSION.Release)
>   IF (version LE 5.5) THEN returnValue = ATAN(imgpart, realpart) $
>   ELSE returnValue = ATAN(complexNum, /Phase)
>
> Then the code won't compile in IDL 5.4, complaining about the PHASE
> keyword not being defined. :-(
>
>
> P.S. The code *does* compile in IDL 5.5, by the way, even though the
> PHASE keyword is not defined there, either.

Which is when _STRICT_EXTRA was first introduced. Coincidence?

My consipiracy theory: RSI switched from checking built-in system routine
keywords at compile to run-time with v5.5. This isn't the first time
they've done this type of thing: around v5.3, they switched from checking
the validity of system variables at compile time to run time (which was
rather convenient, actually).

JD

Subject: Re: no backwards compatibility in IDL 5.6
Posted by David Fanning on Fri, 28 Feb 2003 20:28:16 GMT
View Forum Message <> Reply to Message

JD Smith (jdsmith@as.arizona.edu) writes:

> Which is when _STRICT_EXTRA was first introduced. Coincidence?
>
> My consipiracy theory: RSI switched from checking built-in system routine
> keywords at compile to run-time with v5.5. This isn't the first time
> they've done this type of thing: around v5.3, they switched from checking
> the validity of system variables at compile time to run time (which was
> rather convenient, actually).

Alright, another case closed by the ace detectives
at the IEPA agency. I'll write a report, sir.

Cheers,

David

P.S. There is a new wrapper program on my web page
that uses only a singe EXECUTE statement. :-)

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

Subject: Re: no backwards compatibility in IDL 5.6
Posted by JD Smith on Fri, 28 Feb 2003 20:35:55 GMT
View Forum Message <> Reply to Message

On Fri, 28 Feb 2003 13:12:17 -0700, William Thompson wrote:

> Pavel Romashkin <pavel_romashkin@hotmail.com> writes:
>
>> Why is EXECUTE used in this program? Why can't the value just be
>> returned from each CASE? Execute will slow it down and as far as I can
>> tell, does nothing special. There is no code that follows the CASE to
>> prevent you from returning at any point. Will it not compile in 5.4 with
>> the extra keyword? I thought keyword mismatches are runtime errors. Am I
>> missing something?
>> Pavel
>
>
> Yes, without the execute statement, it will not compile in versions
> earlier than 5.4. You get the error message
>
> IDL> .run atan_complex_wrapper
>
> (version GE 5.6): returnValue = ATAN(complexNum, /Phase)
> ^
> % Keyword parameters not allowed in call.
> At: /disk1/thompson/atan_complex_wrapper.pro, Line 12
> % 1 Compilation errors in module ATAN_COMPLEX_WRAPPER.
>
> However, only the last execute statement is actually required. The
> first two, without the new keyword, can be direct statements.
>

Aha! Another conspiracy laid to rest. IDL apparently checks whether zero, or more than zero keywords are allowed.

Try compiling:

```
pro foo
  empty,/FOO
end
```

and you'll see what I mean. Since ATAN went from having zero to more than zero keywords, this explains it. The compile vs. runtime handling of keywords (perverse as it is) has remained unchanged.

JD

Subject: Re: no backwards compatibility in IDL 5.6
Posted by Pavel Romashkin on Fri, 28 Feb 2003 23:32:26 GMT
View Forum Message <> Reply to Message

And exactly, who of you two folks report to whom? I am getting confused.

It must be those drinks at that banquet from which you, David, got the
headac... I mean, the cold :-)
Pavel

David Fanning wrote:
>
> Alright, another case closed by the ace detectives
> at the IEPA agency. I'll write a report, sir.

Subject: Re: no backwards compatibility in IDL 5.6
Posted by David Fanning on Fri, 28 Feb 2003 23:55:42 GMT
View Forum Message <> Reply to Message

Pavel Romashkin (pavel_romashkin@hotmail.com) writes:

> And exactly, who of you two folks report to whom? I am getting confused.

Oh, I *definitely* report to JD. That guy scares me. :-)

Cheers,

David

P.S. Do you think it's possible to drink TOO much
orange juice? I don't feel so good....

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

Subject: Re: no backwards compatibility in IDL 5.6
Posted by Craig Markwardt on Sat, 01 Mar 2003 02:20:19 GMT
View Forum Message <> Reply to Message

thompson@orpheus.nascom.nasa.gov (William Thompson) writes:

> Pavel Romashkin <pavel_romashkin@hotmail.com> writes:
>
>> Why is EXECUTE used in this program? Why can't the value just be
>> returned from each CASE? Execute will slow it down and as far as I can
>> tell, does nothing special. There is no code that follows the CASE to

>> prevent you from returning at any point. Will it not compile in 5.4 with
>> the extra keyword? I thought keyword mismatches are runtime errors. Am I
>> missing something?
>> Pavel
>
>
> Yes, without the execute statement, it will not compile in versions earlier
> than 5.4. You get the error message
>
>   IDL> .run atan_complex_wrapper
>
>   (version GE 5.6): returnValue = ATAN(complexNum, /Phase)
>   ^
>   % Keyword parameters not allowed in call.
>   At: /disk1/thompson/atan_complex_wrapper.pro, Line 12
>   % 1 Compilation errors in module ATAN_COMPLEX_WRAPPER.
>
> However, only the last execute statement is actually required. The first two,
> without the new keyword, can be direct statements.

And even, the last EXECUTE is not required, if one uses CALL_FUNCTION
instead, which should be a bit faster, not as clunky looking to me.

Craig

--

 -- --------------
Craig B. Markwardt, Ph.D.   EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
 -- --------------

Subject: Re: no backwards compatibility in IDL 5.6
Posted by David Fanning on Sat, 01 Mar 2003 04:40:45 GMT
View Forum Message <> Reply to Message

Craig Markwardt (craigmnet@cow.physics.wisc.edu) writes:

> And even, the last EXECUTE is not required, if one uses CALL_FUNCTION
> instead, which should be a bit faster, not as clunky looking to me.

Yeah, I already changed that when I woke up from my nap. :-)

Cheers,

David
--
David W. Fanning, Ph.D.

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

Subject: Re: no backwards compatibility in IDL 5.6
Posted by JD Smith on Mon, 03 Mar 2003 17:12:24 GMT
View Forum Message <> Reply to Message

On Fri, 28 Feb 2003 13:02:15 -0700, JD Smith wrote:

> On Fri, 28 Feb 2003 12:42:42 -0700, David Fanning wrote:
>
>> Pavel Romashkin (pavel_romashkin@hotmail.com) writes:
>>
>>> Why is EXECUTE used in this program? Why can't the value just be
>>> returned from each CASE? Execute will slow it down and as far as I can
>>> tell, does nothing special. There is no code that follows the CASE to
>>> prevent you from returning at any point. Will it not compile in 5.4
>>> with the extra keyword? I thought keyword mismatches are runtime
>>> errors. Am I missing something?
>>
>> Alright, here is why I am using EXECUTE. If I change the code to this:
>>
>> returnValue = 0.0
>> version = Float(!VERSION.Release)
>> IF (version LE 5.5) THEN returnValue = ATAN(imgpart, realpart) $
>> ELSE returnValue = ATAN(complexNum, /Phase)
>>
>> Then the code won't compile in IDL 5.4, complaining about the PHASE
>> keyword not being defined. :-(
>>
>>
>> P.S. The code *does* compile in IDL 5.5, by the way, even though the
>> PHASE keyword is not defined there, either.
>
> Which is when _STRICT_EXTRA was first introduced. Coincidence?
>
> My consipiracy theory: RSI switched from checking built-in system
> routine keywords at compile to run-time with v5.5. This isn't the
> first time they've done this type of thing: around v5.3, they switched
> from checking the validity of system variables at compile time to run
> time (which was rather convenient, actually).
>
> JD

Just to clarify, for the sake of RSI's hard working engineers: this

conspiracy theory, like many others, is entertaining, but entirely untrue. IDL checks at compile time whether *any* keywords are allowed at all. Since ATAN went from having none to having one keyword, this explains the difference.

JD
