

---

Subject: Re: no backwards compatibility in IDL 5.6  
Posted by [thompson](#) on Thu, 27 Feb 2003 15:58:38 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

"R.G. Stockwell" <[sorry@noemail.now](mailto:sorry@noemail.now)> writes:

> Alexander Rauscher wrote:  
>> 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

> (Perhaps I am missing something here.)

```
> RELEASE      STRING  '5.5a'
> IDL> print,atan(complex(4,4))
> (   1.44452,   0.123674)
> IDL> print,atan(complex(4,4),/phase)
> % Keyword PHASE not allowed in call to: ATAN
> IDL> print,atan(10,10)
>    0.785398
```

```
> RELEASE      STRING  '5.6'
> IDL> print,atan(complex(4,4))
> (   1.44452,   0.123674)
> IDL> print,atan(complex(4,4),/phase)
> 0.785398
> IDL> print,atan(10,10)
>    0.785398
```

> The same thing for the case of atan(complex), and  
> also for the case of atan([float array]). What has broken?  
> Also, in 5.6, in the use of atan(10,10, /phase), the "/phase"  
> is ignored (according to the help file).

> It seems to me that functionality has been added in  
> a way the preserves the previous functionality, so no  
> legacy code should have any effect.

Evidently, the change in behavior was introduced in 5.5 or 5.5a. This is how

it works under 5.4.1 and previous versions of IDL.

```
IDL> print,atan(complex(4,4))  
0.785398
```

So the point is that you need to change your code to \*ADD\* the /PHASE keyword to recover the previous behavior. That's definitely a case of something not being backwardly compatible. To make it worse, you have to put in an IF statement to correctly handle the various cases, e.g. 5.4, 5.5, and 5.6, which all act differently.

A better solution may be to separate out the real and imaginary parts, and pass them to ATAN separately, e.g.

```
IDL> x = complex(3,4)  
IDL> print,atan(x) ;Under 5.4.1 or below  
0.927295  
IDL> print,atan(imaginary(x),float(x))  
0.927295
```

Although you'd still have to worry about the distinction between single and double-precision complex numbers.

Bill Thompson

---