Subject: Re: AND statements

Posted by philaldis on Mon, 01 Mar 1999 08:00:00 GMT

View Forum Message <> Reply to Message

>>   If you write some code like this:
>>
>>   test=Ptr_New()
>>
>>   IF Ptr_Valid(test) AND Size(*test, /type) NE 10 THEN print, *test
>>
>>   ....can you always guarantee that it will not try to evaluate the second
>>   statement if the first one was false - or is this a dangerous tactic to
>>   adopt?
>
> Nope. In fact, I can guarantee that it *will* evaluate the whole
> logical expression. IDL is in this respect totally unlike C.
>
> If the urge is big enough, one could ask RSI nicely to implement
> operators like "AND THEN" and "OR ELSE" used like this:
>
>   IF ptr_valid(test) AND THEN size(*test,/type) ne 10 then print,*test
>
>   IF error_occurred OR ELSE check_for_error() then print,"Error"
>
> "AND THEN" works like C &&
> "OR ELSE" works like C ||, i.e. check_for_error() isn't called
> if "error_occurred" is already true.
>

> Stein Vidar

So, to avoid that you have to do some pretty messy code. Say for
example I've got :

```
IF Ptr_Valid(ThisPointer) THEN BEGIN
 IF Size(*ThisPointer, /type) EQ 10 THEN BEGIN
  .
  .
  .
  .
  .
 ENDIF

ENDIF
```

However, I want to execute the same bit of code if it fails the
Ptr_Valid and the Size(*ThisPointer, type0 EQ 10, so as far as I can
see, (and I realise that I may be missing something pretty blatent),

you have to use flags

```
IF Ptr_Valid(ThisPointer) THEN BEGIN
 IF Size(*ThisPointer, /type) EQ 10 THEN BEGIN
  .
  .
  .
  .

  .
 ENDIF ELSE flag = 1

ENDIF ELSE flag = 1

IF flag ........
```

While obviously this is not the end of the world, there could be more
complex examples, and the code does look messy.

Cheers,
Phil