Subject: Re: bitwise operators in IDL?
Posted by John-David T. Smith on Wed, 23 May 2001 19:46:44 GMT
View Forum Message <> Reply to Message

Craig Markwardt wrote:
>
> JD Smith <jdsmith@astro.cornell.edu> writes:
>> Marc Schellens wrote:
>>> To late for short circuitry.
>>> Consider a case when the second function in the if case
>>> has a side effect (e.g. modifying a global variable).
>>> After once defining the language this way, to change it
>>> would mean to introduce incopatibility.
>>
>> Not if you introduce another operator set all together for short
>> circuiting. People will use them increasingly, and AND,OR will go back
>> to being used primarily for their bitwise function, as they should be.
>> Sort of like C has "&&" and "&". I can't think of good replacement
>> names (I assume RSI won't allow the sensible && and ||). Ideas? (ANDS?
>> ORS?).
>>
>> JD
>
> I like the idea of short-circuiting logical operators as well. Some
> time ago I proposed LAND and LOR (for Logical-AND and Logical-OR). Of
> course what do you do about backwards compatibility? Namely people
> who have written code with their own LAND or ANDS variables. Imagine
> this :-)
>
>   if LAND LAND 1 then ...
>
> What would it mean?
>
> Also, short-circuiting operators are not well-defined for vector
> operands. With vector operands it is possible in an
> element-by-element comparison for one element to evaluate true, while
> the other evaluates false. E.g.
>
>   if [1,1] LAND [0,1] then ...
>
> So, both of these issues would at least have to be dealt with, and my
> guess is that the RSI folks will decide not to deal with it. Still, I
> think it's worth exploring.

Luckily, this is already illegal:

IDL> if [1,1] AND [0,1] then print, 'yay'
% Expression must be a scalar in this context: <INT   Array[2]>.

% Execution halted at: $MAIN$

Only scalars allowed in if's. As long as the new IAND and IOR operators
are only allowed within if statements, no trouble arises.

For your example, proper use of total() is encouraged.

JD