Subject: Re: _overloadMinus: what to do with invalid input?
Posted by Paul Van Delst[1] on Tue, 30 Dec 2014 14:58:57 GMT
View Forum Message <> Reply to Message

Hello,

On 12/29/14 17:27, Paulo Penteado wrote:
> On Monday, December 29, 2014 5:41:49 PM UTC-2, Mike Galloy wrote:
>> I would think an error and halting, like IDL would do if you tried to use
>> an invalid operator with the native types. What happens if you try to add
>> two pointers? (not in front of my computer right now)
>
> To answer Mike's question:
>
> IDL> a=ptr_new(1)
> IDL> b=ptr_new(2)
> IDL> c=a+b
> % Operation illegal with pointer types.
> % Execution halted at: $MAIN$
>
> Regarding Paul's question, I would say that the answer depends on how
> one envisions the object's usage. If one decides it makes no sense to
> do the subtraction, like with pointers, it should throw an error. An
> error should also be raised, instead of returning a value, if such a
> return value could be confused with a valid result. For instance,
> taking IDL's list:
>
> IDL> l=list(1,2)
> IDL> l+3
> % LIST::_OVERLOADPLUS: Arguments must both be lists.
> % Execution halted at: $MAIN$

Oh. That make sense. I guess I should have tested using objects too,
rather than intrinsic types.

An error it is.

Thanks guys.

cheers,

paulv