

---

Subject: Re: Object programming with data...  
Posted by [Jaco van Gorkom](#) on Wed, 22 May 2002 15:39:12 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

"Randall Skelton" <rhskelto@atm.ox.ac.uk> wrote in message  
news:Pine.LNX.4.33.0205192219030.712-100000@moriarty.atm.ox. ac.uk...  
> In response to some of my own questions I found a post from Ken Knighton  
> in 1996 that demonstrates one way of emulating operator overloading using  
> IDL functions... I'm open to other suggestions.

I have used CASE statements in the past, but also occasionally made use of recursion, something like:

```
function Add, p1, p2, p3, p4, p5, p6
  if n_elements(p3) eq 0 then $
    return, p1+p2 $
  else $
    return, Add(p1+p2, p3, p4, p5, p6)
end
```

There is a variant which can be very quick (and sort of ugly) to add to the start of an existing function:

```
function Add, p1, p2, p3, p4, p5, p6
  ; Add this statement for recursive parameter handling
  if n_elements(p3) ne 0 then $
    return, Add( Add(p1,p2), p3, p4, p5, p6)
  ; Original two-parameter function code follows:
  return, p1+p2
end
```

Cheers,  
Jaco

---