On Mar 3, 3:07 pm, wlandsman <wlands...@gmail.com> wrote:
> Well, I don't know how to avoid creating a new structure but you can
> use STRUCT_ASSIGN to copy fields from the old structure to the new
> one.
>
> IDL> str={a: 3l, b: 'v'}
> IDL> str_arr=REPLICATE(str,10)
>
> IDL> str1={a: 3l, b: 'w',c:0l}
> IDL> str1_arr= REplicate(str,10)
>
> IDL> struct_assign,temporary(str),str1
>
> where I changed the initialization values so one can verify that the
> "relaxed structure assignment" is working.

The only other way I see is to use a pointer array instead. Then you
can replace elements by whatever you want. But it is more likely to be
a worse solution, since it will not be possible to use fields as
arrays.

This reminds me of what I currently find most needed in create_struct:
an option to give it an array of field names, and an array of pointers
to the field contents. That way it would not be necessary to loop over
the fields, calling create_struct to add one field at a time, defining
a bunch of temporary structures that will never be used again. But
presumably this will not be necessary in IDL 8, since it will finally
have dictionaries.