
Subject: Re: gurus: example code as main level code ?
Posted by [R.Bauer](#) on Mon, 04 Nov 2002 21:09:15 GMT
[View Forum Message](#) <> [Reply to Message](#)

R.G. Stockwell wrote:

```
> Reimar Bauer wrote:
>> Dear Bob,
>>
>> I am wondering a bit what happens to the example if I do
>>
>> .compile range
>> resolve_all
>> save,file='range.sav',/routines,/compress
>>
>> It seems only the function is saved in the sav binary.
>>
>> First I was thinking this could be a problem for the routine
>> itself, but it works fine. I can miss the example by runtime routines.
>>
>> Normally if we have more complicated routines you need more as one
>> testcase. Or did you think about having a large main program which tells
>> what it is testing and what the result should be and is. At the moment I
>> am using the EXAMPLES: tag in the comments.
>>
>> By graphic routines this is nearly impossible. There is a saved image
>> better for comparisons. For the plotenvironment I have already a large
>> image / testcases which is very useful during beta tests.
>>
>> Your idea will make it very easy to check math functions and to show
>> differences if the output is redirected to a file.
>>
>>
>> regards
>>
>> Reimar
>
> Very good point Reimar, I had not considered what would
> happen in a sav file. I rarely use them, usually only for an
> interim data storage solution in development.
> Perhaps I will switch over to an /examples keyword.
>
> I also include a /help keyword too, to remind the user (i.e. me)
> what keywords are available and how to use the routine, but it
> would probably be better to make a help file or some other real
> documentation.
```

Dear Bob,

For this case may be my call_help routine is interesting too.

http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_source/idl_html/dbase/download/call_help.tar.gz

This routine is able to read out the comments from the description.
So it is not necessary to describe things always twice. And it is much easier to do changes at only one place.

For example if you have a routine which needs two positional parameters
(I have added only a subset of the header and the routine.)

If you enter less parameters always the text from
; CALLING SEQUENCE: is printed out.

```
IDL> x=diff_struct()  
% DIFF_STRUCT:  
result=diff_struct(struct1,struct2,[diff1=diff1],[diff2=diff 2])
```

I am using the example comment too to describe by many examples
what the routine is doing.

This could be extracted too. If you have a look in the catalog
by

http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_source/idl_html/examples/diff_struct_example.pro.html

regards

Reimar

```
;+  
; NAME:  
; diff_struct  
;  
; PURPOSE:  
; This function returns all differences of all values from two structures.  
; Pointer or values could be compared.  
; Only tags which are in both structures are compared.  
; Differences could be returned by ARG_PRESENT by diff1 or diff2.  
;  
; CATEGORY:  
; PROG_TOOLS/STRUCTURES  
;  
; CALLING SEQUENCE:  
; result=diff_struct(struct1,struct2,[diff1=diff1],[diff2=diff 2])  
;-
```

```
FUNCTION diff_struct,struct1,struct2,diff1=diff1,diff2=diff2
```

```
IF N_PARAMS() LT 2 THEN BEGIN
  MESSAGE,call_help(),/cont
  RETURN,-1
ENDIF
return,1
END
```

--

Forschungszentrum Juelich
email: R.Bauer@fz-juelich.de
<http://www.fz-juelich.de/icg/icg-i/>

=====

a IDL library at ForschungsZentrum Juelich
http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_lib_intro.html
