Subject: Getting error when running a routine using @ in IDL.
Posted by msbstar on Wed, 04 Feb 2015 10:53:41 GMT
View Forum Message <> Reply to Message

Hello every body,

When I issued the @reduce.scr command in IDL I got the following error:

```
IDL> @reduce.scr
% Case statement found no matches.
% Execution halted at: $MAIN$
IDL>
```

Below is the content of the "reduce.scr" file:

```
;-------------------------------------------------
;;; Reduce MagE data

; Read in object and calibration file names
templ = {   $
        version: 1.0, $
        datastart: 0, $
        delimiter: ' ', $
        missingvalue: -99.0, $
        commentsymbol: '#', $
        fieldcount: 16, $
        fieldtypes: [7,7,3,3,3,3,3,7,7,7,7,7,7,7,7,7], $
        fieldnames: ['root','type','reduce_lo','reduce_hi','fit_lo',$
                'fit_hi','fiducial','arc','refarc','orderlist',$
             'slit','wave','xe_flat','xe_flatvar','dome_flat',$
             'dome_flatvar'], $
        fieldlocations: [0,14,25,28,36,39,45,55,66,77,98,118,139,166,$
                    196,222], $
        fieldgroups: [0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15] $
       }
data            = read_ascii('reduce_info.dat',template=templ)
rootname        = data.root
exptype         = data.type
reduce_order_lo  = data.reduce_lo
reduce_order_hi  = data.reduce_hi
fit_order_lo     = data.fit_lo
fit_order_hi     = data.fit_hi
fiducial_order   = data.fiducial
arcfile         = data.arc
refarcfile       = data.refarc
orderlist       = data.orderlist
slitfile        = data.slit
wavefile        = data.wave
```

```
xe_flatfile      = data.xe_flat
xe_flatvarfile   = data.xe_flatvar
dome_flatfile    = data.dome_flat
dome_flatvarfile = data.dome_flatvar
n_exposures      = n_elements(rootname)

;;; Change the next three lines to reflect the correct directories on
;;; your system.

rawdir    = '/home/hassan/Desktop/MagE/hassan/'
caldir    = '/home/hassan/mage_reduce/calib/'
reduxdir  = '/home/hassan/Desktop/MagE/hassan/out/'

;;;
;;; Loop through and reduce each frame
;;;

for i=0,n_exposures-1 do begin                          & $

  root    = rootname(i)                                 & $
  raw     = rawdir+root+'.fits'                         & $
  type    = exptype(i)                                  & $
  arc     = rawdir+arcfile(i)+'.fits'                   & $
  refarc  = caldir+refarcfile(i)+'.fits'                & $
  list    = caldir+orderlist(i)+'.dat'                  & $
  wave    = caldir+wavefile(i)+'.fits'                  & $
  slit    = caldir+slitfile(i)+'.fits'                  & $
  xe_flat    = reduxdir+xe_flatfile(i)+'.fits'          & $
  xe_flatvar = reduxdir+xe_flatvarfile(i)+'.fits'       & $
  dome_flat    = reduxdir+dome_flatfile(i)+'.fits'      & $
  dome_flatvar = reduxdir+dome_flatvarfile(i)+'.fits'   & $
  output  = reduxdir+root                               & $

  if (arcfile(i) eq '0') then arc=0                     & $
  if (refarcfile(i) eq '0') then refarc=0               & $

  if (dome_flatfile(i) eq '0') then dome_flat=0         & $
  if (dome_flatvarfile(i) eq '0') then dome_flatvar=0   & $

  case type of                                          & $
    'BRIGHT': begin                                     & $
              spline_profile = 1                        & $
        trace_obj      = 1      & $
           end                                          & $
    'MEDIUM': begin                                     & $
              spline_profile = 0                        & $
        trace_obj      = 1      & $
 end                                 & $
```

```
    'FAINT':  begin                                & $
              spline_profile = 0                   & $
         trace_obj      = 0      & $
 end                                  & $
  endcase                                       & $

  n_to_reduce      = reduce_order_hi(i) - reduce_order_lo(i) + 1  & $
  orders_to_reduce = reduce_order_lo(i) + lindgen(n_to_reduce)    & $
  n_to_fit         = fit_order_hi(i) - fit_order_lo(i) + 1        & $
  orders_to_fit    = fit_order_lo(i) + lindgen(n_to_fit)          & $

  fiducial = fiducial_order(i)                                    & $

  if (!journal ne 0) then journal                                  & $
  journal,output+'.log'                                          & $

  mage_reduce,raw,wave,slit,list,skyref=refarc,skyframe=arc,$
           flatfield=xe_flat,flatvar=xe_flatvar,$
      redflatfield=dome_flat,redflatvar=dome_flatvar,$
           output=output,fiducial=fiducial,maxiter=10,noshow=1,$
           order=orders_to_reduce,fit=orders_to_fit,$
           sprofile=spline_profile,/faint,$
           trace_obj=trace_obj                         & $

  journal                                              & $
  close,/all                                           & $

  ; Uncomment the following line if you want to automatically
  ; gzip the reduced files.
  ;spawn,'gzip -f '+reduxdir+'*.fits'                        & $

endfor

 ;============================================================ ===
```

I would be grateful for any help on this.

With best regards,
Hassan

Subject: Re: Getting error when running a routine using @ in IDL.
Posted by Fabzi on Wed, 04 Feb 2015 11:21:43 GMT
View Forum Message <> Reply to Message

Hi,

at some point in the script you test for variable "type" being either

'BRIGHT','MEDIUM', or 'FAINT'. It seems that at run-time, the variable type was something else than those three possibilities.

As a side note, these kind of scripts are quite bad coding practice, IDL provides functions, procedures and even classes today ;-)

Cheers

Subject: Re: Getting error when running a routine using @ in IDL.
Posted by msbstar on Wed, 04 Feb 2015 13:08:05 GMT
View Forum Message <> Reply to Message

On Wednesday, February 4, 2015 at 2:51:46 PM UTC+3:30, Fabien wrote:
> Hi,
>
> at some point in the script you test for variable "type" being either
> 'BRIGHT','MEDIUM', or 'FAINT'. It seems that at run-time, the variable
> type was something else than those three possibilities.
>
> As a side note, these kind of scripts are quite bad coding practice, IDL
> provides functions, procedures and even classes today ;-)
>
> Cheers

Hi Fabien,

That is great ! Thanks for pointing that out. The problem is now fixed.

Best regards,
Hassan

Subject: Re: Getting error when running a routine using @ in IDL.
Posted by Phillip Bitzer on Wed, 04 Feb 2015 15:25:41 GMT
View Forum Message <> Reply to Message

On Wednesday, February 4, 2015 at 5:21:46 AM UTC-6, Fabien wrote:
> Hi,
>
> at some point in the script you test for variable "type" being either
> 'BRIGHT','MEDIUM', or 'FAINT'. It seems that at run-time, the variable
> type was something else than those three possibilities.
>
> As a side note, these kind of scripts are quite bad coding practice, IDL
> provides functions, procedures and even classes today ;-)

>
> Cheers

I'll add this too: it's ALWAYS a good idea to include a ELSE clause in the a CASE statement. This is exactly why - when no matches are found.

And, I can't agree with Fabien more: this way of coding is a best messy, and at worse poor coding practice. Just as a practical matter, all those line break/continuation characters would drive me crazy ;-)
