## Subject: Re: Fuzzy searching of FITS header
## Posted by Gray on Wed, 22 Dec 2010 13:45:44 GMT

View Forum Message <> Reply to Message

On Dec 22, 8:40 am, Gray <grayliketheco...@gmail.com> wrote:
> On Dec 22, 8:06 am, wlandsman <wlands...@gmail.com> wrote:
>
>
>
>
>
>
>
>
>
>> On Tuesday, December 21, 2010 10:49:30 PM UTC-5, Marc Buie wrote:
>>> Wayne -
>
>>> Why can't this be handled with
>
>>> dexp=sxpar(hdr,'D*DEXP')
>
>>> sxpar already handles
>
>>> naxis=sxpar(hdr,'NAXIS*')
>
>>> It seems to me that this is a simple extension of what sxpar already does.
>
>> Well, there is a FITS convention for reserved keyword names followed by sequential integers (e.g. NAXIS1, NAXIS2, NAXIS3... ), where (with one exception) you can be sure that the returned values will all be of the same type (in this case integers). But for a general wildcard (e.g. 'D*EXP') the returned values might be a mixture of strings, integers and floats. That is why Paulo's list/hash approach seems preferable this case. --Wayne
>
>> P.S. The one exception for reserved keyword names is TSCALi for converting 16 bit integers in a binary table to double/float. In some cases TSCALi returns a float and in other cases it returns a double. That is why MRDFITS currently has a limitation of requiring either all conversions to float or all conversions to double. In some other FITS routines I get around this limitation by using pointers, but it is a pain. It is a nice application for the new LIST datatype.
>
> Here's a very basic writeup using lists and hashes and FXPAR() (since
> I didn't feel like re-writing the keyword parsing rules).
>

```
> FUNCTION hdregex, header, search
>   n = n_elements(search)
>   keys = strmid(header,0,8)
>   if (n lt 2) then begin
>     this = where(stregex(keys,search,/fold,/bool),count)
>     if (count eq 0) then return, !null
>     out = hash(keys[this])
>     for i=0,count-1 do out[keys[this[i]]] = $
>       fxpar(header,keys[this[i]],start=this[i],precheck=0)
>   endif else begin
>     out = list(length=n)
>     foreach key,search,i do begin
>       this = where(stregex(keys,search,/fold,/bool),count)
>       if (count eq 0) then continue
>       temp = hash(keys[this])
>       for j=0,count-1 do out[keys[this[j]]] = $
>         fxpar(header,keys[this[j]],start=this[j],precheck=0)
>       out[i] = temp
>     endforeach
>   endelse
>   return, out
> end
```

Oops, typos (that's what I get for copy/paste):

```
FUNCTION hdregex, header, search
  n = n_elements(search)
  keys = strmid(header,0,8)
  if (n lt 2) then begin
    this = where(stregex(keys,search,/fold,/bool),count)
    if (count eq 0) then return, !null
    out = hash(keys[this])
    for i=0,count-1 do out[keys[this[i]]] = $
      fxpar(header,keys[this[i]],start=this[i],precheck=0)
  endif else begin
    out = list(length=n)
    foreach key,search,i do begin
      this = where(stregex(keys,key,/fold,/bool),count)
      if (count eq 0) then continue
      temp = hash(keys[this])
      for j=0,count-1 do temp[keys[this[j]]] = $
        fxpar(header,keys[this[j]],start=this[j],precheck=0)
      out[i] = temp
    endforeach
  endelse
  return, out
end
```