Subject: Avoiding multiple or complex WHEREs?
Posted by cgguido on Mon, 21 Feb 2011 21:31:00 GMT
View Forum Message <> Reply to Message

```
;Say I have an array of point IDs and times where p[0,*] are IDs and
;p[1,*] are times, something like:

  p=[[1,1],[1,2],[2,3]]

;Typically, the array has many points, many times and many gaps
;I want to be able to get a subset of 'p' that contains a range
;[id0:idn] or a list [id0,id2,id7] of IDs and Ts the most efficient
;way possible.

;Right now, I build an array 'i' with one FOR loop  where i[id,t] is
the row index to 'p'
;with the correct id and t. If that combinations does not exist, then
;i[id,t]=NaN. So:

  i[0,0]=NaN
  i[2,0]=NaN
  i[1,*]=[NaN,0,1,NaN]

;So if I want all instances of particle 1 in array p I do

 i1=i[1,where(i[1,*] ge 0)]
 p1=p[*,i1]

;This gets handy when I want particles 1,3,4 and times [23:100],
;because I don't have to do where(id eq 1 OR id eq 3 OR id EQ 4....
etc.)

   i_cool=i[1,where(i[[1,3,4],[23:100]] ge 0)]
   p_cool=p[*,i_cool]


;Now, I am betting that a histogram/histobin trick might be even
;better... Any ideas?

;PS: This does not deal nicely with IDs or Ts that are bigger than the
;biggest contained in 'p'
```