Subject: Re: Oh what is wrong with WHERE() ?
Posted by Liberum on Sat, 29 Apr 2006 16:15:52 GMT
View Forum Message <> Reply to Message

Hi Bob,

This is good point that you make and I have thought about this at great length before. I have to set the arrays m and p to the equal amount of 255, i.e., no data values so as do a good pixel to pixel comparison over a period of one month. This means that I will do this m and p comparison about 355 times for one month. Now m and p varies when it comes to which pixels are valid and which are not. Some times p has all 255 values and the comparison cannot be done.
Now in the checks that I have placed in the program checks of the the count for 255 in p is greater than the 255 count in m and sets nodata_tot to the greater number:

```
 IF count_m GE count_p THEN nodata_tot = nodata_m ELSE nodata_tot =
nodata_p
```

The results showed that these two amounts were identical:

```
1443953
1443953
```

This is what I did to solve the problem your pointing out. Yet is does not work.
The numbers add up until the last where statement where the one pixel disappears. This is why I am trying now David's approach and using LONG instead of DOUBLE. Still a pixel disappears :(I think that it is something about this array as I tried another and it works. The thing is what? I can shrink the arrays but I will have to exclude some data.
I did just that with the following check (each arrays should only have the values 255, 1 or 2):

```
r = where((p EQ 255.0 OR p EQ 1.0 OR p EQ 2.0), check_p,
COMPLEMENT=compp, NCOMPLEMENT=ncompp)
print, 'check_p:  ', check_p
IF ncompp GT 0 THEN stop, p[compp]
```

Yet this test is passed and ncompp is always zero. If I reduce the array, then what am I looking for in that case? I am lost.

/Sheldon