Subject: Re: idl parallel processing
Posted by siumtesfai on Mon, 22 May 2017 19:06:06 GMT
View Forum Message <> Reply to Message

On Monday, May 22, 2017 at 5:35:49 AM UTC-4, Helder wrote:
> On Monday, May 22, 2017 at 4:59:52 AM UTC+2, Sium T wrote:
>> On Friday, May 19, 2017 at 10:59:07 AM UTC-4, wlandsman wrote:
>>> Yes, you can use the IDL Bridge for this. But if you have IDL 8.4 or later, then more valuable would be using the .HASVALUE() static method. Your code would then be
>>>
>>> result= bytarr(n_elements(siteN))
>>> FOR i= 0,n_elements(siteN)-1 do result[i] = data.hasvalue(siteN[i])
>>>
>>> The reasons this is much faster are (1) you don't need to compute the output vector of WHERE(). All you care about is whether the siteN[i] value is present in the data array-- you don't care where it is. And (2) the .hasvalue() method will return as soon as it finds a single case where the siteN[i] value is present, so you skip having to search the entire data array
>>>
>>> --Wayne
>>>
>>> On Thursday, May 18, 2017 at 6:05:51 PM UTC-4, Sium T wrote:
>>>> Hello,
>>>>
>>>> I have a procedure below. It want to call my procedure in my main program and do parallel processing on the do loop.
>>>>
>>>> How can use the IDL_Bridge . Any suggestion
>>>>
>>>> pro computation,data=data,siteN=siteN,result
>>>>
>>>> result=fltarr(n_elements(siteN))
>>>>
>>>> FOR i= 0,n_elements(siteN)-1 do begin
>>>> y=where(data eq siteN(i))
>>>> if y(0) ge 0 then begin
>>>> result(i)=1
>>>> endif else begin
>>>> result(i)=0
>>>> endelse
>>>> ENDFOR
>>>>
>>>> end
>>
>> Thanks Wayne
>>
>> I tried your method
>> result= bytarr(n_elements(siteN))
>> FOR i= 0,n_elements(siteN)-1 do result[i] = data.hasvalue(siteN[i])

>>
>> However, I got this error message.
>>
>> Object reference type required in this context:
>
> Hi,
> what do you get if you type at the command line:
> help, !version

I have IDL version 8.2.3 . HasValue works with version 8.4 or above.

So I need to use idl_bridge. But it becomes challenging to me

Here is my trial code .
First I have this procedure. It takes for ever to compute Shourly result.

Can you help with how to call this procedure in idl_idlbridge ?

```
;=============================================
pro program1,Rdata,edata,Shourly


  StateN=reform(edata(0,*))
  CountyN=reform(edata(1,*))
  siteN=reform(edata(2,*))
;=======================
scode=reform(Rdata(0,*))
ccode=reform(Rdata(1,*))
snum=reform(Rdata(2,*))
year=reform(Rdata(3,*))
month=reform(Rdata(4,*))
day=reform(Rdata(5,*))
hour=reform(Rdata(6,*))
lats=reform(Rdata(7,*))
lons=reform(Rdata(8,*))
;==============================
Shourly=fltarr(n_elements(siteN),12,31,24)

for s=0,n_elements(stateN)-1 do begin

  z=where(scode eq fix(StateN(s)) and ccode eq fix(CountyN(s)) and snum eq fix(siteN(s)))

  if z(0) ge 0 then begin
    data2=Rdata(*,z)

    FOR mn=1,12 do begin
```

```
    FOR dy=1,31 do begin
      FOR hr=0,23 do begin

        b=where(month eq mn and day eq dy and hour eq hr)

        if b(0) ge 0 then begin
          value=data2(9,b)
          Shourly(s,mn-1,dy-1,hr)=value(0)
        endif else begin
          Shourly(s,mn-1,dy-1,hr)=-9999.0
        endelse

      ENDFOR
    ENDFOR
  ENDFOR

  endif else begin
    Shourly(s,*,*,*)=-9999.0
  endelse

endfor

end

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