Subject: Re: Splitting An Array Of Strings Without Using Loops
Posted by JD Smith on Mon, 28 Jul 2003 23:00:33 GMT
View Forum Message <> Reply to Message

On Mon, 28 Jul 2003 11:24:50 -0700, Rick Towler wrote:

> "Darrick White" wrote...
>
>> It looks like I'm not explaining my problem clearly.
>
>> Is there a way (not knowing what data set input is used) to transform
>> my data into the corresponding result array?
>
> I don't think the issue is one of clarity, but of possibility. Unless
> JD can save you with some magical incarnation of HISTOGRAM you are going
> to have to change your design criteria or use a loop. If performance is
> really that important write this function in C.
>
> -Rick

Come on people. I don't use HISTOGRAM for everything. I use it very rarely, in fact.

How about something like:

```
nums=strsplit(strjoin(data,':'),':',/EXTRACT)
cnts=long(total(byte(data) eq 58b,1))+1L
```

Now you have a list of tuple-counts and the tuples themselves in a long list. You could (yes) use HISTOGRAM or perhaps many other methods to stick these into an array as you describe without looping, but rather than show something you'd forget 5 minutes after dropping it into your code, I'll join Rick in saying that if parsing these strings quickly is this important to you, you'll get better results by re-designing the input format, or pre-parsing them using a language better suited to these manipulations. And on the off chance that you're suffering from the "must-optimize-everything-in-sight" disease, you'll want to make sure a readable and straightforward input loop won't meet your needs before venturing too far into IDL esoterica:

```
b=make_array(/LONG,VALUE=-1,max(cnt),n_elements(data))
for i=0,n_elements(data)-1 do b[0,i]=strsplit(data[i],':',/EXTRACT)
```

Note that there's no integer (long or otherwise) definition of NaN, so I used -1.

JD