Subject: Re: Forcing READ_ASCII output to be a set of strings...
Posted by Jonathan Greenberg on Mon, 14 Mar 2005 21:34:48 GMT
View Forum Message <> Reply to Message

I actually kicked a backwards compatible version of this to the IDL help, and also uploaded it to the RSI user contrib website. You are right, I probably should have renamed the file (we'll see if IDL bounces the upload and asks me to rename it), but here are the mods I made to the "official" read_ascii release:

***

Function call mod:

```
function read_ascii_string, $
   file, $            ; IN:
   RECORD_START=recordStart, $     ; IN: (opt)
   NUM_RECORDS=numRecords, $       ; IN: (opt)
   TEMPLATE=template, $        ; IN: (opt)
   DATA_START=dataStart, $     ; IN: (opt)
   DELIMITER=delimiter, $      ; IN: (opt)
   MISSING_VALUE=missingValue, $   ; IN: (opt)
   COMMENT_SYMBOL=commentSymbol, $ ; IN: (opt)
;    FIELDS=fields, $           ; IN: (opt)    [not implemented]
   VERBOSE=verbose, $          ; IN: (opt)
   HEADER=header, $           ; OUT: (opt)
   COUNT=count, $           ; OUT: (opt)
; **** JONATHAN'S MOD ****
   DATA_TYPE=data_type        ; IDL data type (opt)
```

Modifying the default data_type without needing to use a template:

```
; Keeps the default to floating point
   if n_elements(data_type) eq 0 then data_type=4
   fieldTypesUse   = REPLICATE(data_type, fieldCountUse)
```

***

This would cause zero problems, as far as I can tell, if a user just swapped this in (you'll note that I kept the default a floating point).

The reason I didn't want to use a template was I didn't see a quick way to make it apply a string format to an arbitrarily long ascii file (e.g. If I add a new column to my DB, don't I have to modify the ascii template each time?)

The reason I thought this would be better as a text default is that you can easily go from string -> number, but you can't go backwards. Considering

the input is text, why would they just assume it is filled with floating point numbers? It appears to be completely arbitrary... A string format appears to be the most "generalized" form you could use.

--j

On 3/14/05 12:51 PM, in article 113bua0b4hdv2a2@corp.supernews.com, "Michael Wallace" <mwallace.no.spam@no.spam.swri.edu.invalid> wrote:

> Jonathan Greenberg wrote:
>> I kinda answered my own question -- I took the read_ascii.pro that IDL
>> distributes and simply changed the default behavior on this line:
>>
>> fieldTypesUse = REPLICATE(4L, fieldCountUse)
>>
>> To
>>
>> fieldTypesUse = REPLICATE(7L, fieldCountUse)
>>
>> This appears to work fine... I'm kicking IDL a request to modify this so
>> it'll default to a user-defined format...
>>
>
> Um, this is dangerous. First, if you ever want to distribute this
> application, or at least give it to someone else, you'll have to make
> the same modification in their IDL install. It can be *very* problematic.
>
> Second, why should the default be string rather than float? Just
> because you have a need to use strings for this particular application
> doesn't mean that RSI should change their defaults. For default values,
> a floating point number is the best assumption to make. If the default
> values don't work for you, use a template. This is why templates exist...
>
> -Mike