Subject: Re: Reading complicated ASCII data
Posted by Paul Van Delst[1] on Tue, 29 Jun 2010 14:13:52 GMT
View Forum Message <> Reply to Message

Tone M R wrote:
> Does anyone see a way around these dots?

Use regular expressions to change them to "0.0". I.e. if a "." is not preceded and followed by a digit, then it becomes "0.0".

Although you could, I wouldn't do the above "preprocessing" in IDL. A scripting language like ruby/python/perl would be the go; e.g.

```
#!/usr/bin/env ruby
# Define regular expression for search
re = %r{\s\.\s}
# Inplace edit the file
ARGF.each do |line|
  line.gsub!(re,"0.0")
  puts(line)
end
```

I created a file of text from your example containing:

```
 ------------------------------------------------------------ -----
[block of not-so-interesting information]

   Date   Jan   Feb   Mar   Apr  May   Jun   Jul
      1   0.5   1.4     .   4.7    .     .   0.1
      2   0.6   0.3   3.9     .    .     .     .
      3   5.8   1.6   4.9   0.1  3.1   3.4   4.4
      4   2.0   5.1   1.9   0.2  0.5   6.7   3.3
      5   6.8   0.6   9.7     .  2.7   0.8   1.6
... and so forth, for an entire year. - a 13x31 table of floats.

[new block of non-helpful stuff]

[new block of data for another year]
 ------------------------------------------------------------ ------------
etc..., for a total of ten years.
```

ran it through the above script like so

```
  $ ruby testit.rb blah.txt
```

and got the result:

---------- -----
[block of not-so-interesting information]

Date	Jan	Feb	Mar	Apr	May	Jun	Jul
1	0.5	1.4	0.0	4.7	0.0	0.0	0.1
2	0.6	0.3	3.9	0.0	0.0	0.0	0.0
3	5.8	1.6	4.9	0.1	3.1	3.4	4.4
4	2.0	5.1	1.9	0.2	0.5	6.7	3.3
5	6.8	0.6	9.7	0.0	2.7	0.8	1.6

... and so forth, for an entire year. - a 13x31 table of floats.

[new block of non-helpful stuff]

[new block of data for another year]
---------- ----------
etc..., for a total of ten years.

So there are some spacing issues to be ironed out, but works easypeasy.

cheers,

paulv