Ruby - Bug #12691

CSV performance problem on large files that are misformatted (unclosed quoted field)

08/19/2016 10:09 PM - uberbrady (Brady Wetherington)

Status:	Closed		
Priority:	Normal		
Assignee:			
Target version:			
ruby -v:	ruby 2.3.1p112 (2016-04-26 revision 54768) [x86_64-darwin15]	**Backport:**	2.1: UNKNOWN, 2.2: UNKNOWN, 2.3: UNKNOWN

Description

If you have a large file which has an unclosed quoted field in it, the amount of time it takes for the CSV parser to determine that error increases worse-than-exponentially. My example tests -

60k records - takes 50 seconds to determine 'unclosed quoted field'
120k records - takes 2m45s
240k records - just under 10 minutes

That was from real data that I was running against.

I've attached a simple test.rb script that shows the issue.

The filesize limits prevent me from attaching some sanitzed test files, but I can show how to generate them easily enough.

I start with a file I call "bad_start.csv" -

```
element_one,element_two,element_three
This,is,"a bad start
```

Then I can generate poorly-performing files as follows:

```
yes "This is a very long line that should take up a lot of space in the CSV parser and keep things
 really complicated to make this a better test" |head -n 65535 > 64kblah.txt
cat bad_start.csv 64kblah.txt > bad_64k_blah.txt
```

And that would be a 64k 'bad' file, which I can then test/time as follows:

```
time ./test.rb bad_64k_blah.txt
"Working with file: bad_64k_blah.txt"
"A row is: [\"element_one\", \"element_two\", \"element_three\"]"
/Users/brady/.rbenv/versions/2.3.1/lib/ruby/2.3.0/CSV.rb:1898:in `block in shift': Unclosed quoted
 field on line 2. (CSV::MalformedCSVError)
 from /Users/brady/.rbenv/versions/2.3.1/lib/ruby/2.3.0/CSV.rb:1805:in `loop'
 from /Users/brady/.rbenv/versions/2.3.1/lib/ruby/2.3.0/CSV.rb:1805:in `shift'
 from /Users/brady/.rbenv/versions/2.3.1/lib/ruby/2.3.0/CSV.rb:1747:in `each'
 from /Users/brady/.rbenv/versions/2.3.1/lib/ruby/2.3.0/CSV.rb:1131:in `block in foreach'
 from /Users/brady/.rbenv/versions/2.3.1/lib/ruby/2.3.0/CSV.rb:1282:in `open'
 from /Users/brady/.rbenv/versions/2.3.1/lib/ruby/2.3.0/CSV.rb:1130:in `foreach'
 from ./test.rb:7:in `<main>'

real 0m54.380s
user 0m53.303s
sys 0m0.406s
```

And if you generate larger and larger files, the amount of time that will elapse to determine that the CSV is invalid will increase worse than exponentially.

Another interesting note - when I just used 'yes' by itself (creating lines that just have the text "yes" in them) the problem seemed much, much smaller. So it seems to be related not to a count of lines, but a count of characters.

Associated revisions

Revision 4e5114b0d17aff091267656705c3586ed24b9e1c - 08/22/2016 07:29 AM - nobu (Nobuyoshi Nakada)

csv.rb: performance with very long quoted lines

- lib/csv.rb (CSV#shift): store partial quoted strings in an array
  and join at last, to improve performance with very long quoted
  lines. [ruby-core:76987] [Bug #12691]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55985 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 4e5114b0 - 08/22/2016 07:29 AM - nobu (Nobuyoshi Nakada)

csv.rb: performance with very long quoted lines

- lib/csv.rb (CSV#shift): store partial quoted strings in an array
  and join at last, to improve performance with very long quoted
  lines. [ruby-core:76987] [Bug #12691]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55985 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 08/22/2016 07:30 AM - nobu (Nobuyoshi Nakada)

- Status changed from Open to Closed

Applied in changeset r55985.

---

csv.rb: performance with very long quoted lines

- lib/csv.rb (CSV#shift): store partial quoted strings in an array
  and join at last, to improve performance with very long quoted
  lines. [ruby-core:76987] [Bug #12691]

Files

test.rb	128 Bytes	08/19/2016	uberbrady (Brady Wetherington)