

Ruby - Bug #13941

File.exist? doesn't release gvl while waiting for (f)stat to return

09/26/2017 12:10 AM - graywolf (Gray Wolf)

Status:Closed

Priority:Normal

Assignee:

Target version:

ruby -v:ruby 2.5.0dev (2017-09-26 no_gvl_rb_stat 60025) [x86_64-linux]

Backport:2.2: UNKNOWN, 2.3: UNKNOWN, 2.4: UNKNOWN

Description

When using File.exist?, I noticed that it doesn't release gvl while waiting for system to return from (f)stat call. This causes issues on high-latency network drives, since it freezes whole mri while waiting for system to reach the other end and get (f)stat of the file in question. Attached patched resolved this issue by using rb_thread_call_without_gvl to wrap the (f)stat calls. There is performance hit, but in my opinion not significant one:

```
$ ~/releaseruby_patch/bin/ruby bench.rb
Rehearsal -----
File.exist?:    0.036412    0.056616    0.093028 ( 0.093075)
----- total: 0.093028sec

      user      system      total      real
File.exist?:    0.042953    0.049783    0.092736 ( 0.092804)

$ ~/releaseruby_no_patch/bin/ruby bench.rb
Rehearsal -----
File.exist?:    0.056094    0.026293    0.082387 ( 0.082389)
----- total: 0.082387sec

      user      system      total      real
File.exist?:    0.037250    0.046702    0.083952 ( 0.083956)
```

It's important to stress out the these numbers are over 100000 iterations. Benchmarking script is attached as well as patch to the file.c, simple blocking filesystem to test this thing and test script.

Notice that File.read is not blocking already.

Associated revisions

Revision 66c9d4f55e78765a3dcfbb8d56c1f36fd036c36f - 09/26/2017 04:13 AM - nobu (Nobuyoshi Nakada)

Release gvl while doing (f)stat

At the moment rb_stat function is blocking. This patch changes the behaviour to release the gvl while waiting for OS to return from f(stat).

There is behaviour impact, but not significant (times are for 100000 iterations):

```
$ ~/releaseruby_patch/bin/ruby bench.rb
Rehearsal -----
File.exist?: 0.036412 0.056616 0.093028 ( 0.093075)
----- total: 0.093028sec

      user      system      total      real
File.exist?: 0.042953 0.049783 0.092736 ( 0.092804)

$ ~/releaseruby_no_patch/bin/ruby bench.rb
Rehearsal -----
File.exist?: 0.056094 0.026293 0.082387 ( 0.082389)
----- total: 0.082387sec

      user      system      total      real
```

File.exist?: 0.037250 0.046702 0.083952 (0.083956)

Based on the patch by Wolf wolf@wolfsden.cz at [ruby-core:83012],
with using rb_thread_io_blocking_region for fstat.
[Bug #13941]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@60027 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 66c9d4f5 - 09/26/2017 04:13 AM - nobu (Nobuyoshi Nakada)

Release gvl while doing (f)stat

At the moment rb_stat function is blocking. This patch changes the
behaviour to release the gvl while waiting for OS to return from
f(stat).

There is behaviour impact, but not significant (times are for 100000
iterations):

```
$ ~/releaseruby_patch/bin/ruby bench.rb
Rehearsal -----
File.exist?: 0.036412 0.056616 0.093028 ( 0.093075)
----- total: 0.093028sec
```

user	system	total	real
------	--------	-------	------

File.exist?: 0.042953 0.049783 0.092736 (0.092804)

```
$ ~/releaseruby_no_patch/bin/ruby bench.rb
Rehearsal -----
File.exist?: 0.056094 0.026293 0.082387 ( 0.082389)
----- total: 0.082387sec
```

user	system	total	real
------	--------	-------	------

File.exist?: 0.037250 0.046702 0.083952 (0.083956)

Based on the patch by Wolf wolf@wolfsden.cz at [ruby-core:83012],
with using rb_thread_io_blocking_region for fstat.
[Bug #13941]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@60027 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 6c29f232a733228507a39fbc9f30b89ee960b165 - 10/01/2017 09:29 PM - Eric Wong

NEWS: entries for GVL release in File and Dir

(more to come)

- NEWS: updates for [Bug #13941] and [Feature #13951]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@60091 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 6c29f232 - 10/01/2017 09:29 PM - Eric Wong

NEWS: entries for GVL release in File and Dir

(more to come)

- NEWS: updates for [Bug #13941] and [Feature #13951]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@60091 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision fbb343273666bed162b3190bcfce36c24e89fc6d - 10/03/2017 11:47 PM - Eric Wong

file.c: release GVL around lstat(2)

Like stat(2), lstat(2) can be expensive on slow filesystems and
should not block other threads. There should be a minor, but
not significant slowdowns in single-threaded performance similar
to benchmarks around the more-portable stat(2):
[ruby-core:83012] [Bug #13941]

- file.c (no_gvl_lstat): new function for rb_thread_call_without_gvl
(lstat_without_gvl): new wrapper to replace lstat(2) calls

```
(rb_file_s_lstat): s/lstat/&_without_gvl/  
(rb_file_lstat): ditto  
(rb_file_symlink_p): ditto  
(rb_file_s_fstype): ditto  
(rb_file_expand_path_internal): ditto  
(realpath_rec): ditto  
[ruby-core:83075] [Feature #13963]
```

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@60110 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision fbb34327 - 10/03/2017 11:47 PM - Eric Wong

file.c: release GVL around lstat(2)

Like stat(2), lstat(2) can be expensive on slow filesystems and should not block other threads. There should be a minor, but not significant slowdowns in single-threaded performance similar to benchmarks around the more-portable stat(2):
[ruby-core:83012] [Bug #13941]

- file.c (no_gvl_lstat): new function for rb_thread_call_without_gvl (lstat_without_gvl): new wrapper to replace lstat(2) calls
(rb_file_s_lstat): s/lstat/&_without_gvl/
(rb_file_lstat): ditto
(rb_file_symlink_p): ditto
(rb_file_s_fstype): ditto
(rb_file_expand_path_internal): ditto
(realpath_rec): ditto
[ruby-core:83075] [Feature #13963]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@60110 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 09/26/2017 12:11 AM - graywolf (Gray Wolf)

- File blocking_fs.c added
- File test.rb added
- File bench.rb added

#2 - 09/26/2017 02:21 AM - normalperson (Eric Wong)

wolf@wolfsden.cz wrote:

<https://bugs.ruby-lang.org/issues/13941>

Thanks, I am OK with the minor slowdown to avoid pathological case with slow/dead FS. I'm not sure what others think...

#3 - 09/26/2017 04:13 AM - nobu (Nobuyoshi Nakada)

- Status changed from Open to Closed

Applied in changeset trunk|r60027.

Release gvl while doing (f)stat

At the moment rb_stat function is blocking. This patch changes the behaviour to release the gvl while waiting for OS to return from f(stat).

There is behaviour impact, but not significant (times are for 100000 iterations):

```
$ ~/releaseruby_patch/bin/ruby bench.rb  
Rehearsal -----  
File.exist?: 0.036412 0.056616 0.093028 ( 0.093075)  
----- total: 0.093028sec
```

	user	system	total	real
File.exist?:	0.042953	0.049783	0.092736	(0.092804)

```
$ ~/releaseruby_no_patch/bin/ruby bench.rb
Rehearsal -----
File.exist?: 0.056094 0.026293 0.082387 ( 0.082389)
----- total: 0.082387sec

      user      system      total      real
-----
File.exist?: 0.037250 0.046702 0.083952 ( 0.083956)

Based on the patch by Wolf wolf@wolfsden.cz at \[ruby-core:83012\],
with using rb_thread_io_blocking_region for fstat.
[Bug #13941]
```

Files			
0001-Release-gvl-while-doing-f-stat.patch	3.15 KB	09/26/2017	graywolf (Gray Wolf)
blocking_fs.c	2.12 KB	09/26/2017	graywolf (Gray Wolf)
test.rb	353 Bytes	09/26/2017	graywolf (Gray Wolf)
bench.rb	159 Bytes	09/26/2017	graywolf (Gray Wolf)