Ruby master - Bug #4500

warning: implicit declaration of function 'fdatasync' for OS X

03/14/2011 04:21 AM - Eregon (Benoit Daloze)

Status: Closed
Priority: Normal
Assignee: kosaki (Motohiro KOSAKI)
Target version: 2.0.0
Backport: 

ruby -v:
ruby 1.9.3dev (2011-03-01 trunk 31001) [x86_64-darwin10.6.0]

Description
=begin
While compiling ruby, this warning appears on OS X:

compiling io.c
io.c: In function 'rb_io_fdatasync':
io.c:1433: warning: implicit declaration of function 'fdatasync'

This is a regression caused by the revert (r30821) of r30725: configure.in: Mac OS X wrongly reports it has fdatasync(3).

While fdatasync is defined, there is no header for it. Apparently, it works as expected:
fh = File.open("test.txt", 'w')
fh.write('DATA')
fh.fdatasync # if this line is commented, nothing is written to the file
exit!

But I think we should not trust functions without prototype. So I propose the following patch, which also document the behavior of IO#fdatasync accordingly to r30762.

diff --git a/configure.in b/configure.in
index 01e59f9..8a4e97e 100644
--- a/configure.in
+++ b/configure.in
@@ -940,6 +940,7 @@ AS_CASE("$target_os",
   ac_cv_type_getgroups=gid_t # getgroups() on Rosetta fills garbage
+ac_cv_func_fdatasync=no # Mac OS X wrongly reports it has fdatasync() AC_CACHE_CHECK(for broken crypt with 8bit chars, rb_cv_broken_crypt, [AC_TRY_RUN([ #include din
   ac_cv_lib_crypt_crypt=no
+   ac_cv_type_getgroups=gid_t # getgroups() on Rosetta fills garbage
+   ac_cv_func_fdatasync=no # Mac OS X wrongly reports it has fdatasync() AC_CACHE_CHECK(for broken crypt with 8bit chars, rb_cv_broken_crypt, [AC_TRY_RUN([ #include din
   ac_cv_type_getgroups=gid_t # getgroups() on Rosetta fills garbage
   ac_cv_func_fdatasync=no # Mac OS X wrongly reports it has fdatasync() AC_CACHE_CHECK(for broken crypt with 8bit chars, rb_cv_broken_crypt, [AC_TRY_RUN([ #include din

   static VALUE
   static VALUE

Associated revisions
Revision f4755f21 - 04/03/2011 12:35 PM - kosaki (Motohiro KOSAKI)

- configure.in: disable fdatasync again on Mac OS X. [ruby-core:35493][Bug #4500]
git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@31232 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 31232 - 04/03/2011 12:35 PM - kosaki (Motohiro KOSAKI)

08/07/2021
configure.in: disable fdatasync again on Mac OS X. [ruby-core:35493][Bug #4500]

Revision 31232 - 04/03/2011 12:35 PM - kosaki (Motohiro KOSAKI)
configure.in: disable fdatasync again on Mac OS X. [ruby-core:35493][Bug #4500]

Revision 31232 - 04/03/2011 12:35 PM - kosaki (Motohiro KOSAKI)
configure.in: disable fdatasync again on Mac OS X. [ruby-core:35493][Bug #4500]

Revision 31232 - 04/03/2011 12:35 PM - kosaki (Motohiro KOSAKI)
configure.in: disable fdatasync again on Mac OS X. [ruby-core:35493][Bug #4500]

Revision 31232 - 04/03/2011 12:35 PM - kosaki (Motohiro KOSAKI)
configure.in: disable fdatasync again on Mac OS X. [ruby-core:35493][Bug #4500]

Revision 31232 - 04/03/2011 12:35 PM - kosaki (Motohiro KOSAKI)
configure.in: disable fdatasync again on Mac OS X. [ruby-core:35493][Bug #4500]

History

#1 - 04/03/2011 09:33 PM - kosaki (Motohiro KOSAKI)
- Status changed from Open to Assigned
- Assignee set to kosaki (Motohiro KOSAKI)

=begin
To be honest, I'm not sure Apple's system function support policy. But I'm sure your suggestions were right in past. Then, I'll commit this proposal as is.

Thanks.
=end

#2 - 04/04/2011 07:07 PM - kosaki (Motohiro KOSAKI)
- Status changed from Assigned to Closed

=begin
=end