

Ruby master - Bug #4957

IO.copystream seems make hangup on cygwin (was 2011-07-01 Snapshot build)

07/01/2011 12:13 PM - mfurui (Mitsuru FURUI)

Status:	Closed	
Priority:	Normal	
Assignee:		
Target version:	1.9.3	
ruby -v:	ruby 1.9.3dev (2011-07-01 trunk 32343) [i386-cygwin]	Backport:

Description

2011-07-01 Snapshot → test_io.rb test_thread.rb 12

```
[~/tmp/snapshot] tar xzf snapshot.tar.gz; cd snapshot; ./configure --enable-shared; make ; make test
```

```
test_attr.rb ..
test_autoload.rb .....
test_block.rb .....
test_class.rb .....
test_eval.rb .....
test_exception.rb .....
test_finalizer.rb .
test_flip.rb .
test_flow.rb .....
test_fork.rb ....
test_gc.rb ..
test_io.rb .....FFFFFFFFF.
test_jump.rb .....
test_literal.rb .....
test_load.rb ..
test_marshal.rb .
test_massign.rb .....
test_method.rb
.....
test_objectspace.rb ...
test_proc.rb .....
test_struct.rb .
test_syntax.rb .....
test_thread.rb .....FF.....
```

[#245](#) test_io.rb:

```
at_exit { p :foo }
```

```
megacontent = "abc" * 12345678
#File.open("megasrc", "w") {|f| f << megacontent }

Thread.new { sleep rand*0.2; Process.kill(:INT, $$) }
```

```
r1, w1 = IO.pipe
r2, w2 = IO.pipe
t1 = Thread.new { w1 << megacontent; w1.close }
t2 = Thread.new { r2.read; r2.close }
IO.copy_stream(r1, w2) rescue nil
w2.close
r1.close
t1.join
t2.join
#=> killed by SIGKILL (signal 9) (timeout) megacontent-copy_stream
```

[#246](#) test_io.rb:

```
at_exit { p :foo }
```

```
megacontent = "abc" * 12345678
#File.open("megasrc", "w") {|f| f << megacontent }

Thread.new { sleep rand*0.2; Process.kill(:INT, $$) }

r1, w1 = IO.pipe
r2, w2 = IO.pipe
t1 = Thread.new { w1 << megacontent; w1.close }
t2 = Thread.new { r2.read; r2.close }
IO.copy_stream(r1, w2) rescue nil
w2.close
r1.close
t1.join
t2.join
#=> killed by SIGKILL (signal 9) (timeout) megacontent-copy_stream
```

#247 test_io.rb:
at_exit { p :foo }

```
megacontent = "abc" * 12345678
#File.open("megasrc", "w") {|f| f << megacontent }

Thread.new { sleep rand*0.2; Process.kill(:INT, $$) }

r1, w1 = IO.pipe
r2, w2 = IO.pipe
t1 = Thread.new { w1 << megacontent; w1.close }
t2 = Thread.new { r2.read; r2.close }
IO.copy_stream(r1, w2) rescue nil
w2.close
r1.close
t1.join
t2.join
#=> killed by SIGKILL (signal 9) (timeout) megacontent-copy_stream
```

#248 test_io.rb:
at_exit { p :foo }

```
megacontent = "abc" * 12345678
#File.open("megasrc", "w") {|f| f << megacontent }

Thread.new { sleep rand*0.2; Process.kill(:INT, $$) }

r1, w1 = IO.pipe
r2, w2 = IO.pipe
t1 = Thread.new { w1 << megacontent; w1.close }
t2 = Thread.new { r2.read; r2.close }
IO.copy_stream(r1, w2) rescue nil
w2.close
r1.close
t1.join
t2.join
#=> killed by SIGKILL (signal 9) (timeout) megacontent-copy_stream
```

#249 test_io.rb:
at_exit { p :foo }

```
megacontent = "abc" * 12345678
#File.open("megasrc", "w") {|f| f << megacontent }

Thread.new { sleep rand*0.2; Process.kill(:INT, $$) }

r1, w1 = IO.pipe
r2, w2 = IO.pipe
t1 = Thread.new { w1 << megacontent; w1.close }
t2 = Thread.new { r2.read; r2.close }
IO.copy_stream(r1, w2) rescue nil
```

```
w2.close
r1.close
t1.join
t2.join
#=> killed by SIGKILL (signal 9) (timeout) megacontent-copy_stream
```

[#250](#) test_io.rb:
at_exit { p :foo }

```
megacontent = "abc" * 12345678
#File.open("megasrc", "w") {|f| f << megacontent }
```

```
Thread.new { sleep rand*0.2; Process.kill(:INT, $$) }
```

```
r1, w1 = IO.pipe
r2, w2 = IO.pipe
t1 = Thread.new { w1 << megacontent; w1.close }
t2 = Thread.new { r2.read; r2.close }
IO.copy_stream(r1, w2) rescue nil
w2.close
r1.close
t1.join
t2.join
#=> killed by SIGKILL (signal 9) (timeout) megacontent-copy_stream
```

[#251](#) test_io.rb:
at_exit { p :foo }

```
megacontent = "abc" * 12345678
#File.open("megasrc", "w") {|f| f << megacontent }
```

```
Thread.new { sleep rand*0.2; Process.kill(:INT, $$) }
```

```
r1, w1 = IO.pipe
r2, w2 = IO.pipe
t1 = Thread.new { w1 << megacontent; w1.close }
t2 = Thread.new { r2.read; r2.close }
IO.copy_stream(r1, w2) rescue nil
w2.close
r1.close
t1.join
t2.join
#=> killed by SIGKILL (signal 9) (timeout) megacontent-copy_stream
```

[#252](#) test_io.rb:
at_exit { p :foo }

```
megacontent = "abc" * 12345678
#File.open("megasrc", "w") {|f| f << megacontent }
```

```
Thread.new { sleep rand*0.2; Process.kill(:INT, $$) }
```

```
r1, w1 = IO.pipe
r2, w2 = IO.pipe
t1 = Thread.new { w1 << megacontent; w1.close }
t2 = Thread.new { r2.read; r2.close }
IO.copy_stream(r1, w2) rescue nil
w2.close
r1.close
t1.join
t2.join
#=> killed by SIGKILL (signal 9) (timeout) megacontent-copy_stream
```

[#253](#) test_io.rb:
at_exit { p :foo }

```
megacontent = "abc" * 12345678
```

```
#File.open("megasrc", "w") {|f| f << megacontent }

Thread.new { sleep rand*0.2; Process.kill(:INT, $$) }

r1, w1 = IO.pipe
r2, w2 = IO.pipe
t1 = Thread.new { w1 << megacontent; w1.close }
t2 = Thread.new { r2.read; r2.close }
IO.copy_stream(r1, w2) rescue nil
w2.close
r1.close
t1.join
t2.join
#=> killed by SIGKILL (signal 9) (timeout) megacontent-copy_stream
```

[#254](#) test_io.rb:
at_exit { p :foo }

```
megacontent = "abc" * 12345678
#File.open("megasrc", "w") {|f| f << megacontent }

Thread.new { sleep rand*0.2; Process.kill(:INT, $$) }

r1, w1 = IO.pipe
r2, w2 = IO.pipe
t1 = Thread.new { w1 << megacontent; w1.close }
t2 = Thread.new { r2.read; r2.close }
IO.copy_stream(r1, w2) rescue nil
w2.close
r1.close
t1.join
t2.join
#=> killed by SIGKILL (signal 9) (timeout) megacontent-copy_stream
```

[#911](#) test_thread.rb:
th = Thread.new {sleep 2}
th.join(1)
th.join
#=> not finished in 3 seconds

[#912](#) test_thread.rb:
require 'timeout'
th = Thread.new {sleep 2}
begin
Timeout.timeout(1) {th.join}
rescue Timeout::Error
end
th.join
#=> not finished in 3 seconds

FAIL 12/936 tests failed
make: *** [yes-btest-ruby] Error 1

[~/tmp/snapshot] uname -a
CYGWIN_NT-5.1 kenken 1.7.6(0.230/5/3) 2010-08-16 16:06 i686 Cygwin

[~/tmp/snapshot] ./ruby -v
ruby 1.9.3dev (2011-07-01 trunk 32343) [i386-cygwin]

Related issues:

Related to Ruby master - Bug #5055: ruby-1.9.2-p290 Test Error @Cygwin	Rejected	07/19/2011
Has duplicate Ruby master - Bug #5027: Snapshot(2011-07-14) Build Test Error ...	Closed	07/14/2011

History

#1 - 07/08/2011 05:55 PM - kosaki (Motohiro KOSAKI)

- Priority changed from Normal to 3

#2 - 07/10/2011 01:51 PM - kosaki (Motohiro KOSAKI)

cygwin
close

#3 - 07/21/2011 01:56 PM - kosaki (Motohiro KOSAKI)

- Subject changed from 2011-07-01 Snapshot build to IO.copystream seems make hangup on cygwin (was 2011-07-01 Snapshot build)
- Category set to core

close

#4 - 07/25/2011 08:08 PM - naruse (Yui NARUSE)

- Status changed from Open to Feedback

#5 - 07/26/2011 12:24 AM - naruse (Yui NARUSE)

- Status changed from Feedback to Closed

[#5055](#) close