Ruby master - Bug #3008

Can't compile ruby without ruby

03/25/2010 01:37 AM - docwhat (Christian Höltje)

Status:	Closed		
Priority:	Normal		
Assignee:			
Target version:			
ruby -v:	1.9.1-p376	**Backport:**	

Description

=begin
What happened:
I tried to compile ruby 1.9.1-p376 on a new unix system(s). (Linux and Solaris).
I get the error:
executable host ruby is required. use --with-baseruby option.
make: *** [miniprelude.c] Error 1

What I expected:
I expect that ruby should compile.

There are no docs for how to get around this. How can I get ruby on a system if I can't compile ruby?
=end

History

#1 - 03/25/2010 01:57 AM - mame (Yusuke Endoh)

- Status changed from Open to Rejected

=begin
Hi,

2010/3/25 Christian Holtje redmine@ruby-lang.org:

> I tried to compile ruby 1.9.1-p376 on a new unix system(s). (Linux and Solaris).
> I get the error:
> executable host ruby is required. use --with-baseruby option.
> make: *** [miniprelude.c] Error 1

Yes, compiling trunk requires ruby to preprocess and generate code.

> How can I get ruby on a system if I can't compile ruby?

Please build and install tarball package first. It does not require
ruby to be built.

ftp://ftp.ruby-lang.org/pub/ruby/1.9/ruby-1.9.1-p378.tar.bz2

--
Yusuke ENDOH mame@tsg.ne.jp
=end

#2 - 03/25/2010 03:05 AM - docwhat (Christian Höltje)

=begin
I'm not using "trunk".

I'm using ftp://ftp.ruby-lang.org/pub/ruby/1.9/ruby-1.9.1-p376.tar.bz2
^

I just re-downloaded it to compare and it is exactly the same source code.

I'll try p378 and see if that fixes it.
=end

#3 - 03/25/2010 03:49 AM - docwhat (Christian Höltje)

=begin
Okay. It seems the problem was that my sources didn't have their proper timestamps. I worked around this by adding this step *prior* to running configure && make && make install

This "fixes" the timestamps for the files that would require ruby to be pre-installed. It also prevents autotools from trying to run again.

touch \

.revision.time \

ChangeLog \

configure \

enc/trans/big5.c \

enc/trans/chinese.c \

enc/trans/escape.c \

enc/trans/gb18030.c \

enc/trans/gbk.c \

enc/trans/iso2022.c \

enc/trans/japanese.c \

enc/trans/japanese_euc.c \

enc/trans/japanese_sjis.c \

enc/trans/korean.c \

enc/trans/newline.c \

enc/trans/single_byte.c \

enc/trans/utf_16_32.c \

encdb.h \

ext/ripper/eventids1.c \

ext/ripper/eventids2table.c \

ext/ripper/ripper.c \

ext/ripper/ripper.y \

ext/ripper/y.output \

golf_prelude.c \

insns.inc \

insns_info.inc \

known_errors.inc \

lex.c \

miniprelude.c \

newline.c \

node_name.inc \

opt_sc.inc \

optinsn.inc \

optunifs.inc \

parse.c \

parse.h \

revision.h \

string.c \

transdb.h \

version.h \

vm.inc \

vmtc.inc

=end

#4 - 03/25/2010 03:51 AM - coatl (caleb clausen)

=begin
Some of the ruby 1.9 build scripts are written in ruby. So, you must have an existing ruby already in order to build ruby 1.9. I had thought that the usual workaround was to build a ruby 1.8 release first (which doesn't have this restriction) and then use that to bootstrap 1.9. (I think perhaps Yusuke misspoke when he pointed you at a 1.9 tarball.....?)

This is a known limitation of ruby 1.9. It's not a terribly happy situation, but its not likely to be fixed soon either.
=end

#5 - 03/25/2010 03:54 AM - nielsenj (Jonathan Nielsen)

=begin
I have built ruby 1.9.1 (p376 and p378) on several computers without an existing ruby interpreter. It has always successfully built a 'miniruby' to interpret scripts in the build process when I have used it. I have done this successfully in Cygwin, Ubuntu Linux and Debian Linux.
=end

#6 - 03/25/2010 04:42 AM - nobu (Nobuyoshi Nakada)

- Status changed from Rejected to Open

- Priority changed from 7 to 3

=begin

=end

#7 - 03/25/2010 04:47 AM - nobu (Nobuyoshi Nakada)

- Status changed from Open to Closed

- % Done changed from 0 to 100

=begin
This issue was solved with changeset r27037.
Christian, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

=end