

Re: why can't Ruby load .gem files directly?

Source: <http://newsgroups.derkeiler.com/Archive/Comp/comp.lang.ruby/2005-10/msg00106.html>

- *From:* Eivind Eklund <eeklund@xxxxxxxxxx>
 - *Date:* Mon, 3 Oct 2005 02:34:45 +0900
-

On 10/2/05, Joshua Haberman <joshua@xxxxxxxxxxxxxxxxxx> wrote:

> On Oct 2, 2005, at 9:10 AM, Eivind Eklund wrote:

>> On 10/2/05, Devin Mullins <twifkak@xxxxxxxxxxxxx> wrote:

>>> Joshua Haberman wrote:

>>>> Say that you had two implementations of gems with identical feature

>>>> sets, but one used the "gem" command to do everything and one let

>>>> you

>>>> use .gem files directly. I would *much* prefer using the latter.

>>>>

>>>

>>> Actually, it doesn't sound like they're mutually exclusive. Stick a

>>> little flag in the metadata.gz that says "I don't have a compile

>>> step!"

>>> and then it can be stuck on the \$LOAD_PATH directly. Otherwise, upon

>>> require-ing something from inside the gem, you'll get an error

>>> "Install

>>> meeee!!"

>>>

>>>

>>>> Yes, I don't know the details of the current controversy (ie. why

>>>> Debian is having a problem packaging gems)

>>>>

>>>>

>>> Neither do I. Are they trying to repackage gems as... err..

>>> whatever the

>>> apt-get file extension is? If so, why?

>>>

>>>

>>> They're trying to repackage software that the author assume to be

>>> Gem-distributed as Debian packages (.deb, IIRC). The reason for this

>>> is simple: To provide Ruby software on Debian for Debian users in the

>>> way Debian users are used to. These users do not particularly care

>>> about Ruby, and definately do not want to have to know one packaging

>>> system for handling Ruby software, and another for handling Perl, and

>>> another for handling Java, and another for handling Haskell, and

>>> another for handling Python, and another way for handling C code, and

>>> ...

>>>

>>> Instead, they want they software to Just Work Like All Other Software,

Re: why can't Ruby load .gem files directly?

- >> the way they are used to, and to not have to care about the language
- >> the software is written in at all.
- >
- > May I request that we talk about technical issues like grown-up
- > engineers, not like guests on "Jerry Springer?"

These are user interface issues, and I am trying to explain them in a way that get the overall top level user issues communicated.

Ruby-talk is the wrong list for the detailed technical discussion.

Also, the technical issues are reasonably easy to solve, as long as there is acceptance that they need to be solved, and that they need to be discussed at a technical level.

- > Tell me if this is an accurate list of the problems packaging gems:
- >
- > 1. Gems provide more guarantees than Debian packages can

Different guarantees, and being unable to support the guarantees that Debian packages give. Specifically, Gems do not support using the directory layout that Debian users expect. Gems generally go in a single directory. Some ruby developers end up relying on that single directory because Gems lack the infrastructure for them to access things any other way.

I've posted more exact details of the stuff that's necessary to support to rubygems developer list as references a year and half or so ago. In the last couple of days, I've posted this as extracted and minimized text with very concrete lists, to be able to start building the abstractions necessary. That's under the subject of "Gems and repackaging, hopefully helpful".

There's also the problem that gems are hard to patch, though that's resolvable by using the original source as long as the authors are not led to rely on the RubyGems layout, and end up being just rude to those trying to give them patches to make this work.

Eivind.

- **Follow-Ups:**

- ◆ **[Re: why can't Ruby load .gem files directly?](#)**

◇ From: Joshua Haberman

- **References:**

- ◆ **[why can't Ruby load .gem files directly?](#)**

◇ From: Joshua Haberman

- ◆ **[Re: why can't Ruby load .gem files directly?](#)**

Re: why can't Ruby load .gem files directly?

◇ *From:* Ryan Leavengood

◆ ***Re: why can't Ruby load .gem files directly?***

◇ *From:* Joshua Haberman

◆ ***Re: why can't Ruby load .gem files directly?***

◇ *From:* Devin Mullins

- Prev by Date: ***1.8.3 or 1.8.4: will define method allow blocks, defaults, etc.?***
- Next by Date: ***Re: Eval and block problems***
- Previous by thread: ***Re: why can't Ruby load .gem files directly?***
- Next by thread: ***Re: why can't Ruby load .gem files directly?***
- Index(es):
 - ◆ ***Date***
 - ◆ ***Thread***