

Re: Need a compelling argument to use Django instead of Rails

Re: Need a compelling argument to use Django instead of Rails

Source: <http://coding.derkeiler.com/Archive/Python/comp.lang.python/2006-08/msg00542.html>

- *From:* Bruno Desthuilliers <onurb@xxxxxxxxxxxx>
 - *Date:* Thu, 27 Jul 2006 10:33:02 +0200
-

Jaroslav Zabiello wrote:

On Wed, 26 Jul 2006 18:20:44 +0200, Bruno Desthuilliers wrote:

May I suggest that you learn some Lisp dialect ?

Nope. I hate Lisp syntax.

This should not prevent you from learning it – at least, you'd then avoid making dumb statements...

Of course, I you like, you can freeze every object you want and nobody can be able to open and change it. You can also trace all activity for changing something to objects because Ruby has nice system hooks implemented.

`__getattr__` / `__setattr__` / `__delattr__`

This is not the same. Ruby can also lock access to attributes.

Please re-read the doc for these methods.

Re: Need a compelling argument to use Django instead of Rails

Freezing is different. It can freeze any object so you will not be able to add or delete any method. Once frozen object cannot be unfrozen. So you can be sure that nobody will change your classes if you do not like.

And of course import hooks.

Python?? Where?

RTFM:

<http://www.python.org/doc/2.3.5/lib/built-in-funcs.html>

What about learning Python instead of repeating arguments from clueless people ?

Ruby
has nice security system (private, protected, public scopes for
methods and
attributes,

This is not "security", this is data-hiding.

No. Data hiding are in Python.

Oh, yes ? Where, exactly ?

Ruby uses security similar to Java. If the
class has method marked as private it cannot be used in children classes.

Python methods are attributes, so from a Python POV, this is still
data-hiding. But granted, I should have used "language-enforced access
restriction" instead. So let me rephrase:

This is not "security", this is language-enforced access restriction.

And IIRC, Ruby's attributes are always private

Re: Need a compelling argument to use Django instead of Rails

Re: Need a compelling argument to use Django instead of Rails

Yes and no. Yes, because you cannot read them directly without accessors.
And no, because you can read them if you set proper accessor.

Adding accessors doesn't make the attributes public.

In past Python had Bastion module
(<http://www.python.org/doc/lib/module-Bastion.html>)
but it was rejected and now Python has nothing.

Seems like Zope is not doing so bad...

Zope is only a great application.

Zope is a framework and an application server.

Invision Power Board or ezPublish are
also great application, but it does not mean that PHP is great language. :)

You can't say that Python is dumb when it comes to security and reject
an example of a Python framework that address the problem.

Python has no security at all. I has only convention and mangling.

Please define "security". I fail to see how language-enforced access
restriction (and mandatory declarative static typing etc) relates to
'security'. As far as I'm concerned, security is about protecting a
system from piracy, not about inflicting useless pain to programmers.

Of
course somebody can say, it is enough, and maybe it is. But I think, that
this might be another reason why Java guys prefer Ruby to Python.

Agreed – but this all boil down to Ruby having a more canonical object
model than Python.

—

Re: Need a compelling argument to use Django instead of Rails

Re: Need a compelling argument to use Django instead of Rails

bruno desthuilliers

```
python -c "print '@'.join(['.'.join([w[::-1] for w in p.split('.')]) for  
p in 'onurb@xxxxxxxxxxx'.split('@')]"
```

.