

## Ruby trunk - Bug #2601

### BasicObject.initialize should raise an error

01/13/2010 11:55 AM - marcandre (Marc-Andre Lafortune)

<b>Status:</b> Closed	
<b>Priority:</b> Normal	
<b>Assignee:</b>	
<b>Target version:</b> 1.9.2	
<b>ruby -v:</b> ruby 1.9.2dev (2010-01-12 trunk 26295) [x86_64-darwin10.2.0]	<b>Backport:</b>
<b>Description</b>	
<pre>=begin While having fun cloning BasicObject (see issue 2443), I found the amusing bug that :initialize could successfully be sent to BasicObject, creating all sorts of havoc.  \$ rubydev -e "BasicObject.send :initialize; puts Object.ancestors" # =&gt; Loops indefinitely  \$ rubydev -e "BasicObject.send :initialize, String; puts 42.upcase" # =&gt; Segmentation Fault  It was even possible to change the root of all Ruby classes without running into trouble: \$ rubydev -e ' SuperBasicObject = BasicObject.clone BasicObject.send :initialize, SuperBasicObject puts BasicObject.superclass ' # ==&gt; SuperBasicObject  I fixed it, although changes might be required depending on the outcome of issue #2443. =end</pre>	

#### History

##### #1 - 01/13/2010 03:35 PM - matz (Yukihiro Matsumoto)

```
=begin
Hi,
```

In message "Re: [ruby-core:27577] [Bug #2601] BasicObject#initialize should raise an error" on Wed, 13 Jan 2010 11:55:23 +0900, Marc-Andre Lafortune [redmine@ruby-lang.org](mailto:redmine@ruby-lang.org) writes:

[While having fun cloning BasicObject (see issue 2443), I found the amusing bug that :initialize could successfully be sent to BasicObject, creating all sorts of havoc.

Thank you for reporting. Although it's fixed, I have to mention that it is BasicObject.initialize, not BasicObject#initialize.

```
matz.
```

```
=end
```

##### #2 - 01/13/2010 11:40 PM - marcandre (Marc-Andre Lafortune)

- Subject changed from BasicObject#initialize should raise an error to BasicObject.initialize should raise an error

```
=begin
```

```
=end
```