## Ruby master - Feature #4065
### Rename or alias module#append_features to module#include_module

11/17/2010 10:05 PM - Chauk-Mean (Chauk-Mean Proum)

<table>
<thead>
<tr>
<th>Status:</th>
<th>Rejected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td>2.0.0</td>
</tr>
</tbody>
</table>

**Description**

```text
Hi,

For Object#extend, the hook/callbacks methods are extend_object and extended.
For Module#include, the hook/callbacks methods are append_features and included.

The name append_features is a bit confusing / inconsistent.
What about renaming append_features to include_module (similar as extend_object for extend) ?
Alternatively, include_module could be defined as an alias to append_features (the latter could be made deprecated).

Thanks.
```

**History**

### #1 - 11/27/2010 05:14 PM - Chauk-Mean (Chauk-Mean Proum)

```
=begin

Given the discussion from [http://www.ruby-forum.com/topic/469691#new](http://www.ruby-forum.com/topic/469691#new),

I revised my proposal as following:

For Object#extend, the hook/callbacks methods would be :

- pre_extended (instead of extend_object)
- extended (no change)

For Module#include, the hook/callbacks methods would be :

- pre_included (instead of append_features)
- included (no change)

The advantages of this renaming are:

- there is a name relationship between the three related methods e.g. include, pre_included, included.
- the pre prefix clearly means that those callbacks are called before the actual extension/inclusion (if any).
- both included/extended can mean has or was included/extended so this take care of the "English meaning" issue regarding include vs extend.

Thanks.
```

### #2 - 11/27/2010 11:55 PM - matz (Yukihiro Matsumoto)

```
=begin

Hi,

In message "Re: [ruby-core:33419] [Ruby 1.9-Feature#4065] Rename or alias module#append_features to module#include_module" on Sat, 27 Nov 2010 17:14:44 +0900, Chauk-Mean Proum redmine@ruby-lang.org writes:

| Issue #4065 has been updated by Chauk-Mean Proum. |
| Given the discussion from [http://www.ruby-forum.com/topic/469691#new](http://www.ruby-forum.com/topic/469691#new), |
| I revised my proposal as following: |
| For Object#extend, the hook/callbacks methods would be: |
| - pre_extended (instead of extend_object) |
| - extended (no change) |
| For Module#include, the hook/callbacks methods would be: |
| - pre_included (instead of append_features) |
| - included (no change) |

I don't think it is worth breaking many existing programs that use extend_object / append_feature for aesthetic reason. Besides that,
```

09/18/2021
unlike extended, extend_object is not a hook.

=end

#3 - 05/27/2011 10:20 PM - marcandre (Marc-Andre Lafortune)
- Status changed from Open to Rejected