

Bugs & Wish list

PluginModule interface Crash on Custom Module Insert | Tiki Wiki CMS Groupware :: Development

PluginModule interface Crash on Custom Module Insert

Status

✖ Closed

Subject

PluginModule interface Crash on Custom Module Insert

Version

10.x

11.x

Category

- Usability

Feature

Wiki Plugin (extends basic syntax)

Modules

Resolution status

New

Submitted by

jcarter

Volunteered to solve

manivannans

Lastmod by

Pascal St-Jean

Rating

★ ★ ★ ★ ★ (0) ?

Description

Summary

The plugin for inserting modules into a wiki page has a bug where it crashes whenever it encounters custom modules. The following JS applies some duct-tape to the issue, and prevents the crash. I didn't get too deep into the workings of it, just got it to work. Someone who has worked on it before probably has a better understanding of the whole system, and could determine if this was the way to do it or not (the change could have cascading effects on behavior of pluginModule and how it deals with custom modules?)

Note: crash outputs the following error in js console: `Uncaught TypeError: Cannot read property 'params' of undefined tiki-js.js:1304`

Procedure

To fix the issue, one can do the following:

Modify `lib/tiki-js.js` line 1303. Change the else statement to an else if, checking for the undefined object which would break things.

Note that the code in the examples below is from tiki 10, not 11, and may be a little bit different in 11, but the problem remains in 11 and the line changed is the same, and fixes it in 11 too.

So, in this block



```
//For PluginModule, add selected module parameters to the plugin edit form if (type == 'module')
{ //isolate the module parameter object so it will be shown first in the form var onlymod =
{"params":{"module": meta.params.module}}; //user has not changed the module selection since
opening the form if (typeof selectedMod == 'undefined') { //pick up the parameters of the saved
module parameter if (typeof pluginArgs.module != 'undefined') { //this orders the module
parameter first, module related parameters second, other PluginModule parameters besides
module last meta.params = $.extend(onlymod.params,
tiki_module_params[pluginArgs.module].params, meta.params); //Use the module description
meta.params.module.description = tiki_module_params[pluginArgs.module].description;
//otherwise pick up the parameters of the first module option since that will be selected
automatically } else { meta.params = $.extend(onlymod.params,
tiki_module_params[meta.params.module.options[0].value].params, meta.params);
meta.params.module.description =
tiki_module_params[meta.params.module.options[0].value].description; } //user has selected
another module while the form was open - pick up parameters for the selected module } else {
meta.params = $.extend(onlymod.params, tiki_module_params[selectedMod].params,
meta.params); meta.params.module.description = tiki_module_params[selectedMod].description;
} }
```

change this statement



```
} else { meta.params = $.extend(onlymod.params, tiki_module_params[selectedMod].params,
meta.params); meta.params.module.description = tiki_module_params[selectedMod].description;
}
```

to



```
} else if (typeof tiki_module_params[selectedMod] != 'undefined') { meta.params =
$.extend(onlymod.params, tiki_module_params[selectedMod].params, meta.params);
meta.params.module.description = tiki_module_params[selectedMod].description; }
```

Context

The issue is caused because the module plugin tries to get the parameters for the object to display in the interface, but for custom modules there are none. The `tiki_module_params` object stores a parameters object for each module as the value for an attribute named after the module (e.g., `tiki_module_params[mymodule]` would return the params object for `[mymodule]`). The problem is that the if there are no parameters for the object, it sets the value for the module attribute to false, and the unmodified else statement above from `lib/tiki-js.js` will try to read the attribute `.params` from the false object, which will fail, and the js will crash.

Another possible solution to this issue could be to populate the value as an empty object instead of false, in `tiki-jsmodule.php`.

Solution

I had fixed "pluginModule interface Crash" issue in 12.x.

Please check this commit id: 48363

Thanks jcarter.

Updated:

commit backported to 11.x and 10.x

11.x -> commit id: 48657

10.x -> commit id: 48658

Importance

8

Easy to solve?

8

Priority

64

Demonstrate Bug (Tiki 19+)

Please demonstrate your bug on show2.tiki.org

Version: trunk ▼

Demonstrate Bug (older Tiki versions)

Please demonstrate your bug on show.tikiwiki.org

Version: 18.x ▼

Ticket ID

4668

Created

Monday 19 August, 2013 19:50:26 GMT-0000

by jcarter

LastModif

Saturday 23 November, 2013 22:33:33 GMT-0000

Comments



Marc Laporte 21 Nov 13 19:29 GMT-0000

I don't understand why <http://jcarter-11387-4668.show.tikiwiki.org/tiki-admin.php?page=general> is on 11.x

Is the issue on 11.x or 12.x?

If it's both, commit should be backported to 11.x

Thanks!



jrcarter 21 Nov 13 20:27 GMT-0000

I submitted the bug because I encountered it on 10 and 11, and submitted the bug for 11, demonstrating it at that time. If it has not been pushed to 11, then I don't see why it wouldn't still be there, and you're right, it should probably be backported.

I tested on 12 too, just now, pulling the latest code from 12 with SVN it is still broken for me, but with an undefined call on a different line. Had the same issue on trunk.



Pascal St-Jean 23 Nov 13 22:34 GMT-0000

See show instance for fix demo. I tried various custom modules and they all worked. If your use case still breaks then there is probalby something in your custom module causing the issue at which point we need to evaluate that specific use case

thank you

Attachments

filename	created	hits	comment	version	filetype
----------	---------	------	---------	---------	----------

No attachments for this item

The original document is available at
<https://dev.tiki.org/item4668-PluginModule-interface-Crash-on-Custom-Module-Insert>