

Bugs & Wish list

Packages Duplicate Vendors | Tiki Wiki CMS Groupware :: Development

Packages Duplicate Vendors

Status

Open

Subject

Packages Duplicate Vendors

Version

20.x

21.x

Category

- Conflict of two features (each works well independently)

Feature

Packages

Resolution status

New

Submitted by

drsassafras

Lastmod by

drsassafras

Rating

(0)

Description

Packages run composer independently of the regular composer dependencies.

This creates a situation where package dependencies might conflict.

For example, if you install media alchemy, it requires Symfony console, so Symfony console gets installed via our main dependencies, (in vendor_bundled) also as a dependency of the packages install, (in vendor folder).

Now there two different versions of the same vendor that are both getting auto-loaded. Right now the Package-required Symfony console version is significantly older than that of the main Symfony Console and it overrides the one that Tiki has been tested with.

This, of course, provides a huge opportunity for difficult to find errors.

Perhaps as a solution, we can remove in a VendorsClean equivalent of packages, each dependency that the main composer has already installed. This way it won't severely cripple Tiki when new packages are installed.

Importance

9

Priority

45

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

7177

Created

Monday 16 September, 2019 21:29:41 GMT-0000
by drsassafras

LastModif

Monday 16 September, 2019 21:29:41 GMT-0000

Comments



Marc Laporte 27 Feb 20 17:09 GMT-0000

I suspect I hit this here: [item7294-Upgrade-from-20-to-21-PHP-Fatal-error-Uncaught-Error-Call-to-undefined-method-Tiki-Command-ProfileExport-ActivityRuleSet-setHidden-in-lib-core-Tiki-Command-ConsoleApplicationBuilder-php-543](#)

And as you say, "difficult to find errors."

Workaround: I renamed my 3 vendor directories, did svn up, and then proceeded normally.



Marc Laporte 27 Feb 20 22:50 GMT-0000

But that was short-lived

I ran into other issues after

I removed alchemy package and:

Fatal error: Interface

'Symfony\Component\HttpFoundation\File\MimeType\MimeTypeGuesserInterface' not found in /var/www/virtual/21x.avan.tech/html/lib/Alchemy/Guesser.php on line 12



drsassafras 28 Feb 20 20:54 GMT-0000

hmm. Ya, in dependency management, other packages could be relying on the older version of Symfony, so the other packages would need to know a newer version of Symfony was installed and it would update to a compatible version. 😞 So it won't be as easy as just removing the duplicate dependencies. The second composer will need to know all the existing packages (and versions) that are installed to be able to select and install the right software.



drsassafras 28 Feb 20 21:14 GMT-0000

And let's not forget that the current Tiki software needs to match specific dependency versions. So when one is updating Tiki, if the package requirements change and the package has not been updated, it could cause fatal errors. So all the code in Tiki that interacts with a package, would need to be behind a version check (determined by Tiki) so that code can not execute if the package version is incorrect, or to ensure that packages are updated to a compatible version while upgrading Tiki.

Although this might not be enough. If Tiki updates vendor files to a new version and it is fine in Tiki, but incompatible with a package, then even if the package version is correct, it could still cause a fatal error.

So this brings us back to dependency management. All the installed packages need to be checked not only against its compatible version but also that the new vendor files match and are compatible with the installed package.

I wonder if we might be better to do the last check manually, by including as a part of the release process a check that all the packages can be installed and that they do not cause any fatal errors. They can then all be updated with that Tiki Version that they work with. Any packages that won't work can be marked as incompatible and the related code prevented from executing.



Marc Laporte 10 Mar 20 21:17 GMT-0000

For the record, on one site upgrade from 20.x to 21.x, I emptied vendor to be able to get console.php to tun

Attachments

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

No attachments for this item

The original document is available at <https://dev.tiki.org/item7177-Packages-Duplicate-Vendors>