XOOPS is back, and with a vengeance! :) - XOOPS

NEWS_PDF_AUTHOR: Mamba

NEWS_PDF_DATE: 2013/9/16 6:30:00

I don't think, we had so much excitement about XOOPS as we have right now, in a very very long time! 😁

We're now starting to work with full power on **XOOPS 2.6.0**!

After incorporating the **[Doctrine DB Abstraction Layer](#)**, that puts us on par in the DB area with projects like Laravel and Symfony, which are also using Doctrine, **Richard Griffith, our [new Core Team leader](#)**, continues to deliver amazing things!

Building up on the awesome work on XOOPS 2.6.0 by **Trabis**, **ForMuss** , **Mage** and **Dugris**, Richard just did some major redesign of our Debugging/Logging system, making XOOPS the CMS with probably the best debugging/logging capabilities in the world! Something what all serious developers will truly appreciate!

- Our new logging mechanism in Core is now **[PSR-3 compliant](#)**, which currently only very, very few projects can claim!!!

- he has replaced our outdated Debug view with the awesome **[PhpDebugBar](#)** module, that will provide our developers with tons of useful information.

As you can see above, our new PhpDebugBar is actually a "PhpDebugBar on steroids" because it merges it with information that XOOPS provides. So now we'll have the same information as the old Debug, but much, much more, incl. Smarty info and the cool timeline! Check out the original PhpDebugBar to see the original limited version of the [PhpDebugeBar](#), but if you want to see what Richard did, you need to download XOOPS 2.6.0 😬

- we've converted our Legacy Logger to the new logging system

- and we've included **Monolog** module for professional logging capabilities!

- and Richard is almost done with reworking of Trabis' XMF (XOOPS Module Framework). We plan to use XMF as a migration tool for existing XOOPS modules. Once converted to XMF, the current modules should be able to run on XOOPS 2.6.x, with no or only very minimal modifications, which will keep with our tradition of easy migration for our users.

You can fork it from our **GitHub XOOPS Repository**

Oh, did we mentioned that we're already using **PHP Composer** and that in the future we'll be also fully utilizing Packagist?

Another work that we're doing is to have the whole XOOPS Core tested using PHPUnit. **Alain91 from France**, who is leading this effort, has written already majority of tests, and we also have already set up a dedicated website with Jenkins. This is still in a testing phase, but once we go live with it, all submitted Core code will have to go through Jenkins to be accepted into the central repository.

And Richard and his team are just warming up 😁

So expect more cool things coming soon!

But why wait? Just join the XOOPS Development Team and start contributing to XOOPS, either in the Core area or in the Module Development area, or both!

Please join the conversion in **this thread**!

XOOPS is Powered by YOU!!!

Viva XOOPS! 😁

I don't think, we had so much excitement about XOOPS as we have right now, in a very very long time! 😁

We're now starting to work with full power on **XOOPS 2.6.0**!

After incorporating the **Doctrine DB Abstraction Layer**, that puts us on par in the DB area with projects like Laravel and Symfony, which are also using Doctrine, **Richard Griffith, our** new Core Team leader, continues to deliver amazing things!

Building up on the awesome work on XOOPS 2.6.0 by **Trabis**, **ForMuss** , **Mage** and **Dugris**, Richard just did some major redesign of our Debugging/Logging system, making XOOPS the CMS with probably the best debugging/logging capabilities in the world! Something what all serious developers will truly appreciate!

- Our new logging mechanism in Core is now **PSR-3 compliant**, which currently only very, very few projects can claim!!!

- he has replaced our outdated Debug view with the awesome **PhpDebugBar** module, that will provide our developers with tons of useful information.

As you can see above, our new PhpDebugBar is actually a "PhpDebugBar on steroids" because it merges it with information that XOOPS provides. So now we'll have the same information as the old Debug, but much, much more, incl. Smarty info and the cool timeline! Check out the original PhpDebugBar to see the original limited version of the PhpDebugeBar, but if you want to see what Richard did, you need to download XOOPS 2.6.0 😃

- we've converted our Legacy Logger to the new logging system

- and we've included **Monolog** module for professional logging capabilities!

- and Richard is almost done with reworking of Trabis' XMF (XOOPS Module Framework). We plan to use XMF as a migration tool for existing XOOPS modules. Once converted to XMF, the current modules should be able to run on XOOPS 2.6.x, with no or only very minimal modifications, which will keep with our tradition of easy migration for our users.

You can fork it from our **[GitHub XOOPS Repository](#)**

Oh, did we mentioned that we're already using **[PHP Composer](#)** and that in the future we'll be also fully utilizing [Packagist](#)?

Another work that we're doing is to have the whole XOOPS Core tested using [PHPUnit](#). **Alain91 from France**, who is leading this effort, has written already majority of tests, and we also have already set up a [dedicated website with Jenkins](#). This is still in a testing phase, but once we go live with it, all submitted Core code will have to go through Jenkins to be accepted into the central repository.

And Richard and his team are just warming up 😁

So expect more cool things coming soon!

But why wait? Just join the XOOPS Development Team and start contributing to XOOPS, either in the Core area or in the Module Development area, or both!

Please join the conversion in **[this thread](#)**!

XOOPS is Powered by YOU!!!

Viva XOOPS! 😁