

## Phorum Support Forums / Developers Corner

Collection of thoughts on future Phorum architecture and development

---

Thomas Seifert / November 12, 2012 10:50AM

[Collection of thoughts on future Phorum architecture and development](#)

Just some collection on thoughts on the future Phorum architecture that we discussed in the IRC channel or other ways.

We don't know yet which version that'll be, maybe Phorum 6, maybe 7, who knows. That's why I'll call it Phorum X.

Phorum 5 will be the last mostly procedural code base. Future versions will make more use of OOP (just not for the sake of it ;)).

One main architecture target for Phorum X is to keep the main idea or even extend it further: to have a lean and fast core with lots of hooks and entry points for extensions.

More tests. Maybe it won't be a 100% test coverage but we will code with tests in mind or even as TDD (test driven development).

The admin **could** be based on Zend Framework and EasyUI to make things focused on admin tasks, instead of DB layering.

Backward Compatibility will NOT be a goal of Phorum X but conversion scripts / extensions will be provided. We have to get rid of old stuff to bring something shiny new.

It **could** follow the MVC pattern for better separation. Routing all requests through a single index.php - which makes it easier to integrate.

Getting rid of comma separated parameters. Its working fine for Phorum and its completely by the RFC's but still some other apps fail to parse that correctly :-).

Based on features available from PHP 5.3.x on?

Having a separation between styling and fuctional UI would ge great.

-> Most people don't change the function UI, they only want to change headers, footers, colors. It would be cool to have that styling separated from how the interface itself is implemented. (Quote Maurice)

Full API accessible through web service protocols like SOAP, XMLRPC ... you name it. The Phorum GUI could even be just one of the possible web access methods.

Better organization and handling of extensions / modules. Web install-/upgradeable for those who want it.

**Want to join us in this effort? Get in contact through the [Development Forum!](#)**

---

## Thomas Seifert

Edited 1 time(s). Last edit at 11/12/2012 10:52AM by Thomas Seifert.

---

Maurice Makaay / November 22, 2012 03:16PM

[Re: Collection of thoughts on future Phorum architecture and development](#)

One thing related to OO programming that I really want to leverage, is the use of exceptions for error handling. This allows for really clean coding, where exception cases do not have to be part of data that is returned by functions.

More importantly: you cannot overlook an error case that is propagated as an exception, while errors in return data can easily be overlooked.

One thing that I find worth considering, is making the current PHP version the one to target. That allows us to make use of new features like namespaces, closures and for example availability of multibyte processing.

Regarding charsets: using UTF-8 at least internally seems non-optional to me. If we want to support other charsets for output, then I say this would be part of the output processing. At the view level, we might add conversion from UTF-8 to other charsets. I would not shed tears when other charsets were not to be used at all. UTF-8 is the standard nowadays and charset issues are a bitch.

---

## Maurice Makaay

Phorum Development Team

[my blog](#) [linkedin profile](#) [secret sauce](#)

---

Thomas Seifert / November 23, 2012 04:02AM

[Re: Collection of thoughts on future Phorum architecture and development](#)

For exceptions I just got a big YES :).

You mean making PHP 5.4 the target or even 5.5?

I'd personally go for 5.4. Got quite some mails from hosters which enable that for their customers right now.

I'm not sure how PHP handles strings internally right now. Is it utf-8 or does that still need additional libraries? When PHP6 was planned the idea was to have UTF-8 internally everywhere.

---

## Thomas Seifert

Maurice Makaay / November 23, 2012 03:54PM

[Re: Collection of thoughts on future Phorum architecture and development](#)

PHP 5.4 or in time 5.5 when development is ongoing then ;) When starting a new implementation, I say: go for the latest and greatest. New features are added to PHP and not seldom, these solve some programming pain, taking away the need for work-arounds.

With internally, I mainly focused at Phorum itself. So this would mean that we only use UTF-8 storage in the database and multibyte string processing in the code layers. At the PHP-level, I thought that the mbstring support was standard when building PHP now. If not, then we can still be ass-backwards about it and require the mbstring module to be loaded. Being more strict about this might make installation for some users harder because of having to communicate with a hosting provider, but this brings a lot in trustworthy charset support and not having to implement fallback libraries (like the `compat_*` modules in the 5.3 code tree).

Other charsets are strictly bound to the representation layer this way. When admins really want a different charset, then a conversion to that charset can be done just before putting the data in the resulting pages.

---

## Maurice Makaay

Phorum Development Team

[my blog](#) [linkedin profile](#) [secret sauce](#)

---