XOOPS 2.6.0 Alpha 2 Released for Testing - XOOPS

NEWS_PDF_AUTHOR: trabis

NEWS_PDF_DATE: 2013/1/15 3:00:00

Quote:

> **UPDATE: If you're reading this for the first time, please be aware that the XOOPS 2.6.0 Development is taking place on GitHub and that there has been tons of changes between this Alpha 2 and the upcoming Alpha 3. You can read about some of them here:**
>
> **https://xoops.org/modules/news/article.php?storyid=6538**
> **https://xoops.org/modules/news/article.php?storyid=6557**
>
> **Please fork the XOOPS 2.6.0 code from GitHub and help us in testing and development!**

The XOOPS Core Development Team is pleased to announce the release of XOOPS 2.6.0 Alpha 2.

This is a brand new XOOPS series, with several major changes and enhancements to the Core.

You need PHP 5.3+ to run this version!

You can review the current **XOOPS 2.6.0 Roadmap here**

The main goal of the 2.6 series is to update the XOOPS Core and all classes to PHP5 (public, protected, static) and E_STRICT, and to remove / clean up old legacy code and remove all HTML code found in the PHP files.

Some of the main changes in Alpha 2:

Improved API and Modularization:

We have extracted from Core several functionalities/behaviors that previously were hard-coded into Core. This will make Core smaller and easier to use. At the same time, modules won't have to go for everything to the Core or to the System module.

One of the complaints in the past was that it was hard to extend XOOPS Core. Therefore we've

created several new Abstract classes, and as a result, we have decoupled several of the hard coded dependencies from the Core, making it now easier to extend the Core and letting the modules to develop their own implementations.

New classes like 'Xoops_Plugin_Abstract' along with the already known 'XoopsPreload' are giving modules new and exciting possibilities.

Improved Performance through redesigned Cache system:

XOOPS 2.6.0 Alpha 1 was already faster then 2.5.5. In the Alpha 2, we wanted it to be even faster. We've added new cache layers that greatly reduced the number of queries required on each page load. We have also re-factored the cache classes, added new cache engines, and documented it to allow developers to "actually" use it directly in their modules. The users will definitely appreciate the improved performance and response times

Improved Productivity through several New Classes

We have added several new classes to make development of XOOPS modules faster and much easier. XoopsFormTab, Xoops_Request, Xoops_Module_Helper, and XoopsLoad::addMap(), are some of the new classes/methods introduced in this version.

- You can use XoopsLoad::addMap() to allow class lazy loading, No need to include your classes when you are not using them.

- You can use Xoops_Request to get $_GET,params, $_POST params, cookies, client IP, URI, and many other items. You want to know if the request comes from a mobile device? It supports it too! And you can add many new items. You can also extend the mobile list without hacking the class.

```
$request = Xoops_Request::getInstance();

Xoops_Utils::dumpVar($request->getParam());
$result['id'] = $request->asInt('id', 13);
$result['string'] = $request->asStr('string', 'defaultValueHere');
$result['bool'] = $request->asBool('bool', false);
$result['order'] = $request->asStr('order', 'ASC', array('ASC', 'DESC'));
$result['url'] = $request->getUrl();
$result['uri'] = $request->getUri();
$result['referer'] = $request->getReferer();
$result['phpsessid_cookie'] = $request->getCookie('PHPSESSID');
$result['ip'] = $request->getClientIp();
$result['isget'] = $request->is('get');
$result['ispost'] = $request->is('post');
$result['ismobile'] = $request->is('mobile');
$result['isrobot'] = $request->is('robot');
```

```
$result['files'] = $request->getFiles('file_identifier');
```

- You can use <u>Xoops_Module_helper</u> if you don't want to type the module name every time you want a Config, a Handler or a Form. It also allows you to get other modules configs, handlers and forms with only one line of code.

```
 /**
  * Using Xoops, the verbose way
  */
if ($xoops->isActiveModule('search')) {
    $config = $xoops->getModuleConfig('keyword_min', 'search');
    $xoops->loadLanguage('main', 'search');
    $url = $xoops->url('modules/search/index.php');
    $obj = $xoops->getModuleByDirname('search');
    //etc
}
/**
  * Using the Helper
  */
if ($helper = Xoops_Module_Helper::getHelper('search')) {
    $config = $helper->getConfig('keyword_min');
    $helper->loadLanguage('main');
    $url = $helper->url('index.php');
    $obj = $helper->getModule();
    //etc
}
```

- <u>XoopsFormTab</u>? Yes, we have it now to provide Tabbed forms!

New Codex Module as Tutorial for new Features

one of the shortcomings in the past was that we didn't have good documentation for new features. This time we've included a new module Codex, that will show exactly how to use the new features, with well documented code and examples.

New Modules/Plugins

Since 2.6.0, the backend functionality uses a 'Plugin' interface.
The new class Xoops_Module_Plugin is the class that makes using plugins simple and effective!

- Menus: we have incorporated the advanced menu module, which will make easier to create menus in XOOPS. Admin will have a full control over menus and menu groups. The 'Menus' module provides a 'Plugin' interface that other modules should implement. Each module in XOOPS 2.6.0 will be now able to add its own menu items by hooking into the menus module

- Page: this is our new simple Content module, that will be included in the basic installation of XOOPS.

- QRCode - can be used by other modules to create QRCodes

- PDF: other modules will be able to use it to provide PDF functionality

- Notifications: another extracted extension from the Core

- Search: Since 2.6.0, the search functionality was removed from core.
Now you need to install the 'Search' Module to get search functionality in other modules. The 'Search' module provides a 'Plugin' interface that modules should implement.

- System Plugin: The 'System' module provides a 'Plugin' interface to access the System functionality, e.g. to synchronize user's number of posts, populate the Waiting block, or to access User menus.

- User Config: in the past, many of the configuration items were stared in different locations. Now all user-related preferences will be stored in one location.

These are just few of the Plugins/Modules. There are more like the Logger, Banners, Avatars, Notifications, xCaptcha, or the xLanguage. And in Alpha 3 we'll add some more.

Focus on Code Quality and Consistency

The Core Team has put a lot of effort to ensure the highest source code quality in XOOPS 2.6.0.

The whole code base has been refactored to ensure consistency of function names and functionality, so module developers can expect the same behavior from functions with similar names

The XOOPS 2.6.0 will have no error messages and no red flags in the advanced PHP editors, which will greatly reduce the probability of bugs arising due to typos, variables not set, returning of unexpected values and other frequent mistakes.

Higher Productivity through Advanced Code Inspection

XOOPS 2.6.0 has now total support for advanced PHP editors, such as phpStorm, which will be now able to deeply understand the code, provide smart code completion for methods, functions

and definitions, and quick navigation and on-the-fly error checking. It was possible thanks to implementation of PHPDocs in all classes/functions and by making the code PHP 5.4 compliant. This will improve productivity of our developers who will be able to take advantage of the advanced features of modern PHP editors.

Twitter Bootstrap

We have further improved the implementation of [Twitter Bootstrap](#) in the Core, which is our main advanced HTML/CSS engine. This will make developmet of themes much easier, and will provide "[Responsive Web Design](#)" for mobile devices "out of the box".

If you are a designer looking to develop new Bootstrap based themes, or a module developer wanting to reduce the size of your framework and have auto-completion at your fingertips, or a webmaster looking for a truly extensible and easy to use CMS, please take XOOPS 2.6.0 for a test drive!

The Core Team will now focus on Alpha 3. We have several ambitious goals for Alpha 3 - see our Roadmap.

Please provide us with feedback, suggestions - Alpha development is for testing concepts to see what works and what doesn't. We can still modify things in the Core, but for that we need your help and feedback.

Please remember: This is **Alpha Release** for features testing only!!!!

DO NOT install it on a production site and **DO NOT** upgrade any production site with it!!!

Please post and discuss all issues related to this release in **[this Forum](#)**

System requirements

PHP:
Any PHP version >= **5.3+ (PHP 5.4+** is strongly recommended)

MySQL:
MySQL server 5.0+

Web server:
Any server supporting the required PHP version (Apache highly recommended)

Downloading XOOPS 2.6.0 Alpha 2

Your can get this release package from the [SourceForge repository](#).

Installing XOOPS

----

1. Copy the content of the htdocs/ folder where it can be accessed by your server
2. Ensure mainfile.php and uploads/ are writable by the web server
3. For security considerations, you are encouraged to move directories "/xoops_lib" (for XOOPS libraries) and "/xoops_data" (for XOOPS data) out of Document Root, and change the folder names.
4. Make the directory xoops_data/ writable; Create (if not already present) and make the directories xoops_data/caches/, xoops_data/caches/xoops_cache/, xoops_data/caches/smarty_cache/ and xoops_data/caches/smarty_compile/ writable.
5. Access the folder where you installed the htdocs/ files using your web browser to launch the installation wizard

Installing Protector in XOOPS

We also highly recommend the installation of the PROTECTOR extension which will bring additional security protection and logging capabilities to your site.

Upgrading from a previous version

NOT available in Alpha

Debug information

Please note: to see Debug info, you need to install and activate the "Logger" extension.

Files integrity check

The full XOOPS package is released with a script able to check if all the system files have been correctly uploaded to the server. To use it, follow these instructions:

1. Upload the checksum.php and checksum.md5 files located in the XOOPS package root to your XOOPS server folder (putting them next to mainfile.php).
2. Execute checksum.php with your browser
3. If necessary, re-upload the missing or corrupted system files
4. Remove checksum.php and checksum.md5 from your server

Modules

This release contains only the "system-related modules and extensions".

Unless specifically stated by the module Author, current modules will NOT work properly with XOOPS 2.6.0 Alpha.

How to contribute

Bug report:http://sourceforge.net/tracker/?group_id=41586&atid=430840
Patch and enhancement:http://sourceforge.net/tracker/?group_id=41586&atid=430842
Feature design:http://sourceforge.net/tracker/?group_id=41586&atid=430843
Release announcement:https://lists.sourceforge.net/lists/listinfo/xoops-announcement

DuGris, Mage, Nicolas, and Trabis
XOOPS Core Development Team
January 15th, 2013

Quote:

> **UPDATE: If you're reading this for the first time, please be aware that the XOOPS 2.6.0 Development is taking place on GitHub and that there has been tons of changes between this Alpha 2 and the upcoming Alpha 3. You can read about some of them here:**
>
> **https://xoops.org/modules/news/article.php?storyid=6538**
> **https://xoops.org/modules/news/article.php?storyid=6557**
>
> **Please fork the XOOPS 2.6.0 code from GitHub and help us in testing and development!**

The XOOPS Core Development Team is pleased to announce the release of XOOPS 2.6.0 Alpha 2.

This is a brand new XOOPS series, with several major changes and enhancements to the Core.

You need PHP 5.3+ to run this version!

You can review the current **XOOPS 2.6.0 Roadmap** here

The main goal of the 2.6 series is to update the XOOPS Core and all classes to PHP5 (public, protected, static) and E_STRICT, and to remove / clean up old legacy code and remove all HTML code found in the PHP files.

Some of the main changes in Alpha 2:

Improved API and Modularization:

We have extracted from Core several functionalities/behaviors that previously were hard-coded into Core. This will make Core smaller and easier to use. At the same time, modules won't have to go for everything to the Core or to the System module.

One of the complaints in the past was that it was hard to extend XOOPS Core. Therefore we've created several new Abstract classes, and as a result, we have decoupled several of the hard coded dependencies from the Core, making it now easier to extend the Core and letting the modules to develop their own implementations.

New classes like 'Xoops_Plugin_Abstract' along with the already known 'XoopsPreload' are giving modules new and exciting possibilities.

Improved Performance through redesigned Cache system:

XOOPS 2.6.0 Alpha 1 was already faster then 2.5.5. In the Alpha 2, we wanted it to be even faster. We've added new cache layers that greatly reduced the number of queries required on each page load. We have also re-factored the cache classes, added new cache engines, and documented it to allow developers to "actually" use it directly in their modules. The users will definitely appreciate the improved performance and response times

Improved Productivity through several New Classes

We have added several new classes to make development of XOOPS modules faster and much easier. XoopsFormTab, Xoops_Request, Xoops_Module_Helper, and XoopsLoad::addMap(), are some of the new classes/methods introduced in this version.

- You can use XoopsLoad::addMap() to allow class lazy loading, No need to include your classes when you are not using them.

- You can use Xoops_Request to get $_GET,params, $_POST params, cookies, client IP, URI, and many other items. You want to know if the request comes from a mobile device? It supports it too! And you can add many new items. You can also extend the mobile list without hacking the class.

```php
$request = Xoops_Request::getInstance();

Xoops_Utils::dumpVar($request->getParam());
$result['id'] = $request->asInt('id', 13);
$result['string'] = $request->asStr('string', 'defaultValueHere');
$result['bool'] = $request->asBool('bool', false);
$result['order'] = $request->asStr('order', 'ASC', array('ASC', 'DESC'));
$result['url'] = $request->getUrl();
$result['uri'] = $request->getUri();
$result['referer'] = $request->getReferer();
$result['phpsessid_cookie'] = $request->getCookie('PHPSESSID');
$result['ip'] = $request->getClientIp();
$result['isget'] = $request->is('get');
$result['ispost'] = $request->is('post');
$result['ismobile'] = $request->is('mobile');
$result['isrobot'] = $request->is('robot');
$result['files'] = $request->getFiles('file_identifier');
```

- You can use Xoops_Module_helper if you don't want to type the module name every time you want a Config, a Handler or a Form. It also allows you to get other modules configs, handlers and forms with only one line of code.

```php
/**
 * Using Xoops, the verbose way
 */
if ($xoops->isActiveModule('search')) {
    $config = $xoops->getModuleConfig('keyword_min', 'search');
    $xoops->loadLanguage('main', 'search');
    $url = $xoops->url('modules/search/index.php');
    $obj = $xoops->getModuleByDirname('search');
    //etc
}
/**
 * Using the Helper
 */
if ($helper = Xoops_Module_Helper::getHelper('search')) {
    $config = $helper->getConfig('keyword_min');
    $helper->loadLanguage('main');
    $url = $helper->url('index.php');
    $obj = $helper->getModule();
    //etc
}
```

- XoopsFormTab? Yes, we have it now to provide Tabbed forms!

New Codex Module as Tutorial for new Features

one of the shortcomings in the past was that we didn't have good documentation for new features. This time we've included a new module Codex, that will show exactly how to use the new features, with well documented code and examples.

New Modules/Plugins

Since 2.6.0, the backend functionality uses a 'Plugin' interface.
The new class Xoops_Module_Plugin is the class that makes using plugins simple and effective!

- Menus: we have incorporated the advanced menu module, which will make easier to create menus in XOOPS. Admin will have a full control over menus and menu groups. The 'Menus' module provides a 'Plugin' interface that other modules should implement. Each module in XOOPS 2.6.0 will be now able to add its own menu items by hooking into the menus module

- Page: this is our new simple Content module, that will be included in the basic installation of XOOPS.

- QRCode - can be used by other modules to create QRCodes

- PDF: other modules will be able to use it to provide PDF functionality

- Notifications: another extracted extension from the Core

- Search: Since 2.6.0, the search functionality was removed from core.
Now you need to install the 'Search' Module to get search functionality in other modules. The 'Search' module provides a 'Plugin' interface that modules should implement.

- System Plugin: The 'System' module provides a 'Plugin' interface to access the System functionality, e.g. to synchronize user's number of posts, populate the Waiting block, or to access User menus.

- User Config: in the past, many of the configuration items were stared in different locations. Now all user-related preferences will be stored in one location.

These are just few of the Plugins/Modules. There are more like the Logger, Banners, Avatars, Notifications, xCaptcha, or the xLanguage. And in Alpha 3 we'll add some more.

Focus on Code Quality and Consistency

The Core Team has put a lot of effort to ensure the highest source code quality in XOOPS 2.6.0.

The whole code base has been refactored to ensure consistency of function names and functionality, so module developers can expect the same behavior from functions with similar names

The XOOPS 2.6.0 will have no error messages and no red flags in the advanced PHP editors, which will greatly reduce the probability of bugs arising due to typos, variables not set, returning of unexpected values and other frequent mistakes.

Higher Productivity through Advanced Code Inspection

XOOPS 2.6.0 has now total support for advanced PHP editors, such as phpStorm, which will be now able to deeply understand the code, provide smart code completion for methods, functions and definitions, and quick navigation and on-the-fly error checking. It was possible thanks to implementation of PHPDocs in all classes/functions and by making the code PHP 5.4 compliant. This will improve productivity of our developers who will be able to take advantage of the advanced features of modern PHP editors.

Twitter Bootstrap

We have further improved the implementation of Twitter Bootstrap in the Core, which is our

main advanced HTML/CSS engine. This will make developmet of themes much easier, and will provide "Responsive Web Design" for mobile devices "out of the box".

If you are a designer looking to develop new Bootstrap based themes, or a module developer wanting to reduce the size of your framework and have auto-completion at your fingertips, or a webmaster looking for a truly extensible and easy to use CMS, please take XOOPS 2.6.0 for a test drive!

The Core Team will now focus on Alpha 3. We have several ambitious goals for Alpha 3 - see our Roadmap.

Please provide us with feedback, suggestions - Alpha development is for testing concepts to see what works and what doesn't. We can still modify things in the Core, but for that we need your help and feedback.

Please remember: This is **Alpha Release** for features testing only!!!!

DO NOT install it on a production site and **DO NOT** upgrade any production site with it!!!

Please post and discuss all issues related to this release in **this Forum**

System requirements

PHP:
Any PHP version >= **5.3+ (PHP 5.4+** is strongly recommended)

MySQL:
MySQL server 5.0+

Web server:
Any server supporting the required PHP version (Apache highly recommended)

Downloading XOOPS 2.6.0 Alpha 2

Your can get this release package from the **SourceForge repository**.

Installing XOOPS
----

1. Copy the content of the htdocs/ folder where it can be accessed by your server
2. Ensure mainfile.php and uploads/ are writable by the web server
3. For security considerations, you are encouraged to move directories "/xoops_lib" (for

XOOPS libraries) and "/xoops_data" (for XOOPS data) out of Document Root, and change the folder names.
   4. Make the directory xoops_data/ writable; Create (if not already present) and make the directories xoops_data/caches/, xoops_data/caches/xoops_cache/, xoops_data/caches/smarty_cache/ and xoops_data/caches/smarty_compile/ writable.
   5. Access the folder where you installed the htdocs/ files using your web browser to launch the installation wizard

Installing Protector in XOOPS

We also highly recommend the installation of the PROTECTOR extension which will bring additional security protection and logging capabilities to your site.

Upgrading from a previous version

NOT available in Alpha

Debug information

Please note: to see Debug info, you need to install and activate the "Logger" extension.

Files integrity check

The full XOOPS package is released with a script able to check if all the system files have been correctly uploaded to the server. To use it, follow these instructions:

   1. Upload the checksum.php and checksum.md5 files located in the XOOPS package root to your XOOPS server folder (putting them next to mainfile.php).
   2. Execute checksum.php with your browser
   3. If necessary, re-upload the missing or corrupted system files
   4. Remove checksum.php and checksum.md5 from your server

Modules

This release contains only the "system-related modules and extensions".

Unless specifically stated by the module Author, current modules will NOT work properly with XOOPS 2.6.0 Alpha.

How to contribute

Bug report:http://sourceforge.net/tracker/?group_id=41586&atid=430840
Patch and enhancement:http://sourceforge.net/tracker/?group_id=41586&atid=430842
Feature design:http://sourceforge.net/tracker/?group_id=41586&atid=430843
Release announcement:https://lists.sourceforge.net/lists/listinfo/xoops-announcement

DuGris, Mage, Nicolas, and Trabis
XOOPS Core Development Team
January 15th, 2013