

XOOPS 2.30 Final Is Released - XOOPS

NEWS_PDF_AUTHOR: phppp

NEWS_PDF_DATE: 2008/9/22 10:40:00

Finally the wait is over!!! 🎁 The XOOPS Development Team is pleased to announce the release of **XOOPS 2.30 Final**.

This is the **most significant XOOPS release in years**, as it merges the two existent branches of 2.0 and 2.2 and is building a bridge towards our next generation of system XOOPS 3.0.

With XOOPS 2.3 releases, users will be able to learn some of new features that will be available in XOOPS 3.0, while developers can understand better the direction of XOOPS development, and designers can learn what features will be supported in the future.

Both backward and forward compatibility are guaranteed as much as possible. Modules written for 2.0.* and 2.2.* will work with 2.3; applications developed for 2.3 will be compatible with XOOPS 3.0 automatically.

Developers are encouraged to write their application modules on XOOPS 2.3 so they can incorporate some of the new features and ensure easy migration to XOOPS 3.0.

IMPORTANT: Please read the Installation and Upgrade instructions!!!
As always, before you upgrade, make a backup of your current Web site!!!

Download from [Sourceforge repository](#).

What's New for Users ----- Some of the new features that might interest users: - Default character set encoding has been changed to UTF-8; - Enabled "auto login" or "remember me"; - CAPTCHA is enabled for registration and comment system; - Block clone enabled so that webmasters have more flexibility; - XOOPS editor package is provided and most visual editors are available, e.g. FCKeditor, TinyMCE, etc.; - Side-wide editor; - Mask is added for start page; - Implemented new installer with user experience improvement experiments; - Upgrading scripts from 2.0* and 2.2* to 2.3; - Permissions for access to user profile page is configurable; - Extensible Profile module; - Private Messages Module; - Extended DHTML as default, with inline preview enabled; - EXM/ThAdmin Admin GUI as options. What's New for Developers ----- 1 Improved XOOPS ORM by enabling extensible persistable object handler: - The improved persistable object handler supports almost all common methods for database access and data treatment, so that developers do not need to write their own classes or methods; - Custom handler is supported. Developers can add their own handler if they want more functionality or to replace XOOPS native handlers. 2 Improved

text sanitizer: - Custom extension is supported. Developers can add new filters or methods to sanitizer based on their specific needs. - Built-in extensions, which can be disabled or enabled (e.g. for Flash, YouTube, images, Wiki, syntax highlight, text filter, etc.) - Combined with improved persistable object handler (see above), text input from users will be filtered automatically for security considerations.

3 Custom Frameworks support: - Module developers can add their own library or frameworks (located in XOOPS_ROOT_PATH/Frameworks) which can be shared by other modules. - PEAR, Geshi have been provided as custom libraries in Frameworks. - Once the code inside Frameworks is proven stable and valuable, they will be adopted by XOOPS core and moved into XOOPS core package. This way we can build a standard, efficient channel for third-party developers to contribute.

4 XOOPS editor packages are now supported officially: - A XoopsEditor handler is built to accept third-party editors; - Three editors, including Textarea, Dhtmltextarea, and Extended Dhtmltextarea are provided in XOOPS core. FCKeditor, TinyMCE, Koivi are now provided as third-party add-ons.

5 XOOPS CAPTCHA is built into XOOPS core: - CAPTCHA has been added to registration and comment forms. - Module developers can add CAPTCHA very easily to their applications.

6 XOOPS Control Panel GUI is introduced: - Old administration interface has been kept as legacy - A new interface is designed, although very rough at this stage - Custom interfaces are allowed, ThAdmin by MusS and Kris is the first one introduced

7 XOOPS file and folder engine is added, based on CakePHP.

8 XOOPS Cache engine is built based on CakePHP.

9 Some other improvements and new resources: - XoopsLocal for localization; - XoopsLoad emulating SPL autoload in PHP5; - XoopsUtility as a collection for miscellaneous methods used by XOOPS; - Block clone feature is activated to save module developers' efforts; - Enabled character set and collation support for MySQL 4.1+; - Upgraded Smarty to 2.6.19; - Upgraded phpmailer to 2.0.0; - Added XOOPS data and library folders for forward compatibility with XOOPS 3.0; - Added XOOPS_TRUST_PATH for compatibility with some modules, e.g. protector

10 For more changes, please check [changelog.txt](#)

What's New for Theme/Template developers

1 Rel attribute for anchors is now supported. Developers/designers are encouraged to use (X)HTML compliant "rel" attribute instead of "target".

2 Control Panel GUI handler is introduced thus designers now get the power to make beautiful backend interfaces.

System requirements -----

PHP: Any PHP version >= 4.3 (PHP 4.2.x may work but is not officially supported, PHP 5.0+ is strongly recommended)

MySQL: MySQL server 3.23+ (MySQL 5.0+ is strongly recommended)

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

Downloading XOOPS 2.30 Final -----

Your can get this release package from the sourceforge.net file repository. Both .zip and .gz archives are provided.

Installing XOOPS 2.30 Final -----

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, or even change the folder names.

4. Make the directory of xoops_data/ writable; Create and make the directories of 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

We also highly recommend the installation of the PROTECTOR module which will bring additional security protection and logging capabilities to your site. To install protector module with a fresh new installation of XOOPS 2.30, copy /extras/mainfile.dist.php.protector to /htdocs/mainfile.dist.php

BEFORE installing XOOPS. To install protector module with upgrading from a previous version to XOOPS 2.30, copy /extras/mainfile.dist.php.protector to /upgrade/upd-2.0.18-to-2.3.0/mainfile.dist.php BEFORE upgrading XOOPS. Upgrading from a previous version ----- Upgrading from XOOPS 2.0.* above 2.0.14 and 2.2.* (using the full package) 1. Move the "upgrade" folder inside the "htdocs" folder (it's been kept out as it's not needed for full installs) on your local machine 2. Delete htdocs/mainfile.php, htdocs/install/, htdocs/cache/, htdocs/extras/, htdocs/template_c/, htdocs/themes/ and htdocs/uploads/ from the "htdocs" folder on your LOCAL machine 3. Upload the content of the htdocs folder on your LOCAL machine over your existing files on your server 4. For security considerations, you are encouraged to move directories xoops_lib (for XOOPS libraries) and xoops_data (for XOOPS data) out of document root, or even change the folder names. 5. Make the directory of xoops_data/ writable; Create and make the directories of xoops_data/caches/, xoops_data/caches/xoops_cache/, xoops_data/caches/smarty_cache/ and xoops_data/caches/smarty_compile/ writable. 6. Ensure the server can write to mainfile.php 7. Access /upgrade/ with a browser, and follow the instructions 8. Follow the instructions to update your database 9. Write-protect mainfile.php again 10. Delete the upgrade folder from your server 11. Update the "system" module from the modules administration interface, other modules are recommended to update as well Upgrading from any XOOPS ranging from 2.0.7 to 2.0.13.2 (using the full package): 1. Move the "upgrade" folder inside the "htdocs" folder (it's been kept out as it's not needed for full installs) on your LOCAL machine 2. Delete htdocs/mainfile.php, htdocs/install/, htdocs/cache/, htdocs/extras/, htdocs/template_c/, htdocs/themes/ and htdocs/uploads/ from the "htdocs" folder on your LOCAL machine 3. Upload the content of the htdocs folder on your LOCAL machine over your existing files on your server 4. Delete the following folders and files from your server (they belong to an old version): * class/smarty/core * class/smarty/plugins/resource.db.php 5. Ensure the server can write to mainfile.php 6. For security considerations, you are encouraged to move directories xoops_lib (for XOOPS libraries) and xoops_data (for XOOPS data) out of document root, or even change the folder names. 7. Make the directory of xoops_data/ writable; Create and make the directories of xoops_data/caches/, xoops_data/caches/xoops_cache/, xoops_data/caches/smarty_cache/ and xoops_data/caches/smarty_compile/ writable. 8. Access /upgrade/ with a browser, and follow the instructions 9. Write-protect mainfile.php again 10. Delete the upgrade folder from your server 11. Update the "system" module from the modules administration interface, other modules are recommended to update as well Upgrading a non UTF-8 site: UTF-8 encoding has been introduced into XOOPS 2.30 as default charset. However, there might be some problems with converting existent websites from non UTF-8 charset to UTF-8. Before there is a good enough solution for this conversion, following settings are recommended when you upgrade an existent website if you are not an experienced user: - Select "Do not change" option in "Database character set and collation" step during upgrade process - Modify /languages/yourlanguage/global.php to use existent _CHARSET value if it has been changed to UTF-8 in your new global.php file as

```
define('_CHARSET', 'UTF-8');
```

Upgrading XoopsEditor package: In the XOOPS 2.30 Final package, there are five editors included: dhtmltextarea and textarea for plain text, fckeditor, tinymce and koivi for WYSIWYG HTML. Since there are some directory structure changes in both fckeditor and tinymce editors, you are recommended to remove existent editors before uploading the new additors. And if you are using fckeditor for modules, please modify module specific configs following the files in

/ckeditor/modules/, especially if you use "article" module. 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". You are invited to browse the XOOPS modules repository to if you need additional functionality. Note: as a new repository is being built, the current repository is not up-to-date, PLEASE VISIT INDIVIDUAL DEVELOPERS' WEBSITES TO MAKE SURE YOU ARE USING LATEST VERSION OF MODULES. 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> XOOPS Development Team

September 20th, 2008

Finally the wait is over!!! 🎁 The XOOPS Development Team is pleased to announce the release of **XOOPS 2.30 Final**.

This is the **most significant XOOPS release in years**, as it merges the two existent branches of 2.0 and 2.2 and is building a bridge towards our next generation of system XOOPS 3.0.

With XOOPS 2.3 releases, users will be able to learn some of new features that will be available in XOOPS 3.0, while developers can understand better the direction of XOOPS development, and designers can learn what features will be supported in the future.

Both backward and forward compatibility are guaranteed as much as possible. Modules written for 2.0.* and 2.2.* will work with 2.3; applications developed for 2.3 will be compatible with XOOPS 3.0 automatically.

Developers are encouraged to write their application modules on XOOPS 2.3 so they can incorporate some of the new features and ensure easy migration to XOOPS 3.0.

IMPORTANT: Please read the Installation and Upgrade instructions!!!
As always, before you upgrade, make a backup of your current Web site!!!

Download from [Sourceforge repository](#).

What's New for Users ----- Some of the new features that might interest users: - Default character set encoding has been changed to UTF-8; - Enabled "auto login" or "remember me"; - CAPTCHA is enabled for registration and comment system; - Block clone enabled so that webmasters have more flexibility; - XOOPS editor package is provided and most visual editors are available, e.g. FCKeditor, TinyMCE, etc.; - Side-wide editor; - Mask is added for start page; - Implemented new installer with user experience improvement experiments; - Upgrading scripts from 2.0* and 2.2* to 2.3; - Permissions for access to user profile page is configurable; - Extensible Profile module; - Private Messages Module; - Extended DHTML as default, with inline preview enabled; - EXM/ThAdmin Admin GUI as options. What's New for Developers ----- 1 Improved XOOPS ORM by enabling extensible persistable object handler: - The improved persistable object handler supports almost all common methods for database access and data treatment, so that developers do not need to write their own classes or methods; - Custom handler is supported. Developers can add their own handler if they want more functionality or to replace XOOPS native handlers. 2 Improved text sanitizer: - Custom extension is supported. Developers can add new filters or methods to sanitizer based on their specific needs. - Built-in extensions, which can be disabled or enabled (e.g. for Flash, YouTube, images, Wiki, syntax highlight, text filter, etc.) - Combined with improved persistable object handler (see above), text input from users will be filtered automatically for security considerations. 3 Custom Frameworks support: - Module developers can add their own library or frameworks (located in XOOPS_ROOT_PATH/Frameworks) which can be shared by other modules. - PEAR, Geshi have been provided as custom libraries in

Frameworks. - Once the code inside Frameworks is proven stable and valuable, they will be adopted by XOOPS core and moved into XOOPS core package. This way we can build a standard, efficient channel for third-party developers to contribute. 4 XOOPS editor packages are now supported officially: - A XoopsEditor handler is built to accept third-party editors; - Three editors, including Textarea, Dhtmltextarea, and Extended Dhtmltextarea are provided in XOOPS core. FCKeditor, TinyMCE, Koivi are now provided as third-party add-ons. 5 XOOPS CAPTCHA is built into XOOPS core: - CAPTCHA has been added to registration and comment forms. - Module developers can add CAPTCHA very easily to their applications. 6 XOOPS Control Panel GUI is introduced: - Old administration interface has been kept as legacy - A new interface is designed, although very rough at this stage - Custom interfaces are allowed, ThAdmin by MusS and Kris is the first one introduced 7 XOOPS file and folder engine is added, based on CakePHP. 8 XOOPS Cache engine is built based on CakePHP. 9 Some other improvements and new resources: - XoopsLocal for localization; - XoopsLoad emulating SPL autoload in PHP5; - XoopsUtility as a collection for miscellaneous methods used by XOOPS; - Block clone feature is activated to save module developers' efforts; - Enabled character set and collation support for MySQL 4.1+; - Upgraded Smarty to 2.6.19; - Upgraded phpmailer to 2.0.0; - Added XOOPS data and library folders for forward compatibility with XOOPS 3.0; - Added XOOPS_TRUST_PATH for compatibility with some modules, e.g. protector 10 For more changes, please check [changelog.txt](#) What's New for Theme/Template developers

----- 1 Rel attribute for anchors is now supported. Developers/designers are encouraged to use (X)HTML compliant "rel" attribute instead of "target". 2 Control Panel GUI handler is introduced thus designers now get the power to make beautiful backend interfaces. System requirements ----- PHP: Any PHP version >= 4.3 (PHP 4.2.x may work but is not officially supported, PHP 5.0+ is strongly recommended) MySQL: MySQL server 3.23+ (MySQL 5.0+ is strongly recommended) Web server: Any server supporting the required PHP version (Apache highly recommended) Downloading XOOPS 2.30 Final ----- You can get this release package from the sourceforge.net file repository. Both .zip and .gz archives are provided. Installing XOOPS 2.30 Final

----- 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, or even change the folder names. 4. Make the directory of xoops_data/ writable; Create and make the directories of 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 We also highly recommend the installation of the PROTECTOR module which will bring additional security protection and logging capabilities to your site. To install protector module with a fresh new installation of XOOPS 2.30, copy /extras/mainfile.dist.php.protector to /htdocs/mainfile.dist.php BEFORE installing XOOPS. To install protector module with upgrading from a previous version to XOOPS 2.30, copy /extras/mainfile.dist.php.protector to /upgrade/upd-2.0.18-to-2.3.0/mainfile.dist.php BEFORE upgrading XOOPS. Upgrading from a previous version ----- Upgrading from XOOPS 2.0.* above 2.0.14 and 2.2.* (using the full package) 1. Move the "upgrade" folder inside the "htdocs" folder (it's been kept out as it's not needed for full installs) on your local machine 2. Delete htdocs/mainfile.php, htdocs/install/, htdocs/cache/, htdocs/extras/, htdocs/template_c/, htdocs/themes/ and

htdocs/uploads/ from the "htdocs" folder on your LOCAL machine 3. Upload the content of the htdocs folder on your LOCAL machine over your existing files on your server 4. For security considerations, you are encouraged to move directories xoops_lib (for XOOPS libraries) and xoops_data (for XOOPS data) out of document root, or even change the folder names. 5. Make the directory of xoops_data/ writable; Create and make the directories of xoops_data/caches/, xoops_data/caches/xoops_cache/, xoops_data/caches/smarty_cache/ and xoops_data/caches/smarty_compile/ writable. 6. Ensure the server can write to mainfile.php 7. Access /upgrade/ with a browser, and follow the instructions 8. Follow the instructions to update your database 9. Write-protect mainfile.php again 10. Delete the upgrade folder from your server 11. Update the "system" module from the modules administration interface, other modules are recommended to update as well Upgrading from any XOOPS ranging from 2.0.7 to 2.0.13.2 (using the full package): 1. Move the "upgrade" folder inside the "htdocs" folder (it's been kept out as it's not needed for full installs) on your LOCAL machine 2. Delete htdocs/mainfile.php, htdocs/install/, htdocs/cache/, htdocs/extras/, htdocs/template_c/, htdocs/themes/ and htdocs/uploads/ from the "htdocs" folder on your LOCAL machine 3. Upload the content of the htdocs folder on your LOCAL machine over your existing files on your server 4. Delete the following folders and files from your server (they belong to an old version): * class/smarty/core * class/smarty/plugins/resource.db.php 5. Ensure the server can write to mainfile.php 6. For security considerations, you are encouraged to move directories xoops_lib (for XOOPS libraries) and xoops_data (for XOOPS data) out of document root, or even change the folder names. 7. Make the directory of xoops_data/ writable; Create and make the directories of xoops_data/caches/, xoops_data/caches/xoops_cache/, xoops_data/caches/smarty_cache/ and xoops_data/caches/smarty_compile/ writable. 8. Access /upgrade/ with a browser, and follow the instructions 9. Write-protect mainfile.php again 10. Delete the upgrade folder from your server 11. Update the "system" module from the modules administration interface, other modules are recommended to update as well Upgrading a non UTF-8 site: UTF-8 encoding has been introduced into XOOPS 2.30 as default charset. However, there might be some problems with converting existent websites from non UTF-8 charset to UTF-8. Before there is a good enough solution for this conversion, following settings are recommended when you upgrade an existent website if you are not an experienced user: - Select "Do not change" option in "Database character set and collation" step during upgrade process - Modify /languages/yourlanguage/global.php to use existent _CHARSET value if it has been changed to UTF-8 in your new global.php file as

```
define('_CHARSET', 'UTF-8');
```

Upgrading XoopsEditor package: In the XOOPS 2.30 Final package, there are five editors included: dhtmltextarea and textarea for plain text, fckeditor, tinymce and koivi for WYSIWYG HTML. Since there are some directory structure changes in both fckeditor and tinymce editors, you are recommended to remove existent editors before uploading the new additors. And if you are using fckeditor for modules, please modify module specific configs following the files in /fckeditor/modules/, especially if you use "article" module. 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". You are invited to browse the XOOPS modules repository to if you need additional functionality. Note: as a new repository is being built, the current repository is not up-to-date, PLEASE VISIT INDIVIDUAL DEVELOPERS' WEBSITES TO MAKE SURE YOU ARE USING LATEST VERSION OF MODULES. 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> XOOPS Development Team

September 20th, 2008