

Chamilo LMS - Feature #5871

Make Chamilo debian-packageable

14/01/2013 02:59 - Yannick Warnier

Status:	Assigned	Start date:	14/01/2013
Priority:	High	Due date:	
Assignee:	Erick Ocrospoma	% Done:	50%
Category:	System	Estimated time:	0.00 hour
Target version:	2.0	Spent time:	6.50 hours
Complexity:	Wizard-level	SCRUM pts - complexity:	?

Description

To enable Chamilo to be packaged for Debian, we need:

1. a single variable directory (unifying courses, home, main/lang, main/css, archive, cache, main/upload/users, main/default_course_documents/)
2. an independent, SQL-only, install script (or something close)
3. ...something else?

Related issues:

Related to Chamilo LMS - Feature #5754: Rename "archive" folder name to "cache"	Rejected - Abandoned	26/01/2012
Related to Chamilo LMS - Feature #6100: Move default_platform_document and d...	Assigned	19/04/2013
Related to Chamilo LMS - Feature #6636: Move "temp" folder (old archive fold...	Feature implemented	25/08/2013
Related to Chamilo LMS - Feature #6919: Move the logs folder inside of data	Feature implemented	19/04/2014
Related to Chamilo LMS - Feature #7522: data/ directory in root dir to store...	Feature implemented	16/09/2015
Related to Chamilo LMS - Feature #983: Repository for updates - A LA DEBIAN	New	15/04/2010

Associated revisions

Revision 2d2702b6 - 17/04/2013 12:34 - Julio Montoya

Creating new folders due a new structure see #5871

Revision d9e17597 - 17/04/2013 12:44 - Julio Montoya

Paths changes due the new structure see #5871 (pending confirmation)

Revision 6cc22982 - 19/04/2013 13:36 - Julio Montoya

Moving courses to data/courses see #5871

Revision 9ae08dce - 19/04/2013 13:37 - Julio Montoya

Removing main/upload/users see #5871

Revision e27d2855 - 19/04/2013 13:51 - Julio Montoya

Adding htaccess for blocking access in data + adding htaccess in the root in order to redirect course/ to web/courses see #5871

Revision b9267dea - 21/04/2013 20:47 - Yannick Warnier

Fixed issues with documents templates not loading the right directory - refs #5871

Revision b1dd9ebe - 21/04/2013 21:40 - Yannick Warnier

Added REL_DATA_PATH to api_get_path() - refs #5871

Revision 051cccea - 22/04/2013 02:25 - Yannick Warnier

Update get_user_picture_path to use new folders hierarchy - refs #5871

Revision 3a817aa4 - 22/04/2013 12:45 - Julio Montoya

Changing paths in the fckeditor ajaxmanager to deal with the new data folder changes. See: #5871

History

#1 - 14/01/2013 03:00 - Yannick Warnier

- Description updated

#2 - 14/01/2013 03:00 - Yannick Warnier

update test

#3 - 12/04/2013 16:07 - Julio Montoya

- Status changed from New to Needs more info

What about this structure?

```
|— app
|   |— data
|   |   |— config (new location for the configuration.php, mail, etc) *
|   |   |— courses
|   |   |— temp (instead of archive) *
|   |   |— home (instead of home)
|   |   |— css
|   |   |— lang
|   |   |— upload
|   |   |— logs *
|— console.php *
```

(*) Means that it will be easy to change it right now (some work was already done)

#4 - 13/04/2013 10:52 - Bas Wijnen

To install this in Debian, the most logical way to package it is given below.

How this is implemented in Chamilo is up to you, of course. If you want to provide an "opt-style" package (everything in one directory), then Debian needs to be able to split it into several parts. This may be done with symbolic links, so it doesn't have to be very invasive.

For the mysql databases, the locations are handled by mysql. A setup script which can run non-interactively (with just some questions asked at the start) is required.

```
/etc/chamilo/*      configuration files1
/etc/apache2/conf.d/chamilo apache settings2
/usr/share/chamilo/* All read-only files (php code)
/var/lib/chamilo/*  All writable non-temporary files (courses)
/var/cache/chamilo/* All temporary files (archive). Chamilo should keep this place clean.
```

To make the difference between the directories clear, it is good to think of what would happen if the data got lost:

/usr: I have to reinstall the package.

/etc: I have to reconfigure the package (and possibly look up passwords etc).

/var/lib: I have to restore my backups.

/var/cache: I have to ... do other things; stop bothering me.

[1] All configuration files must be under /etc; special care is taken by the system to never overwrite user changes to those files.

[2] This file should best be a symlink to /etc/chamilo/apache.conf, or something like that. That way, all chamilo config files are under /etc/chamilo.

#5 - 16/04/2013 12:16 - Julio Montoya

- Assignee set to Yannick Warnier

I need feedback for this new folder structure, because many process relies on this one: chash, chamilo installation, etc

#6 - 16/04/2013 17:34 - Yannick Warnier

- Assignee changed from Yannick Warnier to Julio Montoya

What are the "app" folder and the "console.php" file?

Does the "app" folder also contain the rest of Chamilo or is it just for the composer/symfony stuff?

The "console.php" stuff confuses me :-)

If the "app" folder is our current "chamilo" folder, then I would be much more inclined to accept, but I'm not too keen on changing the whole structure "just to make it closer to Symfony", if that is the intention.

If the "data" folder can be put in the root level (chamilo/) without the "app" intermediate, that would make me quite happy.

```
├── chamilo/
│   ├── app/
│   │   ├── css/ (permanent, provided by package)
│   │   ├── lang/ (permanent, provided by package)
│   │   └── (the rest of the "main" folder without the things from above)
│   ├── config/ (new location for the configuration.php, mail, etc)
│   ├── console.php (!?)
│   ├── data/
│   │   ├── courses/
│   │   ├── home/ (instead of home)
│   │   ├── css/ (created/uploaded by users through CSS editor)
│   │   ├── lang/ (created/uploaded by users through sublanguage editor)
│   │   ├── upload/
│   │   ├── logs/
│   └── temp/ (instead of archive)
```

This way, the config/, data/ and temp/ folders, provided by the package, could then be moved away independently to the right places (described by Bas) or left there (if not a .deb package).

#7 - 16/04/2013 22:12 - Julio Montoya

- Assignee changed from Julio Montoya to Yannick Warnier

Sorry for the confusion, in fact my changes were more conservative than yours :)

```
├── app
│   ├── data
│   │   ├── config (new location for the configuration.php, mail, etc) *
│   │   ├── courses
│   │   ├── temp (instead of archive)
│   │   ├── home (instead of home)
│   │   ├── css
│   │   ├── lang
│   │   ├── upload
│   │   └── logs
│   └── console.php
├── documentation
├── main
├── vendor
├── web
└── ...etc
```

The console.php allows the installation/upgrade/etc it also includes chash commands (I think that this script should not be shipped in the zip package).

About your structure, I have a 2nd suggestion and some questions

```
├── chamilo/
│   ├── app/
│   │   ├── css/ (permanent, provided by package)
│   │   ├── lang/ (permanent, provided by package)
│   │   ├── main
│   │   └── (the rest of the "main" folder without the things from above)
│   │   └── console.php
│   ├── data/
│   │   ├── config/ (new location for the configuration.php, mail, etc)
│   │   ├── courses/
│   │   ├── home/ (instead of home)
│   │   ├── css/ (created/uploaded by users through CSS editor)
│   │   ├── lang/ (created/uploaded by users through sublanguage editor)
│   │   ├── upload/
│   │   ├── logs/
│   │   └── temp/ (instead of archive)
│   ├── src/ the ChamiloLMS namespace is located here
│   └── vendor (third party libs)
```

- Why putting the temp outside the data folder?

- I'm a little bit annoyed (because of time issues) about moving the "main" folder inside the "app" folder in Chamilo 1.10 because URLs will be something like:

my.chamilo.net/app/main/admin/session_list.php

If you don't matter about this I can go with the structure above... until I don't find any big drawback.

Waiting for your feedback again :)

#8 - 17/04/2013 00:44 - Yannick Warnier

- Assignee changed from Yannick Warnier to Julio Montoya

Oh, then we are mixing crazy thought and making it two times as crazy.

I was confused by the "app" name. If we can keep it "main", then it's all OK. Let's focus on the rest and **not** change "main" to "app" :-)

I'm splitting the data and temp directories because, as Bas explained, a Debian package assumes temporary files will be somewhere else (in the hierarchy), so splitting it from the start will make it a lot easier afterwards (because we will not assume that the temp/ directory has the same root path as the cache directory).

Actually, the "logs" directory should also be out of data/, because it should go in /var/log/ in the end :-)

So... this would be the ideal directories tree (if I'm not forgetting anything):

```
├── chamilo/
│   ├── main/
│   │   ├── css/ (permanent, provided by package)
│   │   ├── lang/ (permanent, provided by package)
│   │   └── ... (the rest of the "main" folder without the things from above)
│   ├── data/
│   │   ├── courses/
│   │   ├── home/ (instead of home)
│   │   ├── css/ (created/uploaded by users through CSS editor)
│   │   ├── lang/ (created/uploaded by users through sublanguage editor)
│   │   └── upload/ (users uploads)
│   ├── config/ (new location for the configuration.php, mail, etc)
│   ├── logs/
│   ├── temp/ (instead of archive)
│   ├── src/ the ChamiloLMS namespace is located here (current inc/lib/*.php ?)
│   └── vendor (third party libs) (current inc/lib/*/?)
└── console.php
```

This way:

- there is not big change to main/
- data/ contains all the things meant to **stay** but built/uploaded by users (and can be moved to /var/lib/chamilo/ on Debian)
- config/ can easily be moved out of the Chamilo tree (to increase security **and** put it into /etc/chamilo on Debian)
- logs/ can easily be moved to put it in /var/log/chamilo/ on Debian
- temp/ can easily be moved to /var/cache/chamilo/ on Debian

The src/ dir is not really important to me at the moment.

I understand the "vendor" directory is a necessity for Symfony and Composer, but I'd rather have it in main/inc/lib/ to avoid confusing people with stuff that doesn't mean anything to non English natives.

console.php can stay out of the package, definitely.

#9 - 17/04/2013 09:46 - Julio Montoya

- Assignee changed from Julio Montoya to Yannick Warnier

Yannick Warnier wrote:

Oh, then we are mixing crazy thought and making it two times as crazy.

I was confused by the "app" name. If we can keep it "main", then it's all OK. Let's focus on the rest and **not** change "main" to "app" :-)

Ok, no problem for me.

[...]

This way:

- there is not big change to main/

ok

- data/ contains all the things meant to **stay** but built/uploaded by users (and can be moved to /var/lib/chamilo/ on Debian)

ok

- config/ can easily be moved out of the Chamilo tree (to increase security **and** put it into /etc/chamilo on Debian)

I prefer to move this inside "data" folder in order to have less folders to change permissions, but having the config folder will be clear for new users instead of lurking in the main/inc/conf folder. The both reason you mention abover can also work if the folder is located in "data/config" or not?

- logs/ can easily be moved to put it in /var/log/chamilo/ on Debian
- temp/ can easily be moved to /var/cache/chamilo/ on Debian

The src/ dir is not really important to me at the moment.

I understand the "vendor" directory is a necessity for Symfony and Composer, but I'd rather have it in main/inc/lib/ to avoid confusing people with stuff that doesn't mean anything to non English natives.

The "vendor" folder is now very popular in a lot of modern PHP projects and yes, it's used in symfony2 and composer. It can be configured to locate in main/inc/lib but I think that "main/inc/lib" is worst than "vendor". Why this folder is going to confused people? Normally nobody have to modified that folder and if somebody wants to, it means that he knows what he's doing...

console.php can stay out of the package, definitely.

ok

So... there will be 4 folders with write permissions:

```
config
data
logs
temp
```

#10 - 17/04/2013 15:58 - Yannick Warnier

- Assignee changed from Yannick Warnier to Julio Montoya

Julio Montoya wrote:

Yannick Warnier wrote:

- config/ can easily be moved out of the Chamilo tree (to increase security **and** put it into /etc/chamilo on Debian)

I prefer to move this inside "data" folder in order to have less folders to change permissions, but having the config folder will be clear for new users instead of lurking in the main/inc/conf folder. The both reason you mention abover can also work if the folder is located in "data/config" or not?

The config folder is really not like the other ones, in the sense that it shouldn't allow writing by the web-server-user after the installation is complete. As such, I fear that putting it inside "data" will make everyone do `chmod -R 0777` on the data folder and leave it like that afterwards. One of the (important) risks that could be a consequence of such permissions is that someone could change permissions to access the database and suddently overwrite stuff in another database on the same host.

In this sense, the config/ directory is quite unique.

The src/ dir is not really important to me at the moment.

I understand the "vendor" directory is a necessity for Symfony and Composer, but I'd rather have it in main/inc/lib/ to avoid confusing people with stuff that doesn't mean anything to non English natives.

The "vendor" folder is now very popular in a lot of modern PHP projects and yes, it's used in symfony2 and composer. It can be configured to locate in main/inc/lib but I think that "main/inc/lib" is worst than "vendor". Why this folder is going to confused people? Normally nobody have to modified that folder and if somebody wants to, it means that he knows what he's doing...

The explanation makes sense. "Vendor" in itself is a confusing term (remember, I lived for 4 years in an English-speaking country), that's all. The fact that it is used in many other project and I'm not against the change in that respect, but this change will have wide-scale implications I think (more than the config directory, which is called in a single place) because you will move **some** of the main/inc/lib/ directory there but not all of it, or will you move all of it? (that's just a question)

#11 - 17/04/2013 16:21 - Julio Montoya

- Assignee changed from Julio Montoya to Yannick Warnier

The explanation makes sense. "Vendor" in itself is a confusing term (remember, I lived for 4 years in an English-speaking country), that's all. The fact that it is used in many other project and I'm not against the change in that respect, but this change will have wide-scale implications I think (more than the config directory, which is called in a single place) because you will move **some** of the main/inc/lib/ directory there but not all of it, or will you move all of it? (that's just a question)

The "vendor" folder is only updated by composer, it loads the libs added in composer.json file. Our "classic" libraries main.api.lib, usermanager.lib.php, course.lib.php, etc will stay in "main/inc/lib" because those libs can't live outside Chamilo because there are very old and have a lot of dependencies, etc. I don't want to move those libraries at least not in 1.10. We could move those libs (main/inc/lib) in another parallel universe not here :) Wait... maybe we could move it here src/ChamiloLMS/lib!

There are other libraries (xjax, fckeditor, pclzip) in "main/inc/lib" that are old (unmaintained?) and doesn't have a composer way (packagist.org) to install it, so we will leave it there.

I'm ok with the "config" folder outside "data".

#12 - 17/04/2013 19:33 - Julio Montoya

- % Done changed from 0 to 50

#13 - 19/04/2013 14:26 - Julio Montoya

I have a question:

User images will be located here:

`data/upload/users/1/my_photo.jpg`

but this folder is not accessible to the web (we assume that) so we need a proxy to dump the picture:

something like:

`web/user/1/my_photo.jpg`

The same with the group images, course files, etc.

#14 - 20/04/2013 05:11 - Yannick Warnier

- Assignee changed from Yannick Warnier to Julio Montoya

What do you mean by "proxy"?

As for any files with some kind of user ownership, it should go through a controller that checks the permissions, like main/document/download.php does.

On one side, it makes sense to have a single file to manage all these uploads, but on the other side a single script can really be a bottleneck for performances if it has to do too many checks before it decides if permissions are OK.

What is the logic behind prefixing things with "web/", again? Couldn't we just use http://wxyz.org/user/1/my_photo.jpg ?

#15 - 20/04/2013 17:18 - Yannick Warnier

I think the only problem requiring the main/default_course_document/images/ directory to be writable is that thumbnails are generated once you use the default course images for the first time. If we pre-generate them and store them into the Chamilo package, I think we could remove this requirement from the install procedure...

#16 - 20/04/2013 18:10 - Yannick Warnier

We need to figure out what happens when having several Chamilo installations on the same server (so far we only thought about having a single installation for this package situation).

I guess Chamilo could handle it by having named installations

For example, the first one would always be "default" (which we would need to put in the root directory or something like that), and we would have this:

```
/etc/chamilo/configuration.default.php
/etc/apache2/conf.d/chamilo.default
/usr/share/chamilo/ (normal code, but the same for all installs)
/var/lib/chamilo/default/
/var/cache/chamilo/default/
/var/log/chamilo/default/
```

Then, if installing a second time (through the install scripts, which should then preventing overwriting an existing database), we would use another instance name (to be included in the install process form) like "utp" and

```
/etc/chamilo/configuration.utp.php
/etc/apache2/conf.d/chamilo.utp
/usr/share/chamilo/ (normal code, but the same for all installs)
/var/lib/chamilo/utp/
/var/cache/chamilo/utp/
/var/log/chamilo/utp/
```

Even then, that would not suit our case as we have many customers having different versions of Chamilo on one server of ours...

I'm trying to get my head around it. Any suggestion is appreciated.

This is not urgent and should definitely not slow down packaging efforts, but we might have to consider it in the near future.

#17 - 21/04/2013 09:31 - Jérôme Warnier

Yannick Warnier wrote:

I think the only problem requiring the main/default_course_document/images/ directory to be writable is that thumbnails are generated once you use the default course images for the first time. If we pre-generate them and store them into the Chamilo package, I think we could remove this requirement from the install procedure...

Why was it done that way in the first place, then? In case someone choses another theme as default? To spare some KB?

#18 - 21/04/2013 09:49 - Jérôme Warnier

Yannick Warnier wrote:

We need to figure out what happens when having several Chamilo installations on the same server (so far we only thought about having a single installation for this package situation).
I guess Chamilo could handle it by having named installations

For example, the first one would always be "default" (which we would need to put in the root directory or something like that), and we would have this:

```
/etc/chamilo/configuration.default.php
/etc/apache2/conf.d/chamilo.default
/usr/share/chamilo/ (normal code, but the same for all installs)
/var/lib/chamilo/default/
/var/cache/chamilo/default/
/var/log/chamilo/default/
```

Then, if installing a second time (through the install scripts, which should then preventing overwriting an existing database), we would use another instance name (to be included in the install process form) like "utp" and

```
/etc/chamilo/configuration.utp.php
/etc/apache2/conf.d/chamilo.utp
/usr/share/chamilo/ (normal code, but the same for all installs)
/var/lib/chamilo/utp/
/var/cache/chamilo/utp/
/var/log/chamilo/utp/
```

Even then, that would not suit our case as we have many customers having different versions of Chamilo on one server of ours...

I'm trying to get my head around it. Any suggestion is appreciated.

This is not urgent and should definitely not slow down packaging efforts, but we might have to consider it in the near future.

No package in Debian is meant for multiple installations (so forget about multiple versions at the same time).
Though, some of them have tweaks to run several times as distinct *identities* (I mean, the code base is the same, but some configuration changes from one instance to another). Examples: programs to analyse web server logs (e.g. Webalizer if I'm not mistaken) or web servers with their conf.d directories (Apache).

And may I remind you of the way we are doing it for Dolibarr?

Isn't that possible at all for Chamilo?

In the past, there was a concern about user template customization. I guess the situation should be improved now that there is a templating system (Symfony?) underneath.

#19 - 21/04/2013 19:59 - Yannick Warnier

Jérôme Warnier wrote:

Yannick Warnier wrote:

I think the only problem requiring the `main/default_course_document/images/` directory to be writable is that thumbnails are generated once you use the default course images for the first time. If we pre-generate them and store them into the Chamilo package, I think we could remove this requirement from the install procedure...

Why was it done that way in the first place, then? In case someone chooses another theme as default? To spare some KB?

No, these are just fixed images, but the developer who included the feature works on Windows, so he didn't realize when including them that they would require writing a thumbnail the first time they were looked at. Adding the permissions to that directory was a quick fix because it was already too late in the release process to make structural changes, then it was left this way because there were other, more urgent things to deal with.

#20 - 21/04/2013 21:36 - Yannick Warnier

Reasoning about the (packaged) languages files being in `main/lang/` or `data/lang/`

My idea was to clearly separate the languages "approved" by Chamilo LMS and the languages uploaded by the users, but the more I think about it, the more I think they should all be in `data/lang/`, because we want the users to be able to modify their language files anyway, and we can easily set them as writeable by `www-data` if necessary (it can still break a portal to have a parse error in a frequently used English translation file, but I doubt this would be a common attack vector for crackers).

The reason for this "simplification" is I see that the language-selector code in `global.inc.php` is already about 100 lines long, and adding another directory will only slow things down (looking into 3 directories instead of two as it is currently doing: English then specific language then if not found look into other directory).

Any thought on that?

#21 - 22/04/2013 13:03 - Julio Montoya

Yannick Warnier wrote:

What do you mean by "proxy"?

By proxy I mean what you said: having a file that gets a document from `"data/courses"` and show it to the user in the `"web/courses"` URL. Something like the `download.php`. I sent some changes in order to be managed by a controller see: `IndexController::getDocumentAction()` (there are some todos for that action).

As for any files with some kind of user ownership, it should go through a controller that checks the permissions, like `main/document/download.php` does.

On one side, it makes sense to have a single file to manage all these uploads, but on the other side a single script can really be a bottleneck for performances if it has to do too many checks before it decides if permissions are OK.

What is the logic behind prefixing things with `"web/"`, again? Couldn't we just use http://wxyz.org/user/1/my_photo.jpg ?

The idea behind the `"web"` folder is that the `"data/"` folder should not be public.

The URL that you proposed can't work because there's nothing in the `"user"` folder. You need to use the `"web"` folder because there's the `index.php` that contains the `$app->run()` that makes magic happen and contains routes that are parsed when something starts with `"web"`.

For example:

In `global.inc.php` I have this route:

```
// Username
$app->match('/user/{username}', 'user.controller:indexAction', 'GET');
```

This means the URL `"http://localhost/chamilo/web/user/admin"` will be parsed by the `UserController::indexAction()`

We need the `web` because of the `$app->run()`;

#22 - 22/04/2013 13:04 - Julio Montoya

Yannick Warnier wrote:

I think the only problem requiring the `main/default_course_document/images/` directory to be writable is that thumbnails are generated once you use the default course images for the first time. If we pre-generate them and store them into the Chamilo package, I think we could remove this requirement from the install procedure...

I created a task for that folder: [#6100](#)

#23 - 22/04/2013 13:55 - Julio Montoya

I did more changes so now doc files are redirected this way:

Classic way:

```
http://localhost/chamilo/courses/MATHS/document/folder/image.jpg
```

to

```
http://localhost/chamilo/web/courses/MATHS/document/?file=folder/image.jpg
```

If we use this approach we can parse gently the file using a controller.

Is not possible as far as I know using silex routes to parse a URL like this:

```
http://localhost/chamilo/web/courses/MATHS/document/folder1/folder2/image.jpg
```

because the route must be "fixed" and do not accept wildcards.

What do you think?

#24 - 22/04/2013 13:56 - Julio Montoya

- Assignee changed from Julio Montoya to Yannick Warnier

#25 - 27/04/2013 01:42 - Diego E

This broke Chamilo under localhost/chamilo/ for me:

```
commit e27d28559f583ee6ed4afdf96412103eb4f7eff
Author: Julio Montoya <gugli100@gmail.com>
Date: Fri Apr 19 13:51:37 2013 +0200
```

```
Adding htaccess for blocking access in data + adding htaccess in the root in order to redirect course/ to web/
courses see #5871
```

I think the problem is RewriteBase / in the top .htaccess. This is assuming that chamilo is not in any subdirectory of the domain.

#26 - 29/04/2013 15:08 - Julio Montoya

Diego Escalante Urrelo wrote:

This broke Chamilo under localhost/chamilo/ for me:

```
commit e27d28559f583ee6ed4afdf96412103eb4f7eff
Author: Julio Montoya <gugli100@gmail.com>
Date: Fri Apr 19 13:51:37 2013 +0200
```

Adding htaccess for blocking access in data + adding htaccess in the root in order to redirect course/ to web/courses see [#5871](#)

I think the problem is RewriteBase / in the top .htaccess. This is assuming that chamilo is not in any subdirectory of the domain.

Yes, you're right the .htaccess file should be updated like the courses/.htaccess (in chamilo 1.9)

#27 - 01/05/2013 11:28 - Jérôme Warnier

While this is not exactly related, I would like to bring your attention on <https://task.beeznest.com/issues/1594>. Being able to somehow configure dependencies (e.g. mpdf) locations would be great, as they might already be present on the system. Of course, this requires to carefully document if the external dependency has been modified in any way, as well as the acceptable (minimal) versions. If for nothing else, this could avoid having to update Chamilo as a whole on systems with packaging whenever any such dependency publishes a security update.

#28 - 15/04/2014 08:26 - Yannick Warnier

- Priority changed from Normal to High

#29 - 07/05/2014 18:33 - Yannick Warnier

- Status changed from Needs more info to Assigned

- Assignee changed from Yannick Warnier to Erick Ocrospoma

Steps for implementation

Step 1

Prepare a non-configurable debian package that installs Chamilo in /usr/share/chamilo, puts the config file in /etc/chamilo/default.php, and puts the data directory in /var/lib/chamilo/[host]

Step 2

Prepare a configurable install script asking for host, directories, MySQL credentials, admin user/pass

Step 3

Focus on installing multiple Chamilo portals on one server -> multiple configuration files, data dirs and databases (possible through dynamic PHP selection of config file based on host)

#30 - 07/05/2014 21:29 - Bas Wijnen

Yannick Warnier wrote:

Prepare a configurable install script asking for host, directories, MySQL credentials, admin user/pass

These questions should be asked using debconf. It's probably easiest to make debconf update /etc/default/chamilo or some other config file, which just contains key=value pairs. Note that packages must not overwrite local changes in /etc (unless requested by the admin; giving a new answer is a request. But the current values should be the defaults when the question is asked again, regardless of how those current values got there. I'm working on something to make that easier).

Focus on installing multiple Chamilo portals on one server -> multiple configuration files, data dirs and databases (possible through dynamic PHP selection of config file based on host)

This one may be harder to do with debconf; it's not designed for this. But it is possible, and probably a good idea. However, it doesn't seem unreasonable for the administrator to edit a config file in that case either. You could just start with the question "manage the installation with debconf?" and only allow a single portal if the answer is yes.

#31 - 21/05/2014 13:52 - Julio Montoya

Please check

<http://symfony.com/doc/current/cookbook/deployment-tools.html#the-tools>

<https://github.com/liip/sf2debpkg>

<http://capifony.org/>