

Manual configuration of eZ Publish

This section describes how to manually configure eZ Publish instead of using the setup wizard to do all the work. Keep in mind that the manual installation method is for expert users only. It should only be used by people who know what they are doing. The following steps will work on both Linux/UNIX and Windows environments.

Database initialization

A clean eZ Publish database is created using two very important SQL scripts: "kernel_schema" and "cleandata" (note that an empty database should be created before launching these scripts). The first of them initializes the necessary database structure and the second one imports the pre-defined data to the database. While the "kernel_schema" script differs for each database engine, the "cleandata" script is the same for all solutions.

MySQL

Use the following command to run the MySQL specific "kernel_schema" script:

```
$ mysql -u USERNAME -pPASSWORD DATABASE <
PATH/ezpublish_legacy/kernel/sql/mysql/kernel_schema.sql
```

In eZ Publish 4.0.1 and later versions, the script will use the InnoDB storage engine when creating new tables. This storage engine is recommended (and will be required in the future) for running eZ Publish on a MySQL database. Contact your database administrator if you are unsure about whether InnoDB is available on your server.

In eZ Publish 4.0.0, the CREATE TABLE statements in the "kernel_schema" script do not specify which storage engine to use (no ENGINE or TYPE option), and thus the default storage engine will be used. Normally, it is MyISAM. Because of this, it is highly recommended to set the default storage engine to InnoDB before you run the "kernel_schema" script (refer to the [MySQL documentation](#) for information about how to set the default engine). Alternatively, you can run the "kernel_schema" script first and then convert the newly created tables to InnoDB. You can either use the "bin/php/ezconvertmysqltabletype.php" script for database conversion (recommended) or convert the tables individually by using the following SQL query for each table:

```
ALTER TABLE <name_of_table> TYPE = innodb;
```

Use the following command to run the generic "cleandata" script:

```
$ mysql -u USERNAME -pPASSWORD DATABASE <
PATH/ezpublish_legacy/kernel/sql/common/cleandata.sql
```

USERNAME	The MySQL user (if no user is set up, use "root").
PASSWORD	The password that belongs to the username.
DATABASE	The name of the database, for example "my_database".
PATH	The full path to the root directory of your eZ Publish installation, for example "/opt/ezp".

File permissions

Windows users can skip this part.

Several cache, log and config folders must be writable both by the web server and the command line user, use *one* of the following alternatives to do this:

- **Using ACL on a system that supports chmod +a**

These shell commands will give proper permission to the web server and command line users:

```
$ cd /<ezp5-root>/
$ rm -rf ezpublish/cache/*
$ rm -rf ezpublish/logs/*
$ sudo chmod +a "www-data allow
delete,write,append,file_inherit,directory_inherit" \
  ezpublish/{cache,logs,config}
ezpublish_legacy/{design,extension,settings,var} web
$ sudo chmod +a "`whoami` allow
delete,write,append,file_inherit,directory_inherit" \
  ezpublish/{cache,logs,config}
ezpublish_legacy/{design,extension,settings,var} web
```

- **Using ACL on a system that does not support chmod +a**

Some systems don't support chmod +a, but do support another utility called setfacl. You may need to enable ACL support on your partition and install setfacl before using it (as is the case with Ubuntu), like so:

```
$ cd /<ezp5-root>/
$ sudo setfacl -R -m u:www-data:rwx -m u:www-data:rwx \
  ezpublish/{cache,logs,config}
ezpublish_legacy/{design,extension,settings,var} web
$ sudo setfacl -dR -m u:www-data:rwx -m u:`whoami`:rwx \
  ezpublish/{cache,logs,config}
ezpublish_legacy/{design,extension,settings,var} web
```

- **Using chown on systems that don't support ACL**

Some systems don't support ACL at all. You will either need to set your web server's user as the owner of the required directories.

```
$ cd /<ezp5-root>/
$ sudo chown -R www-data:www-data ezpublish/{cache,logs,config}
ezpublish_legacy/{design,extension,settings,var} web
$ sudo find
{ezpublish/{cache,logs,config},ezpublish_legacy/{design,extension,settings,var},web} -type d | sudo xargs chmod -R 775
$ sudo find
{ezpublish/{cache,logs,config},ezpublish_legacy/{design,extension,settings,var},web} -type f | sudo xargs chmod -R 664
```

- **Using chmod**

If you can't use ACL and aren't allowed to change owner, you can use chmod, making the files writable by everybody. Note that this method really isn't recommended as it allows any user to do anything.

```
$ cd /<ezp5-root>/
$ sudo find
{ezpublish/{cache,logs,config},ezpublish_legacy/{design,extension,set
tings,var},web} -type d | sudo xargs chmod -R 777
$ sudo find
{ezpublish/{cache,logs,config},ezpublish_legacy/{design,extension,set
tings,var},web} -type f | sudo xargs chmod -R 666
```

Link assets

To be able to run eZ Publish 5 correctly, assets need to be exposed in the public "web" folder.

The following commands will first symlink eZ Publish 5 assets in "Bundles" and the second will symlink assets (design files like images, scripts and css, and files in var folder) from eZ Publish Legacy

```
cd /<ezp5-root>
php ezpublish/console assets:install --symlink web
php ezpublish/console ezpublish:legacy:assets_install --symlink web

# For 5.1 / 2013.4 and higher, also:
php ezpublish/console assetic:dump --env=prod web
```



Note: In both cases "web" is the default folder and can be skipped from the command. Further information about alternative options is available with `-h` just like it is with `"php ezpublish/console -h"`.



Warning regarding APC

If you are planning to use APC to speed up your site, please be sure to check the available notes [here](#), before entering the setup wizard chapter.

Configuring eZ Publish

The "site.ini.append.php" configuration file located in the "ezpublish_legacy/settings/override" directory of your eZ Publish installation must be changed, or else eZ Publish will not function properly. This file is the global override for the [site.ini](#) configuration file. There are a lot of things that need to be configured (database, mail transport system, var directory, etc.). The following text shows a generic example of a configuration that can be used:

```

<?php /* #?ini charset="utf-8"?

[DatabaseSettings]
DatabaseImplementation=ezmysql
Server=localhost
User=root
Password=
Database=my_database

[FileSettings]
VarDir=var/example

[Session]
SessionNameHandler=custom

[SiteSettings]
DefaultAccess=example
SiteList[]
SiteList[]=example

[SiteAccessSettings]
CheckValidity=false
AvailableSiteAccessList[]
AvailableSiteAccessList[]=example
AvailableSiteAccessList[]=example_admin
RelatedSiteAccessList[]
RelatedSiteAccessList[]=example
RelatedSiteAccessList[]=example_admin
MatchOrder=host;uri

# Host matching
HostMatchMapItems[]=www.example.com/example
HostMatchMapItems[]=admin.example.com/example_admin

[InformationCollectionSettings]
EmailReceiver=webmaster@example.com

[MailSettings]
Transport=sendmail
AdminEmail=webmaster@example.com
EmailSender=test@example.com

[RegionalSettings]
Locale=eng-GB
ContentObjectLocale=eng-GB
TextTranslation=disabled

*/ ?>

```

In the example above the "AvailableSiteAccessList[]" array located in the "[SiteAccessSettings]" section of this file determines the

available [siteaccesses](#) called "example" and "example_admin". The "CheckValidity" setting located in the same section should be set to false, otherwise the setup wizard will be initiated when trying to access the site.

In addition, two siteaccess configurations must be created, a public siteaccess ("example") and an administration siteaccess ("example_admin"). The following sub-directories have to be created in the root of your eZ Publish installation:

- `ezpublish_legacy/settings/siteaccess/example`
- `ezpublish_legacy/settings/siteaccess/example_admin`

Both siteaccesses must have a file called "site.ini.append.php".

The public siteaccess

The following text shows a generic solution for the "example" siteaccess:

```
<?php /* #?ini charset="utf-8"?

[SiteSettings]
SiteName=Example
SiteURL=www.example.com
LoginPage=embedded

[SiteAccessSettings]
RequireUserLogin=false
ShowHiddenNodes=false

[DesignSettings]
SiteDesign=example

[ContentSettings]
ViewCaching=disabled

[TemplateSettings]
TemplateCache=disabled
TemplateCompile=disabled
#ShowXHTMLCode=enabled
#Debug=enabled

[DebugSettings]
DebugOutput=enabled
Debug=inline
#DebugRedirection=enabled

[RegionalSettings]
SiteLanguageList[]
SiteLanguageList[]=eng-GB
ShowUntranslatedObjects=disabled

*/ ?>
```

The admin siteaccess

The following text shows a generic solution for the "example_admin" siteaccess:

```

<?php /* #?ini charset="utf-8"?

[SiteSettings]
SiteName=Example
SiteURL=admin.example.com
LoginPage=custom

[SiteAccessSettings]
RequireUserLogin=true
ShowHiddenNodes=true

[DesignSettings]
SiteDesign=admin

[ContentSettings]
CachedViewPreferences[full]=admin_navigation_content=0;admin_navigation_de
tails=0;admin_navigation_languages=0;admin_navigation_locations=0;admin_na
avigation_relations=0;admin_navigation_roles=0;admin_navigation_policies=0a
dmin_navigation_content=0;admin_navigation_translations=0;admin_children_v
iewmode=list;admin_list_limit=1;admin_edit_show_locations=0;admin_url_list
_limit=10;admin_url_view_limit=10;admin_section_list_limit=1;admin_orderli
st_sortfield=user_name;admin_orderlist_sortorder=desc;admin_search_stats_l
imit=1;admin_treemenu=1;admin_bookmarkmenu=1;admin_left_menu_width=13

[DebugSettings]
DebugOutput=disabled
Debug=inline

[RegionalSettings]
SiteLanguageList[ ]
SiteLanguageList[ ]=eng-GB
ShowUntranslatedObjects=enabled

*/ ?>

```

Note that database settings, mail settings, regional and other settings defined in "ezpublish_legacy/settings/override/site.ini.append.php" will be used for each siteaccess regardless of what is specified in the siteaccess settings. In the example above, the "Database=my_database" is specified under the "[DatabaseSettings]" section of this file so this database will be used for both "example" and "example_admin" siteaccesses. Refer to the "[Site management](#)" and "[Configuration](#)" sections of the "Concepts and basics" chapter for more information.

Activate base extensions

There are some legacy extensions that need to be active, as eZ Publish depends on them. This is usually done in the ezpublish_legacy/settings/override/site.ini.append.php settings file.

Here's an example of the extensions that you should activate:

```
[ExtensionSettings]
ActiveExtensions[]=ezjscore
ActiveExtensions[]=ezoe
ActiveExtensions[]=ezformtoken
ActiveExtensions[]=ezwt
ActiveExtensions[]=ezie
```

Languages

Available languages and their priorities can be controlled per siteaccess using the "SiteLanguageList" configuration setting located under the "[RegionalSettings]" section of the siteaccess "site.ini.append.php" file. If this setting is not specified, the system will use the old "ContentObjectLocale" setting and thus only the default language will be shown. Refer to the "Configuring the site languages" section for more information and examples.

The "cleandata.sql" script creates only one language which is the British English (eng-GB). All other languages should be added using the "Setup - Languages" part of the administration interface (<http://admin.example.com> in the example above).

Dynamic tree menu

If you have a large site with many nodes, it is strongly recommended to enable the "Dynamic" switch for your administration siteaccess. This will make the left tree menu in the administration interface work much faster and decrease the usage of network bandwidth.

Administrator's log-in and password

The following user name and password are set by the "cleandata.sql" script and can be used for logging in to the administration interface.

- User name: admin
- Password: publish

It is strongly recommended to change this password as soon as possible. Note that if you need another user name for site administrator, you can create a new administrator user, log in as this user and remove the old one.

Generate eZ Publish autoloads

The autoloads can be generated by running the `ezpgenerateautoloads.php` as shown in the example below:

```
cd /<ezpublish-5-root-dir>/ezpublish_legacy/
php bin/php/ezpgenerateautoloads.php
```

Generate eZ Publish .yaml configuration

To generate yaml configuration for the new Symfony stack a console command has been provided to cover single site setups.

Perform the following command where `<group>` represents the siteaccess group name which will be created for your configurations in `ezpublish.yaml`, for instance package name like "ezdemo_site", "ezwebin_site" or "ezflow_site". And `<admin_siteaccess>` represents your admin siteaccess, for instance, "ezdemo_site_admin":

```
cd /<ezpublish-5-root-dir>/  
  php ezpublish/console ezpublish:configure --env=prod <group>  
<admin_siteaccess>
```

If you instead would like to manually create your yml config, do the following:

- Copy `ezpublish/config/ezpublish.yml.example` to `ezpublish/config/ezpublish_prod.yml`
- Edit `ezpublish/config/ezpublish_prod.yml`