

# Installing eZ Publish on Windows

The requirements for doing a normal installation must be met. Read the "[Requirements for doing a normal installation](#)" section first!

Proceed only if you have access to a Windows based system with IIS, PHP, MySQL or PostgreSQL already installed and running (do not use Apache 2.x for Windows, at least for production servers).

This section will guide you through the following steps:

- [Setting up a database](#)
  - [MySQL](#)
  - [PostgreSQL](#)
- [Downloading eZ Publish](#)
- [Unpacking eZ Publish](#)
- [Link assets](#)
- [Virtual host](#)
- [Initiating the setup wizard](#)
  - [Document root example](#)
  - [Defining the desired environment \(optional\)](#)

## Setting up a database

**Known issue with running PHP5.3 on MySQL:** Some Windows users with both IPv4 and IPv6 installed experience problems connecting to the database server using host names like "localhost"... If you experience problems, try using IPv4 address like "127.0.0.1". This is due to a connectivity problem when running PHP5.3 on MySQL. So, please replace the database server name "localhost" with the IP address of the machine, or "127.0.0.1", which is reserved for the local host.

A database must be created before running the setup wizard. The following text explains how to set up a database using either MySQL or PostgreSQL.

### MySQL

1. Log in as the root user (or any other MySQL user that has the CREATE, CREATE USER and GRANT OPTION [privileges](#)):

```
mysql --host=<mysql_host> --port=<port> -u <mysql_user>
-p<mysql_password>
```

Note that if MySQL is installed on the same server, the "--host" parameter can be omitted. If the "--port" parameter is omitted, the default port for MySQL traffic will be used (port 3306).

The MySQL client should display a "mysql>" prompt.

2. Create a new database:

```
mysql> CREATE DATABASE <database> CHARACTER SET utf8;
```

3. Grant access permissions:

```
mysql> GRANT ALL ON <database>.* TO <user>@<ezp_host> IDENTIFIED BY
'<password>';
```

Note that if the specified user account does not exist, it will be created.

Reference	Description
<mysql_host>	The hostname of the MySQL database server.
<port>	The port number that will be used to connect to the MySQL database server.
<mysql_user>	The MySQL user (if no user is set up, use "root").
<mysql_password>	The password that belongs to the <mysql_user>.
<database>	The name of the database, for example "my_new_database".
<user>	The username that will be used to access the database.
<ezp_host>	The hostname of the server on which eZ Publish will be running. (may be 'localhost' if MySQL is installed on the same server).
<password>	The password you wish to set in order to limit access to the database.

## PostgreSQL

1. Log in as the postgres user (or any other PostgreSQL user that has sufficient [privileges](#) to create roles and databases):

```
psql -h <psql_host> -p <port> -U <psql_user> -W
```

Note that if PostgreSQL is installed on the same server, the "-h" parameter can be omitted. If the "-p" parameter is omitted, the default port for PostgreSQL traffic will be used (in most cases, port 5432).

The PostgreSQL client will ask you to specify the password that belongs to the <psql\_user>. If the password is correct, the client should display a "<psql\_user>=#" prompt.

2. Create a new database:

```
postgres=# CREATE DATABASE <database> ENCODING='utf8';
```

3. Create a new user:

```
postgres=# CREATE USER <user> PASSWORD '<password>';
```

4. Grant access permissions:

```
postgres=# GRANT ALL PRIVILEGES ON DATABASE <database> TO <user>;
```

5. Import the "pgcrypto" module into the new database:

```
postgres=# \c <database>
<database>=# \i '<path_to_pgcrypto>'
```

Reference	Description
<psql_host>	The hostname of the PostgreSQL database server.
<port>	The port number that will be used to connect to the PostgreSQL database server.

<psql_user>	The PostgreSQL user (if no user is set up, use "postgres").
<database>	The name of the database, for example "my_new_database".
<user>	The username that will be used to access the database.
<password>	The password you wish to set in order to limit access to the database.
<path_to_pgcrypto>	The path to the "pgcrypto.sql" file, for example "C:\\Program Files\\PostgreSQL\\8.2\\share\\contrib\\pgcrypto.sql".

## Downloading eZ Publish

The latest community version of eZ Publish can be downloaded from <http://share.ez.no/downloads>.

## Unpacking eZ Publish

Use your favorite utility to unpack the downloaded eZ Publish archive to a web-served directory (a directory that is reachable using a web browser). The extraction utility will unpack eZ Publish into a subdirectory called "ezpublish-5.x.y". Feel free to rename this directory to something more meaningful, for example "my\_site".

## Link assets

The download file should already have generated these assets for you as of 5.2 / 2013.07, but in case of issues on Windows with symlinks within the download file, make sure to run these commands.

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
php ezpublish/console assetic:dump --env=prod web
```

**Note:** In both cases "web" is the default folder and can be skipped from the command. For symlinks on the first two commands you can either use --relative, --symlink, or none to get the command to copy assets. However make sure to also update "symfony-assets-install" in composer.json if you prefer composer post-cmd to use something else then --relative.

Further information about alternative options is available with -h on each command, just like it is with the console itself using "php ezpublish/console -h".

On Windows, you will need an *elevated prompt*, which means you will need Administrator privileges for the symlink command to work. You may also as explained above skip this argument ("--symlink"), this will make the command copy the asset files instead of symlinking them.

## Virtual host

For help on how to set up a virtual host and rewrite rules on IIS, you will find some information on the community website (e.g. in [this forum thread](#)).

**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.

## Initiating the setup wizard

The setup wizard can be started using a web browser immediately after the previous steps (described in this section) are completed. It will be automatically run the first time someone tries to access/browse the `index.php` file located in the eZ Publish directory. Let's assume that we are using a server with the hostname `"www.example.com"` and that after unpacking, the eZ Publish directory was renamed to `"my_site"`.

You can choose between a [Virtual Host](#) based configuration, or a non Virtual Host configuration by using an `.htaccess` file.

## Document root example

If eZ Publish was unpacked into a directory called `"my_site"` under the document root, the setup wizard can be initiated by browsing the following URL: `http://www.example.com/my_site/index.php`.

Refer to ["The setup wizard"](#) section for a detailed description of the web based setup wizard.

## Defining the desired environment (optional)

This is a procedure to be done for the case you need to set up several multiple environments for development purposes.

Environment configuration is a new feature introduced as of eZ Publish 5.2 and eZ Publish Community Project 2013.06.

You can configure several environments, from production, development or staging, even if for each one of them you need require using different configurations sets.

In the `VirtualHost` example in the [Virtual host setup](#) chapter the required `VirtualHost` configurations have been already included. You can switch to the desired environment by setting the `ENVIRONMENT` environment variable to `"prod"`, `"dev"` or other custom value, as you can see in the following example:

```
# Environment.  
# Possible values: "prod" and "dev" out-of-the-box, other values  
possible with proper configuration (described below)  
# Defaults to "prod" if omitted  
SetEnv ENVIRONMENT "prod"
```

If you want to use a custom environment (something else then `"prod"` and `"dev"`) the next step is to create the dedicated configuration files for your environment:

- `ezpublish/config/config_<env_name>.yaml`
- `ezpublish/config/ezpublish_<env_name>.yaml`

The name used as `<env_name>` will be the one that can be used as value of the `ENVIRONMENT` environment variable.

Those files must import the main configuration file, just like the default `config_dev.yaml` already does. Here's an example:

```
imports:  
  - { resource: config.yaml }
```

This allows you to override settings defined in the main configuration file, depending on your environment (like the DB settings or any other setting you may want to override).