

Best Practices

Structuring an eZ Platform Project

Since release 2015.10, eZ Platform is very close to the [standard Symfony](#) distribution. It comes with default settings that will let you get started in a few minutes.

The AppBundle

Most projects can use the provided `AppBundle`, in the `src` folder. It is enabled by default. The project's PHP code (controllers, event listeners, etc.) can be placed there.

Reusable libraries should be packaged so that they can easily be managed using Composer.

Templates

Project templates should go into `app/Resources/views`.

They can then be referenced in code without any prefix, for example `app/Resources/views/content/full.html.twig` can be referenced in twig templates or PHP as `content/full.html.twig`.

Assets

Project assets should go into the `web` folder, and can be referenced as relative to the root, for example `web/js/script.js` can be referenced as `js/script.js` from templates.

All project assets are accessible through the `web/assets` path.

Configuration

Configuration may go into `app/config`. However, services definitions from `AppBundle` should go into `src/AppBundle/Resources/config`.

Versioning a project

The recommended method is to version the whole eZ platform repository. Per installation configuration should use `parameters.yml`.

eZ Platform Configuration

eZ Platform configuration is delivered using a number of dedicated configuration files. This config covers everything from selecting the content repository to siteaccesses to language settings.

Configuration format

Config files can have different formats. The recommended one is YAML, which is used by default in the kernel (and in examples throughout the documentation). However, you can also have configuration in XML or PHP formats.

In this topic:

- [Structuring an eZ Platform Project](#)
 - [The AppBundle](#)
 - [Templates](#)
 - [Assets](#)
 - [Configuration](#)
 - [Versioning a project](#)
- [eZ Platform Configuration](#)
 - [Configuration format](#)
 - [Configuration files](#)
 - [Specific configuration](#)

Related topics:

[Creating a new design using Bundle Inheritance](#)

Basic configuration handling in eZ Platform is similar to the usual Symfony config. To use it, you define key/value pairs in your configuration files. Internally and by convention, keys follow a dot syntax where the different segments follow your configuration hierarchy. Keys are usually prefixed by a namespace corresponding to your application. Values can be anything, including arrays and deep hashes.

Configuration files

Main configuration files are located in the `app/config` folder.

If you use the provided `AppBundle`, service definitions from it should go into `src/AppBundle/Resources/config`.

- `parameters.yml` contains infrastructure-related configuration. It is created based on the default settings defined in `parameters.yml.dist`.
- `config.yml` contains configuration stemming from Symfony and covers settings such as search engine or cache configuration.
- `ezplatform.yml` contains general configuration that is specific for eZ Platform, like for example `siteaccess` settings.
- `security.yml` is the place for security-related settings.
- `routing.yml` defines routes that will be used throughout the application.

Configuration can be made environment-specific using separate files for each environment. These files will contain additional settings and point to the general (not environment-specific) configuration that will be applied in other cases.

Here you can read more about [how configuration is handled in Symfony](#).

Specific configuration

The configuration of specific aspects of eZ Platform is described in the respective topics in [The Complete Guide to eZ Platform](#).

Selected configuration topics:

- [View provider](#)
- [Logging and debug](#)
- [Content repository](#)
- [Authentication](#)
- [Sessions](#)
- [Siteaccess](#)