

## TASK- JIRA

<https://jira.ez.no/browse/EZP-22209> (As a reader I want to have a dictionary of eZ terms).

CREATE	DELIVER	OPTIMIZE
Author	Personalize	Learn
Capture	Recommend	Adapt
Collaborate	Engage	Convert

**Build your website**  
**Collaborate with your**

### eZ Publish Platform

>>UIs

>>APIs

>>Business Logic

>>Content

<http://www.slideshare.net/jeanvoye/20130527-opening-the-experience-community-ez-unconference2-ez-publish-5-tech-event-montpellier-france?related=1>  
(slide: 53/57)

## PRESENTATIONS:

### eZ publish sources:

<http://www.slideshare.net/search/slideshow?searchfrom=header&q=eZ+publish>

- **eZ Publish Platform 5.4 Release Webinar (by Roland)**

<https://www.youtube.com/watch?v=KSSeHuxVz3Y>

- **eZ Publish Free Developer Basics Training - London - 2011 April 13th by Nicolas Pastorino**

[http://www.slideshare.net/ezcommunity/ez-publish-free-developer-basics-training-london-2011-april-13th?next\\_slideshow=1](http://www.slideshare.net/ezcommunity/ez-publish-free-developer-basics-training-london-2011-april-13th?next_slideshow=1)

- **eZ Publish Platform 5.2 and roadmap**

<http://www.slideshare.net/RolandBenedetti1/ez-publish-platform-52-and-roadmap>

### Requirement for upgrade of eZ documentation

<https://doc.ez.no/pages/viewpage.action?title=Requirement+for+upgrade+of+the+eZ+documentation&spaceKey=PR>

## **Guide to be fluent**

[http://www.ezpedia.org/learning/guide\\_to\\_becoming\\_fluent\\_in\\_ez](http://www.ezpedia.org/learning/guide_to_becoming_fluent_in_ez)

## **Sarah's space**

<https://doc.ez.no/display/SAR3/FAQ>

## **Glossary index - editing**

<https://doc.ez.no/pages/listpages-dirview.action?key=glossary&openId=23528155#selectedPageInHierarchy>

## **eZ Publish Platform Documentation - Technical Doc**

Welcome to the Technical Documentation of the eZ Publish Platform

<https://doc.ez.no/display/EZP6/eZ+Publish+Platform+Documentation>

## **eZ Publish Community Edition & eZ Publish Platform : Enterprise Edition**

<http://ez.no/Products/Compare-Versions>

## **eZ Publish development**

<http://share.ez.no/get-involved/develop#resources>

## **eZ Products**

<https://doc.ez.no/display/EZP6/eZ+Publish+Platform+Documentation>

## **eZ Business Partner**

<http://ez.no/Partners/Success-with-eZ>

## **eZ Marketing Automation**

<https://doc.ez.no/display/EZMA/eZ+Marketing+Automation+documentation>

## **eZ Publish Community Server (API doc and sources, )**

[http://pubsvn.ez.no/#other\\_resources](http://pubsvn.ez.no/#other_resources)

- **eZ Community website** - [share.ez.no](http://share.ez.no)
- **GitHub** - <https://github.com/ezsystems>
  - [PlatformUIBundle](#)
  - [ezpublish-community](#)
  - [ezpublish-docker](#)
  - [ezpublish-kernel](#)
  - [ezpublish-legacy](#)
  - [ezfind](#)

- DemoBundle
- BehatBundle
- PlatformUIAssetsBundle
- ezcomments-ls-extension
- ezdemo-ls-extension
- ezflow-ls-extension
- ezwebin-ls-extension
- ezstarrating-ls-extension
- ezgmaplocation-ls-extension
- ezwt-ls-extension
- ezpublish-spi
- ezpublish-api
- ez-js-rest-client
- satis
- 
- **Jira** - <https://jira.ez.no/secure/Dashboard.jspa>
- **eZ Publish Extensions** - [Projects.ez.no](http://projects.ez.no)
- **eZpedia community wiki** - <http://www.ezpedia.org/>
- **Feeds** -Planet eZ Publish aggregates posts from a large number of eZ Publish blogs.  
<http://www.planetepublish.org/>
- **Documentation Center**
  - **Legacy Doc:** <https://doc.ez.no/eZ-Publish/>
  - **5.x Doc:** <https://doc.ez.no/display/MAIN/eZ+Documentation+Center>
  - **Code Documentation (eZ Publish 5.x Kernel):**  
<https://github.com/ezsystems/ezpublish-kernel>  
<https://github.com/ezsystems/ezpublish-kernel/tree/master/doc>
- **API Doc**

The API documentation: <http://pubsvn.ez.no/doxygen/trunk/html/index.html>.  
An alternative documentation source is <http://ezpedia.org>, which complements the API documentation.  
SVN: <http://pubsvn.ez.no/>
- **Training and Certification**

eZ Systems also provides a large range of training programs  
<http://ez.no/eZPublish/Training-and-Certification>

## **eZ Coding Standards**

[http://www.ezpedia.org/new\\_stack/code\\_standards](http://www.ezpedia.org/new_stack/code_standards)

## **eZ Publish Components- JIRA**

<https://jira.ez.no/browse/EZPFT#selectedTab=com.atlassian.jira.plugin.system.project%3Acomponents-panel>

[eZ Find](#)

[eZ Flow](#)

[eZ Networks](#)

[eZ ODF](#)

[eZ OE](#)

[eZ Publish](#)

[eZ Publish 5 Cloud](#)

[eZ Publish 5 LS](#)

[eZ Publish 5.x - Admin  
interface](#)

[eZ Publish 5.x - Content  
management interface](#)

[eZ Publish 5.x - Demo site](#)

[eZ Publish 5.x - Installation/  
upgrade wizard](#)

[eZ Publish 5.x - kernel](#)

[eZ Publish 5.x - Navigation hub  
interface](#)

[eZ Publish 5.x - Public API](#)

[eZ Publish 5.x - Rest api](#)

[eZ Publish Suppor](#)

[eZ Services](#)

## **eZ PROJECTS**

<http://projects.ez.no/> The community contribution corner

<https://jira.ez.no/secure/BrowseProjects.jspx#10000>

## Departments

Project	Key	Project Lead	URL
<a href="#">Day to day Product Managment</a>	DEPPM	<a href="#">Roland Benedetti</a>	No URL
<a href="#">Internal</a>	INTERNAL	<a href="#">André Rømcke</a>	No URL
<a href="#">IT Support</a>	ITSUPPORT	<a href="#">Jostein Knudsen</a>	No URL
<a href="#">Quality Assurance</a>	QA	<a href="#">André Rømcke</a>	No UR

## Recent Projects

Project	Key	Project Lead	URL
<a href="#">eZ Publish Features</a>	EZPFT	<a href="#">Roland Benedetti</a>	No URL
<a href="#">IT Support</a>	ITSUPPORT	<a href="#">Jostein Knudsen</a>	No URL
<a href="#">Quality Assurance</a>	QA	<a href="#">André Rømcke</a>	No URL
<a href="#">Internal</a>	INTERNAL	<a href="#">André Rømcke</a>	No URL
<a href="#">Day to day Product Managment</a>	DEPPM	<a href="#">Roland Benedetti</a>	No URL
<a href="#">eZ Publish</a>	EZP	<a href="#">André Rømcke</a>	<a href="https://github.com/ezsystems/ezpublish-community">https://github.com/ezsystems/ezpublish-community</a>
<a href="#">eZ Market</a>	EZM	<a href="#">Łukasz Serwatka</a>	No URL
<a href="#">eZ Studio</a>	EZS	<a href="#">Roland Benedetti</a>	No URL

## IT support

## IT support

Component	Lead	Description
<a href="#">Access Administration</a>		Request or remove access to something
<a href="#">Box</a>		Box file storage and collaboration
<a href="#">Confluence</a>		Documentation management system
<a href="#">doc.ez.no</a>		Issues related to the integration of doc.ez.no and confluence
<a href="#">Email</a>		Email related request or issue
<a href="#">Equipment request</a>		Request equipment (PO required)

<a href="#">Google</a>		issue related to Google Apps
<a href="#">Helpdesk (Mac)</a>		Mac-user assistance
<a href="#">Helpdesk (Windows)</a>		Windows-user assistance
<a href="#">Hosting issue</a>		Issue related to existing hosting
<a href="#">Hosting request</a>		Request for hosting services (PO required)
<a href="#">Intranet (i.ez.no)</a>	<a href="#">Jarosław Heba</a>	The eZ Intranet
<a href="#">JIRA</a>		Issue related to JIRA
<a href="#">Login issue</a>		Issue with logging in to any service or resource
<a href="#">login.ez.no (Selfcare site)</a>	<a href="#">Jarosław Heba</a>	The login and Selfcare site
<a href="#">Network</a>		Network issue
<a href="#">Office365</a>		Office365 related issue
<a href="#">Powwownow</a>		Phone conference system
<a href="#">Printer</a>		Printer issue
<a href="#">SalesForce</a>	<a href="#">Maik Seyring</a>	Issue related to SalesForce
<a href="#">support.ez.no</a>	<a href="#">Jarosław Heba</a>	The Service Portal site
<a href="#">TheWire</a>	<a href="#">Jarosław Heba</a>	The internal social channel

## LABELS

### A-L

[Licenses](#)

[Look+Feel](#)

[cloud](#) -

**eZ Publish Cloud** (<http://admin.cloud-test.ez.no/>) Administration Area for your Environments

<https://doc.ez.no/display/EZHE/Architecture+Documents>

[confluence](#)

[documentation](#)

[hosts](#)

[infrastructure](#)

[internal](#)

[internal,](#)

[jira](#)

[login](#)

## M-Z

[Procurement](#) [QA](#) [SSO](#) [Support](#) [networking](#); [qa](#) [support](#) [wIT](#) [www](#);

# Blogs:

## Core Development Team:

<http://share.ez.no/blogs/core-development-team>

## eZ Summer Camp 2014 Rovinj, Croatia by Netgen

- eZ Summer Camp 2014 Workshops Retrospective

<http://share.ez.no/blogs/core-development-team/ez-summer-camp-2014-workshops-retrospective>

- Extending the Summer Camp learning experience with first batch of workshop videos

<http://share.ez.no/blogs/ivo-lukac/extending-the-summer-camp-learning-experience-with-first-batch-of-workshop-videos>

- \* **An Interactive Deep Dive Into eZ Product Backlog**

INTERMEDIATE and ADVANCED level

- \* **Migrating from Legacy, the right way (part 1)**

INTERMEDIATE level

- \* **Migrating from Legacy, the right way (part 2)**

INTERMEDIATE level

- \* **The hybrid approach**

INTERMEDIATE level

- \* **Making a multi language, semantical Web-site/-app (part 1)**

BASIC level

- \* **Making a multi language, semantical Web-site/-app (part 2)**

INTERMEDIATE level

- \* **New features in eZ Publish 2014.07 (5.3 & 5.4)**  
INTERMEDIATE level
- \* **Organising your eZ Publish app**  
ADVANCED level
- \* **Learnings from Real eZ Publish 5 Projects**  
INTERMEDIATE level
- \* **eZ Publish 5 from a Symfony perspective**  
INTERMEDIATE level

## Roland's blog

<http://www.scoop.it/t/ez-publish?page=1>  
<http://www.scoop.it/t/ez-publish?page=2>

## Images

<http://share.ez.no/blogs/core-development-team/images-on-amazon-s3-it-is-coming>  
<https://github.com/brookinsconsulting/ezs3upload>

# Glossary:

## I.

<https://doc.ez.no/pages/viewpage.action?pageId=7438527>

**Caution:** This documentation is for **eZ Publish legacy**, from version 3.x to 5.x.  
 For 5.x documentation covering *Platform* see [eZ Documentation Center](#), for difference between *legacy* and *Platform* see [5.x Architecture overview](#).

<https://doc.ez.no/eZ-Publish/Technical-manual/4.x/Concepts-and-basics/The-internal-structure-of-eZ-Publish/Directory-structure>

## eZ Publish 5 Architecture - Introduction & Overview

### eZ Publish 4.x vs eZ Publish Platform 5.x

#### 4.x feature comparison

<https://doc.ez.no/display/EZP/4.x+feature+comparison>

#### eZ Publish 4 vs eZ Publish 5 feature comparison

[https://docs.google.com/a/ez.no/spreadsheet/ccc?](https://docs.google.com/a/ez.no/spreadsheet/ccc?key=0AgGKn8SQ_akMdfQ3VU0wMIR3cENxTDZ2dkhUcVN2SWc#gid=0)

[key=0AgGKn8SQ\\_akMdfQ3VU0wMIR3cENxTDZ2dkhUcVN2SWc#gid=0](https://docs.google.com/a/ez.no/spreadsheet/ccc?key=0AgGKn8SQ_akMdfQ3VU0wMIR3cENxTDZ2dkhUcVN2SWc#gid=0)

#### Demystifying eZ Publish 5.x for eZ Publish 4.0

[http://www.slideshare.net/Kaliop-slide/kaliop-webinar-ezpublish5?qid=5eec4bdc-ff76-4dd9-aea5-bb48ab27154f&v=qf1&b=&from\\_search=62](http://www.slideshare.net/Kaliop-slide/kaliop-webinar-ezpublish5?qid=5eec4bdc-ff76-4dd9-aea5-bb48ab27154f&v=qf1&b=&from_search=62)



## With the 5.0 release eZ Publish

### Technology

- AccessSite

from installation wizard: Site access configuration

Please choose the access method you want to use for your site. The access method determines how the site will be accessed from within a web browser. If unsure: choose URL.

Access method:

URL (recommended)

Port

Hostname

- API
  - Public API
  - REST API
- eZ Publish
  - Platform
- eZ Systems
- **Symfony**
- **Twig** - a standard template engine to simplify working with templates in eZ Publish
- Storage system - allows scalability, performance and maintainability
- Backward compatibility -
- 
- Customer and User Experience Management (requirements)
  - web content management
  - digital user experience
- Performance and scalability
- The big data situation
- digital services
- **The multichannel vision/delivery** - (from v 5.x) incorporates many of the desirable changes that enable content to live in any screen, any app, any device (tablets, smartphones, desktop, TVs, apps, third-party applications, etc)  
eZ Publish makes it possible to deliver content and experiences anywhere at anytime. This is made possible by a [REST interface](#) (in 5.2) that lets you rise to the omnichannel/multichannel challenge, with an extensible, built-in content exposure system.  
<http://ez.no/Products/Content-Management>
- **Digital Service Excellence**

<http://www.ezpedia.org/ez>

<http://www.netgenlabs.com/Blog/eZ-Publish-CMS-persistent-variables>

## -A-

### API

**New in 5.3.3 and 5.4: Improved API performance**

<http://share.ez.no/blogs/core-development-team/new-in-5.3.3-and-5.4-improved-api-performance>

### Access Control in eZ Publish

<https://doc.ez.no/eZ-Publish/Technical-manual/4.x/Concepts-and-basics/Access-control>

## -B-

### BUL

Business User License (e.g eZ Studio)

### Bundles

Bundles are the new extensions. Studying their source code is also valuable.

[http://www.ezpedia.org/new\\_stack/documentation](http://www.ezpedia.org/new_stack/documentation)

For more information on Bundle types:

- [DemoBundle](#) - represents a demo front-end website with eZ Publish 5
- [PlatformUIBundle](#) - is a bundle for eZ Publish Platform providing a webapp application to manage your content and administrate your eZ Publish Platform install.
- [BehatBundle](#) - Bundle for common reusable sentence implementations & other common needs for Behat testing in eZ bundles/projects.  
Brings the Behavior Driven Development (as known as BDD) into the eZ Publish.
- [PlatformUIAssetsBundle](#) - provides the external frontend dependencies needed by the eZ Platform UI.  
Repo for containing assets for PlatformUIBundle
- [CommentsBundle](#) - is an versatile extension for eZ Publish 5+ making commenting easy.
- [ngsymfonytools-bundle](#) - Integration of netgen/ngsymfonytools as a Legacy Bundle

## -C-

### Cache

Content cache is fully [Http cache](#).

**Q:** How to clear caches for eZ Publish 5? Is there a single action to clear caches for both stacks?

**A:** Content (Http) cache is purged at publish time or from admin interface.

You can use the `cache:clear` command to clear all Symfony cache and in 'dev' environment is disabled (see [Symfony doc on environments](#)) . No unified command yet.

eZ Publish Cache In Details: expiry.php and ezcache.php command

<http://share.ez.no/learn/ez-publish/ez-publish-cache-in-details-expiry.php-and-ezcache.php-command>

New stack uses HTTP cache. It replaces view cache in legacy (However it is still presents in legacy).

You almost always have your pages in cache. First time it is loaded, cached by the reverse proxy (like **Varnish**, or internal one provided by Symfony)

<http://www.netgenlabs.com/Blog/The-eZ-Publish-Show-17-How-to-start-with-eZ-Publish-5-stack>

(18:00)

**3.5 more time faster than 4.7 Publish (JMETER was used)**

**eZ Publish Performance Optimization Part 3 of 3: Practical Cache and Template Solutions** (25 January 2007)

<http://share.ez.no/learn/ez-publish/ez-publish-performance-optimization-part-3-of-3-practical-cache-and-template-solutions/%28page%29/2>

## More info on Caches

### Caches

<http://share.ez.no/content/keyword/Caches>

eZ Publish Cache In Details: Template Compilation And INI Cache

eZ Publish Cache In Details: View Cache

eZ Publish Cache In Details: Template-block Cache

eZ Publish Cache In Details: expiry.php and ezcache.php command

### **New in eZ 5.4: HTTP cache reloaded**

<http://share.ez.no/blogs/core-development-team/new-in-ez-5.4-http-cache-reloaded>

### **Overview of caching in eZ Publish new stack (15.10.2014 - Ivo Lukač)**

<http://www.netgenlabs.com/Blog/Overview-of-caching-in-eZ-Publish-new-stack>

### **Types of caches (slide 36/57)**

(eZ Publish 5 basics training material from Philippe Vincent-Royol & Florent Huck)

<http://www.slideshare.net/ezcommunity/ez-publish-5-basics-philippe-vincentroyol-florent-huck>

=====

**HTTP cache** - [http://symfony.com/doc/current/book/http\\_cache.html](http://symfony.com/doc/current/book/http_cache.html)

=====

### **New in eZ 5.4: HTTP cache reloaded**

<http://share.ez.no/blogs/core-development-team/new-in-ez-5.4-http-cache-reloaded>

### **Context aware HTTP cache**

[https://github.com/ezsystems/ezpublish-kernel/blob/v2014.11.4/doc/specifications/cache/context\\_aware\\_http\\_cache.md](https://github.com/ezsystems/ezpublish-kernel/blob/v2014.11.4/doc/specifications/cache/context_aware_http_cache.md)

## CLA - Contributor Licence Agreement

If you are thinking of developing major features, implying a large set of code changes, you first have to submit your idea to the Board, which ensures a proper synchronization of all initiatives.

Before submitting your contribution you need to sign a Contributor Licensing Agreement (CLA)

<http://share.ez.no/get-involved/develop#ezpublish>

<http://share.ez.no/community-project/contributor-licensing-agreement-cla>

## Cloud

eZ Publish Cloud User Doc: <https://doc.ez.no/display/EZHE/User+Documentation>

## CMS

Content Management System

## CXM

Customer Experience Management

Learn more on eZ Publish Platform [CXM capabilities](#).

## Coding Standards

eZ Coding Standards are the foundations to the eZ consistent, readable code.

They MUST be read, known and followed by every eZ Publish developer & contributor, internal as well as external.

They are based on the **eZ Components** coding standards ([http://ezcomponents.org/contributing/coding\\_standards](http://ezcomponents.org/contributing/coding_standards))

GitHub: <https://github.com/ezsystems/ezpublish-kernel/wiki/codingstandards>

Blog: <http://share.ez.no/blogs/bertrand-dunogier/the-ez-coding-standards-need-you>

## Community

The eZ Community is open to professionals, hobbyists and anyone with a passion for innovative Web platforms

eZ Publish Community has the following resources:

**Info:**

<http://ez.no/Community>

**Website:**

[share.ez.no](http://share.ez.no)

**Projects (contribution corner):**

<http://projects.ez.no/>

**GitHub:**

<https://github.com/>

[ezsystems/ezpublish-community](http://ezsystems/ezpublish-community)

**Server (Release history, Downloads, Other resources):**

<http://pubsvn.ez.no/>

[#other\\_resources](#)

**Wiki:**

<http://www.ezpedia.org>

## Complex Web Setups

Refer to Multi-sites

## Components

eZ Components is a high-quality PHP library of independent [components](#) that aid in the development of Web-centric applications.

<http://ezcomponents.org/introduction>

eZ Components were substituted by **Zeta Components** since July 2010.

## Composer

is an open source PHP packaging system to manage dependancies. This makes it easy to adapt package installs and updates to your workflow, allowing you to test new/updated packages in a development environment, put the changes in your version con

<https://doc.ez.no/display/EZP/Using+Composer>

**For more information on Compose visit** <https://getcomposer.org/>

**Blog: The eZ Publish Show #22: Introducing Composer**

<http://www.netgenlabs.com/Blog/The-eZ-Publish-Show-22-Introducing-Composer>

**Next episode of The eZ Publish Show: Introducing Composer**

<http://share.ez.no/blogs/ivo-lukac/next-episode-of-the-ez-publish-show-introducing-composer>

## Content engine

With the eZ Publish Platform content engine, all content, including articles and images, can be published at multiple locations, as well as embedded into other content.

An image, for example, can be used in different articles, or the same article could be published in different locations. One example might be a single news item appearing in both a Company and Community News section.

<http://ez.no/Products/Content-Management>

## Content caching

Integrated content caching

**content purging (since v 5.2)** - when you publish a content the cash is automatically purged for it

**Content aware (since v 5.2)** - you can have a shared caching

<http://www.netgenlabs.com/Blog/The-eZ-Publish-Show-17-How-to-start-with-eZ-Publish-5-stack>  
(25:00)

**content is gold** - <http://ez.no/Blog/Content-Is-Gold>

## -D-

### Docker

#### Demo Bundle

Refer to Bundles

#### Dependency injection

New in eZ 5.4: Dynamic settings injection

<http://share.ez.no/blogs/core-development-team/new-in-ez-5.4-dynamic-settings-injection>

#### Design system

#### Dual kernel

Refer to Kernel

## -E-

**eZ Platform** (platform for developers and designers to build remarkable experiences)

**eZ Platform** will be fully open source, developed together with the eZ Publish community -- fully supported and commercially maintained by eZ Systems.

It is a product for developers that arms them with a solid framework, strong APIs, unequaled extensibility, user interfaces that fit developer needs, and tools and resources to help them build great websites and content-driven apps.

**eZ Platform** will be the next generation of eZ Publish - something we might have referred to as eZ Publish 6.

We will be leaving the term “publish” on the roadside. “Publish” is part of our legacy.

**eZ Platform** is being built on top of the new eZ Publish 5.x kernel and, on the technical side, relies heavily on the Symfony2 framework.

It won't include any “legacy” code from the 3 and 4.x generations - no more dependency.

The user interface will cover:

**content types**, defining information and service architecture, managing users, roles, permissions, and configuring integration and extensions.

Additionally, it will offer an **editorial interface** that gives the ability to edit content, as this is something developers need when they build applications.

Platform will be more open, more open-source, more focused toward developers.

And upgrading to eZ Platform will be easy, thanks to the effort we put into building

5.x with its dual kernel approach.

**eZ Platform** will:

- provide all the APIs and technical components needed for a developer to build a project, as well as the content repository itself
- produce the generic content management interface, similar to the old administration interface
- enable users to browse the content repository and create, edit and delete content of any type
- feature all of the administrative elements of the user interface, such as the management of users, content model, roles and permissions

**eZ Platform = eZ Publish Enterprise + eZ Publish Community**

Unlike the previous versions, where **eZ Publish Enterprise** and **eZ Publish Community** were two distinct software codebases -- even if very close one another -- there will only be one eZ Platform. It will be developed and maintained openly on Github public repositories and distributed under the GPL license.

<http://ez.no/Blog/eZ-Publish-Platform-5.4-is-Almost-Here-What-s-Coming-Next>

Introduction: <http://share.ez.no/forums/discussions/the-future-ez-publish-platform-backend-ui-is-here>  
Early versions of the [new eZ Platform interface](#) are already available, as [shared on this blog](#).

**GitHub:** [PlatformUIBundle git repository on Github](#).

[PlatformUIBundle/docs/extensibility.md](#)

<https://github.com/ezsystems/PlatformUIBundle/blob/master/docs/>

[extensibility.md](#)

**The future eZ Publish Platform backend UI is "here"!**: <http://share.ez.no/blogs/core-development-team/the-future-ez-publish-platform-backend-ui-is-here>

**Platform UI December 2014 status:** <https://docs.google.com/a/ez.no/document/d/17Kbv3pQ4K88R9cINSyTJWP1QbBBWOu5cI1kt-7FFudE/edit#heading=h.2dxcyb7p5pi7>

- **PlatformUI field type support:** <https://github.com/ezsystems/PlatformUIBundle/blob/master/docs/field-types-support.md>

**Changes to YUI and how it affects eZ Publish Platform**

<http://share.ez.no/blogs/core-development-team/changes-to-yui-and-how-it-affects-ez-publish-platform>

*On August 29th Yahoo! [announced](#) that they are dropping further development of YUI, the javascript framework eZ Publish Platform have been using for several years now.*

## Architecture

The Platform UI Application is based on the [YUI Library](#) and especially on the [YUI App framework](#).

We chose YUI because it provides a full set of consistent and high quality components.

As written before, the PlatformUIBundle provides an application which is executed by the browser and

is divided into two different parts with two different strategies in the way they build the pages:

### **eZ Platform consists of two parts:**

- **Editorial** - uses [our JavaScript REST Client](#) to issue REST calls in order to interact with the eZ Publish Platform installation.  
From those REST calls, it extracts the data it needs and basically executes some templates to generate the resulting HTML code
- **Admin** - *pages* are the result of one regular AJAX request which response is of course generated by a regular Symfony controller and where you can use for instance a Twig template.

### **Frontend**

As already decided in the past we will avoid use of YUI in Frontend (DemoBundle, ..), and continue to use Bootstrap and jQuery here. However we will most likely prefer HTML5+ / ES6+ / CSS3+ features over use of jQuery which with time might get the same fate.

### **Backend**

1. As our future [Platform UI](#) both allows Symfony based (traditional, YUI ajax enhanced) and
  - YUI based (rich single page app) development we will from now on favor the former for new interfaces.
  - This will allow us to potentially speed up the development of the new interface as we have more developers
  - available to help out on this, and as it is not a layer on it's own it does not need the same amount of testing
  - (mainly functional acceptance testing needed).
2. Furthermore for extension points for community/partners we will try to abstracted those from the framework, similarly to how we also do in our backend PHP code.
3. We will in the future prototype and find a replacement for the rich client part of the Platform UI, and we would like to do this together with the community. However our overall main priority is time to market for the new User Interfaces, so we will most likely not finalize such an effort before we launch first couple of versions.

### **=====** **eZ Publish/Platform technical point of view** **=====**

#### **eZ Publish 5 from a technical point of view - part 1**

<http://share.ez.no/blogs/jerome-vieilledent/ez-publish-5-from-a-technical-point-of-view-part-1>

#### **eZ Publish 5 from a technical point of view, Part 2**

<http://share.ez.no/blogs/jerome-vieilledent/ez-publish-5-from-a-technical-point-of-view-part-2>



## **An Explosive Cocktail: Symfony and eZ Publish 5 Joining Forces**

<http://share.ez.no/blogs/ez/an-explosive-cocktail-symfony-and-ez-publish-5-joining-forces>

## **Changes: #1 Opening up, starting with this dev blog**

<http://share.ez.no/blogs/core-development-team/changes-1-opening-up-starting-with-this-dev-blog>

## **Changes #2: Technology Outlook towards eZ Platform**

<http://share.ez.no/blogs/core-development-team/changes-2-technology-outlook-towards-ez-platform>

## **More info on eZ Platform:**

### **Roland's BLOG: eZ Publish Next UI**

<http://share.ez.no/blogs/ez-publish-next-ui>

### **The New eZ Platform User Interface is on its Way**

<http://2012.ez.no/Blog/The-New-eZ-Platform-User-Interface-is-on-its-Way>

### **Introducing eZ Platform And Its New Development Cycle**

<http://ez.no/Blog/Introducing-eZ-Platform-And-Its-New-Development-Cycle>

### **The future eZ Publish Platform backend UI is "here"!**

<http://share.ez.no/blogs/core-development-team/the-future-ez-publish-platform-backend-ui-is-here>

### **eZ Publish 5 Platform**

<http://share.ez.no/forums/ez-publish-5-platform>

### **Stories (by Roland Benedetti):**

#### **Redefining eZ Publish User Interface on eZ Publish 5 UX & UI concepts**

**#1: Basic Edit:** <http://www.slideshare.net/RolandBenedetti1/story-1-17020612>

**#1.1: Basic Edit - sub Items:** <http://www.slideshare.net/RolandBenedetti1/story-11basicedit-subitems>

- **#2: Create a container:** <http://www.slideshare.net/RolandBenedetti1/story-2search-create-container>

**#3: Landing Page with form wall:** <http://www.slideshare.net/RolandBenedetti1/story-3landing-page-with-form-wall?related=1>

**#4: Modify Frontpage:** <http://www.slideshare.net/RolandBenedetti1/story-4-frontpage>

<https://github.com/eZsystems/PlatformUIBundle>

## eZ Projects

Develop and share eZ Publish extensions

<http://projects.ez.no/>

<http://share.ez.no/get-involved/develop#ezpublish>

## eZ Publish Platform

It is an intuitive and simple open source Content Management System (CMS) which provides a powerful digital management solutions across multiple processes and systems.

The eZ Publish Platform offers Content Management (**CMS**) & Customer Experience (**CXM**) solutions

that enable you to build and run a wide range of websites and applications

<http://ez.no/News/eZ-Publish-Platform-5.2-Faster-Stronger-For-Your-Content-Business>

Basically there are 3 main ways to use eZ Publish 5:

- **Pure legacy mode:** using only the legacy part, as you would use eZ Publish 4.7.  
(note: there is also a “legacy mode” in the Hybrid approach that is [discussed here](#) and should not be confused with this “pure legacy” approach)
- **Legacy mode:** only for administrators interface. Not for website
- **New Stack mode:** using the new Symfony based architecture and Twig based templates.

Minimum cost is to migrate your template layout and adapt settings for the new format.

<http://www.netgenlabs.com/Blog/The-eZ-Publish-Show-17-How-to-start-with-eZ-Publish-5-stack> (12:00 - 12:15)

- **Hybrid:** using the legacy within the new stack, authorizing everything that was done in legacy, but going through the new kernel and the Symfony stack.

Pure symphony is compatible with Varnish

## More info on eZ Publish:

### **eZ Publish 5 Architecture - Introduction & Overview**

<https://doc.ez.no/pages/viewpage.action?pageId=12780126>

### **Migrating from eZ Publish 4 to eZ Publish 5: basic concepts (by Łukasz Serwatka)**

<http://share.ez.no/blogs/ez/migrating-from-ez-publish-4-to-ez-publish-5-basic-concepts>

=====

## **eZ Publish version 5.1 and beyond**

=====

### **eZ Publish 5.1 Platform at a glance - Release Notes**

<http://share.ez.no/blogs/ez/ez-publish-5.1-platform-at-a-glance-release-notes>

=====

## **eZ Publish version 5.3 and beyond**

=====

### **5.3 to come with broader platform support and improved performance**

<http://share.ez.no/blogs/core-development-team/5.3-to-come-with-broader-platform-support-and-improved-performance>

### **eZ Publish Platform 5.3 Hitting the Road!**

<http://ez.no/Blog/eZ-Publish-Platform-5.3-Hitting-the-Road>

### **eZ Publish Community Project 2014.05 available now**

<http://share.ez.no/blogs/community-project-board/ez-publish-community-project-2014.05-available-now>

=====

## **eZ Publish version 5.4 and beyond**

=====

### **The eZ Publish Show #21: eZ Publish version 5.4 and beyond**

<http://share.ez.no/blogs/ivo-lukac/the-ez-publish-show-21-ez-publish-version-5.4-and-beyond>

<http://share.ez.no/blogs/damien-pobel/conference-at-the-forum-php-2013-create-an-ez-publish-website-in-45-minutes>

## **ESI (Edge Side Includes)**

### **Using Edge Side Includes**

[http://symfony.com/doc/current/book/http\\_cache.html#edge-side-includes](http://symfony.com/doc/current/book/http_cache.html#edge-side-includes)

**eZ Studio** (for editors and users, built on top of this eZ Platform, for content creators to create, manage and deliver relevant content.)

**eZ Studio** will be a commercial product distributed under the eZ Business User License (BUL).

It will not be developed openly and can only be used if purchased from eZ Systems. eZ Studio will be a product with added value for business users and is really targeted for them.

<http://ez.no/Blog/eZ-Publish-Platform-5.4-is-Almost-Here-What-s-Coming-Next>

## Extensions

### What are eZ Publish extensions

<https://doc.ez.no/Tutorials/Developing-eZ-Publish-extensions/What-are-eZ-Publish-extensions>

- **ezwt extension -**

### Migrating extensions to eZ Publish5 (Symfony 2 bundle system)

- [http://www.slideshare.net/lserwatka/migrating-ez?next\\_slideshow=1](http://www.slideshare.net/lserwatka/migrating-ez?next_slideshow=1)

### List of eZ Publish extensions and their support options

- <https://doc.ez.no/List-of-eZ-Publish-extensions-and-their-support-options>

## Extensions

<http://share.ez.no/forums/extensions>

### An Introduction to Developing eZ Publish Extensions

<http://share.ez.no/learn/ez-publish/an-introduction-to-developing-ez-publish-extensions>

### editable:

<http://www.pdfdrive.net/an-introduction-to-developing-ez-publish-extensions-e1676101.html>

## FOSL

Refer to Service Life

## -F-

### Field type

reference introduced with eZ publish 5.x.

**FiledTypes** were known as **DataTypes** in eZ Publish 4.x, and are still used in "Dual kernel" eZ Publish 5.x.

A FieldType is the smallest possible entity of storage. It determines how a specific type of information should be validated, stored, retrieved, formatted and so on.

eZ Publish comes with a collection of fundamental datatypes that can be used to build powerful and complex content structures.

In addition, it is possible to extend the system by creating custom FieldTypes for special needs.

Custom FieldTypes have to be programmed in PHP.

## **eZ Publish Summer Camp 2013 - Building custom field types**

<http://www.slideshare.net/emodric/ez-publish-summer-camp-2013-building-custom-field-types>

## **eZ Publish 5 Field Type Tutorial**

<https://doc.ez.no/display/EZP/eZ+Publish+5+Field+Type+Tutorial>

## **Field Type API and best practices**

<https://doc.ez.no/display/EZP/Field+Type+API+and+best+practices>

## **Field Types reference**

<https://doc.ez.no/display/EZP/FieldTypes+reference>

## **The eZ Publish 5 FieldType tutorial is available**

<http://share.ez.no/blogs/core-development-team/the-ez-publish-5-fieldtype-tutorial-is-available>

## **eZ Unconference#2 - Fieldtype - A. Rømcke**

<http://www.slideshare.net/ezcommunity/ez-unconference2-fieldtype-a-rmcke-22556452>

## **Fetch functions**

### **Understanding and developing fetch functions**

<http://share.ez.no/learn/ez-publish/understanding-and-developing-fetch-functions>

This tutorial will explain how template fetch functions work and how to develop custom ones.

## **FSM**

Refer to Service Life

## **-G-**

## **-H-**

## **HMVC (Hierarchical Model View Controller) stack**

## **-I-**

## **Image alias system** (revamped -This feature is available since eZ Publish 5.4 / 2014.11)

### **New in eZ 5.4: Image alias system revamped**

<http://share.ez.no/blogs/core-development-team/new-in-ez-5.4-image-alias-system-revamped>

### **Image alias generation**

<https://doc.ez.no/display/EZP/Image+alias+generation>

## **i18n**

Refer to the [translation section in Symfony documentation](#).

## -J-

## -K-

### Kernel

The new 5.x kernel is built using an **HMVC** model. It currently consists of **Public API** (Model), **Symfony2** Fullstack Framework (Hierarchical View Controller) with **Twig** (template engine) enhanced with **eZ Publish** concepts (Core Bundle). On top of these you'll find **REST** v2 API, **Demo Bundle** and a Legacy Bundle allowing some integration between the two kernels. See also:

- **Dual kernel**

It contains both a *legacy kernel* (effectively a "4.y" version) aka "Legacy Stack", and a new *5.x kernel*, aka "Symfony Stack" (because it uses [Symfony2](#) Fullstack framework).

<https://doc.ez.no/pages/viewpage.action?pageId=6291674>

- **Legacy kernel**

For eZ Publish legacy documentation (v 3.x to 5.x) refer to: <https://doc.ez.no/eZ-Publish/> or through <https://doc.ez.no/display/MAIN/eZ+Documentation+Center>

The version 5.4 of eZ Publish is the last one which relies on legacy kernel.

<http://ez.no/Blog/eZ-Publish-Platform-5.4-is-Almost-Here-What-s-Coming-Next>

## -L-

### Legacy kernel

Refer to Kernel

### Legacy mode

Legacy mode is a specific configuration mode where eZ Publish's behavior is the closest to v4.x. It might be used in some very specific use cases, such as **running the admin interface**.

<https://doc.ez.no/display/EZP/Legacy+code+and+features#Legacycodeandfeatures-LegacyMode>  
[http://share.ez.no/blogs/jerome-vieilledent/legacy\\_mode-clarified](http://share.ez.no/blogs/jerome-vieilledent/legacy_mode-clarified)

### LSM

Refer to Service Life

## **LTS**

Refer to Service Life

## **-M-**

### **Multi-sites (and Complex Web Setups)**

With one installation of the eZ Publish Platform one can manage as many websites and content repositories as they want with syndication, aggregation and content re-use policies and no technical constraints

<http://ez.no/Products/Content-Management>

## **-N-**

### **New Stack**

Refer to Stack

## **-O-**

## **-P-**

### **Platform Publish**

Refer eZ Platform Publish

### **Platform UI Bundle**

Refer to Bundles

## **-Q-**

## **-R-**

### **REST API**

Exposes all Content Management features programmatically, making it very easy to retrieve or contribute content from applications using JSON or XML standard content data. Thanks to API, it is possible to deliver content in any context with little effort.

<http://ez.no/Products/Content-Management>

## **-S-**

### **Search (Service)**

**Fetching content in eZ Publish 5 using Search service (08.07.2014, Petar Španja )**

<http://www.netgenlabs.com/Blog/Fetching-content-in-eZ-Publish-5-using-Search-service>

### **Security**

**SSO ( Single Sign On - one authentication for all their applications)**

**Using a SSO in eZ Publish**

<http://share.ez.no/learn/ez-publish/using-a-sso-in-ez-publish>

**Security updates for eZ Publish & extensions**

<http://share.ez.no/blogs/core-development-team/security-updates-for-ez-publish-extensions>

**Keeping your eZ Publish 5.x & it's Symfony install secure**

<http://share.ez.no/blogs/ez/keeping-your-ez-publish-5.x-it-s-symfony-install-secure>

**Security policy limitations**

**Adding custom security policy limitations to your modules**

<http://share.ez.no/learn/ez-publish/adding-custom-security-policy-limitations-to-your-modules>

This tutorial is targeted to anyone who needs to implement security policy limitations for custom modules.

Knowledge of eZ Publish module development is a must to fully understand the interest of this article. If you are not familiar to module and/or extension development in eZ Publish, you may read this excellent article about developing extensions.

You also might read this interesting (old) tutorial about module development for eZ Publish (warning : this article was based on eZ Publish 3.x and PHP4, so read it with PHP5 style in mind).

You can brush-up your knowledge of the basics of Access Control in eZ Publish by reviewing the online documentation on this subject.

### **Service ( Content a Service)**

The whole spectrum of content management functionality has been made available in the form of web services, which are extremely simple and straightforward to integrate into any connected application.

This capability of managing content 'as a service' through a content API and not through a user interface opens a wide range of new opportunities for application developers and integrators.

<http://ez.no/Products/Content-Management>



## Service Life

It is a service life which refers to product's lifetime, or the acceptable period of use on service.

Types of eZ product's Service Life:

- **FSM:** The first month we accept installations to start using this version.
- **LSM:** The last month we accept installations to start using this version.
- **EOSL:** The last month we provide Product Support and Software Maintenance services for a given version.
- **SLE:** Extended Software Maintenance which includes security updates for up to 2 years beyond EOSL. Contact your sales representative for more details and availability of versions not listed with a SLE here.
- **LTS:** Long term release

<https://support.ez.no/Public/Service-Life>

## SLE

Refer to Service Life

## Source code

One of the best sources of documentation is found simply reading the source code.

The source code is split into several repositories:

- [ezpublish-kernel](#)
- [ezpublish-api](#)
- [ezpublish-community](#)
- [ezpublish-spi](#)

Source code API documentation also available from [apidoc.ez.no](#)

## Stack

It refers either the legacy stack or a completely new stack introduced since version 5.x of eZ Publish.

Sites using eZ Publish 5.x base on new Web stack, **Twig** and **Symfony** (and not the legacy system).

<http://ez.no/Blog/High-performance-one-of-the-many-very-good-reason-s-to-move-to-eZ-Publish-5.x>

### eZ Publish 5 - the hybrid stack

<http://share.ez.no/blogs/ivo-lukac/ez-publish-5-the-hybrid-stack>

## -T-

## Tags

**eZ Tags** is an **extension** for eZ Publish, providing more functionality for tagging content objects than ezkeyword datatype included in eZ Publish kernel.

<https://github.com/ezsystems/eztags>

## Template

template system, and the design system.

eZ doc      4.x: <https://doc.ez.no/eZ-Publish/Technical-manual/4.x/Templates>  
5.x: <https://doc.ez.no/display/EZP/Templating>

This part covers Templating in eZ Publish 5 in a general aspect.  
If you want to learn about how to display content and build  
your content templates,

	you might want to check <a href="#">Content view</a> .
--	--

eZpedia:      <http://www.ezpedia.org/ez/templates>

<https://doc.ez.no/display/EZP/Templating>

## Twig

Twig is a templating engine, used in the templates.

eZ Publish Platform provides [Twig helpers](#) and [Twig functions](#).

You can also add your own Twig helpers and functions.

See [Twig documentation](#) and [templating basics in Symfony](#).

- For extending Twig, refer to [Twig developer documentation](#) and [Symfony cookbook](#).

## -U-

### User Interaction

The eZ Publish Platform offers many possibilities for users to interact on the Web. Common interactions include registering on a website for events or services, filling a survey, accessing a private portal or commenting on content, the eZ Publish's capabilities are limitless.

When business goes digital, eZ Publish is the front-end for your digital e-service, ensuring smooth and controlled interaction with your users.

<http://ez.no/Products/Content-Management>

## **-V-**

**Varnish Cache** - intelligent caching for better web performance

**Optimize your eZ Publish with Varnish** (eZ Summer Camp 2012 - eZ Publish and Varnish workshop)

<http://www.slideshare.net/morelseb/varnish-e-zpublish>

**Using Varnish**

<https://doc.ez.no/display/EZP/Using+Varnish>

## **Virtual host**

<https://doc.ez.no/display/EZP/Virtual+host+setup>

## **-W-**

## **-X-**

## **-Y-**

### **YUI - the javascript framework**

On August 29th Yahoo! announced that they are dropping further development of YUI, the javascript framework eZ Publish Platform have been using for several years now.

Changes to YUI and how it affects eZ Publish Platform

<http://share.ez.no/blogs/core-development-team/changes-to-yui-and-how-it-affects-ez-publish-platform>

## **-Z-**

### **Zeta Components (Since 07/2010, formerly known as eZ Components)**

The Zeta Components are used in the kernel of eZ Publish, and can be used when developing your own extensions.

You may want to have a look at this high-quality library of loosely-coupled PHP components.

Visit the Zeta Components website : <http://zetacomponents.org/>

<http://share.ez.no/get-involved/develop#ezpublish>