

# Introduction to the eZ Publish Cloud

This documentation gives you a brief introduction to our Cloud Solution for eZ Publish.

The eZ Publish Cloud Service provides you a scalable, secure and easy way to host your eZ Publish installation.

## Basic Concepts

eZ Publish Cloud is based on three basic concepts:

1. Environments
2. Stages
3. Deployments

## Environments

The first basic concept is the environment. An environment encapsulates all your data and separates it from the data of other customers. An environment is in a one to one relationship with a service subscription. Each environment can hold one productive (public accessible) eZ Publish Installation (which can of course have multiple site accesses).

The environment is also the entity on which permissions are granted. This means, that a user, who has access to a certain environment, can see and edit all settings of the whole environment including all stages and deployments.

## Stages

Each environment is separated into 3 Stages:

- Staging
- Live
- Rollback

The three stages define three areas in your environment, where your eZ Publish installations live. It is quite similar to a classical software distribution process where you have a development environment (not provided by eZ Publish Cloud), a testing environment (staging stage) and a production environment (live stage). The eZ Publish Cloud provides you another stage (Rollback). The Rollback stage keeps older installations for a quick reset, in case something goes wrong with the installation in production.

The stages can be seen as tags for deployments, which define how the deployment is treated and which actions for the Deployment are available. The stages provide no physical or virtual separation of the contained deployments. All deployments are equally treated, no matter in which stage the deployment resides.

## Deployments

The last concept is the deployment. **A deployment is basically a complete and independent eZ Publish installation**, which resides in one of the three stages of an environment. It can be transferred between the stages but it is bound to one environment and can not be transferred to an other one. An environment can hold multiple fully functional deployments (eZ Publish Installations). This implies, that you can have multiple concurrent deployments (installations of eZ Publish) in one environment. But only one of the deployments is public available for your customers at a time.

Every time you change a piece of your code (like a template or a style sheet) you have to create a new Deployment including the changes to make it available to your users. An existing Deployment can't be changed afterwards.

## How It fits together

If you create a new deployment it will always reside first in the Staging stage. There can be only one deployment in staging at a time. Deployments in Staging can be tested to ensure they work as desired before they go Live. As long as the Deployment is in Staging a cache warming can be performed. At the time you move the Deployment to live, the incoming requests are redirected to the new Deployment, which generates the responses from now on.

To create another new deployment you have to push the current staging deployment to the live stage or delete it.

The Deployment in the Live stage is the one public available eZ Publish installation in your environment. The site accesses configured in this deployment can be accessed by your customers through the defined URLs. There can be only one Deployment in the Live stage at a time. If an other deployment (from Staging or Rollback) is pushed to the Live stage, the current live deployment is pushed to the Rollback stage.

The Rollback stage holds deployments which were before in the live stage. These deployments can be pushed back to the Live stage, for example if the current Live deployment works not as expected. The Rollback stage can hold multiple deployments at the same time. Deployments in rollback are "recycled" (deleted with all its according data) after a time, defined in your service contract.

**All deployments in an environment are fully functional installations of eZ Publish** and live parallel in your environment. Each deployment can hold its own database and binary content. All of the the site accesses can be visited via a deployment specific URL. Deployments in Staging and Rollback are protected with basic HTTP authentication to prevent public access to the sites of the deployments. (This prevents also unwanted indexing by web crawlers.)

## Shared Content

In classic eZ Publish installations a one to one relationship exists between a installation, the database and the "var"-folder for binary content like pictures, videos or pdf files. In eZ Publish Cloud we have opened this relationship. Since every change in the code base leads to a new separate Deployment, we provide the possibility to reuse the current live Database and its related binary content for a new deployment. This decoupling of content and code enables a transparent upgrade of the code base while keeping the actual content of the installation. For a certain time you have two installations (one in Staging and one in Live) sharing the same content.

## Development Process

The eZ Publish Cloud is designed for hosting of eZ Publish installations in production and not for development and testing. Development and testing should happen in separate environments outside of eZ Publish Cloud.

To have a productive, fully functional eZ Publish installation running on our Cloud Service several steps are necessary.

### Develop your site locally:

1. Setup the full ez Publish installation incl. database on your local desktop or development server.
2. Apply all the customizations you need to the installation.
3. Create content or import content on your development system

If you are satisfied with your work and it matches all your requirements, upload it to your eZ Publish Cloud environment:

1. Create a dump of database in your local development environment upload the dump to the eZ Publish Cloud
2. make a tar ball of the content folder under `ezpublish/var/`, upload it to the eZ Publish Cloud
3. upload your eZ Publish installation (without the `var` folder) into the git
4. Tag the git revision to make it available in the Cloud Gui

Deploy the Installation in the eZ Publish Cloud (see the UI manual below)