

eZ Publish 5 Field Type Tutorial

About Field Types

They are the most granular building block for content managed by eZ Publish. The system comes with about 30 native types that cover most common needs (Text line, Rich text, Email, Author list, Content relation, Map location, Float...)

FieldTypes are responsible for:

- storing data, either using the native storage engine mechanisms, or specific means
- validating input data
- making the data searchable (if applicable)
- displaying an instance of the type

Custom FieldTypes are a very powerful type of extension, since they allow you to hook deep into the content model.

You can find the in-depth [documentation about FieldTypes and their best practices](#). It describes how each component of a FieldType interacts with the various layers of the system, and how to implement those.

About this tutorial

This tutorial aims at covering the conception and development of a custom eZ Publish 5 FieldType.

We will do this by implementing a Tweet Field Type. It will:

- accept as input the URL of a tweet (<https://twitter.com/<username>/status/<id>>)
- fetch the tweet using the twitter oEmbed API (<https://dev.twitter.com/docs/embedded-tweets>)
- store the tweet's embed contents and url
- display the tweet's embedded version when displaying the field from a template



Getting the code

The code created throughout this tutorial is available on github: <https://github.com/eZsystems/TweetFieldTypeBundle>.

Steps

The bundle

FieldTypes, like any other eZ Publish 5 extensions, must be provided as Symfony 2 bundles. This chapter will cover the creation and organization of this bundle

Read more about [Creating the bundle](#) and [structuring the bundle](#).

API

This part will cover the implementation of the eZ Publish API elements required to implement a custom FieldType

Read more about [implementing the API](#).

Converter

Storing data from any FieldType into the Legacy Storage Engine requires that your custom data is mapped to the data model.

Read more about [Implementing the Legacy Storage Engine Converter](#).

Templating

Displaying a FieldType's data is done through a twig template.

Read more about [implementing the FieldType template](#).