Content Repository configuration

The **content repository** is where all your content is been stored. It is the heart of eZ Platform which you interact with using Public API.

To store data, the content repository **uses a storage engine** that can virtually can work with any kind of storage (RDBMS, NoSQL, etc.). eZ Publish 5 came with a relational storage engine, compatible with v4.x: the **Legacy storage engine**, which is the default.

You can define several repositories within a single application. However, you can only use one per site.

Configuration examples

Using default values

ezplatform.yml

```
ezpublish:
    repositories:
        # Defining repository with alias "main"
        # Default storage engine is used, with default connection
        # Equals to:
        # main: { engine: legacy, connection: <defaultConnectionName> }
        main: ~

    system:
        # All members of my_siteaccess_group will use "main" repository
        # No need to set "repository", it will take the first defined repository by
default
        my_siteaccess_group:
            # ...
```

If no repository is specified for a siteaccess or siteaccess group, the first defined repository (under `ezpublish.repositories`) will be used.

All explicit

ezplatform.yml

```yaml
doctrine:
    dbal:
        default_connection: my_connection_name
        connections:
            my_connection_name:
                driver:   pdo_mysql
                host:     localhost
                port:     3306
                dbname:   my_database
                user:     my_user
                password: my_password
                charset:  UTF8

            another_connection_name:
                # ...

ezpublish:
    repositories:
        first_repository: { engine: legacy, connection: my_connection_name, config: {}
}
        second_repository: { engine: legacy, connection: another_connection_name,
config: {} }

    # ...

    system:
        my_first_siteaccess:
            repository: first_repository

            # ...

        my_second_siteaccess:
            repository: second_repository
```

Legacy storage engine

Legacy storage engine uses Doctrine DBAL (Database Abstraction Layer). Database settings are supplied by DoctrineBundle. As such, you can refer to DoctrineBundle's documentation.

> **ORM**
> Doctrine ORM is **not** provided by default. If you want to use it, you will need to add `doctrine/orm` as a dependency in your `composer.json`.

Field groups configuration

`SINCE 1.3/16.04`

Field groups, used in content and Content Type editing, can be configured from the repository section. Values entered there are field group *identifiers*:

```
repositories:
  default:
    fields_groups:
      list: [content, features, metadata]
      default: content
```

These identifiers can be given human-readable values and translated. Those values will at the moment be used when editing Content Types. The translation domain is `ezplatform_fields_groups`.
This file will define English names for field groups:

app/Resources/translations/ezplatform_fields_groups.en.yml

```
content: Content
metadata: Metadata
user_data: User data
```