

Symfony Integration

Wire the entity manager and the repository factory with Symfony DependencyInjection

Anonymous · 26/10/2026

Symfony Integration

For an application that uses `symfony/dependency-injection`, `resources/config/orm-services.yaml` defines these services:

Service	Notes
EntityManagerFactory and ConnectionParametersFactory	With their contracts, <code>EntityManagerFactoryInterface</code> and <code>ConnectionParametersFactoryInterface</code> , as aliases. An application replaces a factory by pointing the alias to its own service
<code>Doctrine\ORM\EntityManagerInterface</code>	Created by the factory of the entity manager, from the parameters <code>doctrine.dbal</code> and <code>doctrine.orm</code>
<code>DoctrineRepositoryFactory</code>	To register in the repository manager, see repositories
The provider of the translations of the package	Tagged <code>derafu_translation.resource_provider</code>

Import it from the file of services of the application and set the parameters:

```
imports:
  - { resource: '../vendor/derafu/orm/resources/config/orm-services.yaml' }

parameters:
  doctrine.dbal:
    url: '%env(DATABASE_URL)%'
  doctrine.orm:
    paths: ['%kernel.project_dir%/src/Entity']
    naming_strategy: Doctrine\ORM\Mapping\UnderscoreNamingStrategy
```

Their defaults are a SQLite file in `var/app.db` and the entities in `src/Entity`. An application can ignore the file and define its own services, with its own configuration or with Doctrine directly.

Last updated on 26/10/2026