

Wprowadzenie do R2DBC

Doświadczenia praktyczne*

Bartek „Koziołek” Kuczyński/[@koziolek](#)

Reactive Relational Data Base Connectivity

Systemy Reaktywne

Reactive Manifesto

<https://www.reactivemanifesto.org/>

Responsive
Odpowiedź jest sprawna

Resilient

Odpowiedź jest zrozumiała

Nawet jak jest opisem błędu

Elastic

Komponenty są skalowalne

Najlepiej w zautomatyzowany sposób

Message Driven

Nie ma powiązań pomiędzy komponentami

Reactive system

!=

Reactive Programming

Reactive Data Stream

Back Pressure

R2DBC

Okrojone JDBC
ale na sterydach

JDBC + Reactive Wrapper?

Można, ale nie do końca...

Można, ale nie do końca...

- Obsługa błędów

Można, ale nie do końca...

- Obsługa błędów
- Skalowalność

Można, ale nie do końca...

- Obsługa błędów
- Skalowalność
- Back pressure

Połączenie

```
ConnectionFactoryOptions options =
ConnectionFactoryOptions.builder()
    .option(ConnectionFactoryOptions.HOST, "localhost")
    .option(ConnectionFactoryOptions.PORT, 15432)
    .option(ConnectionFactoryOptions.DATABASE, "postgres")
    .option(ConnectionFactoryOptions.USER, "r2dbc")
    .option(ConnectionFactoryOptions.PASSWORD, "r2dbc")
    .option(ConnectionFactoryOptions.DRIVER, "postgresql")
    .build();

ConnectionFactory connectionFactory =
ConnectionFactories.get(options);
Publisher<? extends Connection> conn = connectionFactory.create();
```

Zapytanie

```
Flux.from(conn)
    .map(c -> c.createStatement(ALL_MEMBERS))
    .flatMap(Statement::execute)
    .flatMap(r -> r.map(new MemberExtractor()))
    .map(Member::getMemId)
    .blockLast();
```

Transakcije

ACID

Kto zarządza?

JDBC

```
Flux.from(conn)
    .map(
        c -> {
            c.beginTransaction();
            c.setAutoCommit(true);
            return c;
        }
    ).map(
        c -> c.createStatement(INSERT_BOOKING)
            .bind(0, 1)
            .bind(1, 1)
            .bind(2, now())
            .bind(3, 1)
    )
    .flatMap(Statement::execute)
    .flatMap(Result::getRowsUpdated)
    .blockLast();
```

Typy danych

Praktycznie jak w JDBC

Obsługa błędów

`R2dbcException`

Spring (👟)

Jest, działa

Szablony jak JDBC

Reactive WEB

Czego nie robić?

Nie używać Tomcata

Nie mieszać bezpośrednio z JPA

Nie konfigurować wątków „na pałę”

JPA

Da się

Osobne pakiety!

Osobne modele!

CQRS!

Mikroserwisy

Hexagonal Architecture

Q&A

<https://github.com/Koziolek/wjug-r2dbc>