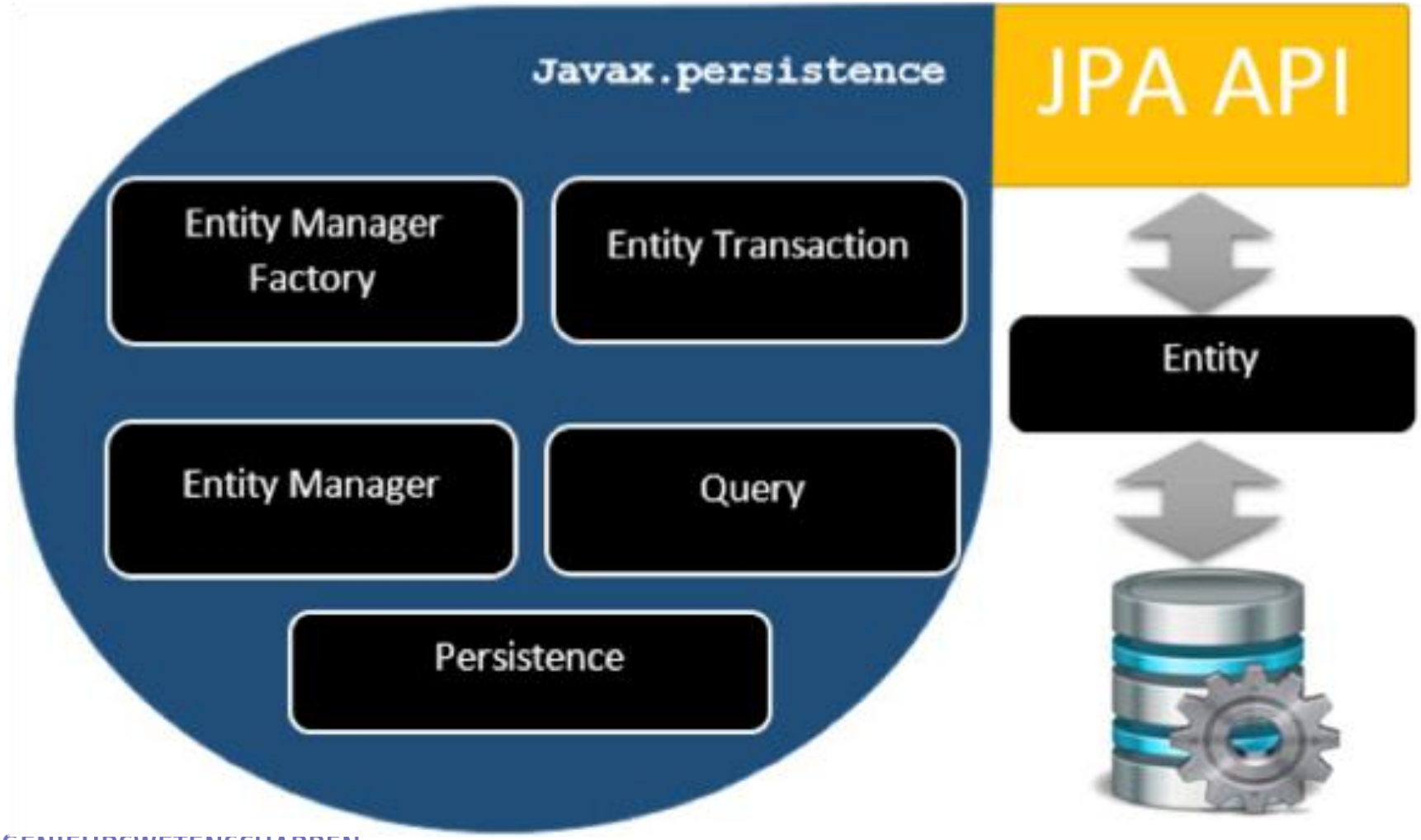
JPA in Spring

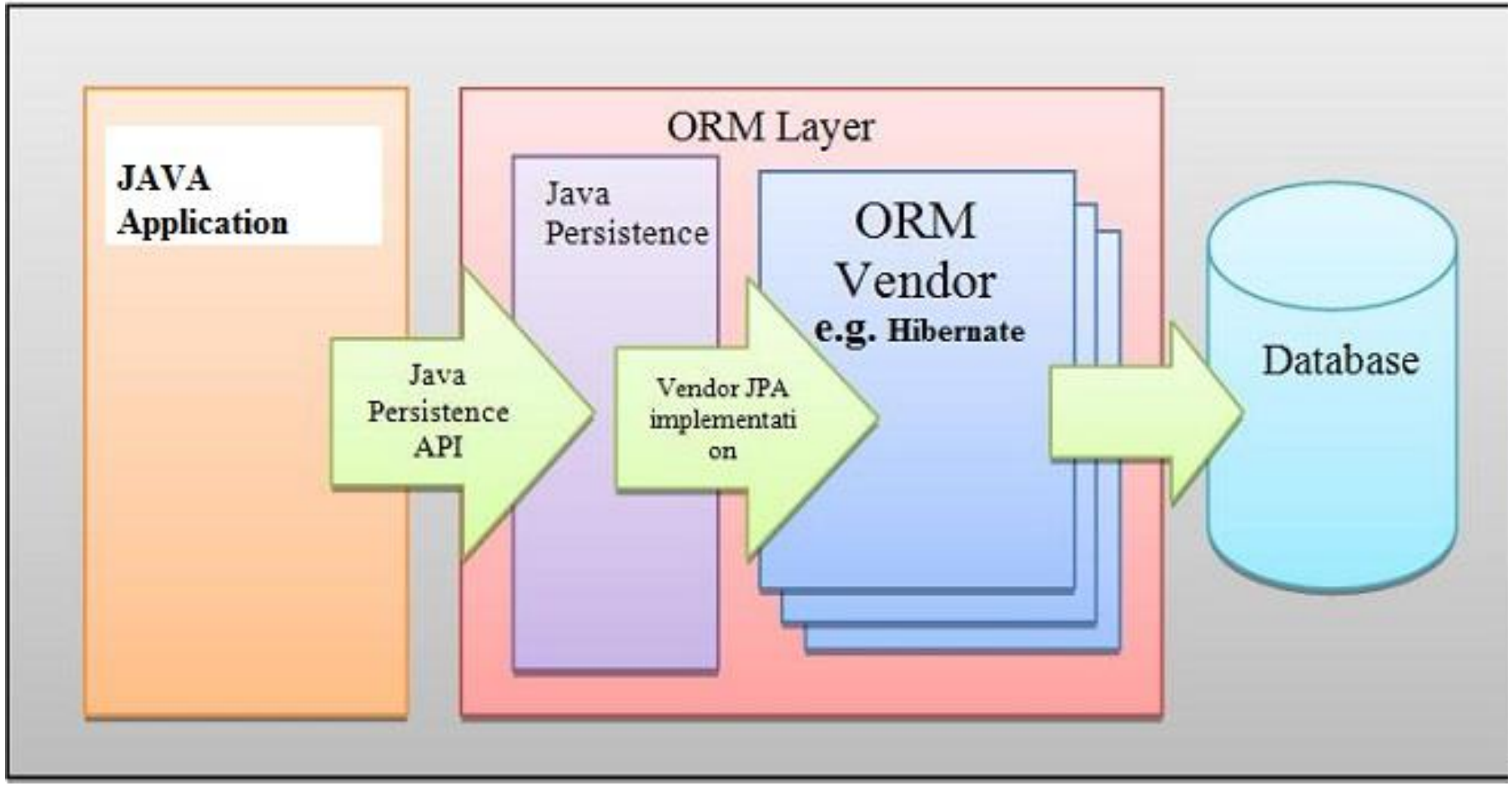
Veerle Ongenae

Configuratie



https://www.tutorialspoint.com/jpa/jpa_architecture.htm

JPA



bron: http://thecafetechno.com/tutorials/hibernate/hibernate-and-java-persistence-apijpa/

Configuratie JPA in Spring (pom.xml)

```xml
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
</dependency>
```

Entiteiten: klassen annoteren

UNIVERSITEIT GENT

- Annotaties
  - Boven getter
  - Boven attribuut
  - Consistent (beide werkwijzen niet combineren)

```
import javax.persistence.*;

@Entity
@Table(name = "sportclub")
public class Sportclub {
    private Long id;
    private String naam;

    @Id
    @GeneratedValue(strategy=GenerationType.AUTO)
    public Long getId() { return id;  }
    public void setId(Long id) { this.id = id;   }

    //@Basic
    //@Column(name = "NAAM")
    public String getNaam() { return naam;  }
    public void setNaam(String naam) { this.naam = naam; }
}
```

Repository

- CRUD-operaties voor entiteiten

Klasse entiteit Type id

```
public interface SportclubRepository extends JpaRepository<Sportclub, Long> {


}
```

- Interface → Spring genereert de implementatie
- JpaRepository (https://docs.spring.io/spring-data/jpa/docs/current/api/org/springframework/data/jpa/repository/JpaRepository.html)
  - Methodes om entiteiten op te halen, te bewaren, aan te passen, …

JpaRepository

UNIVERSITEIT GENT

- Objecten bewaren

```
SportclubRepository repository; // geïnjecteerd

…

repository.save(new Sportclub(“Mijn favoriete sportclub”);
```

- Alle objecten opvragen

```
SportclubRepository repository; // geïnjecteerd

…

repository.findAll().forEach(club -> log.info(club.getNaam());
```


JpaRepository

UNIVERSITEIT GENT

- Een object ophalen op id

```
SportclubRepository repository; // geïnjecteerd

…

Sportclub club = repository.findById(1L);

log.info(club.getNaam());
```

- Andere methodes
  - flush → aanpassingen doorsturen naar de database
  - saveAll → meerdere entiteiten bewaren of aanpassen
  - count → aantal entiteiten
  - delete, deleteAll → één of meerdere entiteiten verwijderen
  - existById → bestaat er een entiteit met de gegeven id?

Extra zoekmethodes

- Repository – interface

```
public interface CustomerRepository extends JpaRepository<Customer, Long> {
    List<Customer> findByLastName(String lastName);
}
```

Eigenschap customer

- Definieert welke zoekmethodes er beschikbaar zijn

Configuratie databank

- application.properties
  - Naam/waarde-paren


```
spring.datasource.url=jdbc:mysql://localhost:3306/iiidb?serverTimezone=UTC
spring.datasource.username=iii
spring.datasource.password=iiipwd
spring.jpa.properties.javax.persistence.schema-generation.database.action=create
spring.jpa.properties.javax.persistence.schema-generation.scripts.action=create
spring.jpa.properties.javax.persistence.schema-generation.scripts.create-target=create.sql
spring.jpa.properties.javax.persistence.schema-generation.scripts.create-source=metadata
```

Locatie databank

Login en wachtwoord

Databankstructuur genereren

Scripts genereren

Databankstructuur genereren

UNIVERSITEIT GENT

- Standaard create

Niet nodig

```
spring.jpa.properties.javax.persistence.schema-generation.database.action=create
```

- Alternatieven

Instelling	Betekenis
none	Niets aanmaken of verwijderen
create	De eerste keer databankstructuur aanmaken
drop-and-create	Telkens structuur verwijderen en opnieuw aanmaken
drop	Structuur verwijderen