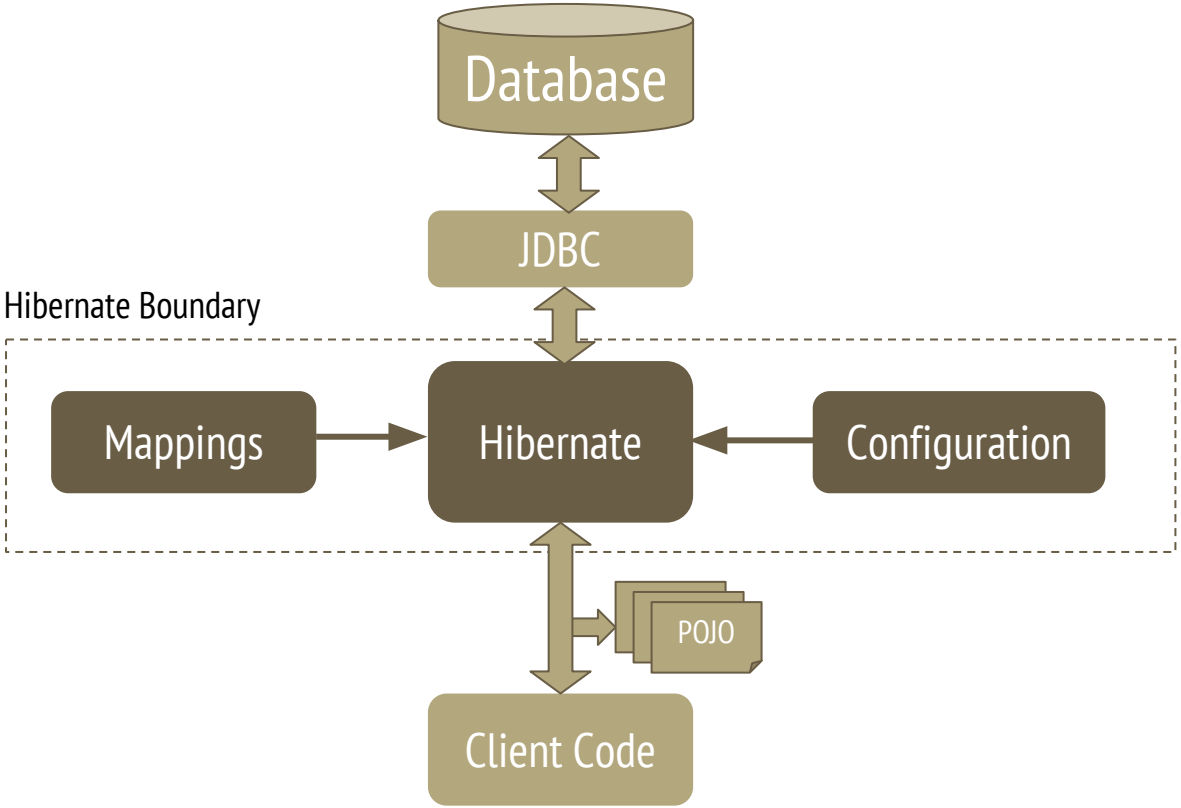
Hibernate ORM 101

Java Frameworks

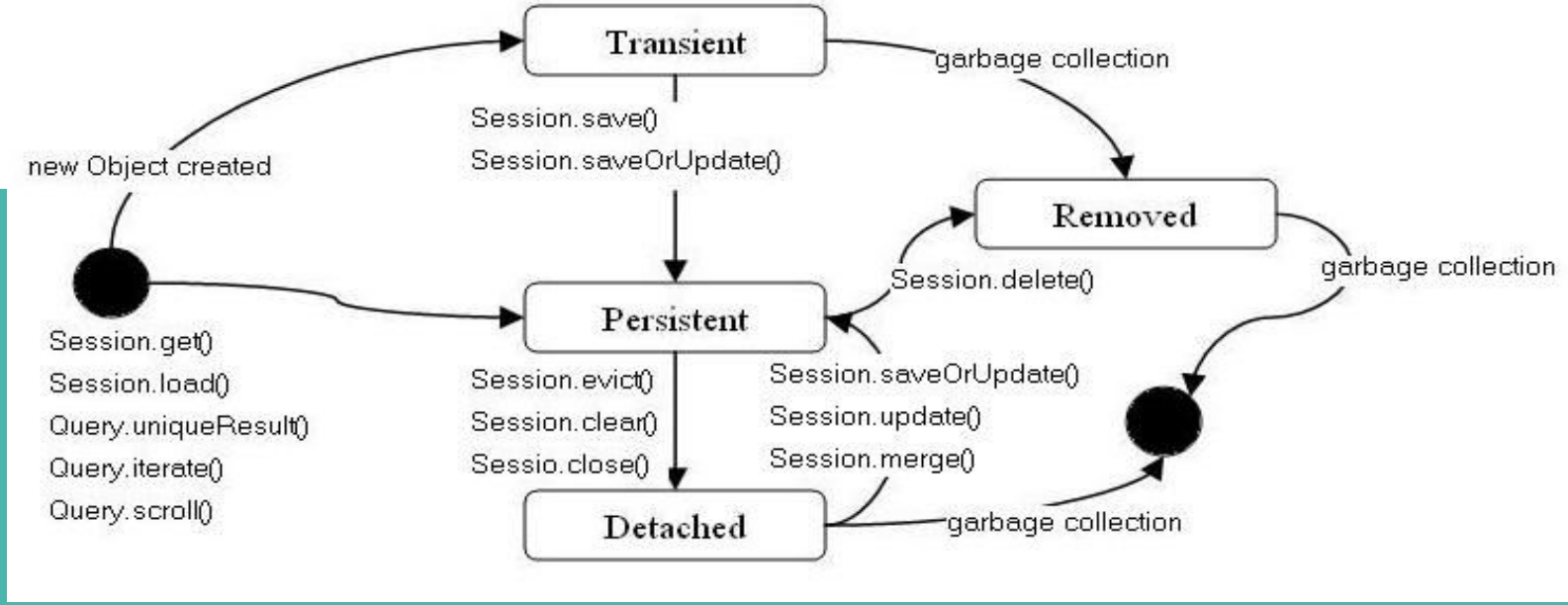
ORM - Object Relational Mapping

- Impedance Mismatch
- Maps objects to relational world
- Unifies data representation

Hibernate / Object Relational Mapping



Object Lifecycle



Basic Annotations

@Entity

@Id

@Table & @Column

@GeneratedValue

@Enumerated & @Temporal

@Transient

@Entity & @Id

```
@Entity
public class Product {

    @Id
    private Integer id;

    private String sku;

    private String name;

    private String description;
}
```

@Table & @Column

```
@Entity @Table(name="PRODUCTS")
public class Product {

    @Id private Integer id;

    @Column(name = "SKU_NBR")
    private String sku;

    @Column(name = "DESC")
    private String description;
}
```

@GeneratedValue

```
@Entity
public class DateEvent  {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;

    private String sku;

    private String name;

}
```

@Enumurated

```
@Entity
public class DateEvent  {

    @Id
    private Long id;

    @Enumerated(EnumType.STRING)
    private EventType type;

    ...
}
```

@Temporal

```
@Entity
public class DateEvent  {

    @Id
    private Long id;

    @Temporal(TemporalType.DATE)
    private Date timestamp;

    ...
}
```

@Transient

```
@Entity
public class Person  {

    @Id private Long id;

    private LocalDate dateOfBirth;
    @Transient private int age;

                                        public          int          age()          {
        age = … // calculate age
                                                                  return        age;
    }
}
```

Relationship Annotations

@ManyToOne

@OneToMany

@OneToOne

@ManyToMany

@OneToMany

```java
@Entity(name = "Person")
public static class Person {

    @Id private Long id;

    @OneToMany(cascade = CascadeType.ALL,
               orphanRemoval = true)
    private List<Phone> phones = new ArrayList<>();

    public List<Phone> getPhones() {
        return phones;
    }
    ...
}
```

```java
@Entity(name = "Phone")
public static class Phone {

    @Id
    private Long id;

    private String number;

    ...
 }
```

@ManyToOne

```java
@Entity(name = "Person")
public static class Person {

    @Id
    @GeneratedValue
    private Long id;

    public Person() {
    }
}
```

```java
@Entity(name = "Phone")
public static class Phone {

    @Id private Long id;

    @ManyToOne
    @JoinColumn(name = "person_id",
      foreignKey = @ForeignKey(name = "PERSON_ID_FK"))
    private Person person;

    public Phone() {
    }
    ...
}
```

Advance Annotations

@EntityListner

@PostLoad

@PreUpdate & @PrePersist

Advance Annotations

```java
@Entity
@EntityListeners( LastUpdateListener.class )
public static class Person {
    @Id private Long id;
    private Date dateOfBirth;
    @Transient private long age;

    @PostLoad
    public void calculateAge() {
        age = ...
    }
}
```

```java
public static class LastUpdateListener {
    @PreUpdate
    @PrePersist
    public void setLastUpdate( Person p ) {
        p.setLastUpdate( new Date() );
    }
}
```

Hibernate Queries

- EntityManager
- HQL - Hibernate Query Language
  - NamedQueries > NamedQuery

Query & TypedQuery

```
Query query = entityManager.createQuery(
    "select p " +
    "from Person p " +
    "where p.name like :name"
);

TypedQuery<Person> typedQuery = entityManager.createQuery(
    "select p " +
    "from Person p " +
    "where p.name like :name", Person.class
);
```

@NamedQueries & @NamedQuery

```
@NamedQueries(
    @NamedQuery(
        name = "get_person_by_name",
        query = "select p from Person p where name = :name"
    )
)

Query query = entityManager.createNamedQuery( "get_person_by_name" );

TypedQuery<Person> typedQuery = entityManager.createNamedQuery(
    "get_person_by_name", Person.class
);
```

Hibernate Fetch Modes

- Join
- Select
- Eager

http://www.mkyong.com/hibernate/hibernate-fetching-strategies-examples/

Future Reading

- Book: Beginning Hibernate, 3rd Edition
- Java Brains: Introduction to Hibernate
  - http://javabrains.koushik.org/courses/hibernate_intro
- http://www.mkyong.com/hibernate/hibernate-fetching-strategies-examples/
- http://www.javatpoint.com/hibernate-tutorial
- http://www.mkyong.com/tutorials/hibernate-tutorials/
- http://www.infoq.com/presentations/orm-hibernate-demo

Thanks!

Like to reach us,

Sunit Parekh
sunit.mangilal.parekh@citi.com
sp18336

Sarthak Makhija
sarthak.makhija@gmail.com

Object Lifecycle (simple view)