



CS544 EA  
Hibernate  
Cascades

# Cascades

- An operation cascades if it **follows references**
- By default non of the operations cascade:
  - em.persist(person) will not persist its car objects
  - em.merge(person) will not re-connect its car objects
  - em.remove(person) will not remove its car objects

```
@Entity
public class Person {
    @Id
    @GeneratedValue
    private Long id;
    private String firstname;
    private String lastname;
    @OneToMany(mappedBy="owner")
    private List<Car> cars =
        new ArrayList<>();

    ...
}
```

```
@Entity
public class Car {
    @Id
    @GeneratedValue
    private Long id;
    private short year;
    private String model;
    private String maker;
    @ManyToOne
    private Customer owner;

    ...
}
```

# CascadeType

- You can specify **which operations** cascade
  - Every association has the cascade option

```
@Entity
public class Customer {
    @Id
    @GeneratedValue
    private Long id;
    private String firstName;
    private String lastName;
    @ManyToMany(mappedBy="customers", cascade= {CascadeType.MERGE, CascadeType.PERSIST})
    private List<SalesPerson> salesPeople = new ArrayList<>();
    @OneToOne(cascade=CascadeType.ALL)
    private Address address;
```

Or as a single value

Can be specified as a list

Persisting a Customer  
now automatically also  
persists all linked SalesPerson  
and Address Objects

## CascadeTypes

ALL

DETACH

MERGE

PERSIST

REFRESH

REMOVE

# Orphan Removal

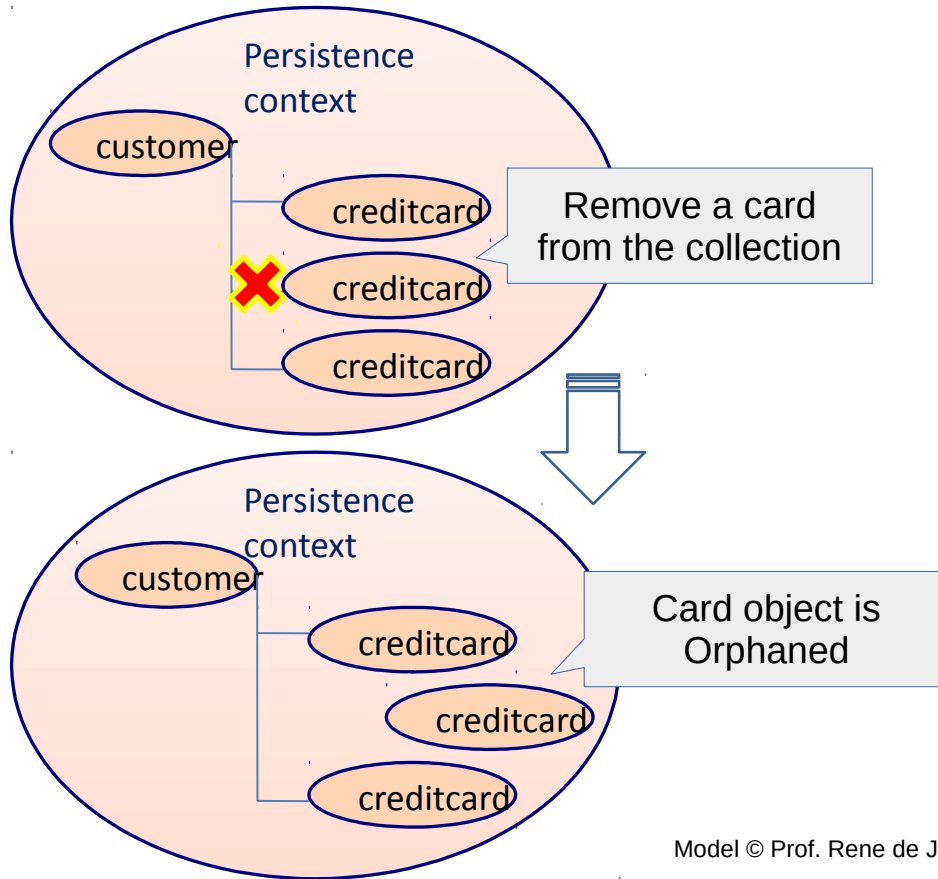
- Orphan removal is a topic related to cascades
  - Option on @OneToMany and @OneToOne
    - Both for Uni-directional and Bi-directional
  - When the connection / **reference is broken**, the entity that was referred to is **automatically removed**

```
@Entity
public class Customer {
    @Id
    @GeneratedValue
    private Long id;
    private String firstname;
    private String lastname;
    @OneToMany(mappedBy="owner", orphanRemoval=true)
    private List<CreditCard> cards =
        new ArrayList<>();

    ...
}
```

# Orphan Removal

One to Many without Orphan Removal



One to Many using Orphan Removal

