

CS544 EA Hibernate

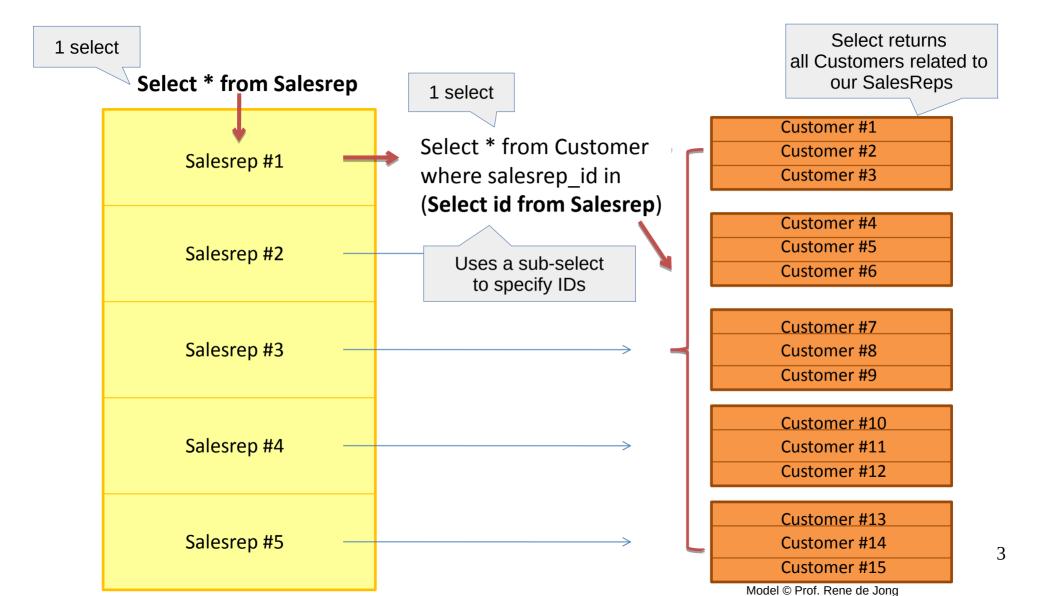
Optimization: FetchMode.SUBSELECT

FetchMode.SUBSELECT

- Hibernate Extension commonly used for N+1
 - Turns N + 1 into 1 + 1
 - Can load too much data

```
import org.hibernate.annotations.Fetch;
import org.hibernate.annotations.FetchMode;

@Entity
public class SalesRep {
    @Id
    @GeneratedValue
    private Long id;
    private String name;
    @OneToMany(mappedBy = "salesRep", cascade=CascadeType.ALL)
    @Fetch(FetchMode.SUBSELECT)
    private List<Customer> customers = new ArrayList<>();
```



Code

```
em.getTransaction().begin();

SalesRep sr1 = new SalesRep("John Willis");
SalesRep sr2 = new SalesRep("Mary Long");

sr1.addCustomer(new Customer("Frank", "Brown"));
sr1.addCustomer(new Customer("Jane", "Terrien"));
sr2.addCustomer(new Customer("John", "Doe"));
sr2.addCustomer(new Customer("Carol", "Reno"));
em.persist(sr1);
em.persist(sr2);
em.getTransaction().commit();
```

```
Before query
Hibernate:
    select
        salesrep0 .id as id1 3 ,
        salesrep0 name as name2 3
    from
        SalesRep salesrep0
                                 SubSelect Ouerv
    where
                                  executes when
        salesrep0 .id<1000
After query
                               first customer needed
Hibernate:
    select
        customers0 .salesRep id as salesRep4 1 1 ,
        customers0 .id as id\overline{1} 1 1 ,
        customers0 .id as id1 1 0 ,
        customers0 .firstName as firstNam2 1 0 ,
        customers0 .lastName as lastName3 1 0 ,
        customers0 .salesRep id as salesRep4 1 0
    from
        Customer customers0
    where
        customers0 .salesRep id in (
            select
                                        Includes
                salesrep0 .id
                                         where
            from
                SalesRep salesrep0
            where
                salesrep0 .id<1000
```

After loop

SubSelect and N+1

- SUBSELECT is like a supercharged BatchSize
 - Works on same principle but goes all the way
 - Turns N + 1 into 1 + 1

- Potential problem:
 - Static, always on, no way to not use it
 - May load everything even if we only needed 1