

CS544 EA Hibernate

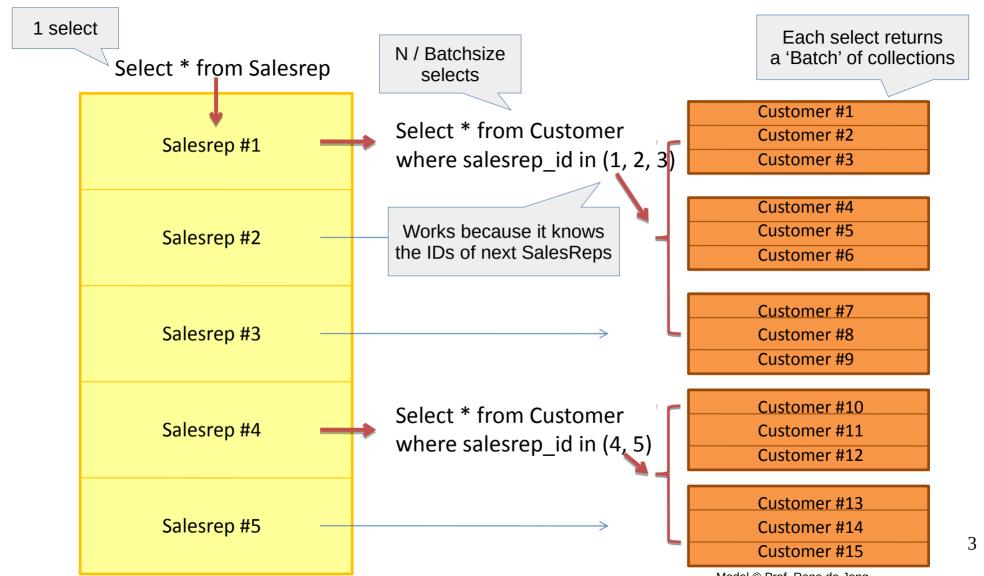
Optimization: BatchSize

@BatchSize

- Hibernate extension commonly used for N+1
 - Helps by loading several collections in one 'batch'
 - Gets another batch when previous batch is empty
 - Turns N + 1 into: ceil(N / Batchsize) + 1

```
import org.hibernate.annotations.BatchSize;

@Entity
public class SalesRep {
    @Id
    @GeneratedValue
    private Long id;
    private String name;
    @OneToMany(mappedBy = "salesRep", cascade=CascadeType.ALL)
    @BatchSize(size=3)
    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
                               Batch loaded when
        SalesRep salesrep0
After query
                              customer first needed
Hibernate:
    select
        customers0 .salesRep id as salesRep4 1 1 ,
        customers 0 .id as id 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 (
            ?. ?
After loop
```

BatchSize and N+1

- @BatchSize does not completely eliminate N+1
 - It does significantly reduce it
 - N+ 1 becomes: [N / batchsize] + 1

- Potential Problems:
 - N+1 effects not completely removed (just reduced)
 - Static, always on, no way to not use it
 - May load (batchsize 1) objects not needed