**[SoleWing](http://www.solewing.org/blog/" \o "SoleWing)**

**Musings of an enterprise software architect**

[Skip to content](http://www.solewing.org/blog/2015/07/hibernate-envers-onetomany-list-and-ordercolumn/#content)

**Hibernate Envers, @OneToMany List, and @OrderColumn**

[1 Comment](http://www.solewing.org/blog/2015/07/hibernate-envers-onetomany-list-and-ordercolumn/#disqus_thread)

Using [Hibernate Envers](http://hibernate.org/orm/envers/) with a @OneToMany association for a List property that uses @OrderColumn requires a little extra glue. Without this extra glue, Envers will throw a NullPointerException in a lazy initializer when you try to load revisions.

Suppose our domain model has a Sponsor entity. A Sponsor can have multiple business names, represented by a Sponsor Name entity. The names are ordered by preference, so we use a @OneToMany with a List. Here’s how we might map the Sponsor entity.

1. @Entity
2. @Audited
3. @Access(AccessType.FIELD)
4. public class SponsorEntity {
5. @OneToMany(mappedBy = "sponsor", fetch = FetchType.LAZY, orphanRemoval = true)
6. @OrderColumn(name = "name\_index")
7. private List<SponsorNameEntity> names = new ArrayList<>();
8. }

And here’s the SponsorNameEntity.

1. @Entity
2. @Audited
3. @Access(AccessType.FIELD)
4. public class SponsorNameEntity {
5. @Column(nullable = false)
6. private String name;
7. @ManyToOne(optional = false, fetch = FetchType.LAZY)
8. private SponsorEntity sponsor;
9. }

If we use these entities with Hibernate Envers, we’ll get an NPE when we try to load revisions of the SponsorEntity. It happens because the name\_index column specified by the @OrderColumn annotation is not included in the audit table for SponsorNameEntity.

We can fix this easily, with an additional field on SponsorNameEntity and an extra annotation on SponsorEntity. The extra field on SponsorNameEntity is used to expose the value of the order column as a field:

1. @Entity
2. @Audited
3. @Access(AccessType.FIELD)
4. public class SponsorNameEntity {
5. // This field is used to capture the value of the column named
6. // in the @OrderColumn annotation on the referencing entity.
7. @Column(insertable = false, updatable = false)
8. private int name\_index;
9. @Column(nullable = false)
10. private String name;
11. @ManyToOne(optional = false, fetch = FetchType.LAZY)
12. private SponsorEntity sponsor;
13. }

In SponsorEntity we use an additional annotation to inform Envers about the field that contains the order column value:

1. @Entity
2. @Audited
3. @Access(AccessType.FIELD)
4. public class SponsorEntity {
5. @OneToMany(mappedBy = "sponsor", fetch = FetchType.LAZY, orphanRemoval = true)
6. @OrderColumn(name = "name\_index")
7. @AuditMappedBy(mappedBy = "sponsor", positionMappedBy = "name\_index")
8. private List<SponsorNameEntity> names = new ArrayList<>();
9. }

The positionMappedBy attribute of the @AuditMappedBy annotation informs Envers that the position of each entry in the list is given by the value of the name\_index field that we added to SponsorNameEntity. It seems a little redundant, but we’re required to also specify the value of the mappedBy attribute, which should be the same as the value given in the @OneToMany annotation.

If you discover that you need this fix after you’ve already got some revisions out there in your audit tables, don’t forget to

1. Add the column specified by @OrderColumn to the appropriate audit table. In our example, this column goes into the audit table for SponsorNameEntity.
2. Initialize the column value in each of the existing rows. You can usually query the table associated with the entity to get the appropriate index value — e.g. in our case we’d query the table for SponsorNameEntity to get the name\_index column value and use it to update the corresponding rows in the audit table.

This entry was posted in [JavaEE](http://www.solewing.org/blog/category/javaee/) and tagged [envers](http://www.solewing.org/blog/tag/envers/), [hibernate](http://www.solewing.org/blog/tag/hibernate/), [javaee](http://www.solewing.org/blog/tag/javaee-2/), [jpa](http://www.solewing.org/blog/tag/jpa/) on [23 July 2015](http://www.solewing.org/blog/2015/07/hibernate-envers-onetomany-list-and-ordercolumn/).