

[Home](#) [All Tutorials](#) [Java Core](#) [JSF](#) [Spring](#) [Hibernate](#) [Struts](#) [Android](#) [Others](#) 

# Different Between Cascade And Inverse



Posted on February 5, 2010 , Last modified : August 30, 2012

By [mkyong](#)

Many Hibernate developers are confuse about the cascade option and inverse keyword. In some ways..they really look quite similar at the beginning, both are related with relationship.

## Cascade vs inverse

However, there is no relationship between cascade and inverse, both are totally different notions.

### 1. inverse

This is used to decide which side is the relationship owner to manage the relationship (insert or update of the foreign key column).

#### Example

In this example, the relationship owner is belong to stockDailyRecords (inverse=true).

```
<!-- Stock.hbm.xml -->
<hibernate-mapping>
  <class name="com.mkyong.common.Stock" table="stock" ...>
    ...
    <set name="stockDailyRecords" table="stock_daily_record" inverse="true">
      <key>
        <column name="STOCK_ID" not-null="true" />
      </key>
      <one-to-many class="com.mkyong.common.StockDailyRecord" />
    </set>
    ...
  </class>
</hibernate-mapping>
```

When you save or update the stock object

```
session.save(stock);
session.update(stock);
```

Hibernate will only insert or update the STOCK table, no update on the foreign key column. [More detail example here...](#)

### 2. cascade

In cascade, after one operation (save, update and delete) is done, it decide whether it need to call other operations (save, update and delete) on another entities which has relationship with each other.

#### Example

In this example, the cascade="save-update" is declare on stockDailyRecords.

```
<!-- Stock.hbm.xml -->
<hibernate-mapping>
  <class name="com.mkyong.common.Stock" table="stock" ...>
    ...
    <set name="stockDailyRecords" table="stock_daily_record"
      cascade="save-update" inverse="true">
      <key>
        <column name="STOCK_ID" not-null="true" />
      </key>
      <one-to-many class="com.mkyong.common.StockDailyRecord" />
    </set>
    ...
  </class>
</hibernate-mapping>
```

#### Recent Posts

[Java - Cron job to run a jar file](#)[Gradle - How to skip unit test](#)[Gradle - Spring MVC Web Project Example](#)[How to configure hot deploy in Eclipse](#)[Gradle - Create a Jar file with dependencies](#)

#### Popular Tutorials

[Android Tutorial](#)[JSF 2.0 Tutorial](#)[Spring Tutorial](#)[Maven Tutorial](#)[Hibernate Tutorial](#)

When you save or update the stock object

```
session.save(stock);  
session.update(stock);
```

It will inserted or updated the record into STOCK table and call another insert or update statement (cascade="save-update") on StockDailyRecord. [More detail example here...](#)

## Conclusion

In short, the "inverse" is decide which side will update the foreign key, while "cascade" is decide what's the follow by operation should execute. Both are look quite similar in relationship, but it's totally two different things. Hibernate developers are worth to spend time to research on it, because misunderstand the concept or misuse it will bring serious performance or data integrity issue in your application.

## Reference

1. [inverse="true" example and explanation](#)
2. [cascade example – save, update and delete](#)

Tags : [cascade](#) [hibernate](#)



### mkyong

Founder of [Mkyong.com](#) and [HostingCompass.com](#), love Java and open source stuff. Follow him on [Twitter](#), or befriend him on [Facebook](#) or [Google Plus](#). If you like my tutorials, consider making a [donation to this charity](#), thanks.

## Related Posts

- ▶ [Cascade – JPA & Hibernate annotation common mistake](#)
- ▶ [Hibernate – Cascade example \(save, update, delete and delete-orphan\)](#)
- ▶ [Spring Security + Hibernate XML Example](#)
- ▶ [java.lang.ClassNotFoundException: org.hibernate.service.jta.platform.spi.JtaPlatform](#)
- ▶ [Hibernate component mapping example](#)

## Popular Posts

- ▶ [Top 8 Java People You Should Know](#)
- ▶ [Top 20 Java Websites](#)
- ▶ [Top 5 Free Java eBooks](#)
- ▶ [Top 10 Java Regular Expression Examples](#)
- ▶ [Top 5 Open Source Q&A Systems](#)

You might also like following tutorials :

