

Vlad Mihalcea

HOME

BLOG

STORE

TRAINING

CONSULTING

TUTORIALS

VIDEOS

TALKS

multiple JPA entities to one database table with Hibernate

Last modified: Aug 24, 2019

Follow @vlad_mihalcea

**Find
Article**Search .

Imagine having a tool that can automatically detect if you are using JPA and Hibernate properly.

[Hypersistence Optimizer](#) is that tool!

Introduction

In this article, we are going to see how we can map multiple JPA entities to one table with Hibernate. Using multiple entities can speed up both read and write operations.

This question has been a recurring theme on [StackOverflow](#), so I decided to explain the advantages

Sponsored**Training
and
Workshops**

Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use. To find out more, including how to control cookies, see here: [Our Cookie Policy](#).

Close and accept

Vlad Mihalcea

HOME

BLOG

STORE

TRAINING

CONSULTING






TUTORIALS

VIDEOS

TALKS

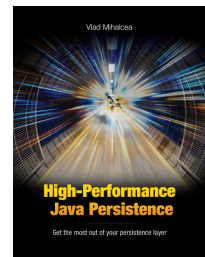
Domain Model

Let's assume we have a book database table that looks as follows:

| book | |
|---|-------------|
|  id | bigint |
|  author | varchar(50) |
|  isbn | varchar(15) |
|  title | varchar(50) |
|  properties | jsonb |

Notice that the properties column is of the jsonb type. Therefore, the properties column value is as large as the containing JSON object. For this reason, we don't want to map the book table to a single Book entity because we would have to fetch the properties

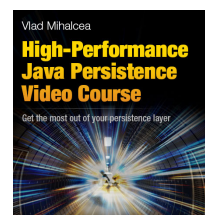
e B o o k



P a p e r b a c k B o o k



V i d e o C o u r s e



Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use. To find out more, including how to control cookies, see here: [Our Cookie Policy](#).

Close and accept

is BOOKsummary, which maps only a subset of book

Optimizer

Vlad Mihalcea

HOME

BLOG

STORE

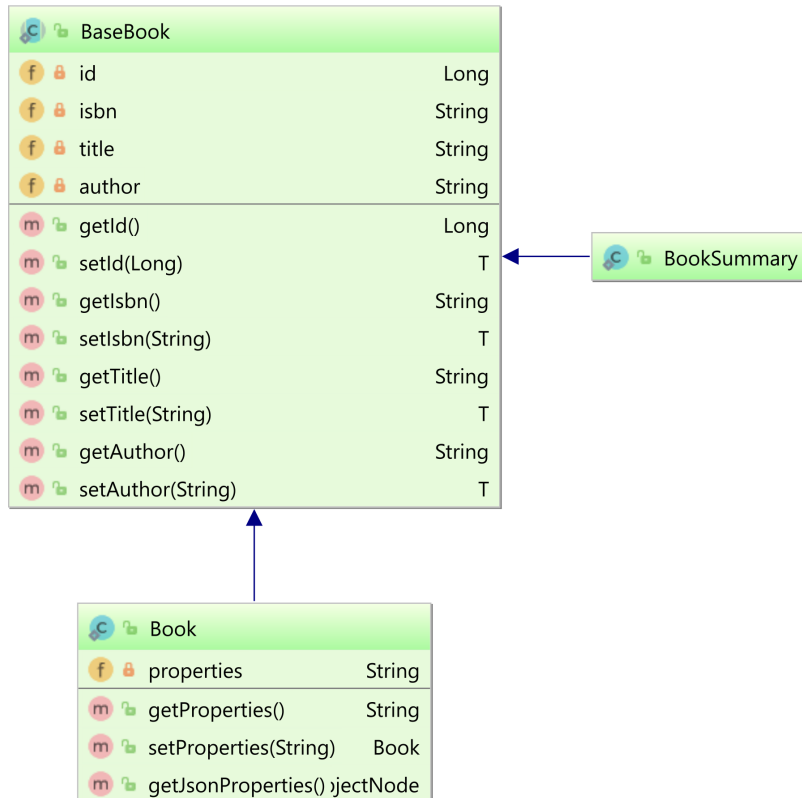
TRAINING

CONSULTING

TUTORIALS

VIDEOS

TALKS



The BaseBook is an abstract class that contains the base attributes that will be shared by all entities mapped to the book table, and it looks as follows:

```

1  @MappedSuperclass
2  public abstract class BaseBook<T> extends
3
4      @Id
5      @GeneratedValue
6      private Long id;
7
8      @NaturalId
9      @Column(length = 15)
10     private String isbn;
  
```

Consulting



Artsoft Consult

ArtSoft Consult

+160 skilled professionals

+300 completed projects

16 years of excellence

- Custom software development
- Software outsourcing
- Mobile app development
- Software customization
- Testing
- Complete software roll-out
- Data and application migration
- IT consulting services
- Business process outsourcing
- UX/UI and graphic design

GET A QUOTE FOR YOUR PROJECT

Sponsored

DATADOG

Visualize Java app performance with end-to-end detailed flame graphs

Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use. To find out more, including how to control cookies, see here: [Our Cookie Policy](#).

Close and accept

Vlad Mihalcea

HOME

BLOG

STORE

TRAINING

CONSULTING

TUTORIALS

VIDEOS

TALKS

```

28         return isbn;
29     }
30
31     public T setIsbn(String isbn) {
32         this.isbn = isbn;
33         return (T) this;
34     }
35
36     public String getTitle() {
37         return title;
38     }
39
40     public T setTitle(String title) {
41         this.title = title;
42         return (T) this;
43     }
44
45     public String getAuthor() {
46         return author;
47     }
48
49     public T setAuthor(String author) {
50         this.author = author;
51         return (T) this;
52     }
53 }

```



The abstract BaseBook is using the `@MappedSuperclass` annotation, as otherwise, base class attributes are not inherited by entities extending the BaseBook class.

For more details about the `@MappedSuperclass` JPA annotation, check out [this article](#).

Notice that the setter methods use the fluent

Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use. To find out more, including how to control cookies, see here: [Our Cookie Policy](#).

Close and accept

reference.

Vlad Mihalcea

HOME

BLOG

STORE

TRAINING

CONSULTING

TUTORIALS

VIDEOS

TALKS

superclass and adds no additional entity attribute.

```

1  @Entity(name = "BookSummary")
2  @Table(name = "book")
3  public class BookSummary extends BaseBoc
4
5  }
```

On the other hand, the Book entity extends the BaseBook superclass and maps the properties attribute.

```

1  @Entity(name = "Book")
2  @Table(name = "book")
3  @TypeDef(
4      name = "jsonb",
5      typeClass = JsonBinaryType.class
6  )
7  @DynamicUpdate
8  public class Book extends BaseBook<Book>
9
10     @Type(type = "jsonb")
11     @Column(columnDefinition = "jsonb")
12     private String properties;
13
14     public String getProperties() {
15         return properties;
16     }
17
18     public Book setProperties(String pr
19         this.properties = properties;
20         return this;
21     }
22
23     public ObjectNode getJsonProperties
24         return (ObjectNode) JacksonUtil
25             .toJsonNode(properties);
26     }
```

Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use. To find out more, including how to control cookies, see here: [Our Cookie Policy](#).

Close and accept

column types. However, you can use the

Vlad Mihalcea

HOME

BLOG

STORE

TRAINING

CONSULTING

TUTORIALS

VIDEOS

TALKS

Testing time

When persisting a Book entity:

```
1 entityManager.persist(  
2     new Book()  
3         .setIsbn("978-9730228236")  
4         .setTitle("High-Performance Java Persistence")  
5         .setAuthor("Vlad Mihalcea")  
6         .setProperties(  
7             "{" +  
8                 "\"publisher\": \"Artec\"" +  
9                 "\"price\": 44.99," +  
10                "\"publication_date\": \"2015-09-24\"" +  
11                "\"dimensions\": \"8x10x2.5\"" +  
12                "\"weight\": \"2.5 pounds\"" +  
13                "\"average_review\": \"4.5\"" +  
14                "\"url\": \"https://www.amazon.com/dp/B01005OZLN\"" +  
15                "}"  
16         )  
17 );
```

Hibernate properly sets all book table row columns:

```
1 INSERT INTO book (  
2     author,  
3     isbn, title,  
4     properties,  
5     id  
6 )  
7 VALUES (  
8     'Vlad Mihalcea',  
9     '978-9730228236',  
10    'High-Performance Java Persistence'
```

Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use.
To find out more, including how to control cookies, see here: [Our Cookie Policy](#).

Close and accept

Vlad Mihalcea

HOME

BLOG

STORE

TRAINING

CONSULTING

TUTORIALS

VIDEOS

TALKS

```
1 entityManager.persist(  
2     new BookSummary()  
3         .setIsbn("978-1934356555")  
4         .setTitle("SQL Antipatterns")  
5         .setAuthor("Bill Karwin")  
6 );
```

And Hibernate will set only the columns defined by the BookSummary entity:

```
1 INSERT INTO book (  
2     author,  
3     isbn,  
4     title,  
5     id  
6 )  
7 VALUES (  
8     'Bill Karwin',  
9     '978-1934356555',  
10    'SQL Antipatterns',  
11    2  
12 )
```

We can fetch the BookSummary for the [High-Performance Java Persistence](#) book as follows:

```
1 BookSummary bookSummary = entityManager  
2     .unwrap(Session.class)  
3     .bySimpleNaturalId(BookSummary.class)  
4     .load("978-9730228236");  
5  
6 assertEquals(  
7     "High-Performance Java Persistence",  
8     bookSummary.getTitle()  
9 );
```

Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use. To find out more, including how to control cookies, see here: [Our Cookie Policy](#).

Close and accept

Vlad Mihalcea

[HOME](#)[BLOG](#)[STORE](#)[TRAINING](#)[CONSULTING](#)[TUTORIALS](#)[VIDEOS](#)[TALKS](#)

```
4      author = vlad mihalcea ,
5      title = 'High-Performance Java Persi
6  WHERE
7      id = 1
```

Notice that the UPDATE is done only for the attributes defined by the BookSummary entity.

Now, in a subsequent Persistence Context, we can also fetch the Book entity associated with the High-Performance Java Persistence book table row.

```
1  Book book = entityManager
2      .unwrap(Session.class)
3      .bySimpleNaturalId(Book.class)
4      .load("978-9730228236");
5
6  assertEquals(
7      "High-Performance Java Persistence,
8      book.getTitle()
9  );
```

Because the Book entity maps the properties attribute as well, we can both read and write the properties attribute.

```
1  ObjectNode jsonProperties = book.getJsc
2
3  assertEquals(
4      "4.7 out of 5 stars",
5      jsonProperties.get("average_review"
6  );
7
8  jsonProperties.put(
```

Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use. To find out more, including how to control cookies, see here: [Our Cookie Policy](#).

[Close and accept](#)

Vlad Mihalcea

HOME

BLOG

STORE

TRAINING

CONSULTING

TUTORIALS

VIDEOS

TALKS

```
3  SET
4      properties = {
5          "url": "https://amzn.com/973
6          "price": 44.99,
7          "weight": "2.5 pounds",
8          "publisher": "Amazon",
9          "dimensions": "8.5 x 1.1 x 11 i
10         "average_review": "4.8 out of 5
11         "publication_date": "2016-20-12
12     }
13  WHERE
14      id = 1
```

This time, Hibernate only sets the `properties` column in the `UPDATE` statement because the `Book` entity uses the `@DynamicUpdate` annotation.

For more details about how the `@DynamicUpdate` annotation works when using JPA and Hibernate, check out [this article](#).

I'm running an [online workshop](#) on the **14th of May** about **The Best Way to Fetch Data with Java Persistence and Hibernate**.

If you enjoyed this article, I bet you are going to love my [Book](#) and [Video Courses](#) as well.

Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use. To find out more, including how to control cookies, see here: [Our Cookie Policy](#).

Close and accept

Vlad Mihalcea

[HOME](#)[BLOG](#)[STORE](#)[TRAINING](#)[CONSULTING](#)[TUTORIALS](#)[VIDEOS](#)[TALKS](#)

properties.

The only drawback of using this approach is that you have to make sure you don't fetch more than one entity type for the same database table record, as otherwise, this can cause inconsistencies when flushing the Persistence Context.

Follow [@vlad_mihalcea](#)



Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use. To find out more, including how to control cookies, see here: [Our Cookie Policy](#).

Close and accept

Vlad Mihalcea

HOME

BLOG

STORE

TRAINING

CONSULTING

TUTORIALS

VIDEOS

TALKS

How to map Oracle JSON columns using JPA and Hibernate In "Hibernate"

How to map SQL Server JSON columns using JPA and Hibernate In "Hibernate"

How to get the entity mapping to database table binding metadata from Hibernate In "Hibernate"

Category: [Hibernate](#)

 Tags: [hibernate](#), [jpa](#), [mapping](#), [multiple entities](#), [table](#)

[← How to limit the SQL query result set to Top-N rows only](#)

[Mapping PostgreSQL Interval to Java Duration with Hibernate](#)

[→](#)

7 Comments on “How to map multiple JPA entities to one database table with Hibernate”

Gothster C

September 10, 2019

Why would BaseBook extend BaseBook?

When I try this in Eclipse I get BaseBook is a raw type. References to generic type BaseBook

Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use. To find out more, including how to control cookies, see here: [Our Cookie Policy](#)

Close and accept

Vlad Mihalcea

| | | | | | | |
|-------|------|-------|----------|------------|-----------|--------|
| HOME | BLOG | STORE | TRAINING | CONSULTING | TUTORIALS | VIDEOS |
| TALKS | | | | | | |

since you can run the test with maven just fine.

As for Eclipse, I haven't used it since 2005. That's when I first discovered IntelliJIdea.

Reply

Gothster C

September 10, 2019

I got it. Thanks

Bukaj

August 26, 2019

Sure. But my point is that let's have bigger entity composed from three parts/embedables. Now we would like to work with entity in every combination of parts. I think it would be imposible with inheritance, right?

Reply

vladmihalcea

August 26, 2019

I don't understand your question

Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use. To find out more, including how to control cookies, see here: [Our Cookie Policy](#).

Close and accept

Vlad Mihalcea

- HOME
- BLOG
- STORE
- TRAINING
- CONSULTING
- TUTORIALS
- VIDEOS
- TALKS

Thanks

Reply

vladmihalcea

August 25, 2019

This is not about composition. The source of truth is the database. It's about fetching and persisting data efficiently.

Reply

Leave a Reply

Your email address will not be published.
Required fields are marked *

Comment

Name *

Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use. To find out more, including how to control cookies, see here: [Our Cookie Policy](#).

Close and accept

Vlad Mihalcea

HOME

BLOG

STORE

TRAINING

CONSULTING

TUTORIALS

VIDEOS

TALKS

.....

how your comment data is processed.

Tutorials

Social
Media

About

Meta

Hibernate

SQL

Spring

Git

FlexyPool

Twitter

Facebook

YouTube

GitHub

LinkedIn

About

FAQ

Archive

Privacy Policy

Terms of Service

Log in

Entries feed

Comments feed

WordPress.org

Vlad Mihalcea

Powered by WordPress.com.

f



in







Privacy & Cookies: This site uses cookies. By continuing to use this website, you agree to their use.
To find out more, including how to control cookies, see here: [Our Cookie Policy](#).

Close and accept