

From Spring Boot 2 To Spring Boot 3

with the Help of Apache Tomcat and More

Ivar Grimstad



Jakarta EE Developer Advocate
Eclipse Foundation



ECLIPSE
FOUNDATION



@ivar_grimstad



@ivar_grimstad@mastadon.social



<https://github.com/ivargrimstad>



<https://www.linkedin.com/in/ivargrimstad>



ivargrimstad_official



Java™
Champions



NEW RELEASE

Jakarta EE 10

Community-Driven

Modernized

Simplified

Lightweight



Final Specification

**Specification
Document**

API

TCK

**Compatible
Implementation(s)**



Jakarta EE 10 Platform

	Authentication 3.0	Persistence 3.1	
Authorization 2.1	Concurrency 3.0	Server Pages 3.1	RESTful Web Services 3.1
Activation 2.1	CDI 4.0	WebSocket 2.1	JSON Processing 2.1
Batch 2.1	Expression Language 5.0	Bean Validation 3.0	JSON Binding 3.0
Connectors 2.1	Faces 4.0	Debugging Support 2.0	Annotations 2.1
Mail 2.1	Security 3.0	Enterprise Beans Lite 4.0	Interceptors 2.1
Messaging 3.1	Servlet 6.0	Managed Beans 2.0	Dependency Injection 2.0
Enterprise Beans 4.0	Standard Tag Libraries 3.0	Transactions 2.0	CDI Lite 4.0

Updated

Not Updated

New

Jakarta EE 10 Web Profile

Authentication 3.0	Persistence 3.1	
Concurrency 3.0	Server Pages 3.1	RESTful Web Services 3.1
CDI 4.0	WebSocket 2.1	JSON Processing 2.1
Expression Language 5.0	Bean Validation 3.0	JSON Binding 3.0
Faces 4.0	Debugging Support 2.0	Annotations 2.1
Security 3.0	Enterprise Beans Lite 4.0	Interceptors 2.1
Servlet 6.0	Managed Beans 2.0	Dependency Injection 2.0
Standard Tag Libraries 3.0	Transactions 2.0	CDI Lite 4.0

Updated

Not Updated

New

Jakarta EE 10 Core Profile

RESTful Web Services 3.1

JSON Processing 2.1

JSON Binding 3.0

Annotations 2.1

Interceptors 2.1

Dependency Injection 2.0

CDI Lite 4.0

Updated

Not Updated

New



Jakarta EE 10

API source level: Java SE 11

API binary level: Java SE 11

TCK run with: Java SE 11+17

App Developers (YOU) can use Java SE 17 features if you like!!



How is this Relevant to





`javax.* -> jakarta.*`





javax.* -> jakarta.*



	Authentication 3.0	Persistence 3.1	
Authorization 2.1	Concurrency 3.0	Server Pages 3.1	RESTful Web Services 3.1
Activation 2.1	CDI 4.0	WebSocket 2.1	JSON Processing 2.1
Batch 2.1	Expression Language 5.0	Bean Validation 3.0	JSON Binding 3.0
Connectors 2.1	Faces 4.0	Debugging Support 2.0	Annotations 2.1
Mail 2.1	Security 3.0	Enterprise Beans Lite 4.0	Interceptors 2.1
Messaging 3.1	Servlet 6.0	Managed Beans 2.0	Dependency Injection 2.0
Enterprise Beans 4.0	Standard Tag Libraries 3.0	Transactions 2.0	CDI Lite 4.0

Updated

Not Updated

New



Hibernate ORM

UUID as Basic Java Type

```
@Entity
public class Item {

    @Id
    @GeneratedValue(strategy=GenerationType.UUID)
    private java.util.UUID id;
    private String description;

    ...
}
```

Jakarta EE 10 Platform

	Authentication 3.0	Persistence 3.1	
Authorization 2.1	Concurrency 3.0	Server Pages 3.1	RESTful Web Services 3.1
Activation 2.1	CDI 4.0	WebSocket 2.1	JSON Processing 2.1
Batch 2.1	Expression Language 5.0	Bean Validation 3.0	JSON Binding 3.0
Connectors 2.1	Faces 4.0	Debugging Support 2.0	Annotations 2.1
Mail 2.1	Security 3.0	Enterprise Beans Lite 4.0	Interceptors 2.1
Messaging 3.1	Servlet 6.0	Managed Beans 2.0	Dependency Injection 2.0
Enterprise Beans 4.0	Standard Tag Libraries 3.0	Transactions 2.0	CDI Lite 4.0

Updated

Not Updated

New



Hibernate Validator

Jakarta EE 10 Platform

	Authentication 3.0	Persistence 3.1	
Authorization 2.1	Concurrency 3.0	Server Pages 3.1	RESTful Web Services 3.1
Activation 2.1	CDI 4.0	WebSocket 2.1	JSON Processing 2.1
Batch 2.1	Expression Language 5.0	Bean Validation 3.0	JSON Binding 3.0
Connectors 2.1	Faces 4.0	Debugging Support 2.0	Annotations 2.1
Mail 2.1	Security 3.0	Enterprise Beans Lite 4.0	Interceptors 2.1
Messaging 3.1	Servlet 6.0	Managed Beans 2.0	Dependency Injection 2.0
Enterprise Beans 4.0	Standard Tag Libraries 3.0	Transactions 2.0	CDI Lite 4.0

Updated

Not Updated

New



Eclipse Jetty



Apache Tomcat



Demo



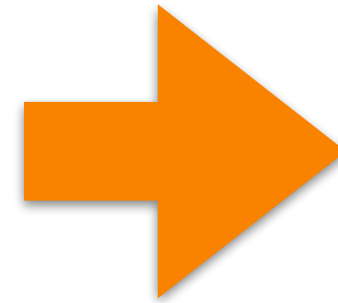
Scenario 1

New Spring App



Spring Boot 3

Tomcat 9



Spring Boot 3

+

Tomcat 10

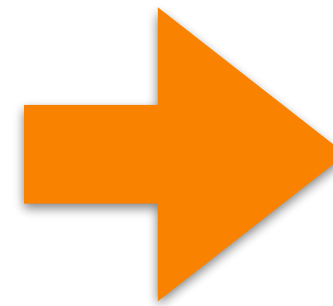


Scenario 2

Upgrade Spring App



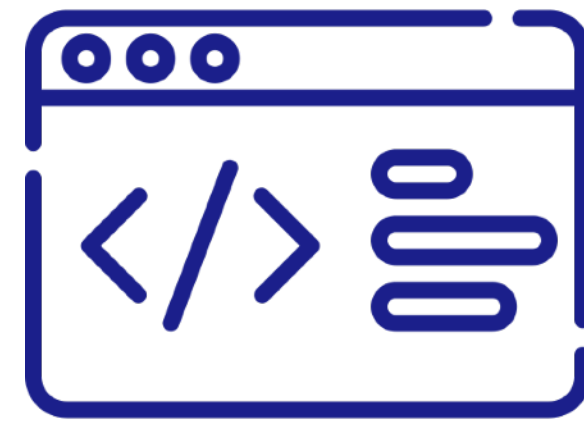
Spring Boot 2



Spring Boot 3



javax.* -> jakarta.*



IDE

Package and Class Migration

Select migration map:

Java EE to Jakarta EE ▼

Edit...

Copy...

New...

Remove

Replaces javax.* imports with jakarta.* imports in Java code.

Does not affect usages of: javax.sql, javax.swing, javax.xml, javax.naming and javax.transaction.xa.

You need to add Jakarta EE libraries before the migration.

In Project

Module

?

Cancel

Run



docs.openrewrite.org/recipes/java/migrate/jakarta



1. Update **Spring** version in pom.xml
2. Update **Jakarta EE** version in pom.xml
3. Fix the imports

<https://github.com/ivargrimstad/jakartaee-spring>





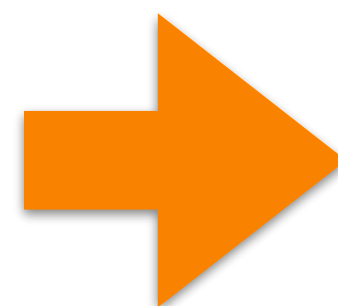
Scenario 3

Upgrade Spring App

with Dependencies



Spring Boot 2

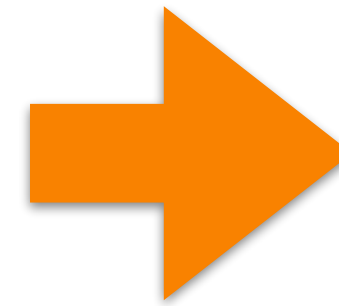


Spring Boot 3





Spring Boot 2



Spring Boot 3



1. Update **Spring** version in pom.xml
2. Update **Jakarta EE** version in pom.xml
3. Fix the imports
4. Repeat for Dependencies

<https://github.com/ivargrimstad/jakartaee-spring>

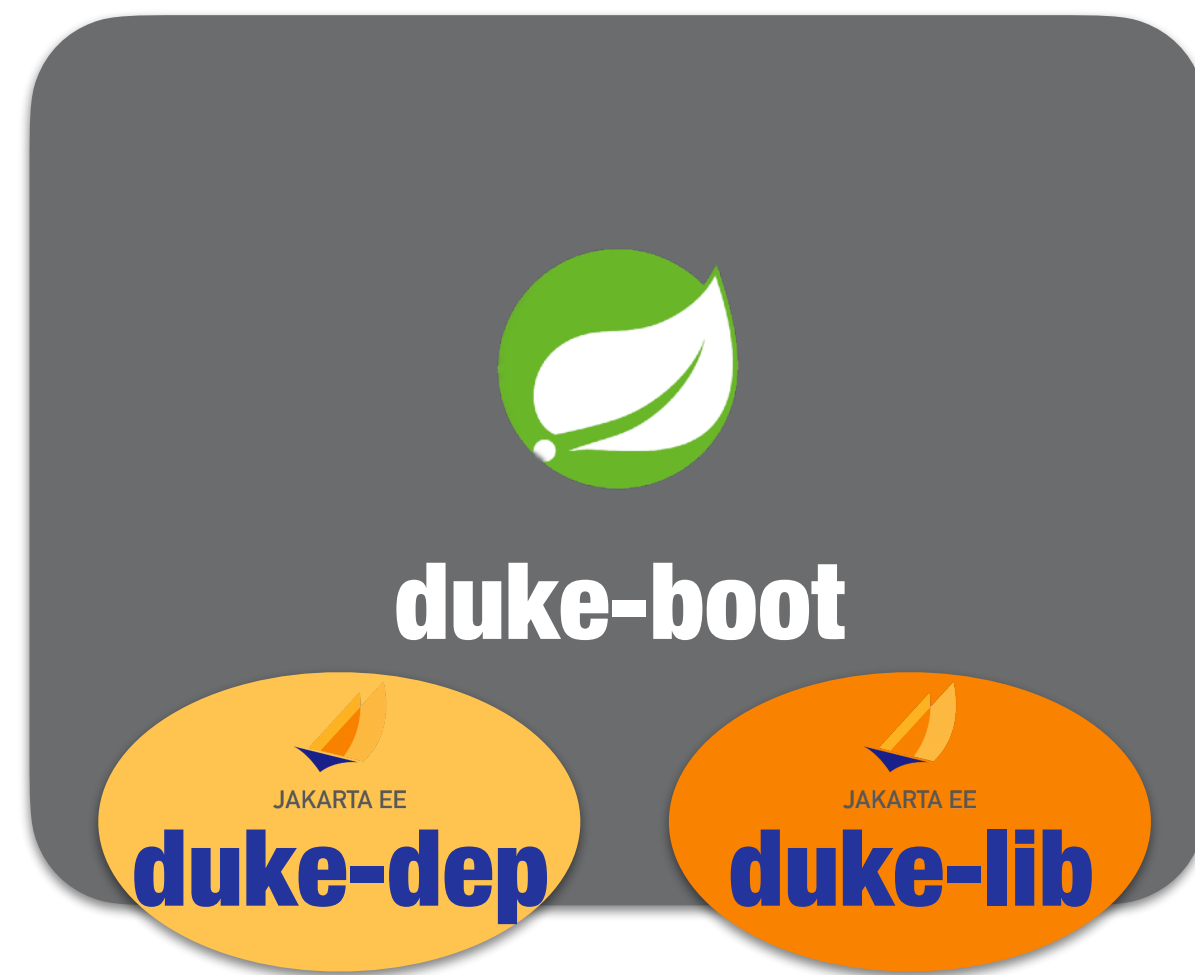




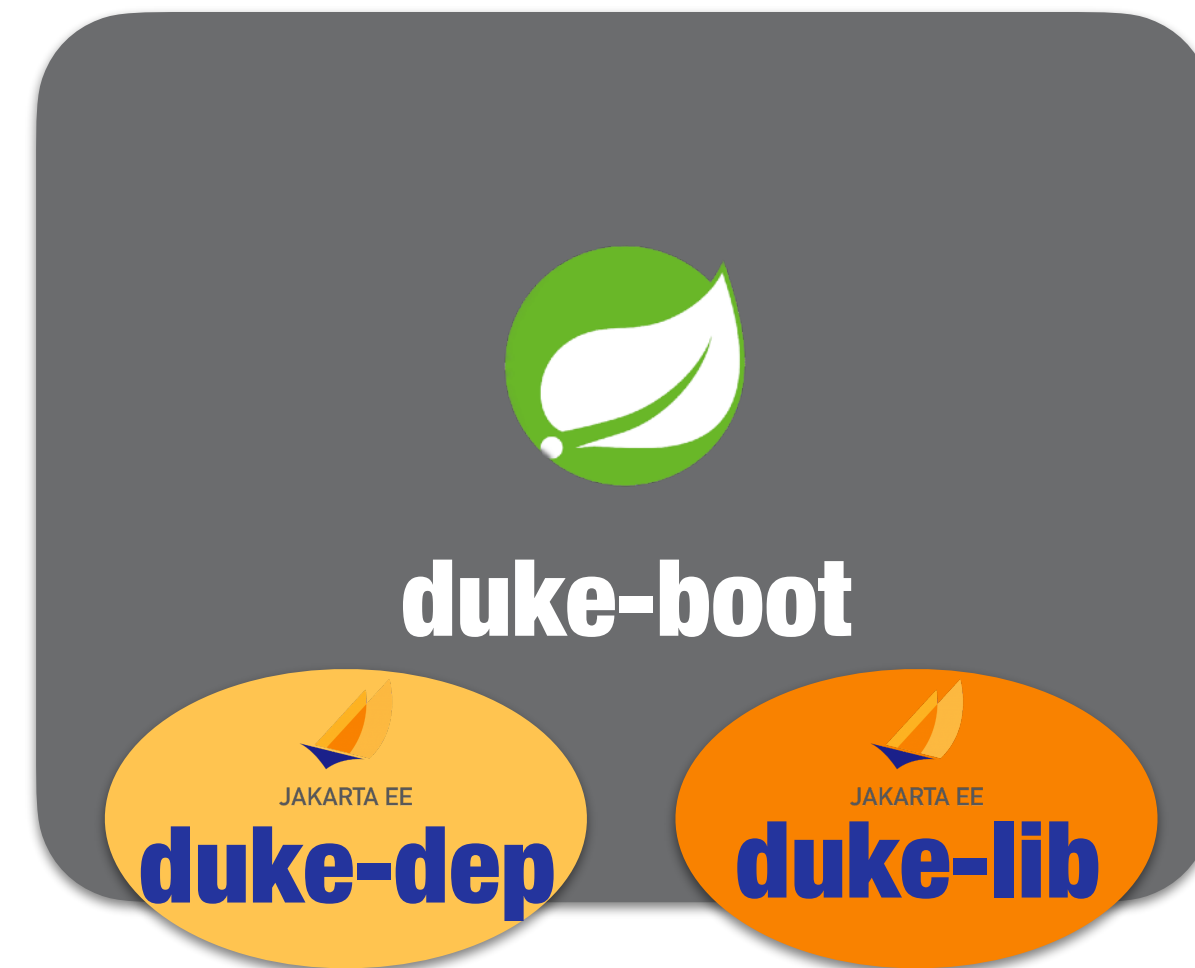
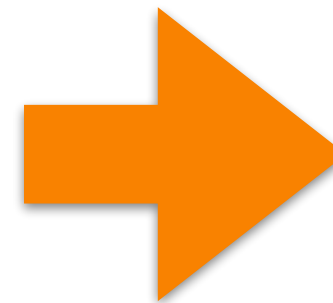
Scenario 4

Upgrade Spring App

with External Dependencies

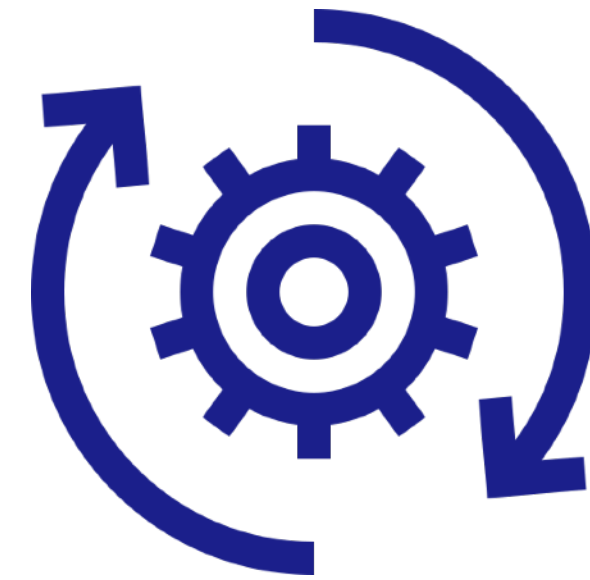


Spring Boot 2



Spring Boot 3





Transformation



Eclipse Transformer

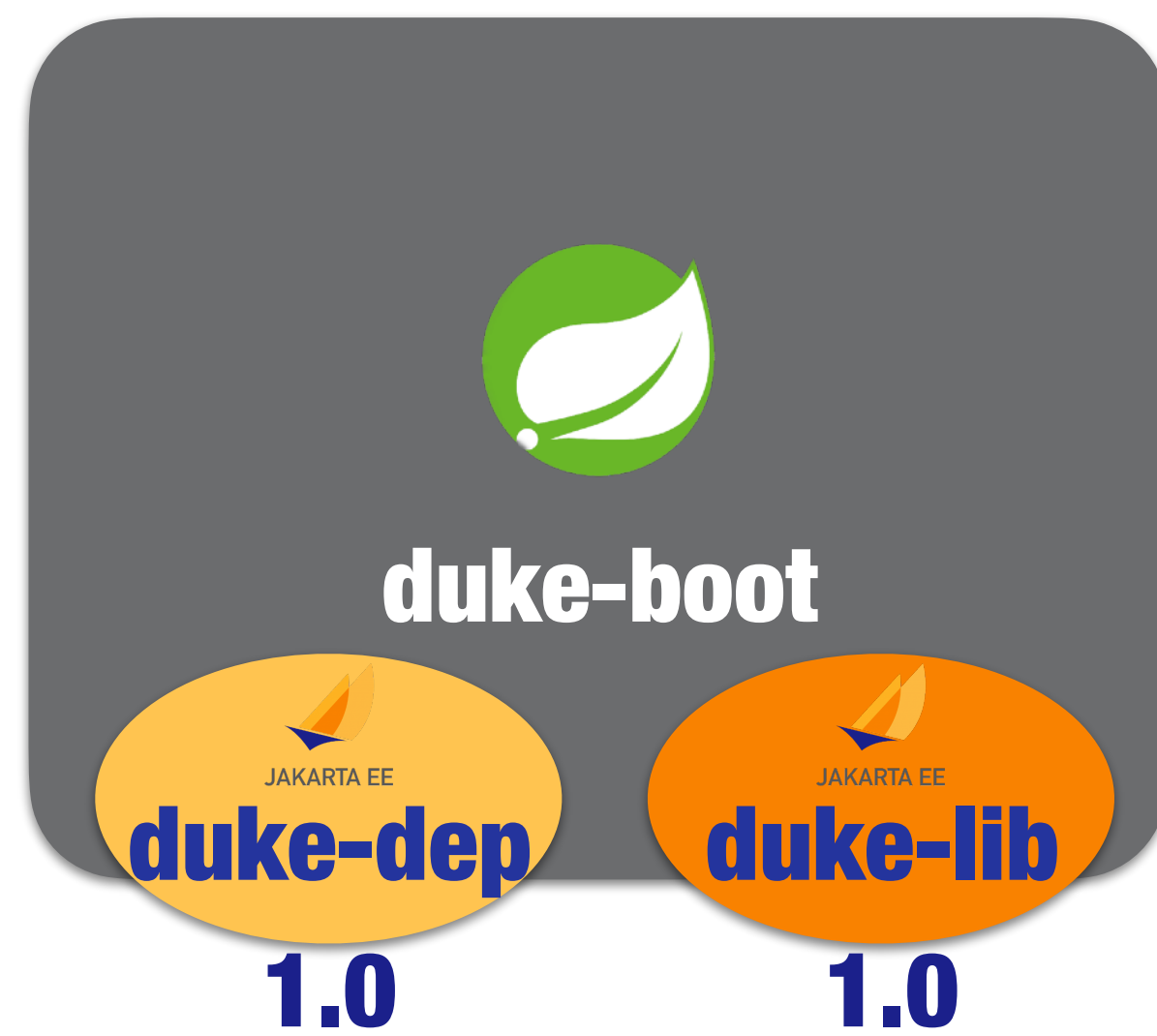


Apache Tomcat Migration Tool

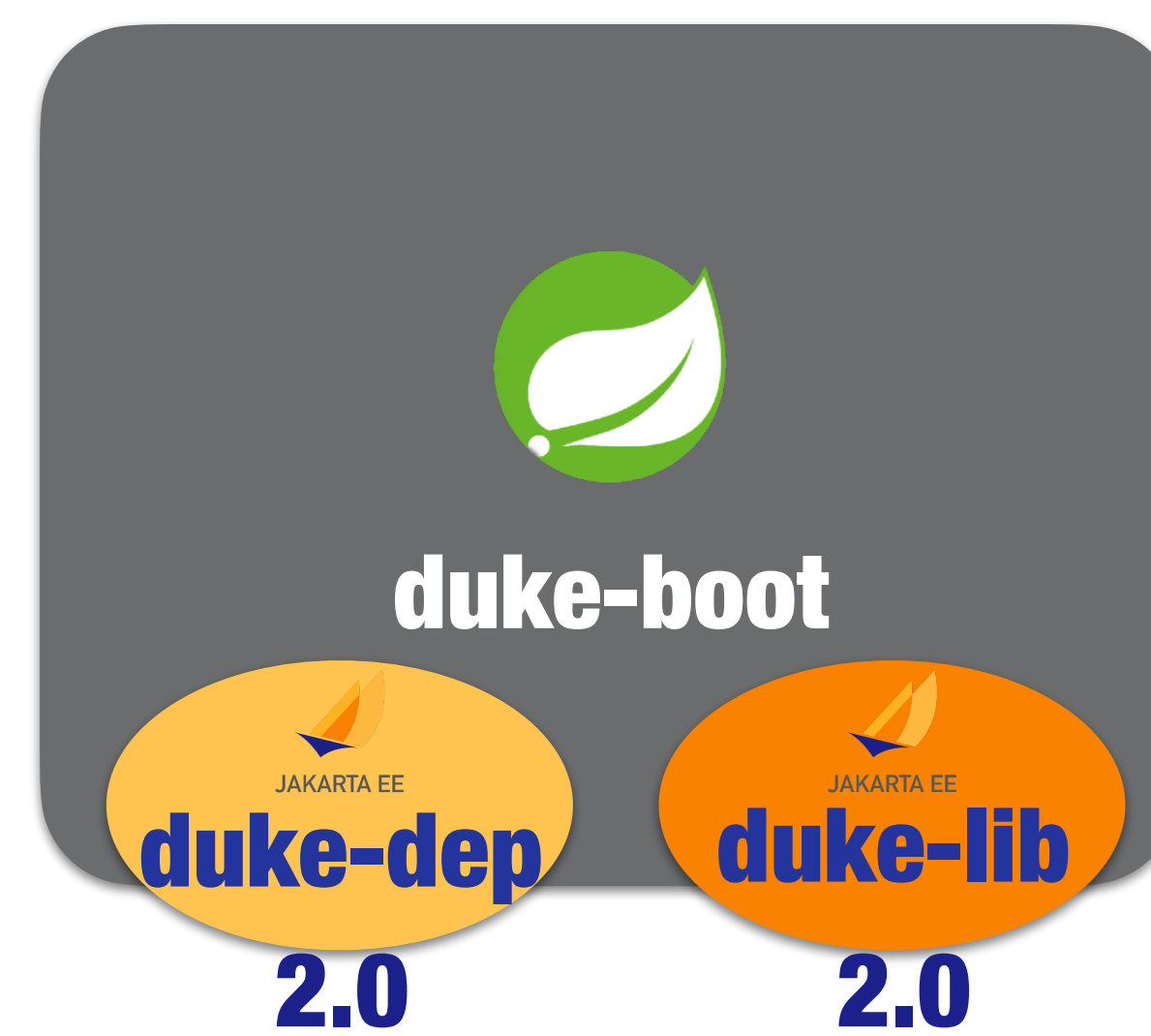
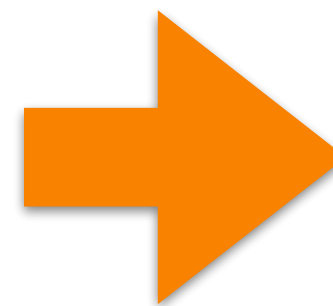


Eclipse Transformer

<https://github.com/eclipse/transformer>



Spring Boot 2



Spring Boot 3



- 1. Update **Spring** version in pom.xml**
- 2. Update **Jakarta EE** version in pom.xml**
- 3. Fix the imports**
- 4. Repeat for Dependencies**
- 5. Transform Libraries if necessary**

<https://github.com/ivargrimstad/jakartaee-spring>



Additional Migration Steps





- 1. Update **Spring** version in pom.xml**
- 2. Update **Jakarta EE** version in pom.xml**
- 3. Fix the imports**
- 4. Repeat for Dependencies**
- 5. Transform Libraries if necessary**
- 6. Rename properties prefixed with javax**
- 7. Verify data and dynamic content**

<https://github.com/ivargrimstad/jakartaee-spring>





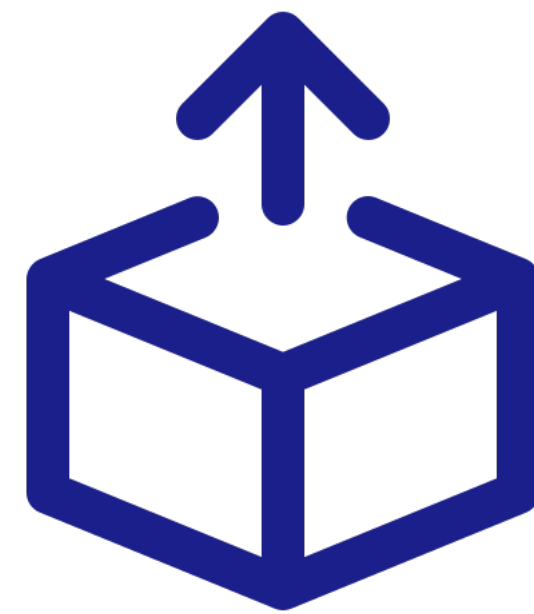
Java 21





Beyond JAKARTA EE 10



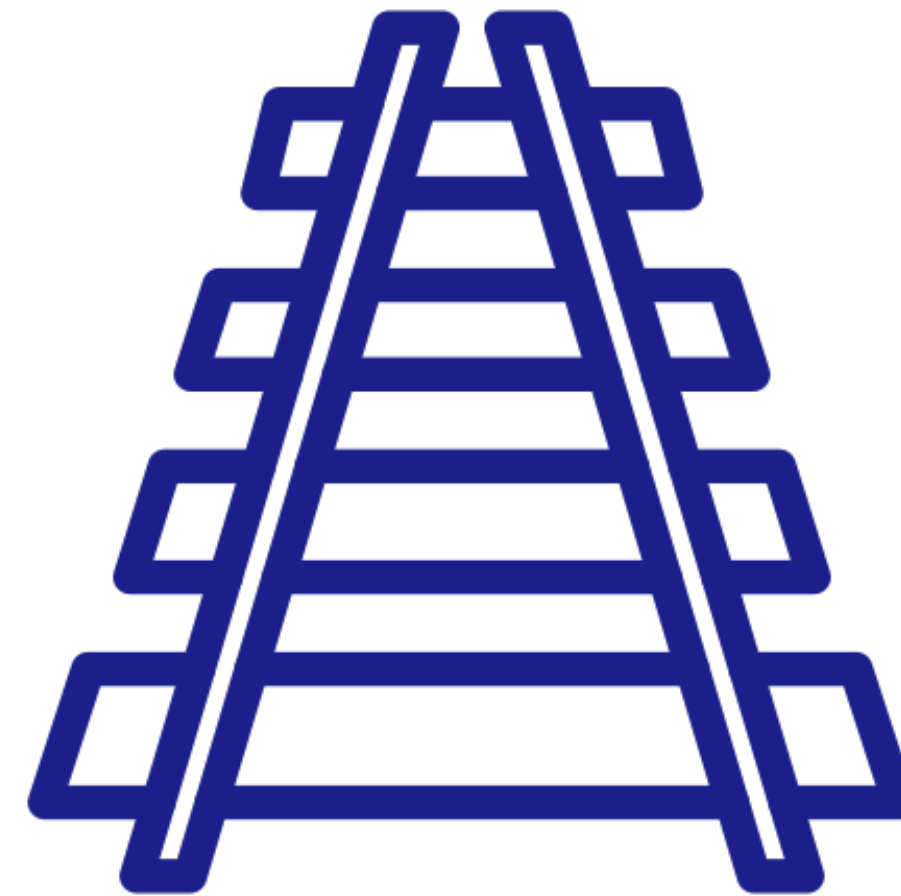


Jakarta EE 11

H1, 2024



JAKARTA EE



Release Cadence

~6-9 Months after the latest LTS of Java



JAKARTA EE



Jakarta Data

Standardizes the repository pattern for data access

<https://jakarta.ee/specifications/>





Jakarta NoSQL

Standardizes integration with NoSQL databases

<https://jakarta.ee/specifications/nosql/>





Jakarta RPC

Standardizes gRPC within Jakarta EE

<https://jakarta.ee/specifications/rpc/>





Jakarta EE 11

API source level: Java SE 21

API binary level: Java SE 21

TCK run with: Java SE 21



Summary

Jakarta EE 10 Platform



	Authentication 3.0	Persistence 3.1	
Authorization 2.1	Concurrency 3.0	Server Pages 3.1	RESTful Web Services 3.1
Activation 2.1	CDI 4.0	WebSocket 2.1	JSON Processing 2.1
Batch 2.1	Expression Language 5.0	Bean Validation 3.0	JSON Binding 3.0
Connectors 2.1	Faces 4.0	Debugging Support 2.0	Annotations 2.1
Mail 2.1	Security 3.0	Enterprise Beans Lite 4.0	Interceptors 2.1
Messaging 3.1	Servlet 6.0	Managed Beans 2.0	Dependency Injection 2.0
Enterprise Beans 4.0	Standard Tag Libraries 3.0	Transactions 2.0	CDI Lite 4.0

Updated

Not Updated

New

Under Development

Jakarta EE 11 Platform



	Authentication 3.1	Data 1.0	
Authorization 3.0	Concurrency 3.1	Persistence 3.2	RESTful Web Services 4.0
Activation 2.1	CDI 4.1	Pages 4.0	JSON Processing 2.1
Batch 2.1	Expression Language 6.0	WebSocket 2.2	JSON Binding 3.0
Connectors 2.1	Faces 5.0	Validation 3.1	Annotations 3.0
Mail 2.1	Security 4.0	Debugging Support 2.0	Interceptors 2.2
Messaging 3.1	Servlet 6.1	Enterprise Beans Lite 4.0	Dependency Injection 2.0
Enterprise Beans 4.0	Standard Tag Libraries 3.0	Transactions 2.0	CDI Lite 4.1

Updated

Not Updated

New



How JAKARTA EE is Relevant to spring®







javax.* -> jakarta.*





- 1. Update **Spring** version in pom.xml**
- 2. Update **Jakarta EE** version in pom.xml**
- 3. Fix the imports**
- 4. Repeat for Dependencies**
- 5. Transform Libraries if necessary**
- 6. Rename properties prefixed with javax**
- 7. Verify data and dynamic content**

<https://github.com/ivargrimstad/jakartaee-spring>



Jakarta EE

<https://jakarta.ee>

Jakartablogs

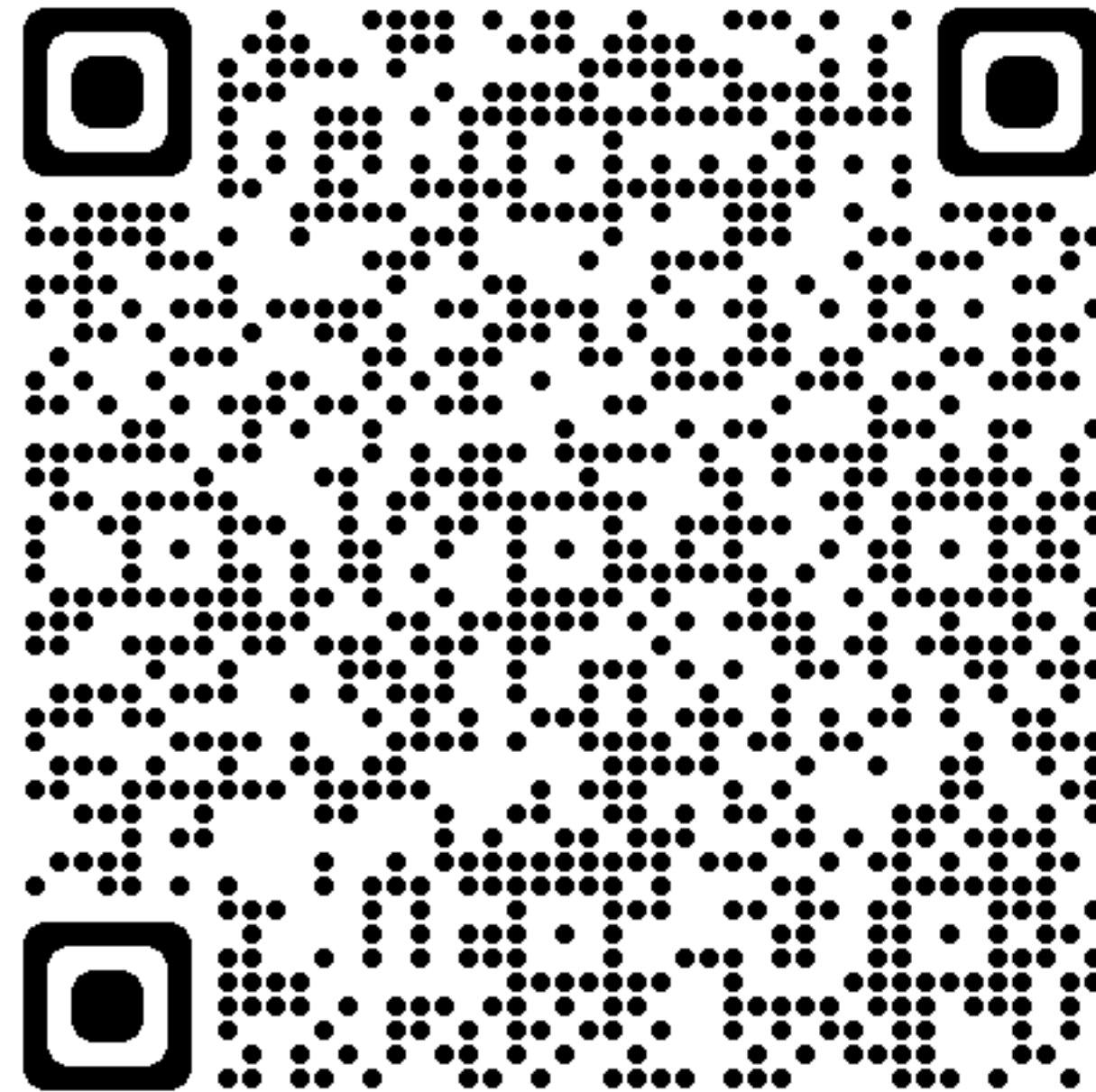
<https://jakartablogs.ee/>

Ivar's Hashtag Jakarta EE

<https://www.agilejava.eu/category/jakarta-ee/>

Demo Code

<https://github.com/ivargrimstad/jakartaee-spring>



Jakarta EE Overview

Course available on [LinkedIn Learning](#)

COMMUNITY

THE ASF CONFERENCE

CODE