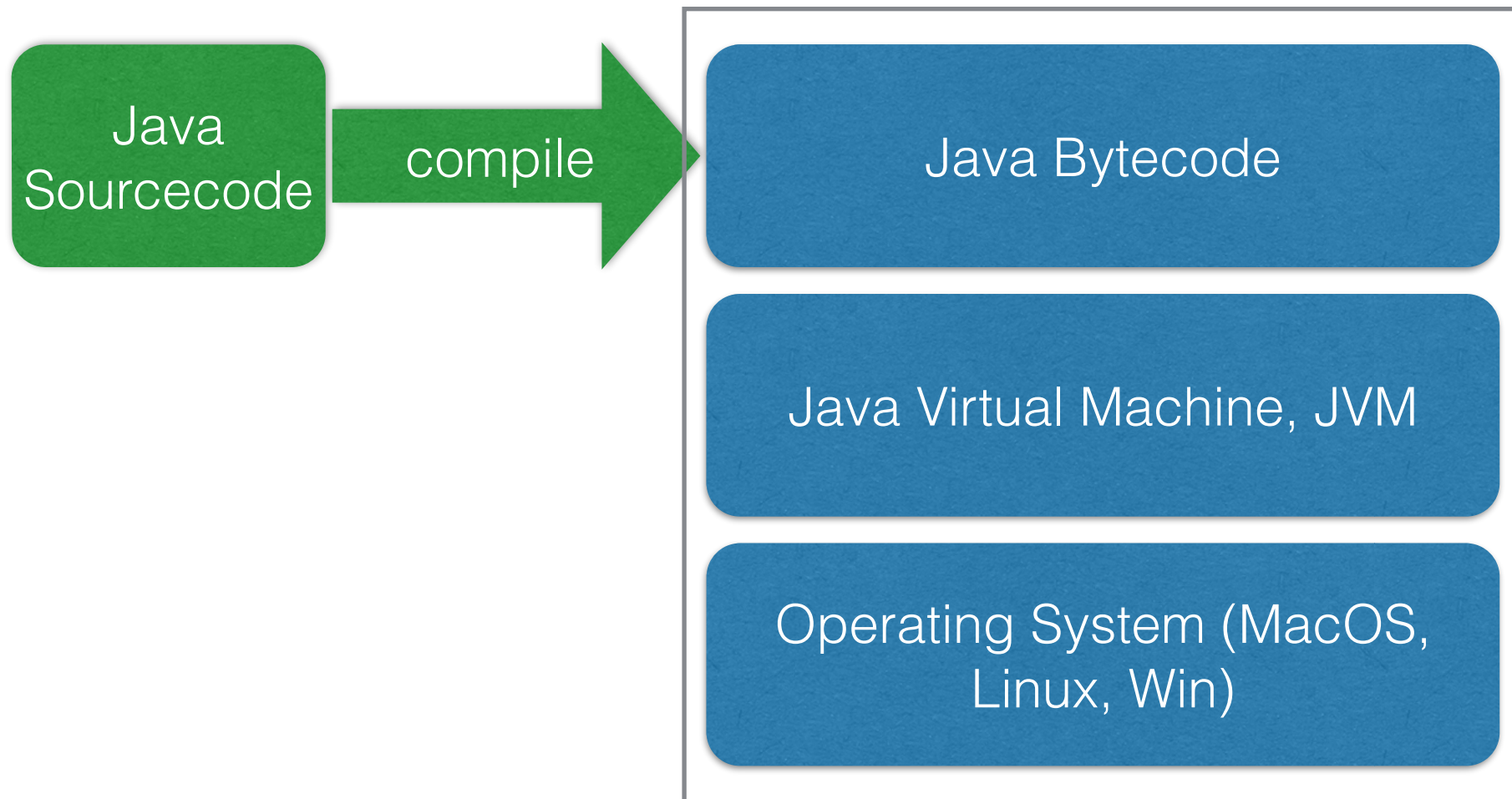


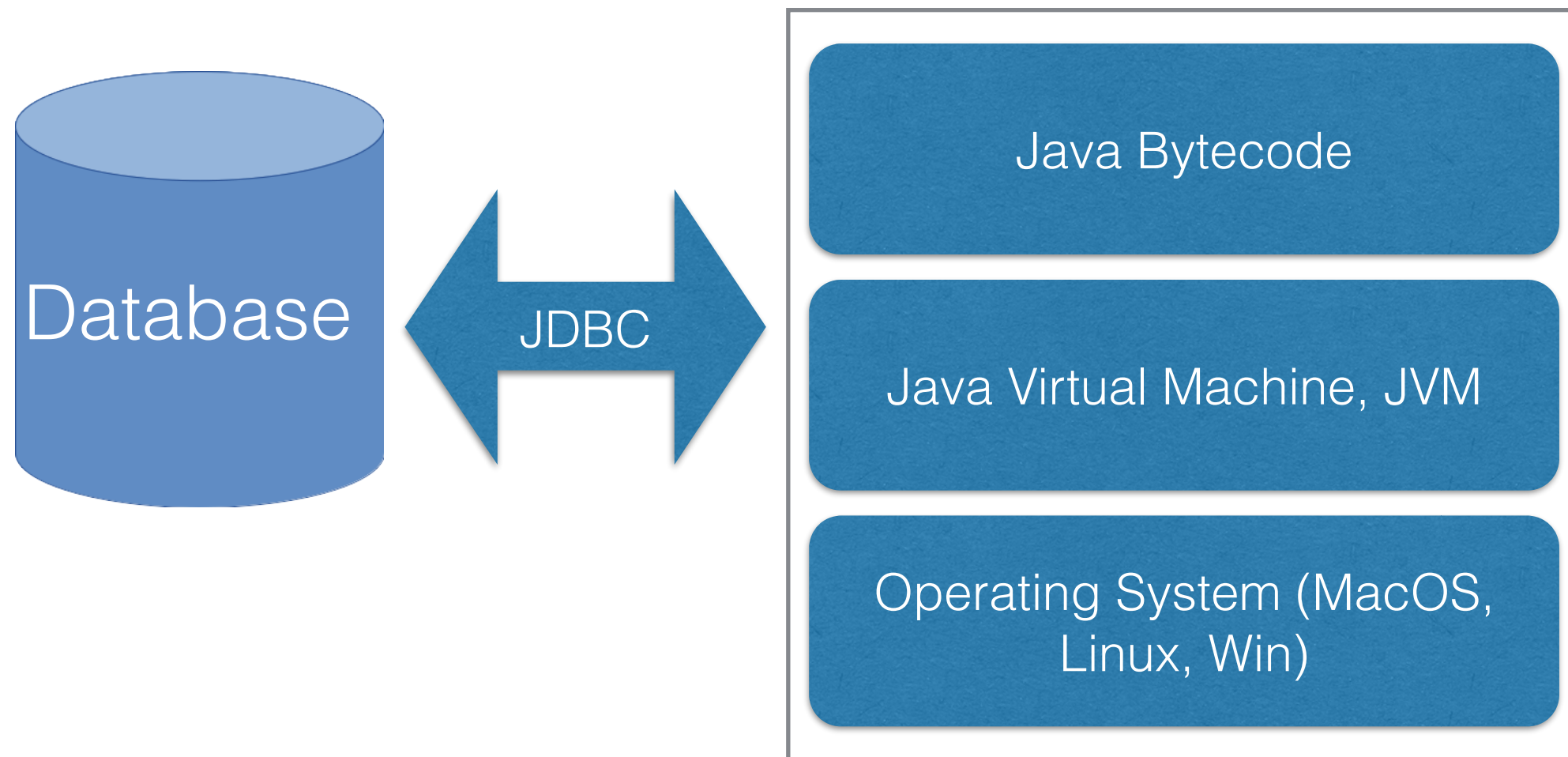
NVS4

Netzwerke und verteilte Systeme, 4. Jg.

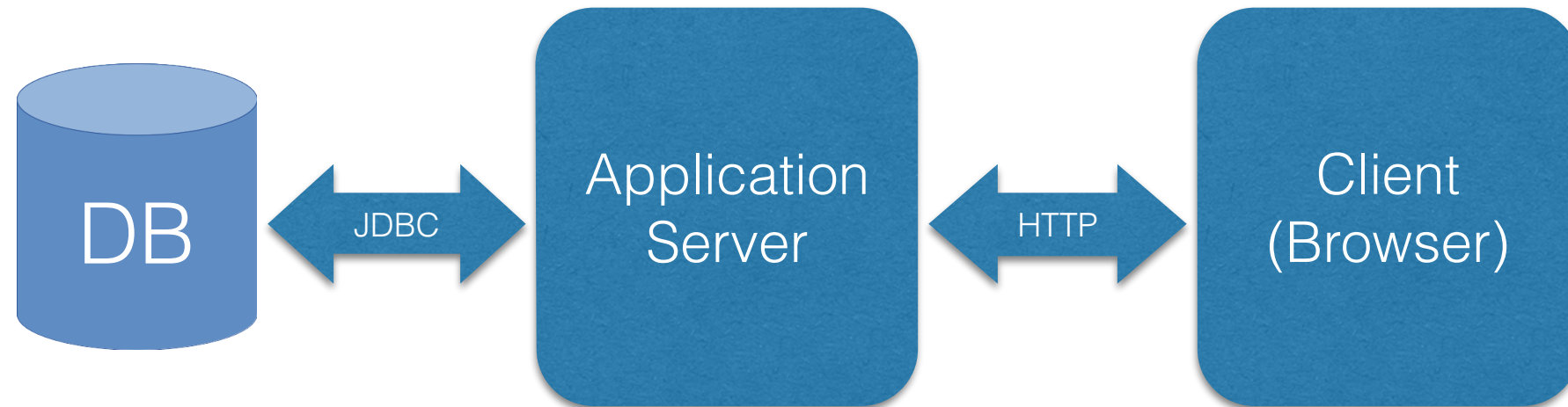
Fat Clients



2-tier Architecture



3-tier Architecture



Oracle, DerbyDB,
MySQL, PostgreSQL,...

Glassfish, Payara,
Wildfly, ...

Firefox, Safari,
Chrome, IE,...

compile + deploy

Java
Sourcecode

Server-zentriert:
JSF
PHP

Business Logik
am Server

Client-zentriert:
Javascript
und JS-Frameworks
(zB AngularJS)

Business Logik
am Client
Persistierung per REST

NVS teilt sich auf in

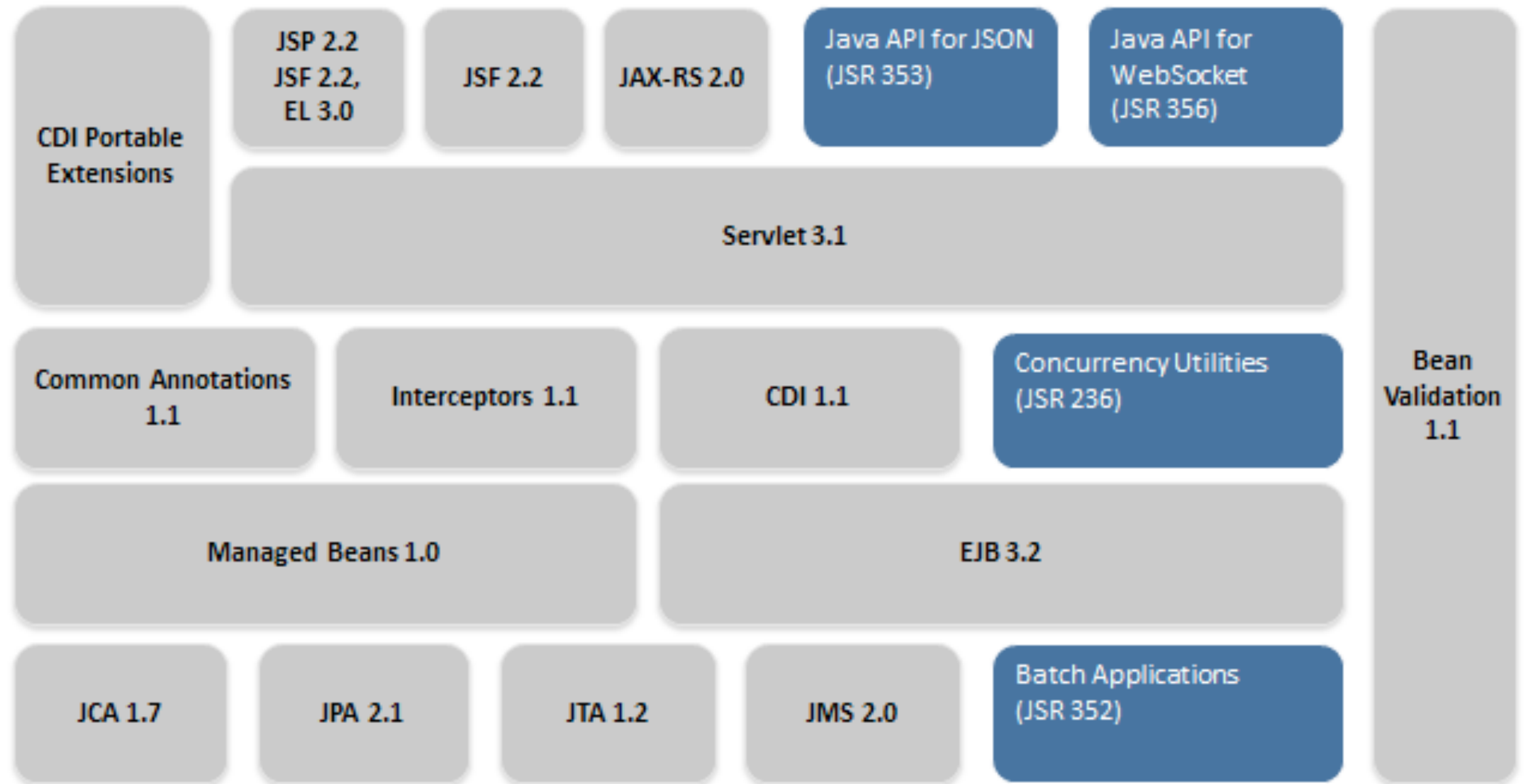
- Server-Technologien



- Client-Technologien

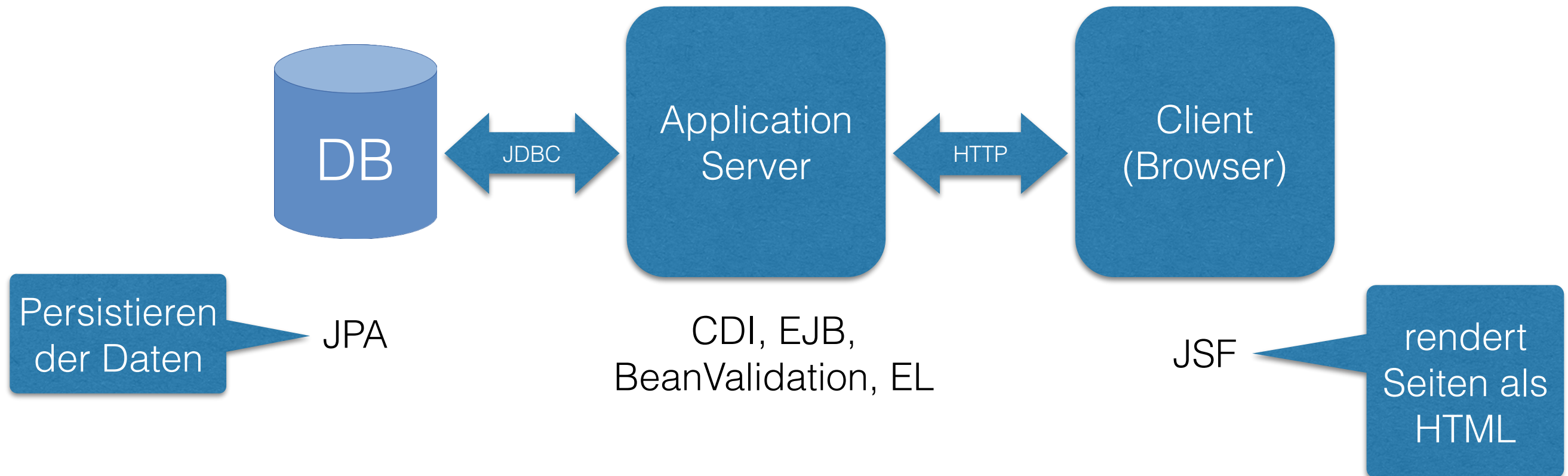


Bestandteile eines Application Servers



Java Specification Request

im Rahmen des Java Community Process



JPA ... Java Persistence API

CDI ... Context and Dependency Injection

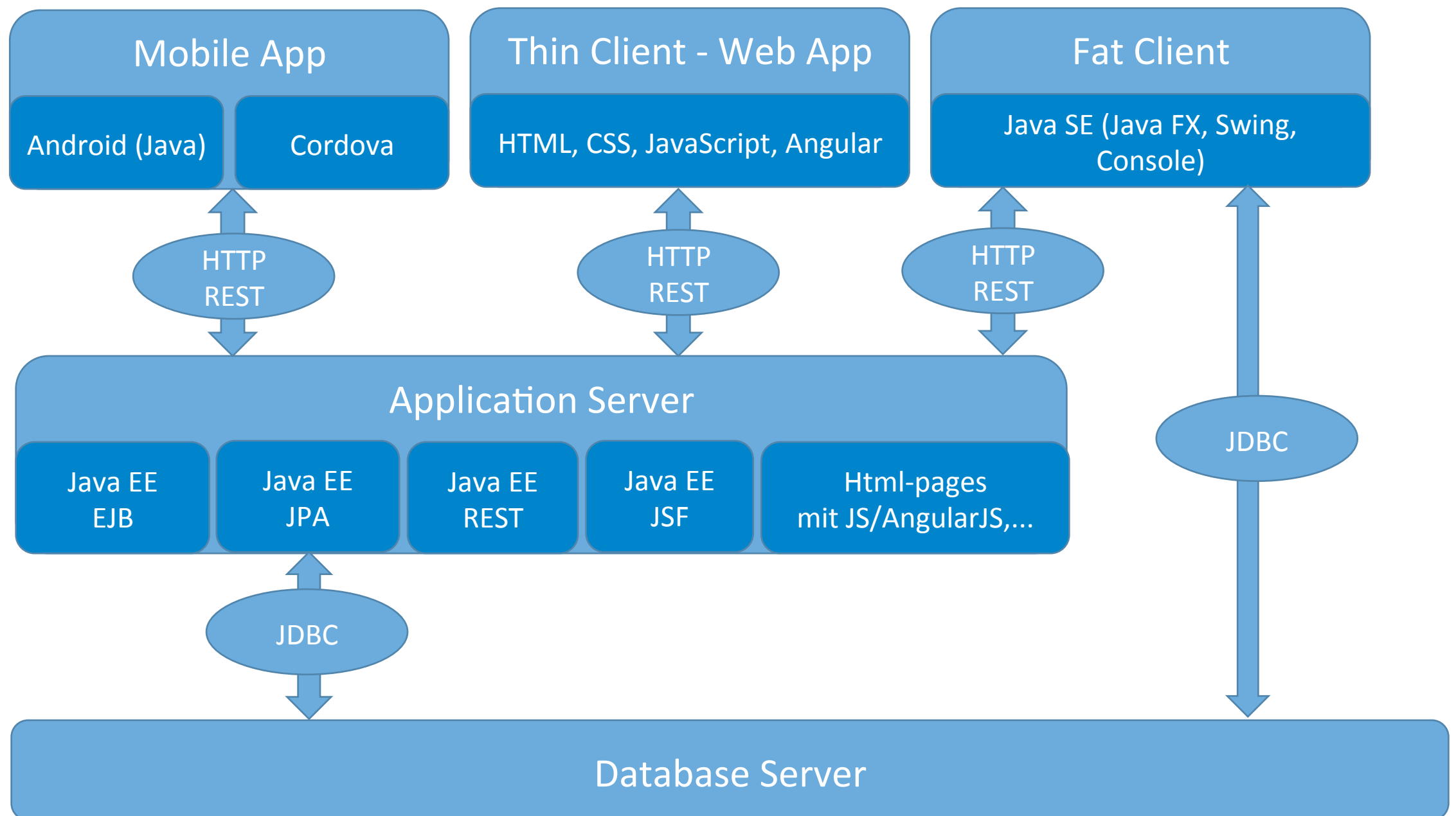
EJB ... Enterprise Java Beans

JSF ... JavaServer Faces

JTA ... Java Transaction API

BeanValidation, Servlets, ...

Big Picture



Voraussetzungen für das Arbeiten mit JEE

IDE + Application Server + Datenbank

- | IDE | | Application Server | | Datenbank |
|--------------------------|--|--------------------------|--|-------------------------|
| • Netbeans ! | | • Oracle Glassfish | | • Oracle DB |
| • IntelliJ IDEA ! | | • Payara ! | | • Apache Derby ! |
| • Eclipse | | • JBoss Wildfly ! | | • Oracle MySQL |
| • Oracle JDeveloper | | • Apache Geronimo | | • PostgreSQL |
| • ... | | • Jetty | | • H2 |
| | | • Oracle WebLogic | | • ... |
| | | • IBM WebSphere, ... | | |



Build-Systeme

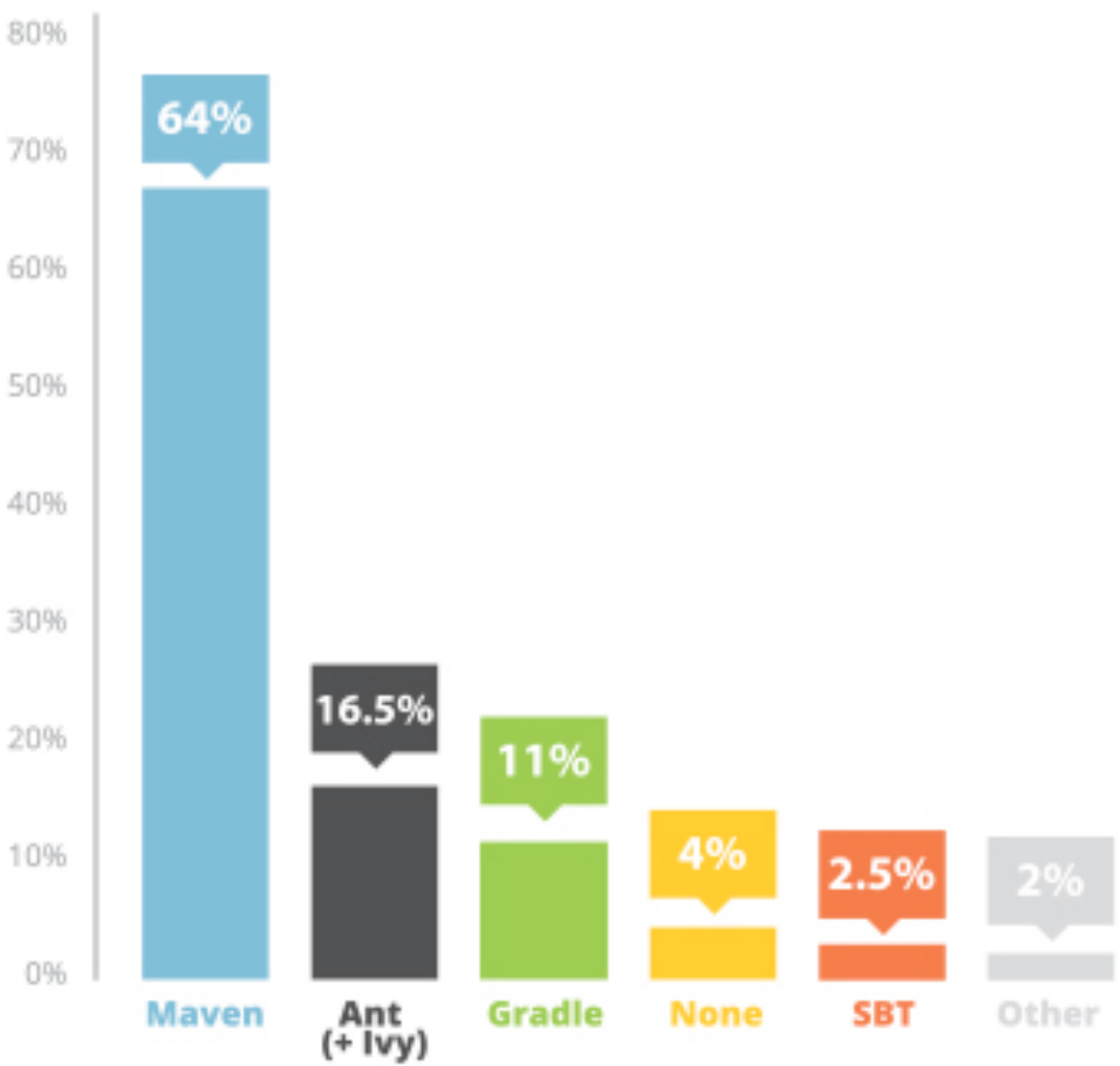


- Unter dem Begriff Build-Prozess werden alle Aktivitäten zusammengefasst, die für die Produktion und Bereitstellung von lauffähiger Software notwendig sind.
- Build-Systeme wie Ant, **Maven**!, Gradle aber auch MS-Build haben viele Aufgaben. Wir betrachten nur die (für uns derzeit) wichtigsten:
 - Download der benötigten Libraries
 - Kompilieren der Source-Codes
 - Paketieren zu .jar, .war, .ear, ...
 - Deployment auf den Application Server
- Die Build-Systeme sind in den IDE's verfügbar

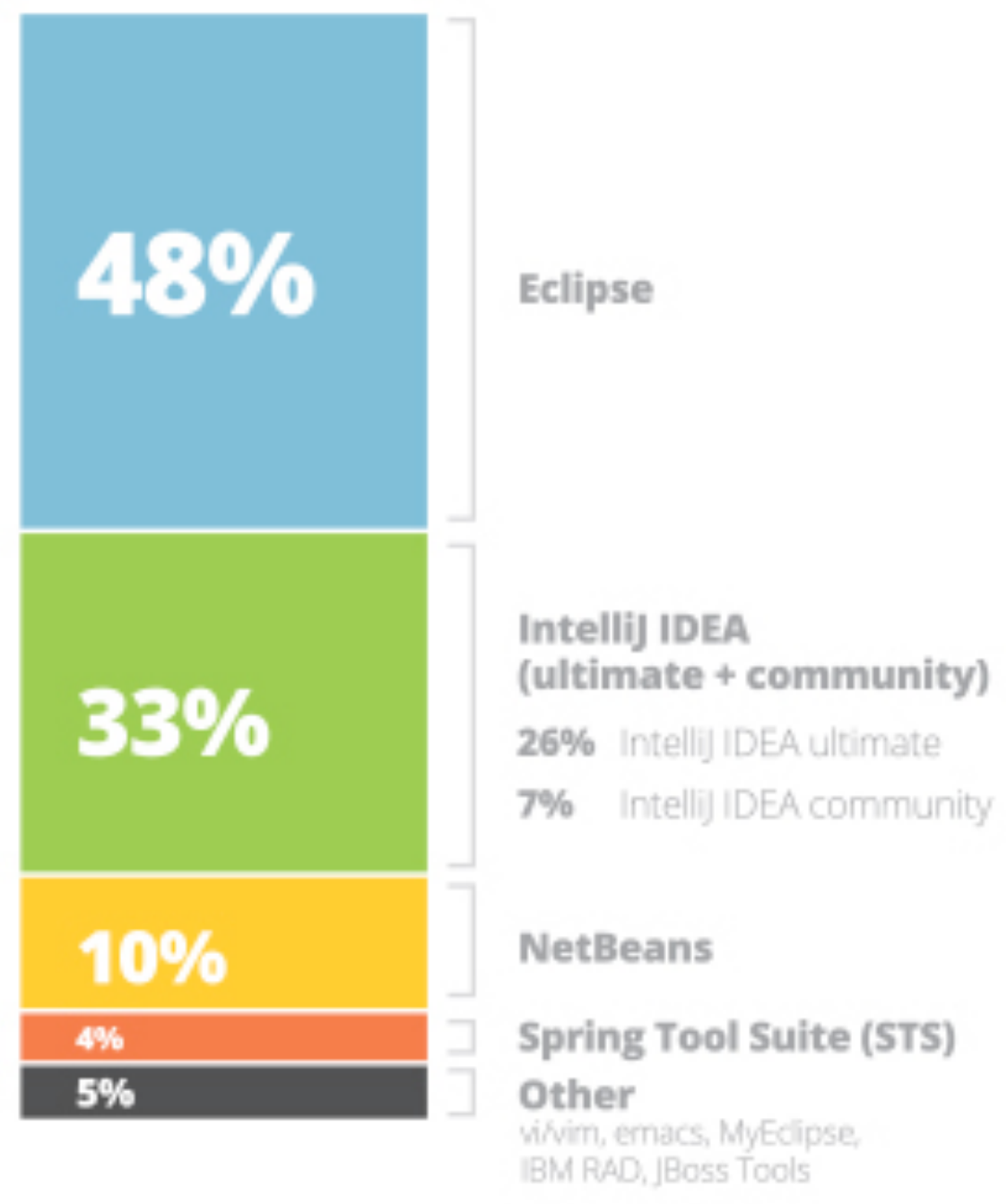
Einsatz im Unterricht

Maven: wird für JavaSE
und JavaEE verwendet
Gradle: wird für Android
verwendet

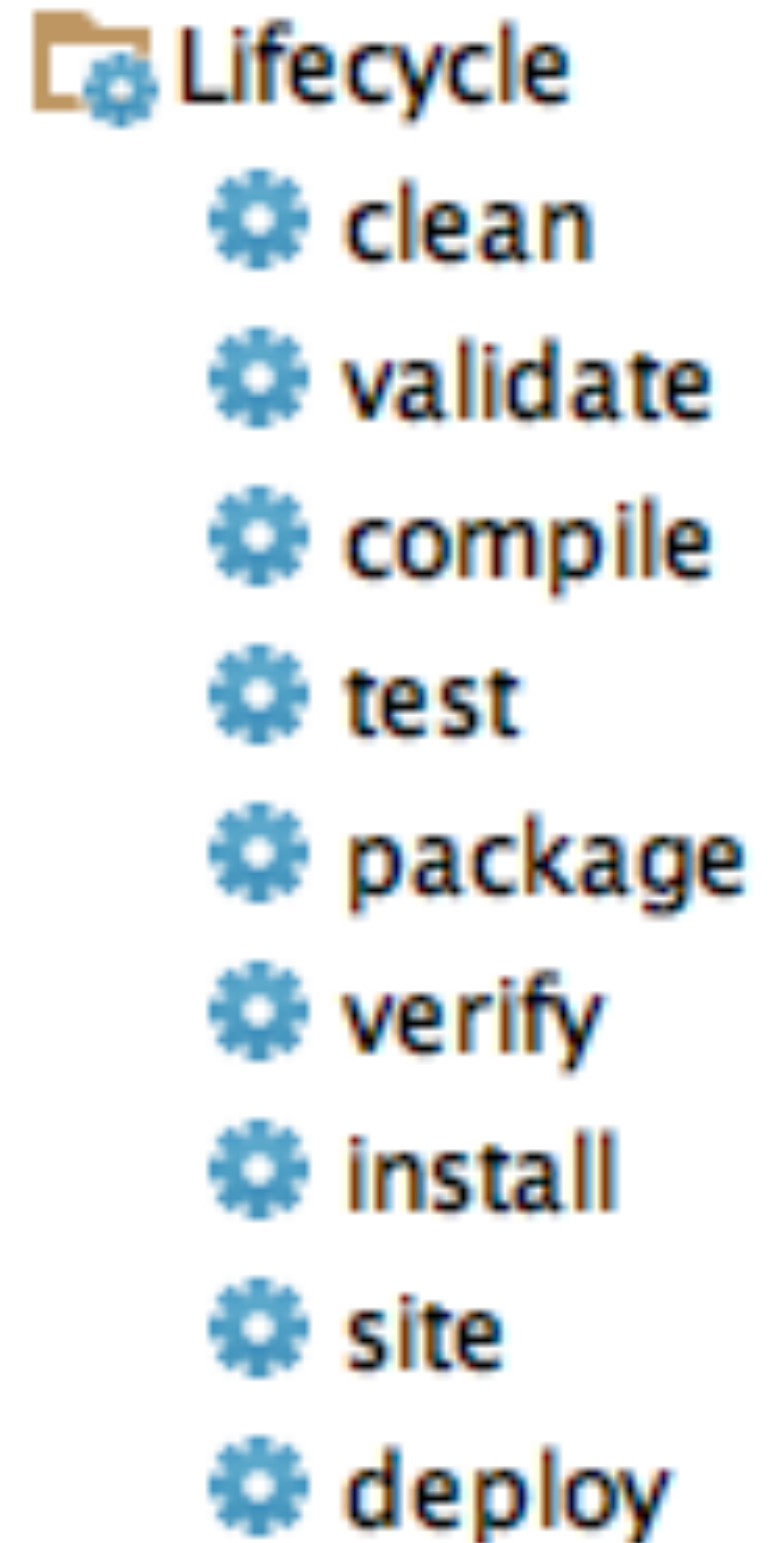
Build tool used most often



IDE used most often



- Project Object Model (POM)
- Maven besitzt ein Super POM
- Zentrales Konzept sind Lebenszyklen
 - Definieren für jeden Schritt im Build-Prozess eine Phase
 - Es gibt drei Lebenszyklen: clean, default und site
 - Es müssen nicht zwangsweise alle Phase des jeweiligen Zyklus genutzt werden
 - Entwickler wählt eine Phase aus und Maven sorgt dafür, dass alle vorherigen Phasen ausgeführt werden



Installation der Software

- **IntelliJ IDEA Ultimate:** Download vom Duke; Lizenznr am Duke verfügbar

https://duke.html5online.de/ftp_dl.php?file=VTI5bWRI2GhjbV2SVZCaWlyZHIzF0YVdWeWRHOXZiSE1nS0VsRVJYTWdkVzVrSUZORVMzTXBMMHBsZEVKeVlXbHVjeUJKYm5SbGJHeHBTaUJKUkVWQkiERTBMMHhKUTBWT1UwVXVtMEZaWDFWc2RHbHRZWFJsTG5SNGRBPT0=

- **Wildfly und/oder Payara:** Download vom Duke

https://duke.html5online.de/browse_ftp.php?dir=Software/!Programmiertools%20%28IDEs%20und%20SDKs%29/WildFly%209.0.1

[https://duke.html5online.de/browse_ftp.php?dir=Software/!Programmiertools%20\(IDEs%20und%20SDKs\)/Glassfish%20Payara%204.1](https://duke.html5online.de/browse_ftp.php?dir=Software/!Programmiertools%20(IDEs%20und%20SDKs)/Glassfish%20Payara%204.1)

In Verzeichnis /opt bzw \opt entpacken

- **DerbyDb:** Ist im Java JDK enthalten.
\$JAVA_HOME/db/bin/startNetworkServer
- **JavaJDK:** <http://www.oracle.com/technetwork/java/javase>

mind. Java 8 update 60