

CLOUDZ LABS

Spring Boot Introduction



Table of Contents

01 | **Spring Boot**

02 | **Spring Boot Features**

03 | **Spring VS Spring Boot**

PART 01 Spring Boot

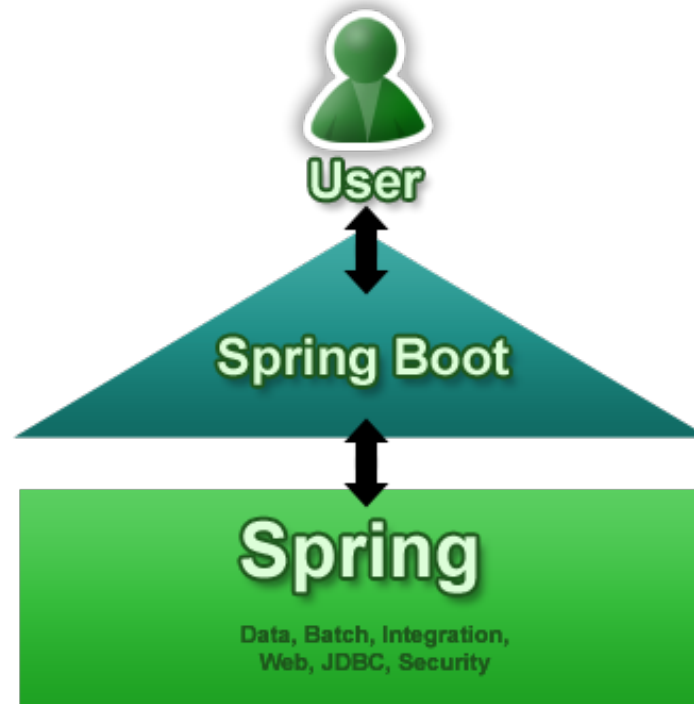




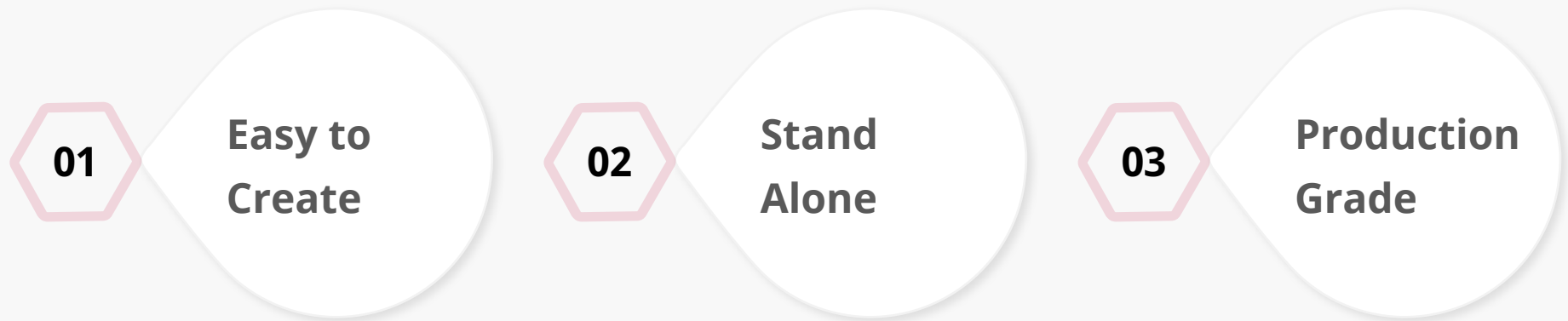
The Spring Framework is a Java platform that provides comprehensive infrastructure support for developing Java application.

Spring handles infrastructure so you focus on **your application.**

Spring Boot is designed to get **you** up and running as **quickly** as possible.



PART 02 Spring Boot Features



Spring Boot makes it easy to create
stand-alone, production-grade Spring
based Application that

you can **“just run”**.

Spring Boot makes it **easy to create**
stand-alone, production-grade Spring
based Application that

you can **“just run”**.

project Application Configuration

project

Application

Configuration

project

```

<!-- Spring -->
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-context</artifactId>
  <version>4.3.10.RELEASE</version>
  <exclusions>
    <exclusion>
      <groupId>commons-logging</groupId>
      <artifactId>commons-logging</artifactId>
    </exclusion>
  </exclusions>
</dependency>
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-webmvc</artifactId>
  <version>4.3.10.RELEASE</version>
</dependency>
<dependency>
  <groupId>org.springframework.data</groupId>
  <artifactId>spring-data-jpa</artifactId>
  <version>1.11.0.RELEASE</version>
</dependency>
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-aspects</artifactId>
  <version>4.3.10.RELEASE</version>
</dependency>

```

```

<!-- AspectJ -->
<dependency>
  <groupId>org.aspectj</groupId>
  <artifactId>aspectjrt</artifactId>
  <version>1.8.10</version>
</dependency>

<!-- JPA -->
<dependency>
  <groupId>org.hibernate</groupId>
  <artifactId>hibernate-core</artifactId>
  <version>5.0.12.Final</version>
  <exclusions>
    <exclusion>
      <groupId>org.apache.geronimo.specs</groupId>
      <artifactId>geronimo-jta_1.1_spec</artifactId>
    </exclusion>
  </exclusions>
</dependency>
<dependency>
  <groupId>org.hibernate</groupId>
  <artifactId>hibernate-entitymanager</artifactId>
  <version>5.0.12.Final</version>
  <exclusions>
    <exclusion>
      <groupId>org.apache.geronimo.specs</groupId>
      <artifactId>geronimo-jta_1.1_spec</artifactId>
    </exclusion>
  </exclusions>
</dependency>
<!-- dbms, logging, test, servlet & etc... -->

```

project

```

<!-- Spring -->
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-context</artifactId>
  <version>4.3.10.RELEASE</version>
  <exclusions>
    <exclusion>
      <groupId>commons-logging</groupId>
      <artifactId>commons-logging</artifactId>
    </exclusion>
  </exclusions>
</dependency>
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-webmvc</artifactId>
  <version>4.3.10.RELEASE</version>
</dependency>
<dependency>
  <groupId>org.springframework.data</groupId>
  <artifactId>spring-data-jpa</artifactId>
  <version>1.11.6.RELEASE</version>
</dependency>
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-aspects</artifactId>
  <version>4.3.10.RELEASE</version>
</dependency>

```

```

<!-- AspectJ -->
<dependency>
  <groupId>org.aspectj</groupId>
  <artifactId>aspectjrt</artifactId>
  <version>1.8.10</version>
</dependency>

<!-- JPA -->
<dependency>
  <groupId>org.hibernate</groupId>
  <artifactId>hibernate-core</artifactId>
  <version>5.0.12.Final</version>
  <exclusions>
    <exclusion>
      <groupId>org.apache.geronimo.specs</gro
upId>
      <artifactId>geronimo-jta_1.1_spec</arti
factId>
    </exclusion>
  </exclusions>
</dependency>
<dependency>
  <groupId>org.hibernate</groupId>
  <artifactId>hibernate-entitymanager</artifactId>
  <version>5.0.12.Final</version>
  <exclusions>
    <exclusion>
      <groupId>org.apache.geronimo.specs</gro
upId>
    </exclusion>
  </exclusions>
</dependency>
<!-- dbms, logging, test, servlet & etc... -->

```

spring-boot-starter-project

```
<parent>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-parent</artifactId>
  <version>1.5.0.RELEASE</version>
  <relativePath/> <!-- lookup parent from repository -->
</parent>
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>
  <dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <scope>runtime</scope>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-test</artifactId>
    <scope>test</scope>
  </dependency>
</dependencies>
```

spring-boot-starter-project

```
<parent>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-parent</artifactId>
  <version>1.5.6.RELEASE</version>
  <relativePath/> <!-- lookup parent from repository -->
</parent>
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>
  <dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <scope>runtime</scope>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-test</artifactId>
    <scope>test</scope>
  </dependency>
</dependencies>
```

spring-boot-starter-project

Application

Configuration

web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd">

  <!-- The definition of the Root Spring Container shared by all Servlets and Filters -->
  <context-param>
    <param-name>contextConfigLocation</param-name>
    <param-value>/WEB-INF/spring/root-context.xml</param-value>
  </context-param>

  <!-- Creates the Spring Container shared by all Servlets and Filters -->
  <listener>
    <listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>
  </listener>

  <filter>
    <filter-name>EncodingFilter</filter-name>
    <filter-class>org.springframework.web.filter.CharacterEncodingFilter</filter-class>
    <init-param>
      <param-name>encoding</param-name>
      <param-value>UTF-8</param-value>
    </init-param>
  </filter>
```


web.xml

```
<!-- next... -->

<filter-mapping>
  <filter-name>EncodingFilter</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>

<!-- Processes application requests -->
<servlet>
  <servlet-name>appServlet</servlet-name>
  <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
  <init-param>
    <param-name>contextConfigLocation</param-name>
    <param-value>/WEB-INF/spring/appServlet/servlet-context.xml</param-value>
  </init-param>
  <load-on-startup>1</load-on-startup>
</servlet>

<servlet-mapping>
  <servlet-name>appServlet</servlet-name>
  <url-pattern>/</url-pattern>
</servlet-mapping>
</web-app>
```

root-context.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:util="http://www.springframework.org/schema/util"
       xmlns:context="http://www.springframework.org/schema/context"
       xmlns:p="http://www.springframework.org/schema/p"
       xmlns:jdbc="http://www.springframework.org/schema/jdbc"
       xmlns:tx="http://www.springframework.org/schema/tx"
       xmlns:jpa="http://www.springframework.org/schema/data/jpa"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
                           http://www.springframework.org/schema/beans/spring-beans.xsd
                           http://www.springframework.org/schema/util http://www.springframework.org/schema/util/spring-
util.xsd
                           http://www.springframework.org/schema/context http://www.springframework.org/schema/data/jpa
                           http://www.springframework.org/schema/context/spring-context.xsd
                           http://www.springframework.org/schema/jdbc http://www.springframework.org/schema/jdbc/spring-
jdbc.xsd
                           http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx.xsd
                           http://www.springframework.org/schema/data/jpa/spring-jpa.xsd">
  <!-- Root Context: defines shared resources visible to all other web components -->
  <bean id="dataSource" class="org.springframework.jdbc.datasource.SimpleDriverDataSource">
    <property name="driverClass" value="com.mysql.jdbc.Driver" />
    <property name="url" value="jdbc:mysql://localhost:3306/testdb" />
    <property name="username" value="root" />
    <property name="password" value="" />
  </bean>
```

root-context.xml

```
<!-- next... -->

<!-- JPA settings -->
<bean id="transactionManager" class="org.springframework.orm.jpa.JpaTransactionManager">
  <property name="entityManagerFactory" ref="entityManagerFactory" />
</bean>
<bean class="org.springframework.dao.annotation.PersistenceExceptionTranslationPostProcessor" />

<bean id="entityManagerFactory"
class="org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean">
  <property name="dataSource" ref="dataSource" />
  <property name="packagesToScan" value="com.sample.spring" />
  <property name="jpaVendorAdapter">
    <bean class="org.springframework.orm.jpa.vendor.HibernateJpaVendorAdapter" />
  </property>
  <property name="jpaProperties">
    <props>
      <prop key="hibernate.hbm2ddl.auto">update</prop>
    </props>
  </property>
</bean>

<jpa:repositories base-package="com.sample.spring" entity-manager-factory-ref="entityManagerFactory" />
</beans>
```

servlet-context.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans:beans xmlns="http://www.springframework.org/schema/mvc"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:beans="http://www.springframework.org/schema/beans"
  xmlns:context="http://www.springframework.org/schema/context"
  xsi:schemaLocation="http://www.springframework.org/schema/mvc
    http://www.springframework.org/schema/mvc/spring-mvc.xsd
    http://www.springframework.org/schema/beans
    http://www.springframework.org/schema/beans/spring-beans.xsd
    http://www.springframework.org/schema/context
    http://www.springframework.org/schema/context/spring-context.xsd">

  <!-- DispatcherServlet Context: defines this servlet's request-processing infrastructure -->

  <!-- Enables the Spring MVC @Controller programming model -->
  <annotation-driven />

  <!-- Handles HTTP GET requests for /resources/** by efficiently serving up static resources
  in the ${webappRoot}/resources directory -->
  <resources mapping="/resources/**" location="/resources/" />
</beans:beans>
```

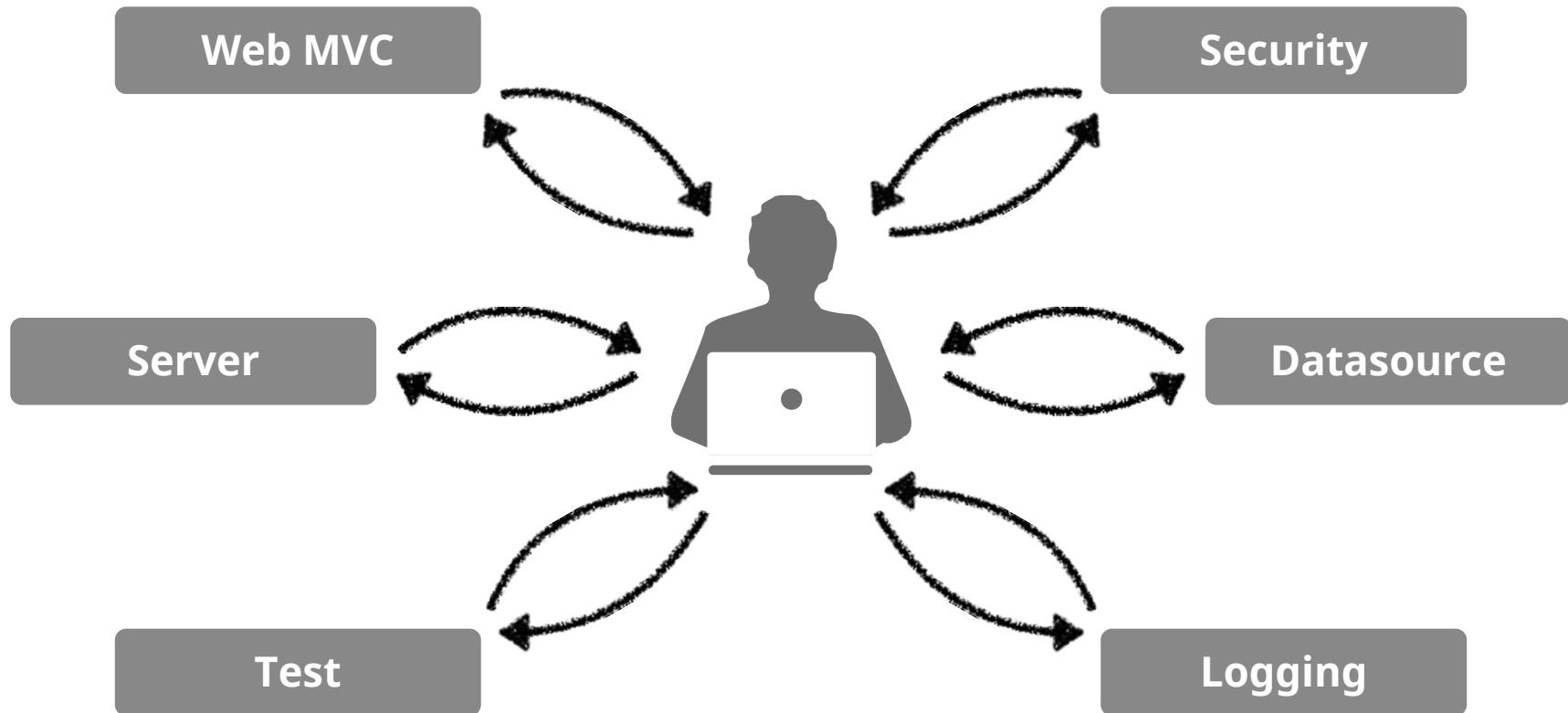
servlet-context.xml

```
<!-- next... -->

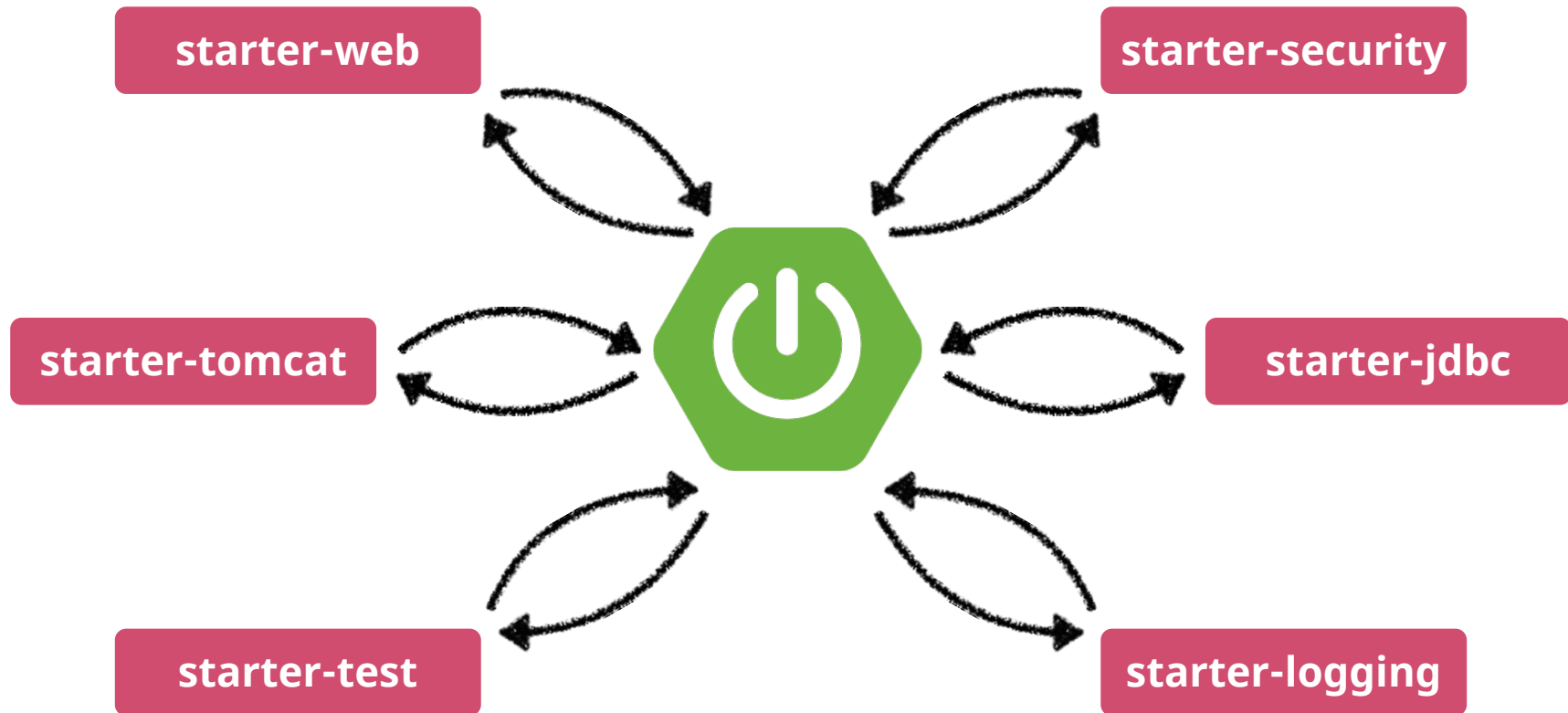
    <!-- Resolves views selected for rendering by @Controllers to .jsp resources in
    the /WEB-INF/views directory -->
    <beans:bean class="org.springframework.web.servlet.view.ContentNegotiatingViewResolver">
        <beans:property name="viewResolvers">
            <beans:list>
                <beans:bean class="org.springframework.web.servlet.view.BeanNameViewResolver"/>
                <beans:bean
class="org.springframework.web.servlet.view.InternalResourceViewResolver">
                    <beans:property name="prefix" value="/WEB-INF/jsp/" />
                    <beans:property name="suffix" value=".jsp" />
                </beans:bean>
            </beans:list>
        </beans:property>
        <beans:property name="defaultViews">
            <beans:list>
                <beans:bean
class="org.springframework.web.servlet.view.json.MappingJackson2JsonView"/>
            </beans:list>
        </beans:property>
    </beans:bean>

    <context:component-scan base-package="com.sample.spring" />
</beans:beans>
```

Application



SpringBootApplication



@SpringBootApplication

```
package com.sample.boot;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class SpringBootApplicationSample {

    public static void main(String[] args) {
        SpringApplication.run(SpringBootApplicationSample.class, args);
    }

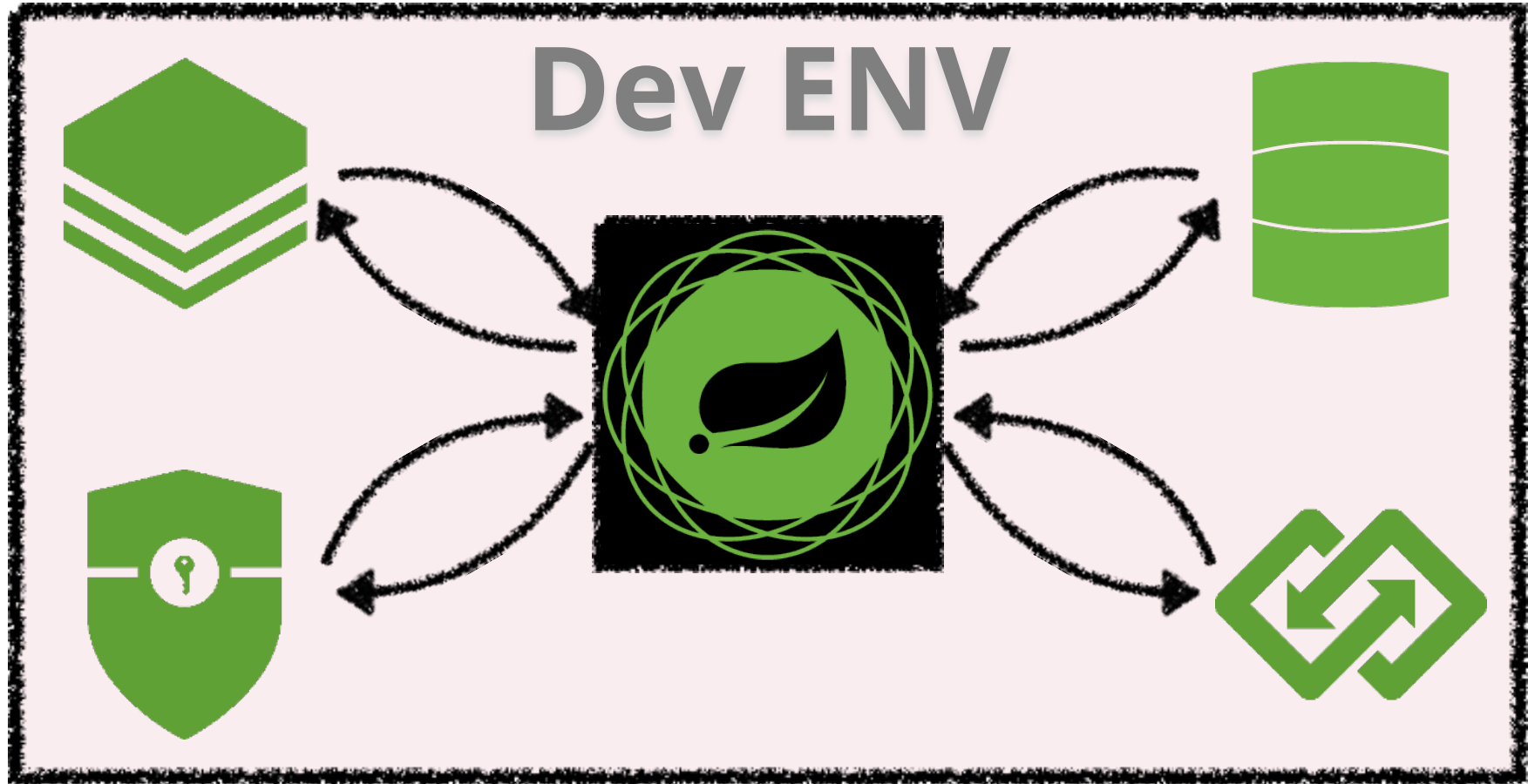
}
```


spring-boot-starter-project

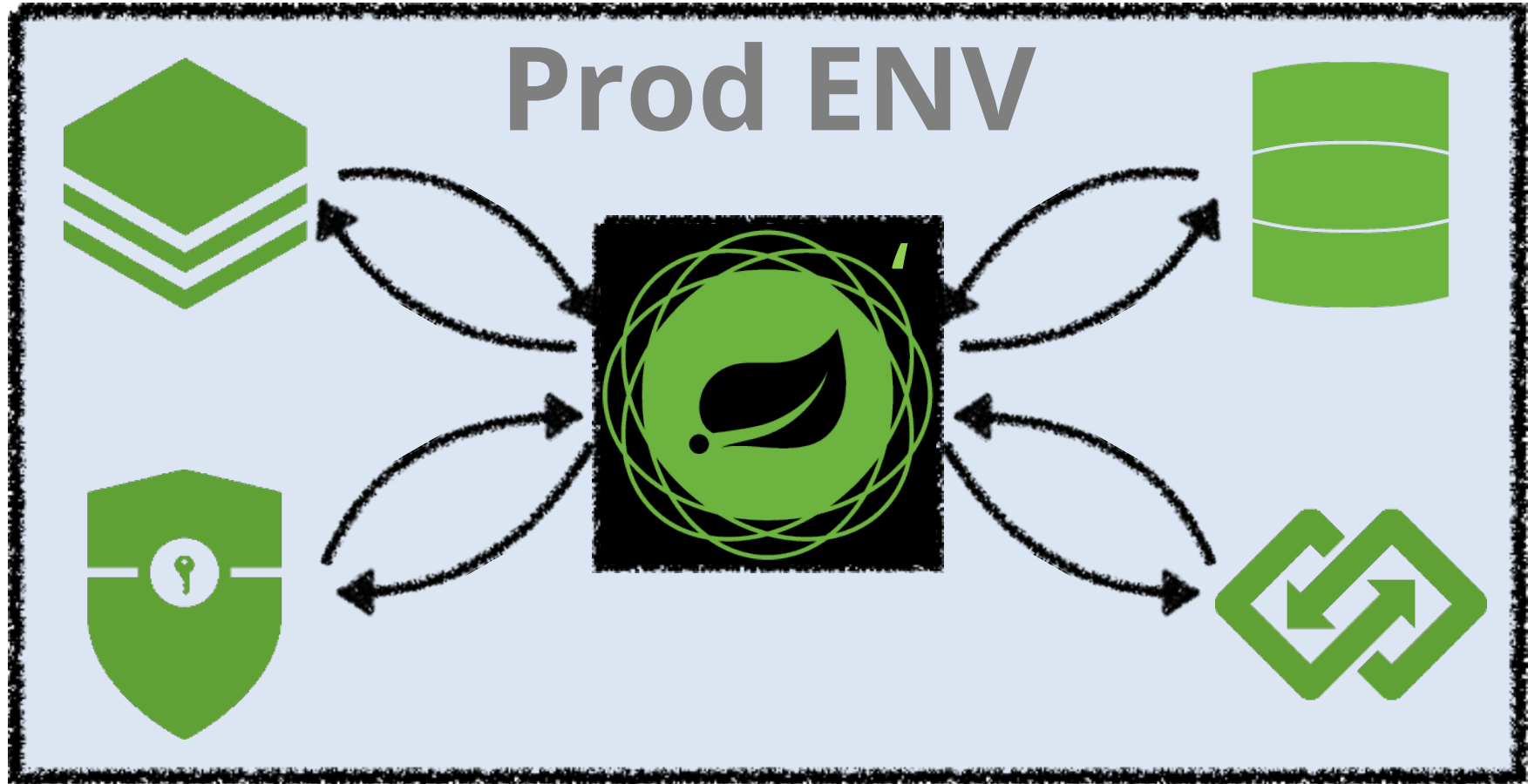
@SpringBootApplication

Configuration

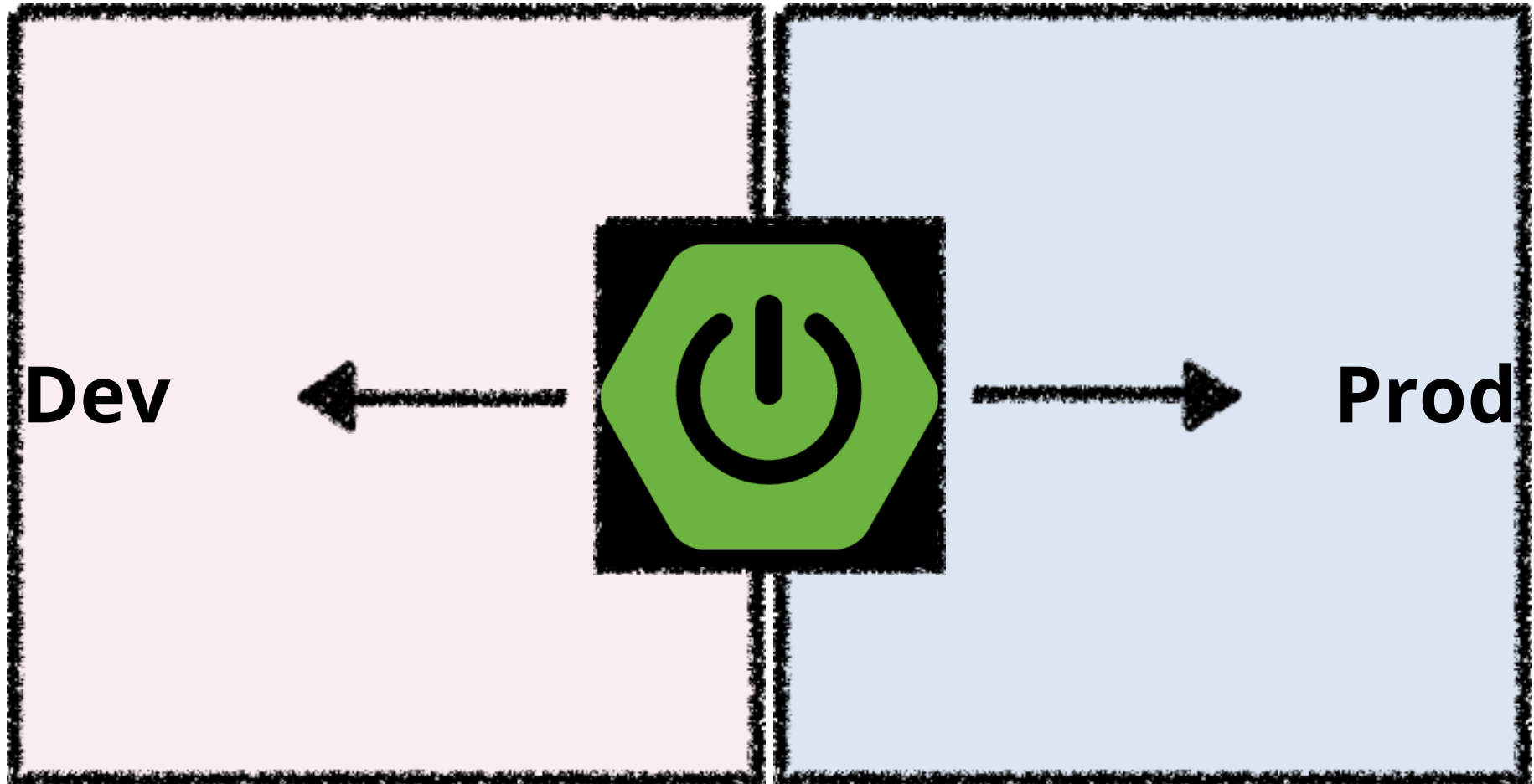
Configuration



Configuration



application-{profile}.yml



application-{profile}.yml

```
# Common
spring:
  jpa:
    hibernate:
      ddl-auto: update
---
# Localhost
spring:
  profiles: default
  datasource:
    url: jdbc:mysql://localhost:3306/testdb
    username: root
    password:
    driver-class-name: com.mysql.jdbc.Driver
---
# Production
spring:
  profiles: prod
  datasource:
    url: jdbc:mysql://${DB_SERVICE_HOST}:${DB_SERVICE_PORT}/${MYSQL_DATABASE}
    username: ${MYSQL_USER}
    password: ${MYSQL_PASSWORD}
    driver-class-name: com.mysql.jdbc.Driver
```

spring-boot-starter-project

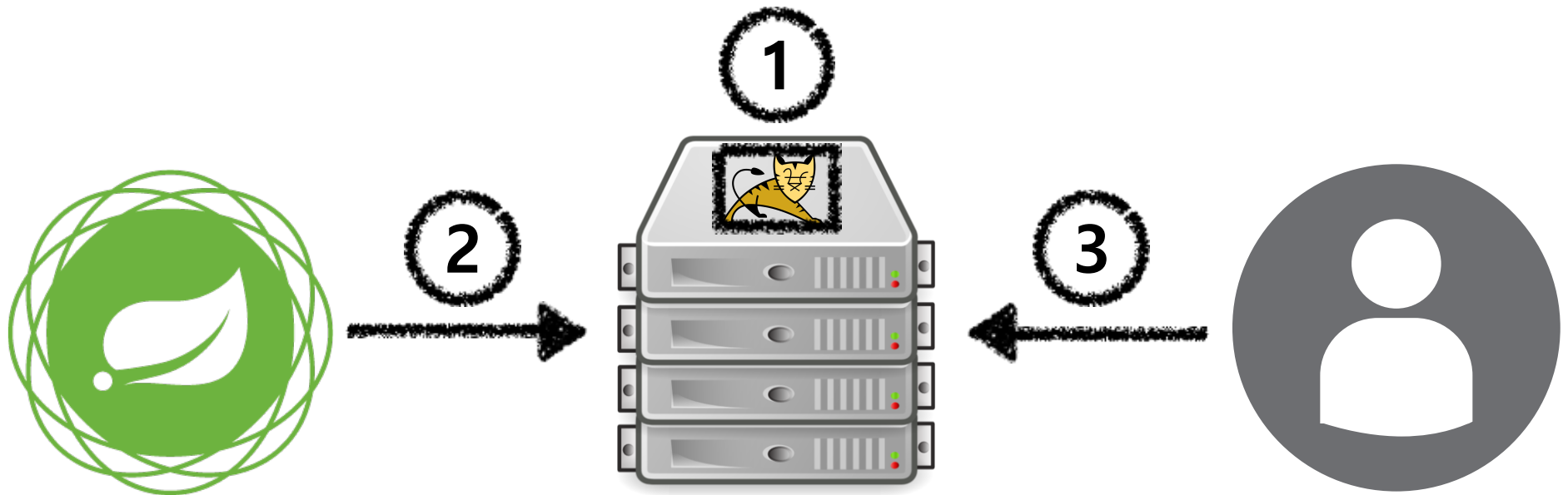
@SpringBootApplication

application-{profile}.yml

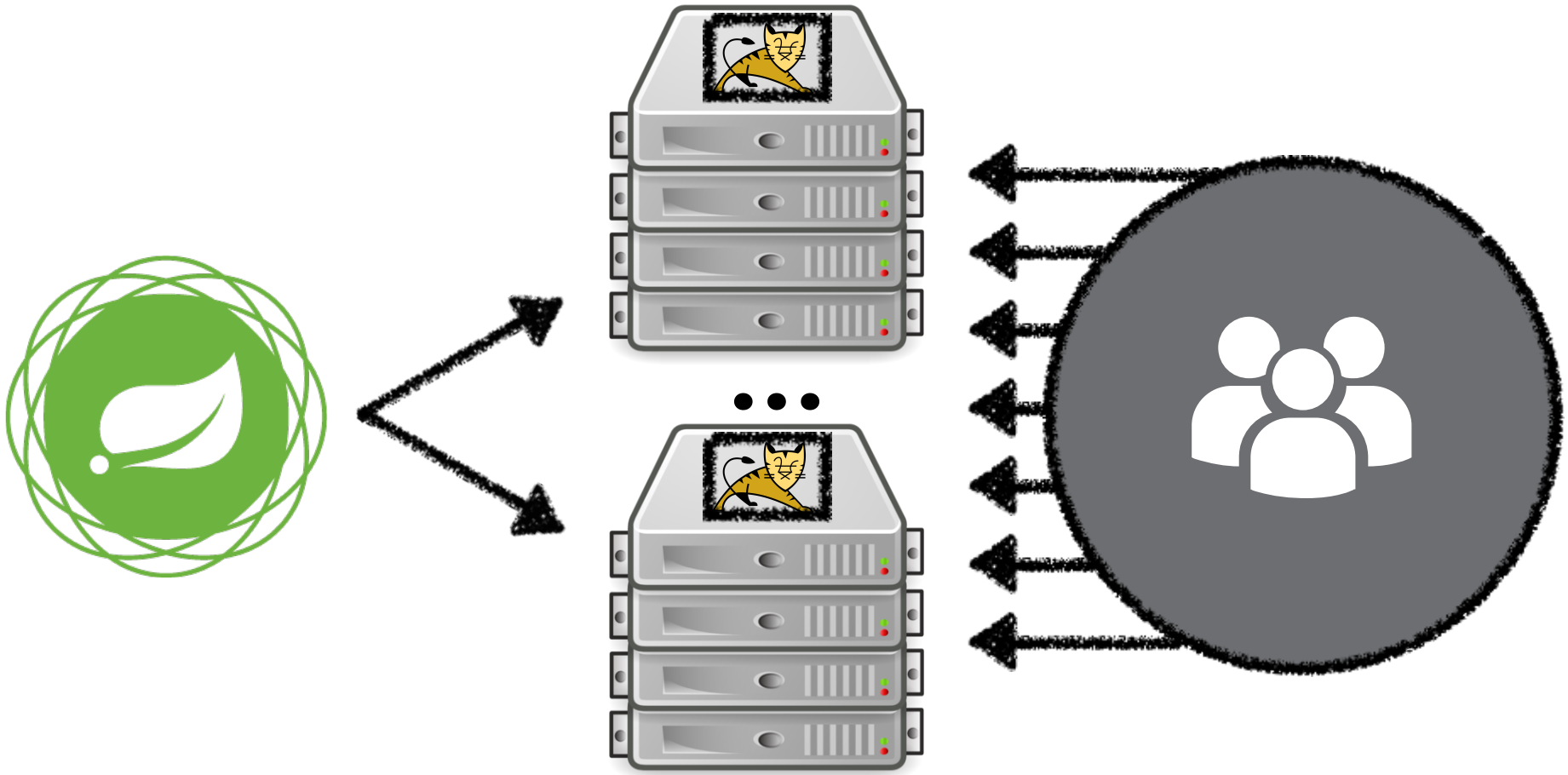
Spring Boot makes it easy to create **stand-alone**, production-grade Spring based Application that

you can **“just run”**.

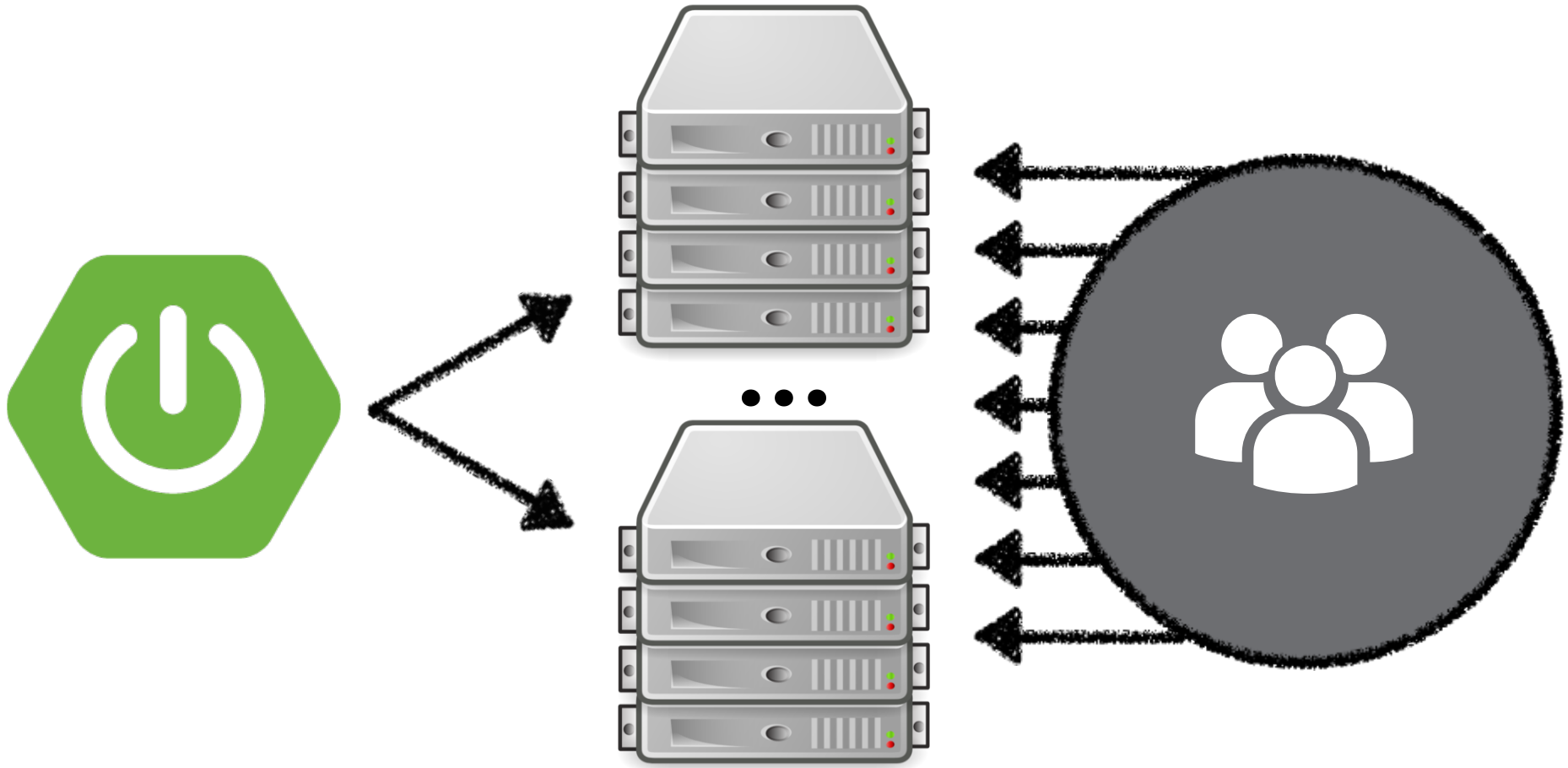
WAS Setting



n times WAS Setting



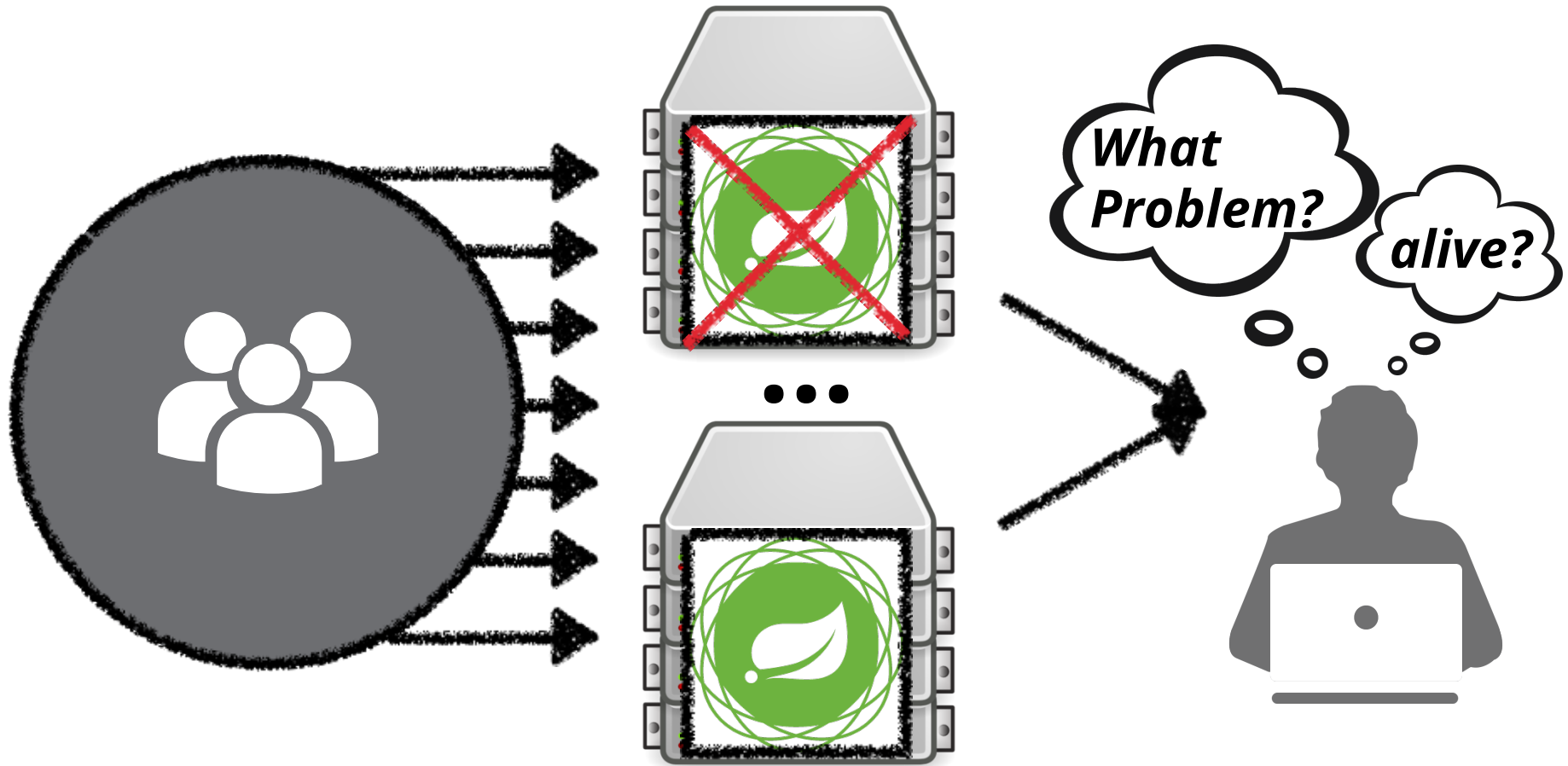
1 times WAS Setting



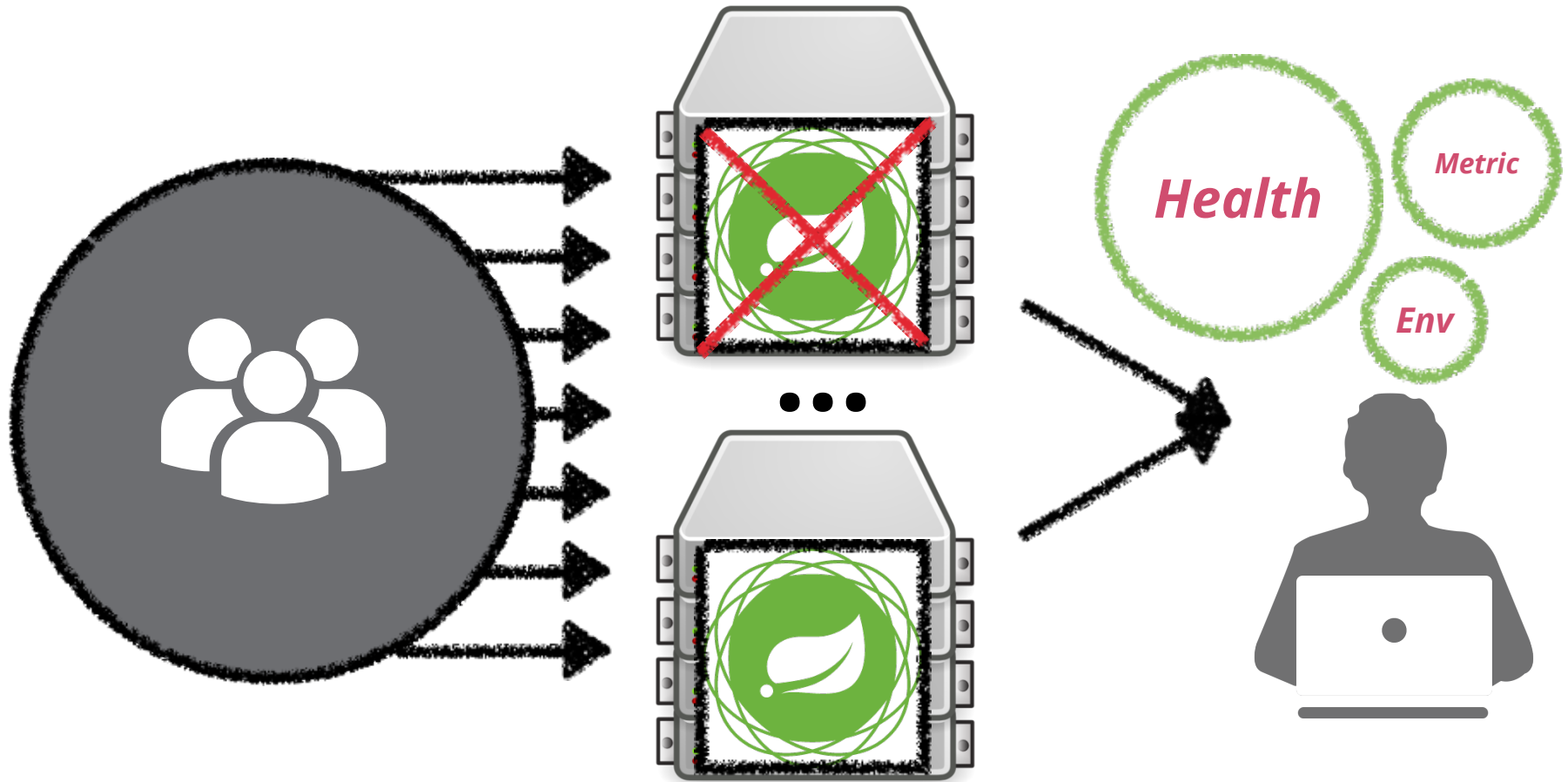
Spring Boot makes it easy to create stand-alone, **production-grade** Spring based Application that

you can **“just run”**.

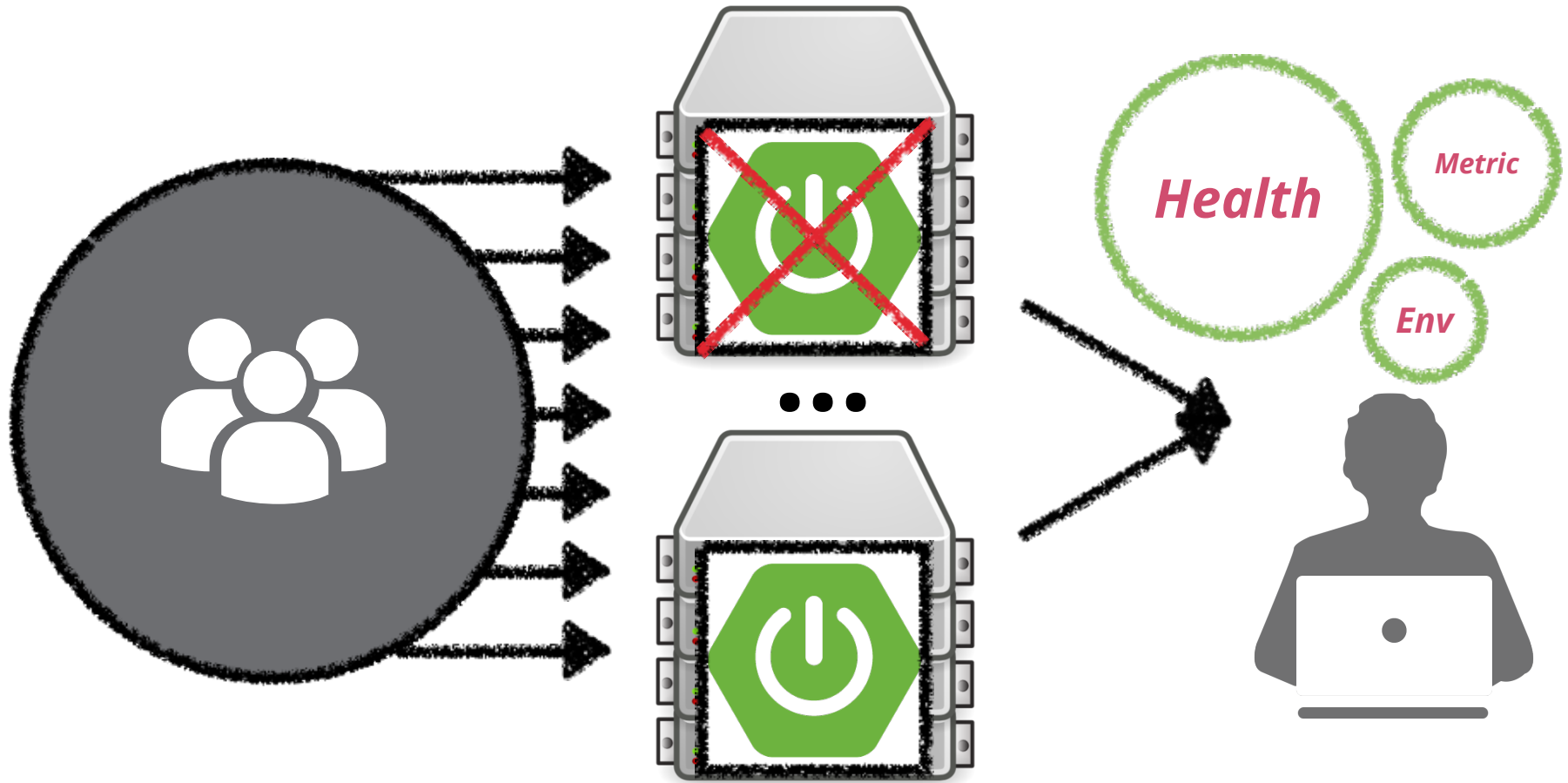
a number of additional features



Production-ready features



spring-boot-starter-actuator



PART 03 Spring VS Spring Boot



SPRING FRAMEWORK





you can “**just run**”.



Anywhere

감사합니다

