

Java EE 6: Easy & Extensible





Dan Allen

@mojavelinux 

Seam Community Liaison
JBoss, by Red Hat Inc.

Lincoln Baxter III

@lincolnthree 

Forge Lead
JBoss, by Red Hat Inc.

What is **Java EE**?

What **benefits** does it offer?

What **improvements** have been made?

What makes it **extensible**?

Is it really **easy**?

Enterprise application development





HUGE!

It's gonna be big...

$f(x)$

Managed
component

Managed
component

Managed
component

Managed
component

Services

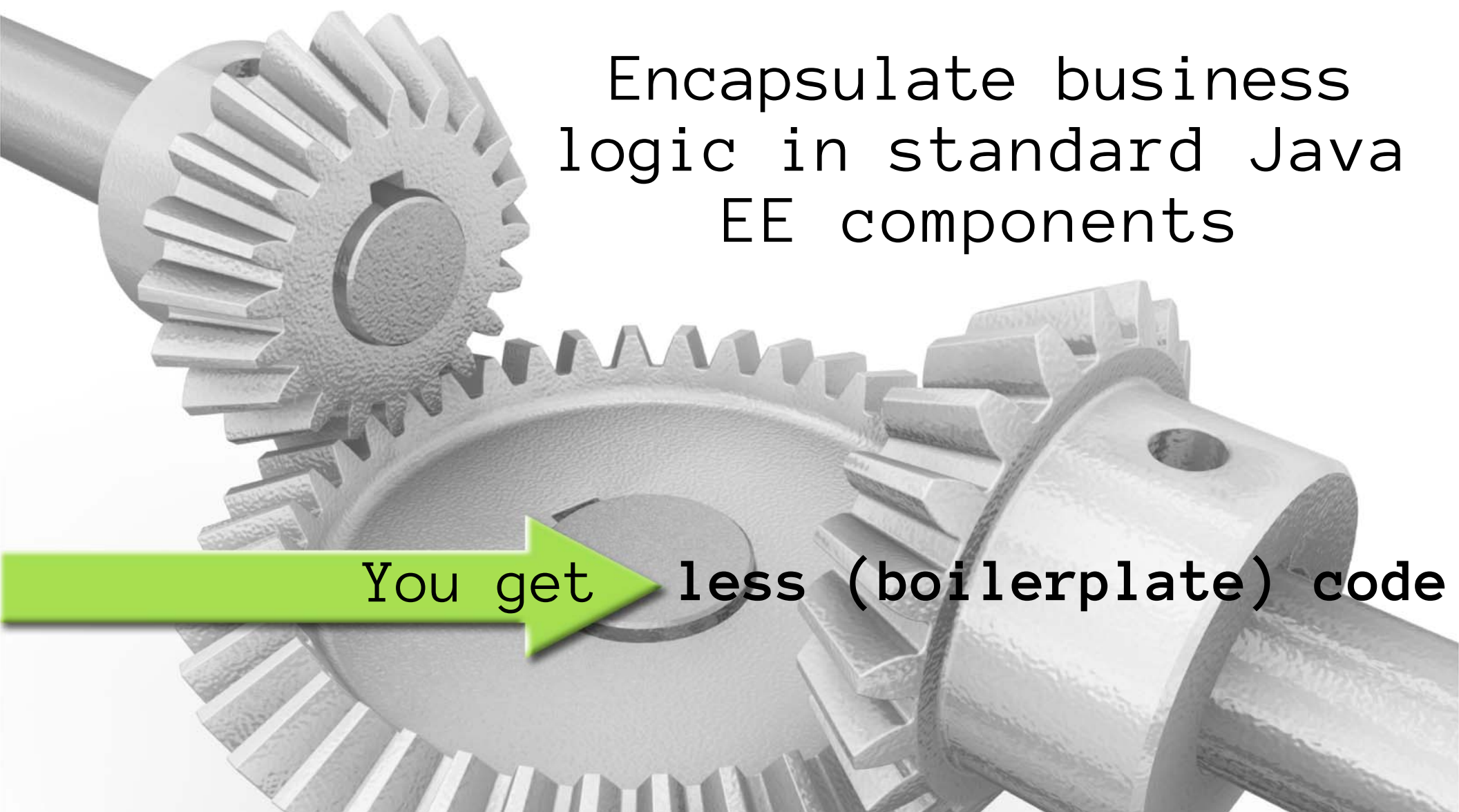
Standard Platform

What is Java EE?

Programming
Model

A word cloud of Spring annotations on a black background. The annotations are arranged in a roughly circular pattern, with some overlapping. The colors of the text vary, including shades of yellow, orange, red, and teal. The font is a bold, sans-serif typeface. The annotations include:

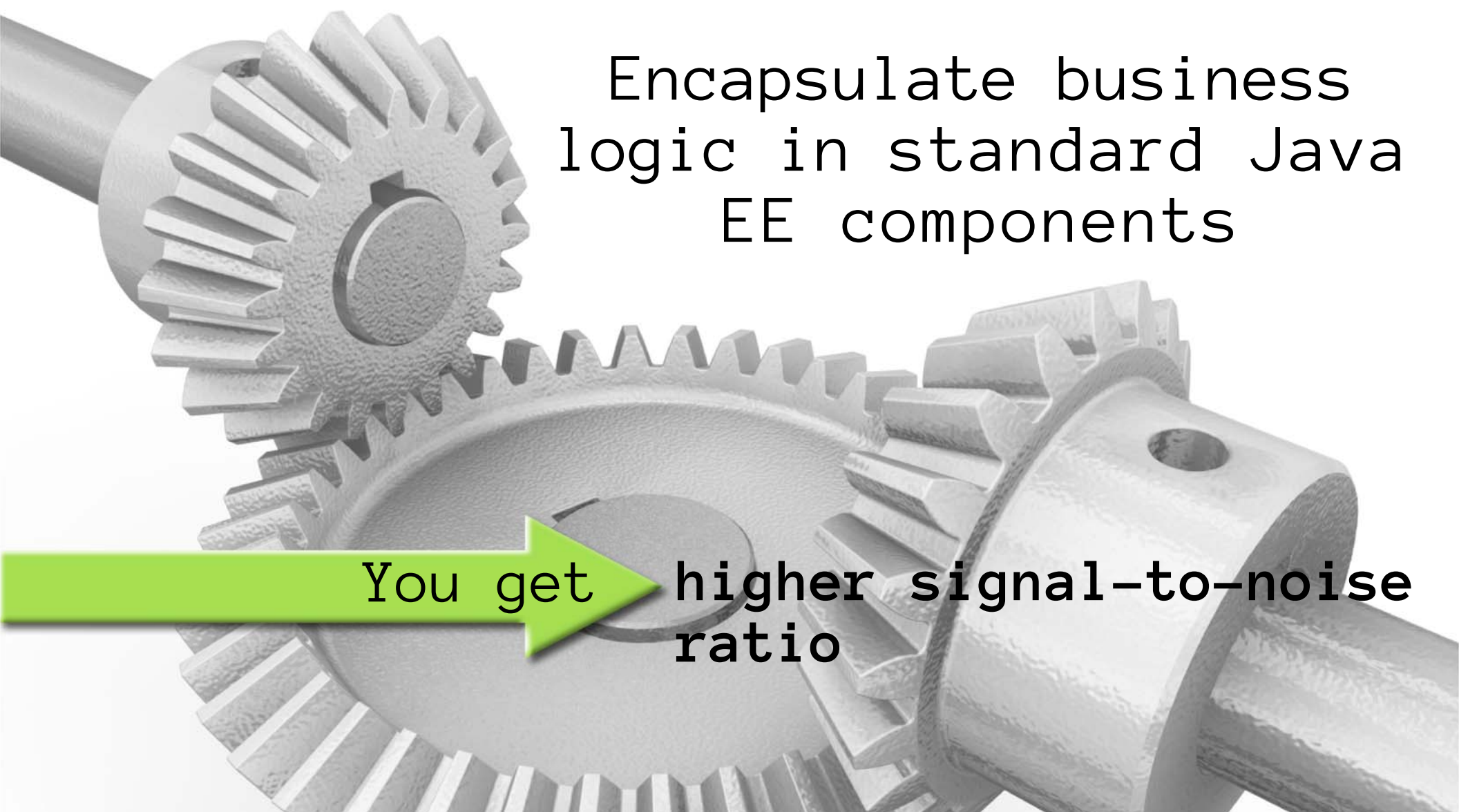
- @Inject** (large, yellow, center)
- @PersistenceContext** (large, orange, top-right)
- @Stateless** (large, yellow, top-left)
- @Path** (small, red, top-left)
- @Observes** (medium, red, top-left)
- @SessionScoped** (medium, yellow, top-right)
- @TransactionAttribute** (medium, orange, top-right)
- @RequestScoped** (medium, red, top-right)
- @GET** (small, yellow, top-right)
- @ApplicationScoped** (medium, orange, left)
- @Qualifier** (medium, orange, left)
- @Named** (medium, teal, bottom-left)
- @Produces** (medium, teal, bottom-left)
- @Stateful** (medium, teal, bottom-right)
- @PostConstruct** (small, red, bottom-right)
- Entity** (large, yellow, vertical, center)



Encapsulate business
logic in standard Java
EE components

You get

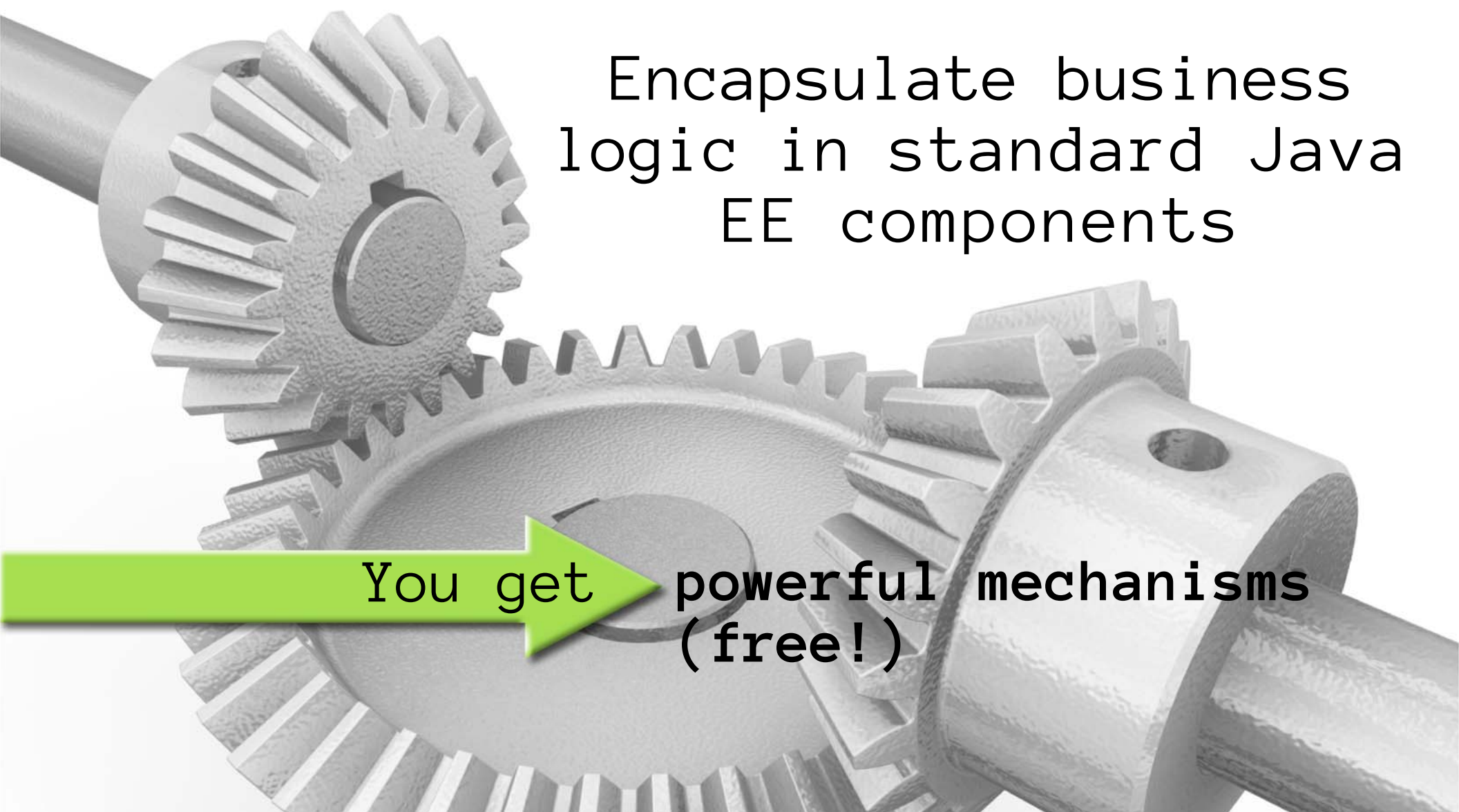
less (boilerplate) code



Encapsulate business
logic in standard Java
EE components

You get

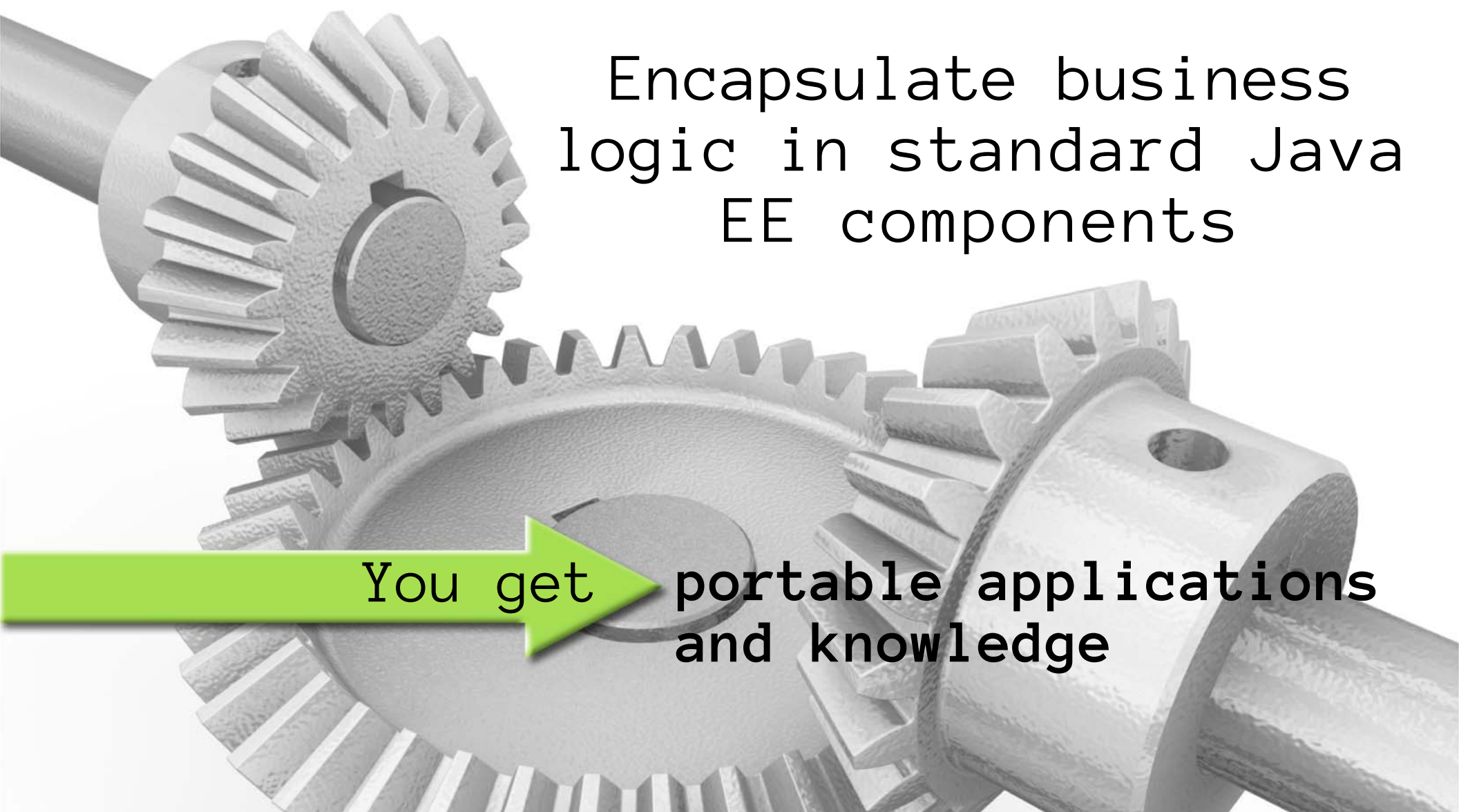
higher signal-to-noise
ratio



Encapsulate business
logic in standard Java
EE components

You get

**powerful mechanisms
(free!)**



Encapsulate business
logic in standard Java
EE components

You get

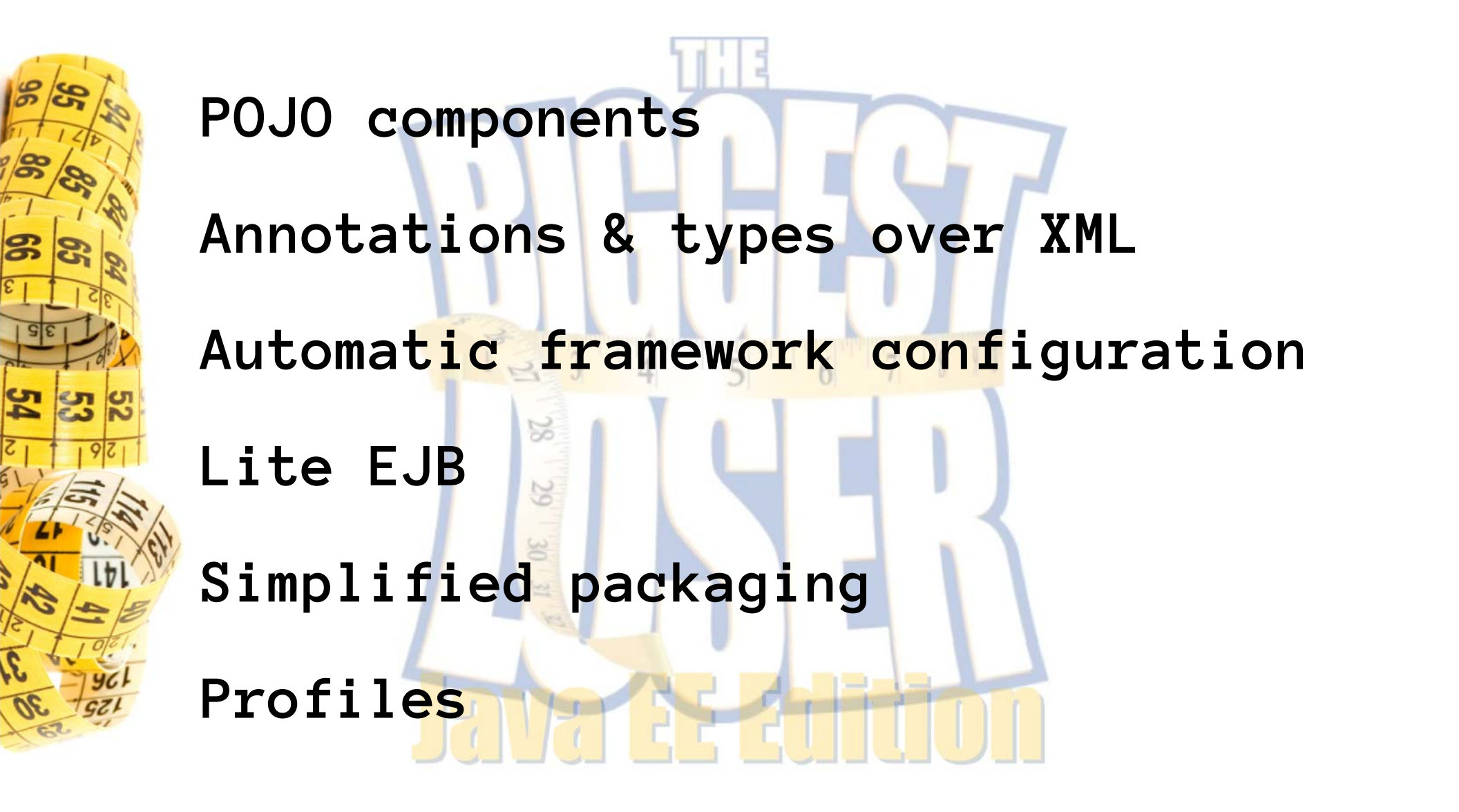
portable applications
and knowledge

**DOES THIS JAVA EE
MAKE ME LOOK
FAT?**



THE BIGGEST LOSER Java EE Edition

A yellow measuring tape is wrapped around the word 'LOSER' in the graphic. The tape is positioned horizontally across the middle of the word, with the numbers 1 through 9 visible on the top edge and 27 through 31 visible on the bottom edge. The tape is slightly curved, following the shape of the letters.



POJO components

Annotations & types over XML

Automatic framework configuration

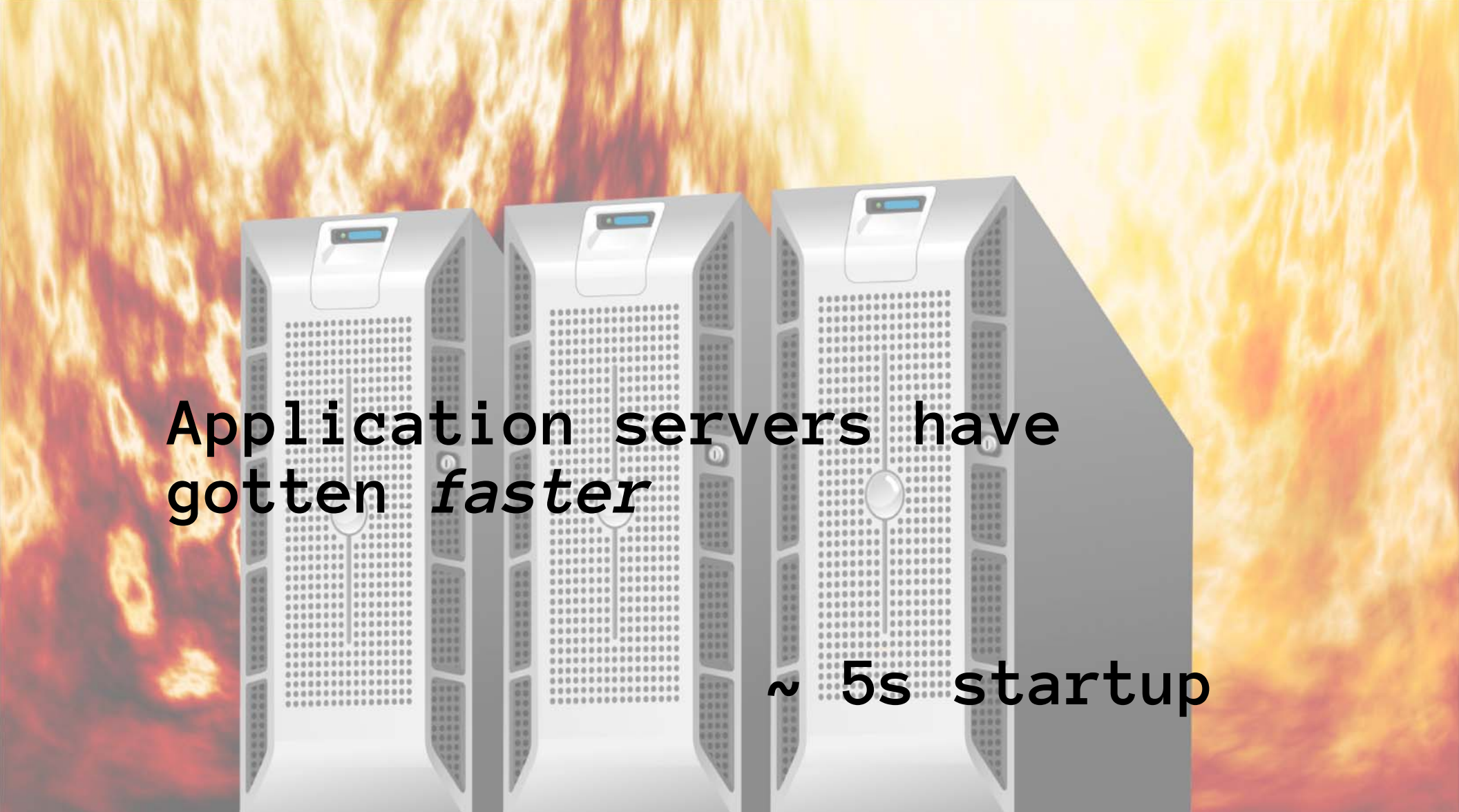
Lite EJB

Simplified packaging

Profiles



Blazing Servers

The image features three vertical server racks in the foreground, rendered in a light gray, metallic style. Each rack has a series of horizontal ventilation grilles and a small blue indicator light at the top. The background is a vibrant, abstract depiction of flames in shades of orange, yellow, and red, creating a sense of intense heat. Overlaid on the left side of the image is the text 'Application servers have gotten *faster*' in a bold, black, sans-serif font. On the right side, near the bottom, is the text '~ 5s startup' in the same font style.

Application servers have
gotten *faster*

~ 5s startup

Servlet container

(AFTER LOTS OF TWEAKING)

vs.

Java EE 6



Bean Validation

JSF 2

Servlet 3

EL

CDI

Managed Beans

EJB 3.1

JAX-RS

JMS

JPA 2

Web Profile





JPA & JAXB Entity

@Entity

@XmlRootElement

```
public Speaker implement Serializable {  
    private Long id;  
    private String name;
```

@Id @GeneratedValue

```
public Long getId() { return id; }  
public void setId(Long id) { this.id = id; }
```

```
public String getName() { return name; }  
public void setName(String name) { this.name = name; }  
}
```

Stateful Named Bean

@Stateful

@Model

public SpeakerController {

@Inject

private Speaker speaker;

@PersistenceContext

private EntityManager em;

public String add() {

em.persist(speaker);

return "speakers";

}

}

JAX-RS Resource

```
@Stateless
```

```
@Path("/speakers")
```

```
public class SpeakerResource {
```

```
    @PersistenceContext
```

```
    private EntityManager em;
```

```
    @GET
```

```
    public List<Speaker> getSpeakers() {
```

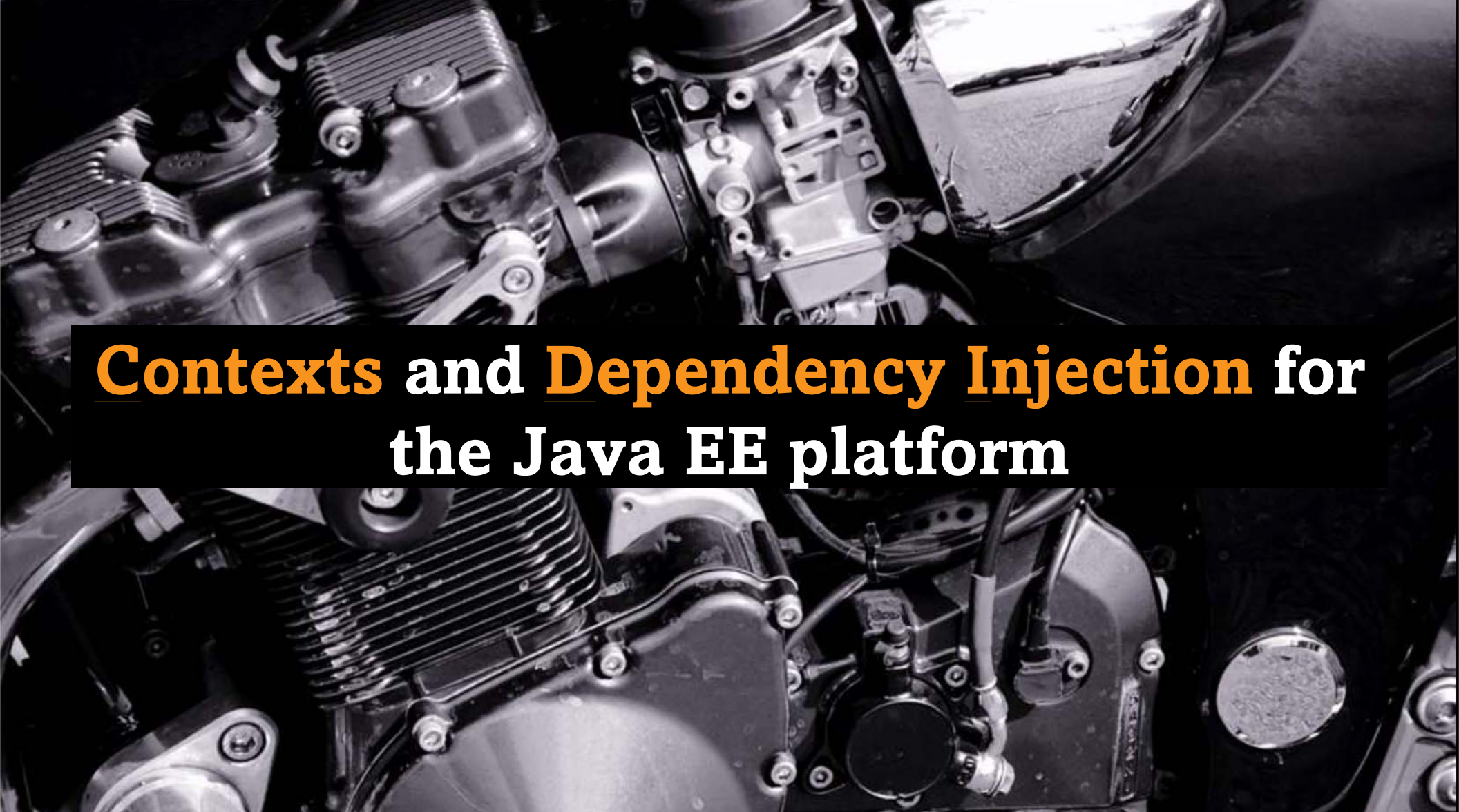
```
        return em.createQuery(
```

```
            "select s from Speaker s", Speaker.class)
```

```
            .getResultList();
```

```
    }
```

```
}
```

Contexts and Dependency Injection for the Java EE platform

Loose coupling

STRONG TYPING

INJECTION 101

```
public class StatusUpdater {  
    @Inject  
    private UrlShortener shortener;  
  
    public void post(String username, String status) {  
        String sStatus = shortener.shortenUrls(status);  
        System.out.println(username + " said " + sStatus);  
    }  
}
```



"What makes CDI
unique?"

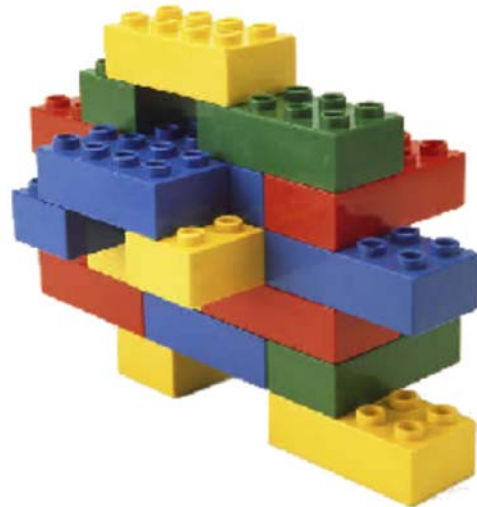
STANDARD
TYPE-SAFE
EXTENSIBLE

A brown and black portable equipment case, possibly for a video camera, is shown from a three-quarter perspective. The case has a black textured top and sides, with brown leather-like trim along the edges. It features a silver-colored metal handle on the front, a silver latch on the left side, and a silver latch on the front. The case is set against a plain white background.

Portable extensions

SPI for hacking Java EE:

- ✓ register additional beans
- ✓ satisfy injection points
- ✓ introduce custom scope with backing context
- ✓ augment or override bean metadata



Extensions



**"All we have to decide is
what to do with the beans
that are given to us."**



portable + flexible + extensible



=



Java EE is your oyster



**I need one of those cloud thing-a-
ma-jigs.
And I need it in 10 minutes for a
marketing meeting.**



Startup cost
Gotchas
Integration
Details

Forge

Carve out a project,
work it into shape



Let 's get started

```
Terminal
[dallen@saguaro:jdk6 demos]$ ~/opt/forg/bin/forg
Using Forge at /home/dallen/opt/forg

Seam Forge >>

[no project] demos $ new-project --named conftrack --topLevelPackage org.conftrack
Use [/home/dallen/demos/conftrack] as project directory? [Y/n]
Wrote /home/dallen/demos/conftrack/src/main/resources/META-INF/forg.xml
***SUCCESS*** Created project [conftrack] in new working directory [/home/dallen/demos/conftrack]
[conftrack] conftrack $
```

0 to Java EE!

**WHAT JUST
HAPPENED?**





**WE CREATED A
WEBAPP, YAY!**



**FORGE DIDN'T
LEAVE \$#@! IN
YOUR PROJECT!**

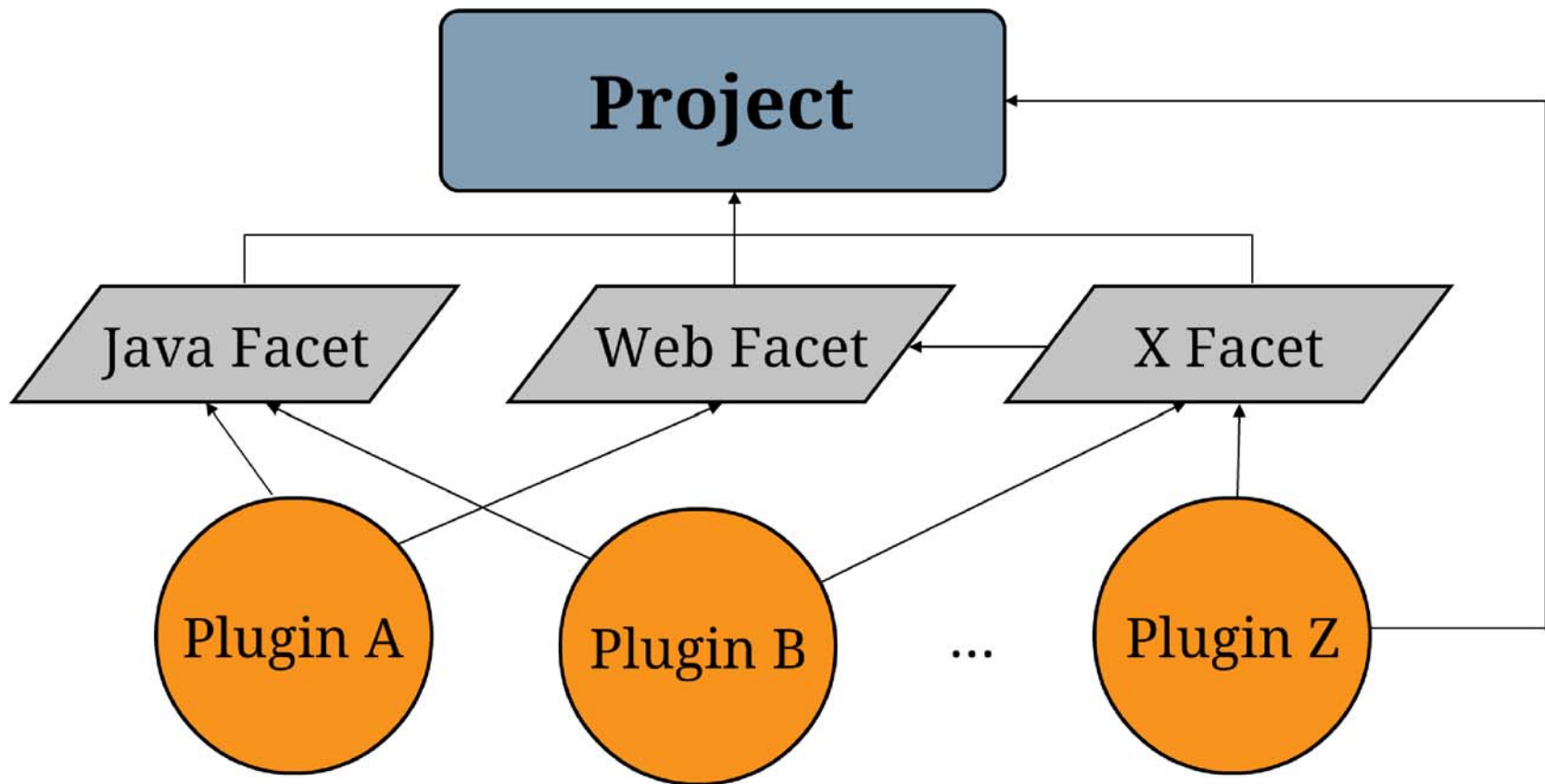


**WE USED
PLUGINS!**



**REMIND ME.
WHAT'S A PLUGIN
AGAIN?**





Gets you started quickly



Takes care of the gotchas



**Adds/activates technologies via
plugins**



Handles details, gives you perspective



A **powerful** programming model.

Less code, **portability**.

Optimized for productivity & automation.

CDI and **Forge** plugins.

Absolutely. **You saw it** first hand.



Java Easy & Extensible

Download a Java EE 6 container:

JBoss AS 6 & 7 - <http://jboss.org/jbossas>

GlassFish 3.1 - <http://glassfish.org>

Create a Java EE 6 project:

Archetypes - <http://tinyurl.com/gojavaee>

Use tools, feel hacker:

Forge - <https://github.com/forge>

JBoss Tools - <http://jboss.org/tools>

Arquillian - <http://jboss.org/arquillian>

Add a Seam 3 module:

<http://seamframework.org/Seam3>

