# EE 6 Programming

## *Immediate Productivity*

Pete Muir Dan Allen

JBoss Community

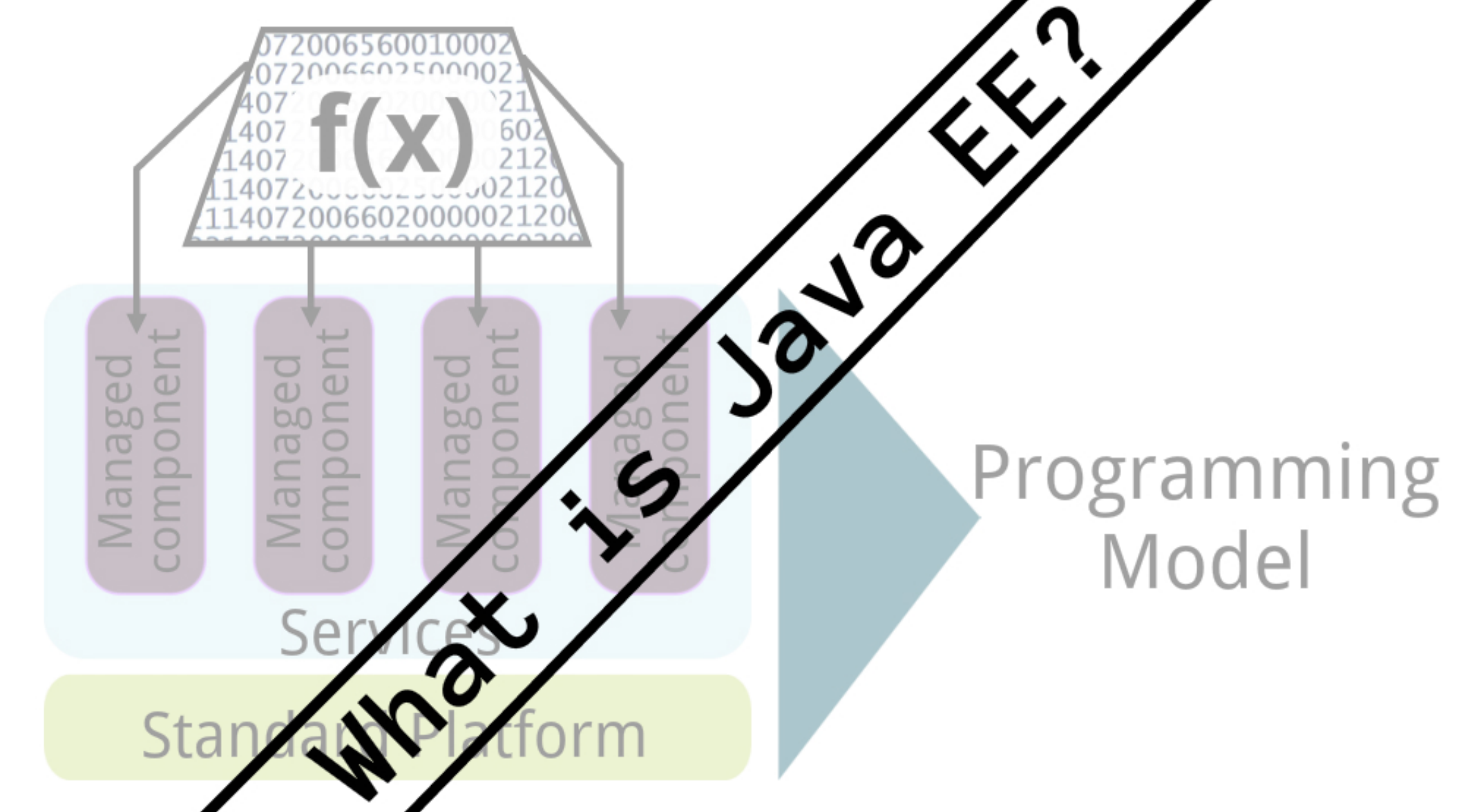
What is Java EE?

What benefits does it offer?

What improvements have been made?

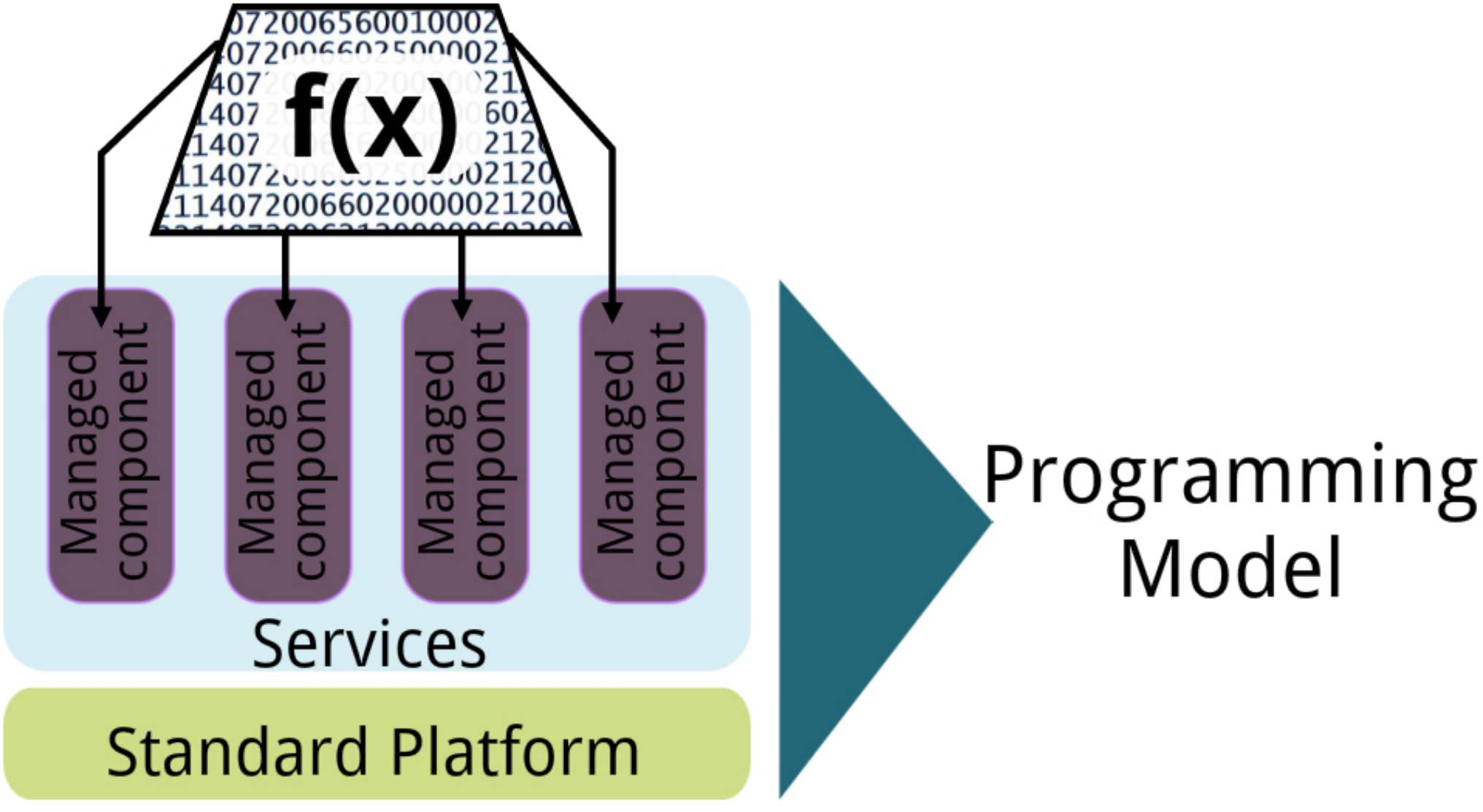
What makes it extensible?

Is it really easy?


f(x)
Managed component
Managed component
Managed component
Managed component
Services
Standard Platform
What is Java EE?
Programming Model

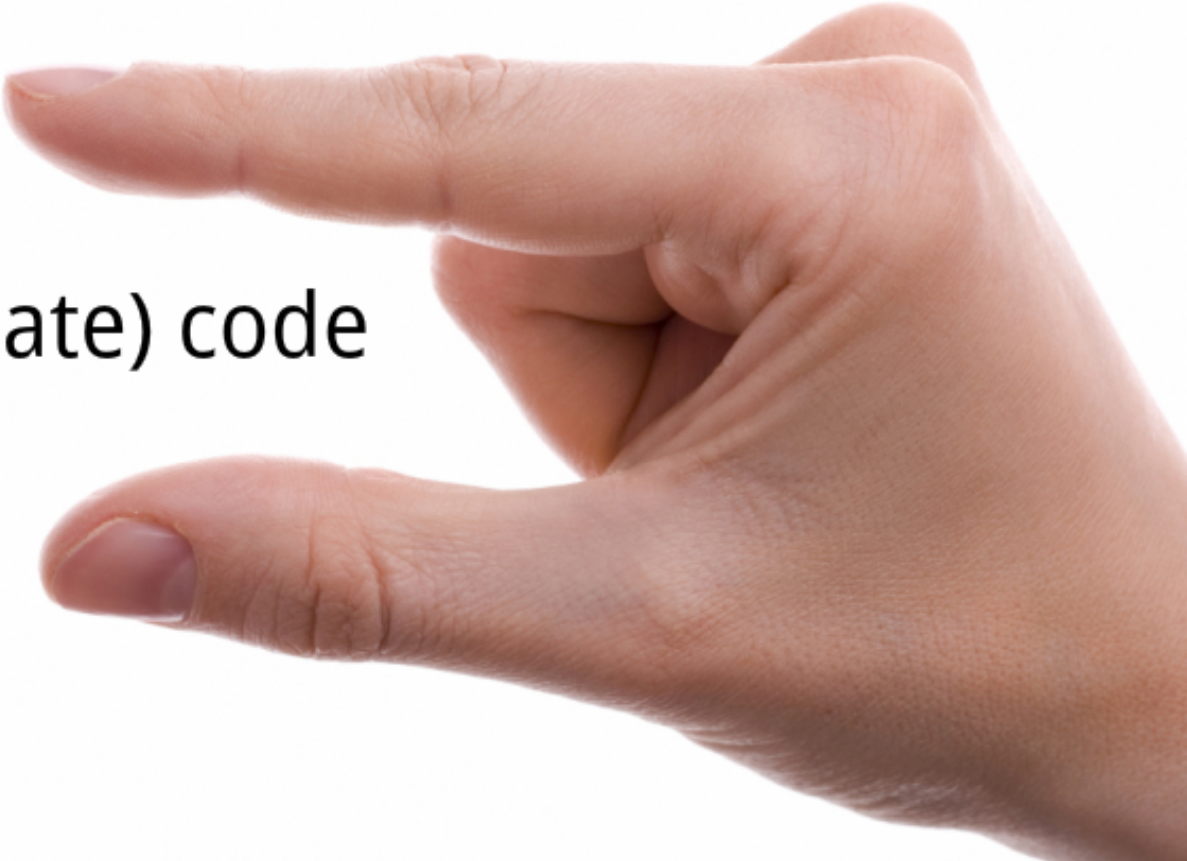
# Enterprise application development

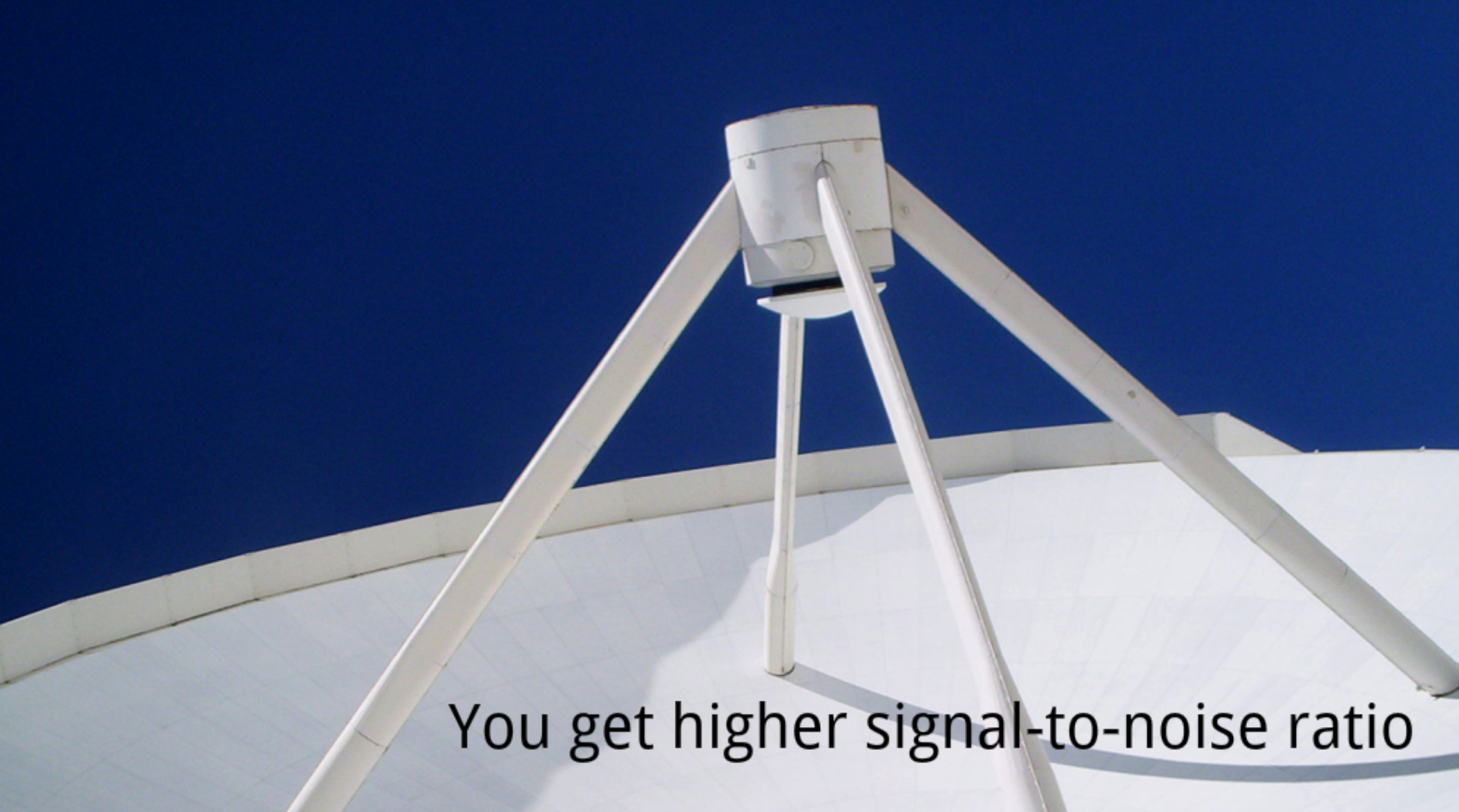

HUGE!
It's gonna be big...


f(x)
Managed component
Managed component
Managed component
Managed component
Services
Standard Platform
Programming Model


Encapsulate
business logic
in standard Java EE components

You get less (boilerplate) code




You get higher signal-to-noise ratio

You get powerful mechanisms for free!

You get portable
applications & knowledge


DOES THIS JAVA EE MAKE ME LOOK FAT?
THIS IS A TRICK QUESTION, ISN'T IT?


THE
BIGGEST
LOSER
Java EE Edition

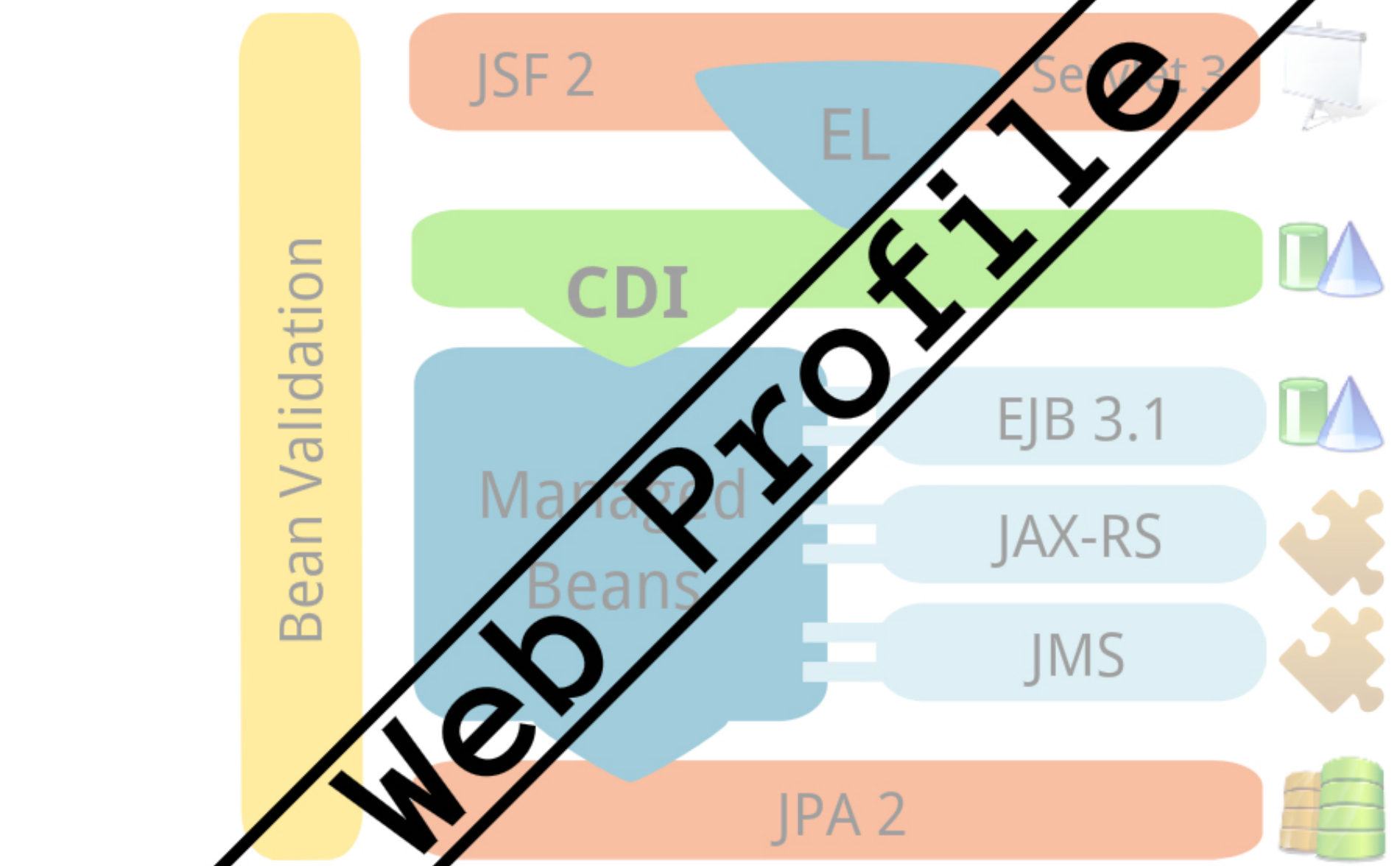
- POJO components
- Annotations & types over XML
- Automatic framework configuration
- Lite EJB
- Simplified packaging
- Profiles

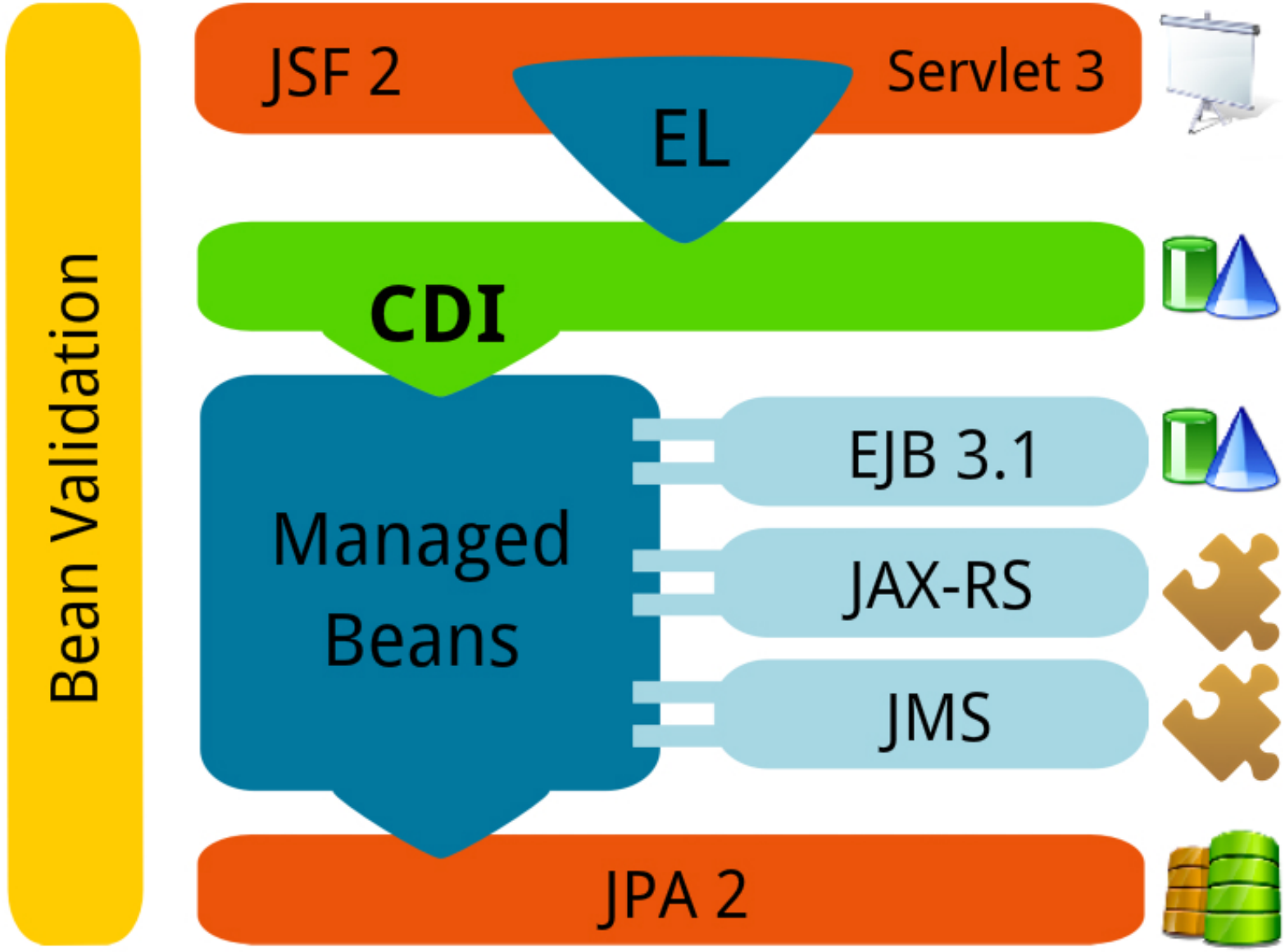

Blazing Servers

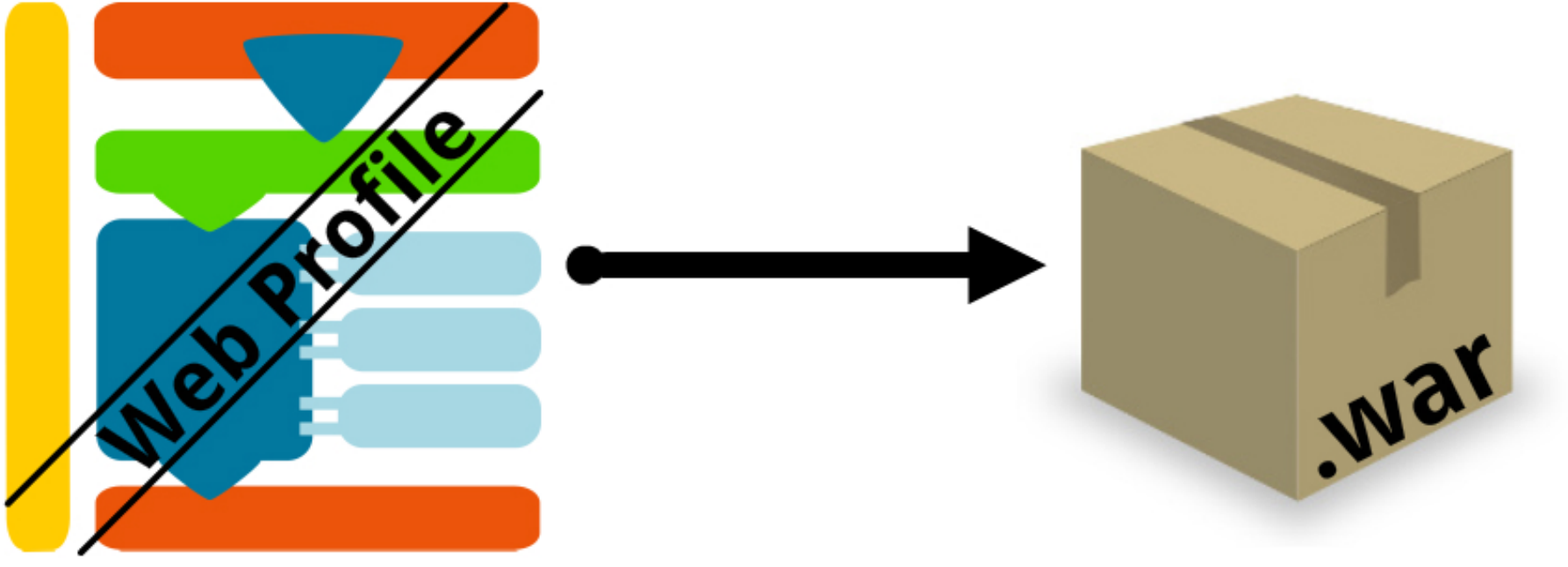

How does 3 seconds to startup sound?

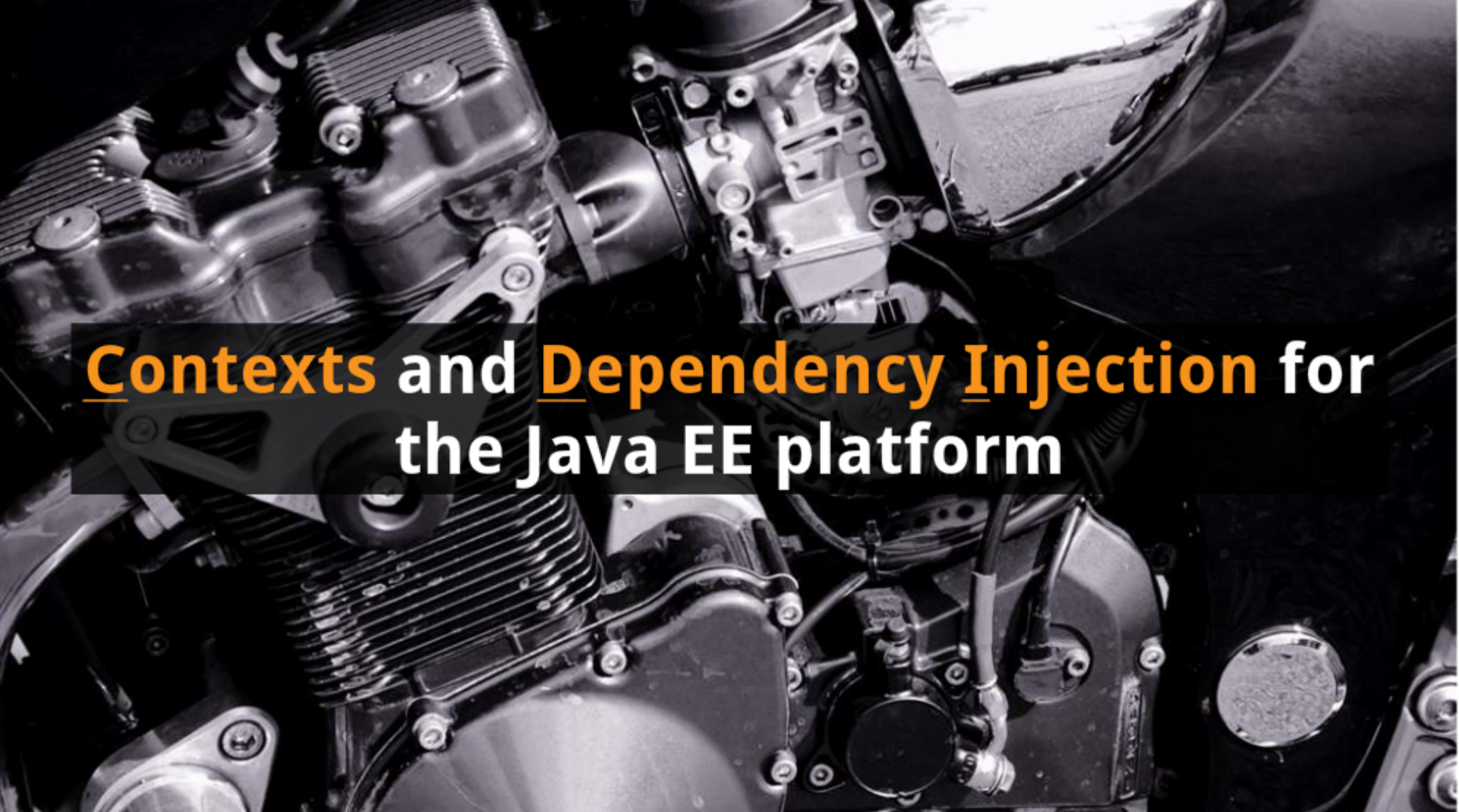
# Servlet container vs. Java EE 6

(AFTER LOTS OF TWEAKING)


Bean Validation
JSF 2
Servlet 3
EL
CDI
Managed Beans
EJB 3.1
JAX-RS
JMS
JPA 2
Web Profile


Bean Validation
JSF 2
Servlet 3
EL
CDI
Managed Beans
EJB 3.1
JAX-RS
JMS
JPA 2


Web Profile
.war


Contexts and Dependency Injection for
the Java EE platform

*Loose coupling*

**STRONG TYPING**


@Path
@Observes
@Entity
@GET
@SessionScoped
@TransactionAttibute
@Stateless
@RequestScoped
@ApplicationScoped
@PersistenceContext
@Qualifer
@Named
@Inject
@Stateful
@PostConstruct
@Produces

# 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);
    }
}
```

# JPA & JAXB Entity w/ constraints

```
@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; }

    @NotNull @Size(min = "3", max = "50")
    public String getName() { return name; }
    public void setName(String name) { this.name = name; }
}
```

# Stateful Named Bean

```
@Stateful
@Model
public SpeakerController {
   @Inject @Current
   private Speaker speaker;

   @PersistenceContext
   private EntityManager em;

   public String add() {
      em.persist(speaker);
      return "speakers";
   }
}
```

# JAX-RS Resource w/ JPQL

```
@Stateless
@Path("/speakers")
public class SpeakerResource {
    @PersistenceContext
    private EntityManager em;

    @GET
    public List<Speaker> getSpeakers() {
        return em.createQuery(
            "select s from Speaker s order by s.name asc", Speaker.class)
            .getResultList();
    }
}
```

# JAX-RS Resource w/ JPA criteria

```
@Stateless
@Path("/speakers")
public class SpeakerResource {
    ...

    @GET
    public List<Speaker> getSpeakers() {
        CriteriaBuilder cb = em.getCriteriaBuilder();
        CriteriaQuery<Speaker> cq = cb.createQuery();
        Root<Speaker> speaker = cq.from(Speaker.class);
        return em.createQuery(cq.select(speaker)
                .orderBy(cb.asc(speaker.get(Speaker_.name))));
    }
}
```

# JAX-RS Resource w/ injection

```
@Stateless
@Path("/speakers")
public class SpeakerResource {
    @Inject
    private SpeakerRepository dao;

    @GET
    public List<Speaker> getSpeakers() {
        return dao.findAll();
    }
}
```


Portable
extensions

# SPI for hacking Java EE:

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

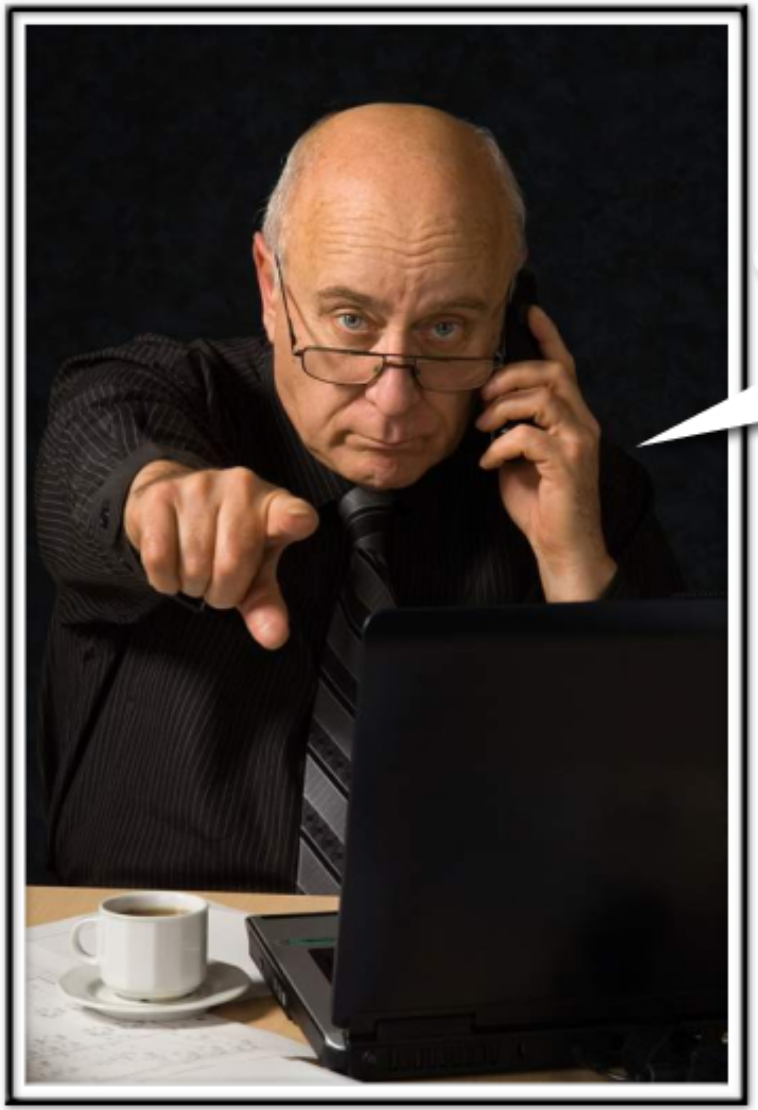
# Extensions

**"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 to demo it to a customer in 10 minutes.



Let's get started

# Fire up the demo

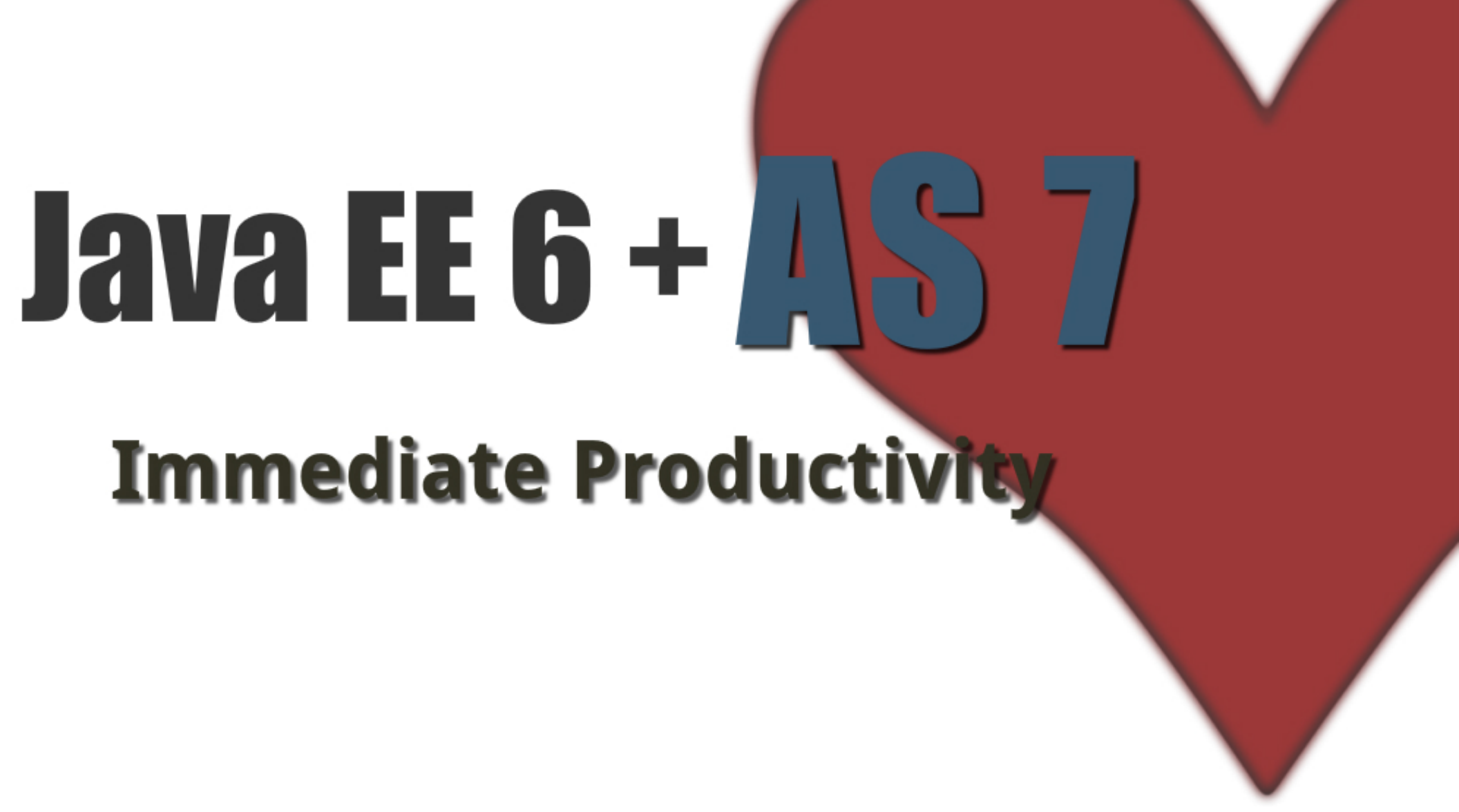
A powerful programming model.

Less code, portability.

Optimized for productivity & automation.

CDI extensions.

Absolutely. You saw it first hand.


Java EE 6 + AS 7
Immediate Productivity

## Download a Java EE 6 container:

JBoss AS 7 - http://jboss.org/as7

## Create a Java EE 6 project:

Quickstart guide - http://bit.ly/as7tutorials

## Use tools, feel hacker:

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

Seam Forge - http://bit.ly/seamforge

## Watch for AS 7 news:

Twitter hashtag: #JBossAS7

Follow: @JBossNews

# Q & A Live!

**Pete Muir**
@plmuir
http://in.relation.to/Bloggers/Pete

**Dan Allen**
@mojavelinux
http://community.jboss.org/people/dan.j.allen/blog