

JUDCon

JBoss Users & Developers Conference

2012:India

Java EE on Google App Engine: CDI to the rescue!

Aleš Justin
JBoss by Red Hat

Agenda

- What is GAE and CDI?
- Why GAE and CDI?
- Running JavaEE on GAE
 - Other JavaEE technologies
 - Development vs. Production
 - Problems vs. Solutions
 - Testing
- CapeDwarf Project

About me

- Old school JBoss-ian
 - Microcontainer co-author
 - Weld lead
 - CapeDwarf founder
 - OpenShift “guinea pig”
- Eager PaaS user
 - While still GAE amateur

GAE

- Forked Jetty
- Many (logical) restrictions
- RDBMS → BigTable
- Simple Map(Reduce)
- Good administration
- (Less) Free (very) basic usage

CDI

- Context and Dependency Injection
- JSR-299 → JBoss Weld RI
- “type-safe” programming
- “contexts” concept
- Easy to extend: “Extensions”

Why GAE and CDI?

- Simple reason: sandbox app
- Android app with server back-end
 - Tattletale (Tožibaba)
- Zero administration / configuration
- Reliable up-time
- No more “boiler-plate” code → CDI
 - Rescue #1

CDI #2

- @Inject
- @Qualifier
- @Named
- @Scope
- Interceptors
- InjectionPoint
- Instance

- Plenty of CDI ...

- @Produces
- @Observes
- @ConversationS.
- @Decorator
- @Alternative
- @Sterotype

Other technologies

- JPA
- JSF2
- BeanValidation
- javax.cache
- JSON
- ShrinkWrap + Arquillian

Project Kick-Off

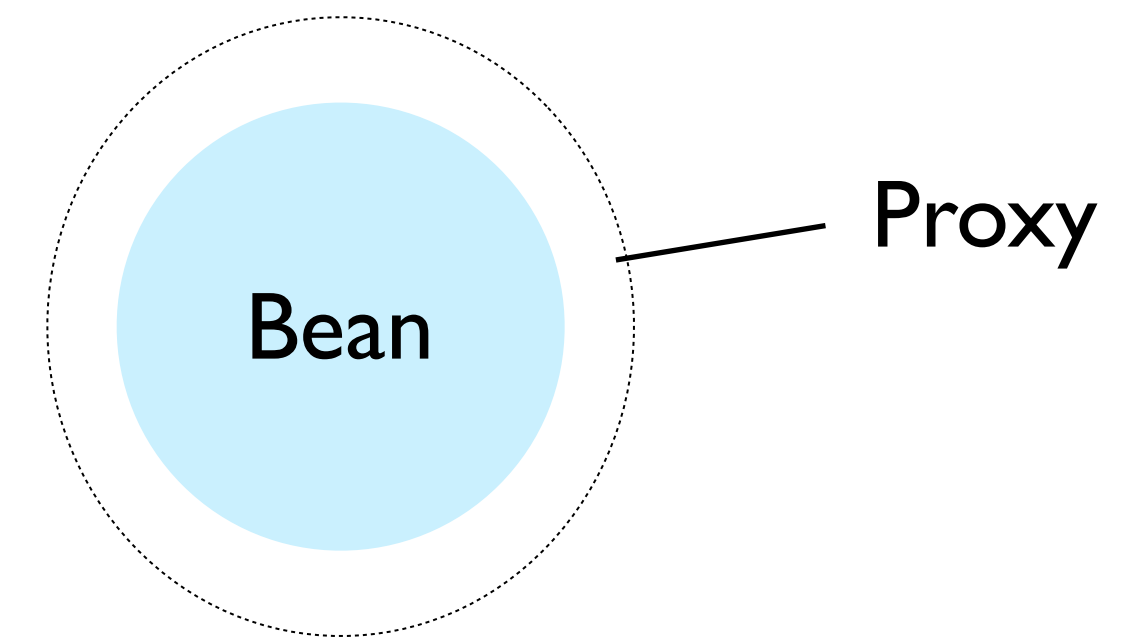
- Know (all) restrictions!
 - 30sec, white-list, no threads, ...
- How portable should my app be?
- Development vs. Production Env!
- Frameworks have problems
 - But they mostly have (easy) solutions → fork?

The Goal

- Abstract away GAE API
 - Pure JavaEE
 - Port app to JBossAS (6 & 7), GlassFish, ...
- Use CDI as much as possible
- Test it all, same as other JavaEE apps

Initial problems

- CDI
 - “Entry” mechanism
 - **WeldServlet**
 - Request / response binding
 - **WeldFilter**
 - Lack of “lifecycle” handling
 - **EMF**



Initial problems

- JPA
 - Limited relationships (Entity groups)
 - Proxying layer vs. Custom “ORM”
 - **ProxyingEntityManager**
 - Transaction per “Entity group”
 - Adding XG just recently (cross-group)
 - Old DataNucleus version
 - 3.0 Beta testing?
 - Long vs. Key entity key
 - “Enhanced” entities

Initial problems

- JSF2
 - Context parameter settings
 - “new InitialContext” patch
 - “built-in” EL 1.0 → workaround

Initial problems

- BeanValidation
 - Static usage with JPA
 - **ValidatorListener**
 - Size
 - Configuration

Initial problems

- javax.cache (GCache)
 - No partition by default → Namespaces!
 - NUCCORE-540 → fork → fixed in 3.0
- JSON
 - Different versions; client vs. server
- Arquillian
 - Missing GAE container

“Production” problems

- CDI
 - “Bean scanning” → exact listing
 - **bean-classes.txt**
 - “Lazy beans” → MVC re-write
 - GZIP + Base64 usage
 - **GzipFilter**
 - “Fall-back / repeat” mechanism
 - **ServerProxyHandler**

“Production” problems

- JPA
 - Explicit entity listing
 - Maximize cache usage
 - Be aware of the limits, hollow entities (GAE-JDO 1.0.9 ?)
 - **TimestampedListKeyStrategy**
 - “Caching” → Lazy EMF
 - **LazyEntityManagerFactory**
- javax.cache
 - serialVersionUID
 - Non-primitive variables
 - Cache evict / clear mechanism
 - **CacheCommand**

Testing

- ShrinkWrap
 - Binaries abstraction
- Arquillian
 - Runtime / Env abstraction
 - <https://github.com/alesj/arquillian-container-gae>
- “Testable development ...”

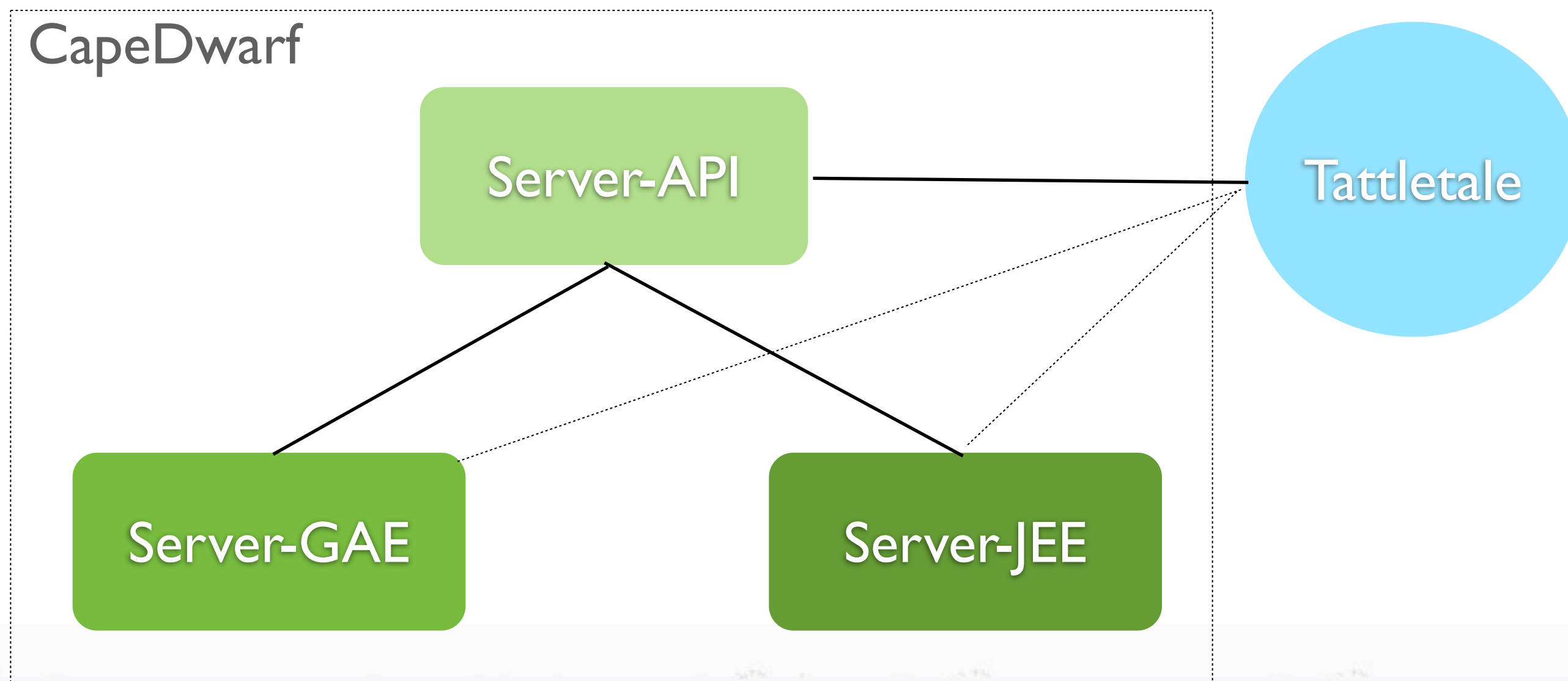
CapeDwarf Project



- Two colors / flavors
 - Green
 - Simple GAE to JavaEE abstraction
 - Android to server-side layer
 - Project's modules
 - Blue – previous talk :-)
 - JBoss API on top of GAE API
- <https://github.com/capedwarf>

Tattletale

- Choosing the right implementation based on the runtime environment
 - Rescue #2



Using CapeDwarf Green

- Integrated & tested
 - GAE, JBossAS 6&7, GlassFish3.1
- Important differences
 - Libraries usage
- Configuration ↗
- Bugs, ...
 - @Entity on @MappedSuperclass
 - **TimestampedEntity**
 - Enhanced entities + JDO

Q&A

- ales.justin@jboss.org

