

JUDCon

JBoss Users & Developers Conference

2012:India

Android and the Cloud: A perfect fit!

Aleš Justin
JBoss by Red Hat

Agenda

- Background of this talk
- Project modularization
- Tips and tricks
- Even simpler approach to client-server architecture
- Q&A

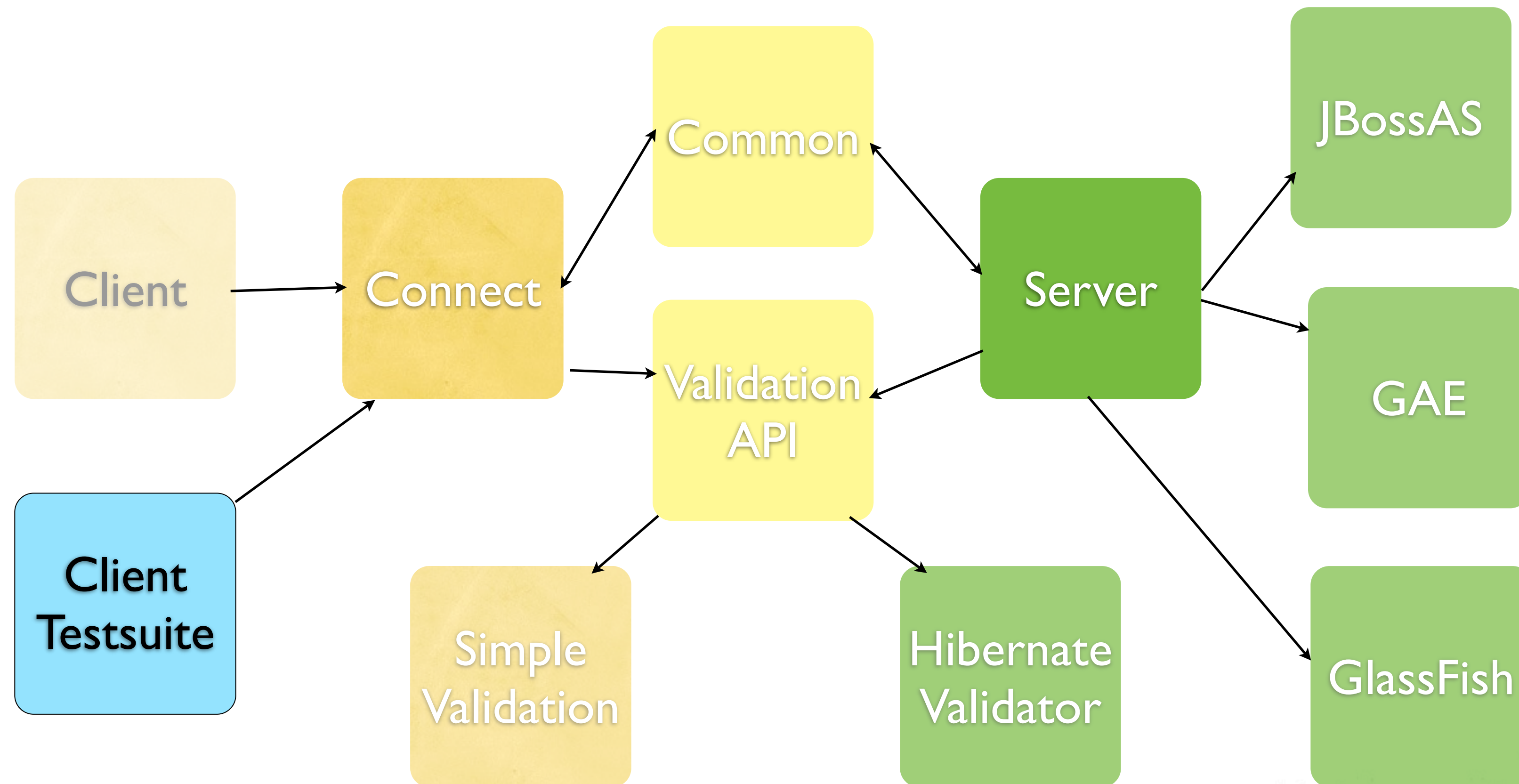
About me

- Old school JBoss-ian
 - Microcontainer co-author
 - Weld project lead
 - CapeDwarf founder
 - Ceylon core developer
 - OpenShift “guinea pig”

Background

- Buzzing ...
 - Android
 - prototype application with server-side backend (X)
 - Google App Engine
 - JBossAS7 / OpenShift
- Simple client-to-server communication
 - JSON
 - Apache HttpComponents
- Scalable (and portable) server-side
 - **CapeDwarf Green**
- Let's make it even simpler -> LeanEngine

Project modularization



Project modularization

- Common
 - Data (DTOs)
 - Environment abstraction
 - Android vs. testsuite vs. server-side
 - Serialization logic
 - Data annotations and utils
- Connect
 - ServerProxy handler factory
 - Feed an interface to the factory == Communication API
 - Primitive, JSON, streaming

Project modularization

- Validation API
 - JSR-303 based
 - Client vs. server-side impl
 - Simple impl on client, full Hibernate Validator on server-side
- Client-testsuite
 - Mock all app interaction
 - No need for Android emulator, etc
- Server
 - Using Java EE; CDI, JPA, JTA, ...
 - Abstracting away custom API
 - CapeDwarf Project

Tips and tricks

- (X) more detailed server “tricks” at my JavaEE+GAE talk
- Let’s go back to previous slides with more details
 - custom fine-grained server-side caching
 - simple SQLite ORM
- Running the tests
 - Diff server-side environments == diff configuration
 - App servers have bins already there
 - Web servers need those bins bundled
 - Bugz ...

Even simpler, leaner

- Lean == minimum “investment”
 - Into infrastructure
 - Into architecture
- Lean-Engine
 - Open Source: www.lean-engine.com

LeanEngine

- Familiar with GAE and its DatastoreService?
 - It takes generic Entity
 - Which takes generic properties
- Client side
 - Mocks this DatastoreService
 - But it does it remote -> via REST / JSON
 - Google, Facebook, OpenID auth
- Server side -> simply pushes this into DatastoreService
 - Thin REST layer, delegating to DatastoreService
 - Zero code needed, simply deploy a prepared .war
 - It has a simple datastore UI atm

LeanEngine cont.

- Use case?
 - Really simple sync app
 - But more complex logic cloud be easily added
- TODOs
 - Data sharing
 - Custom groups vs. Google+ or Facebook friends?
 - Server-side caching
 - Objectify persistence
- Stuck on GAE?
 - Not really, if you use CapeDwarf Blue ;-)

Demo

- CapeDwarf -> TattleTale
- LeanEngine

Q&A

- ales.justin@jboss.org