



OSGifying the repository

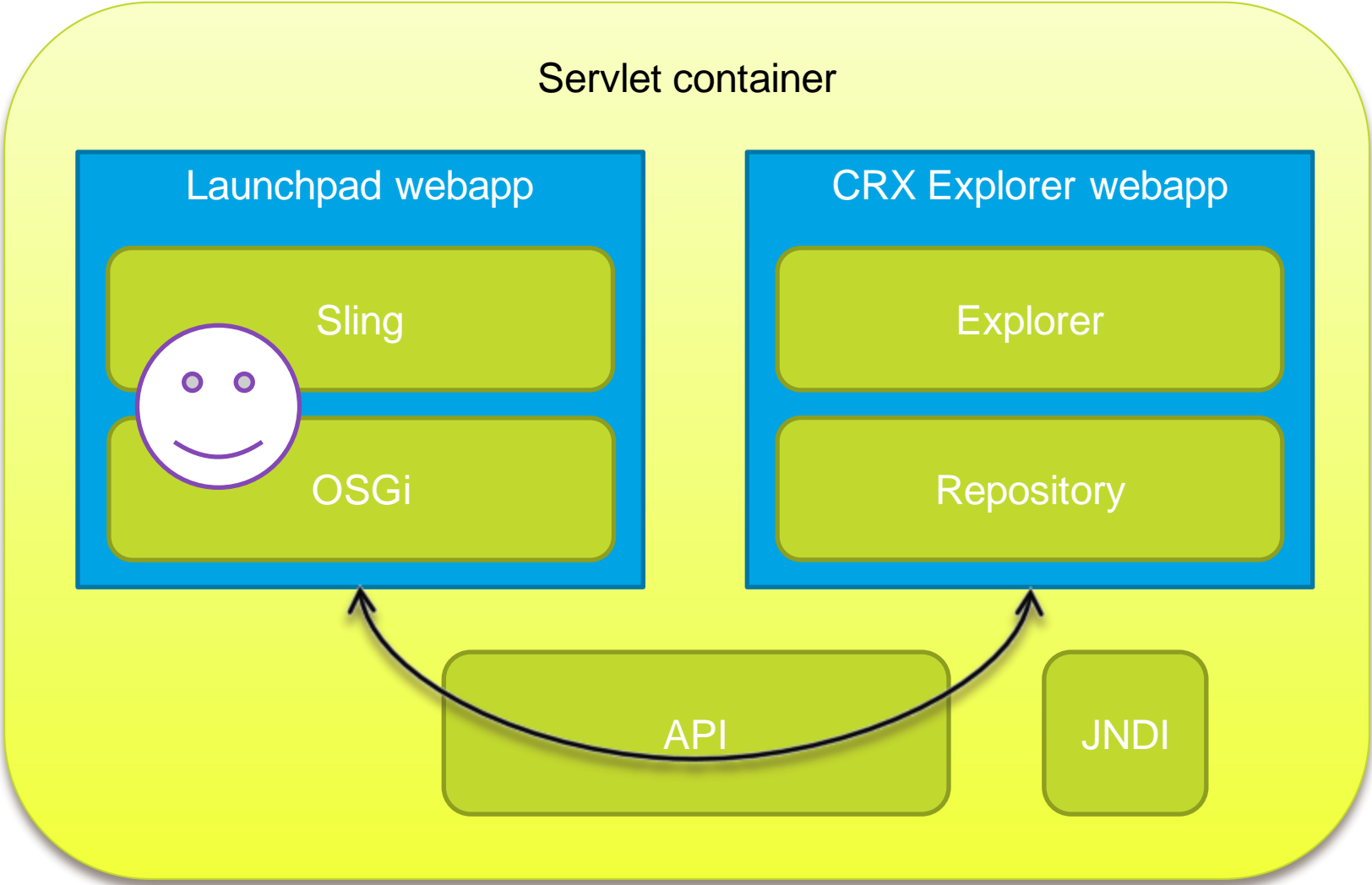
Jukka Zitting | Senior Developer



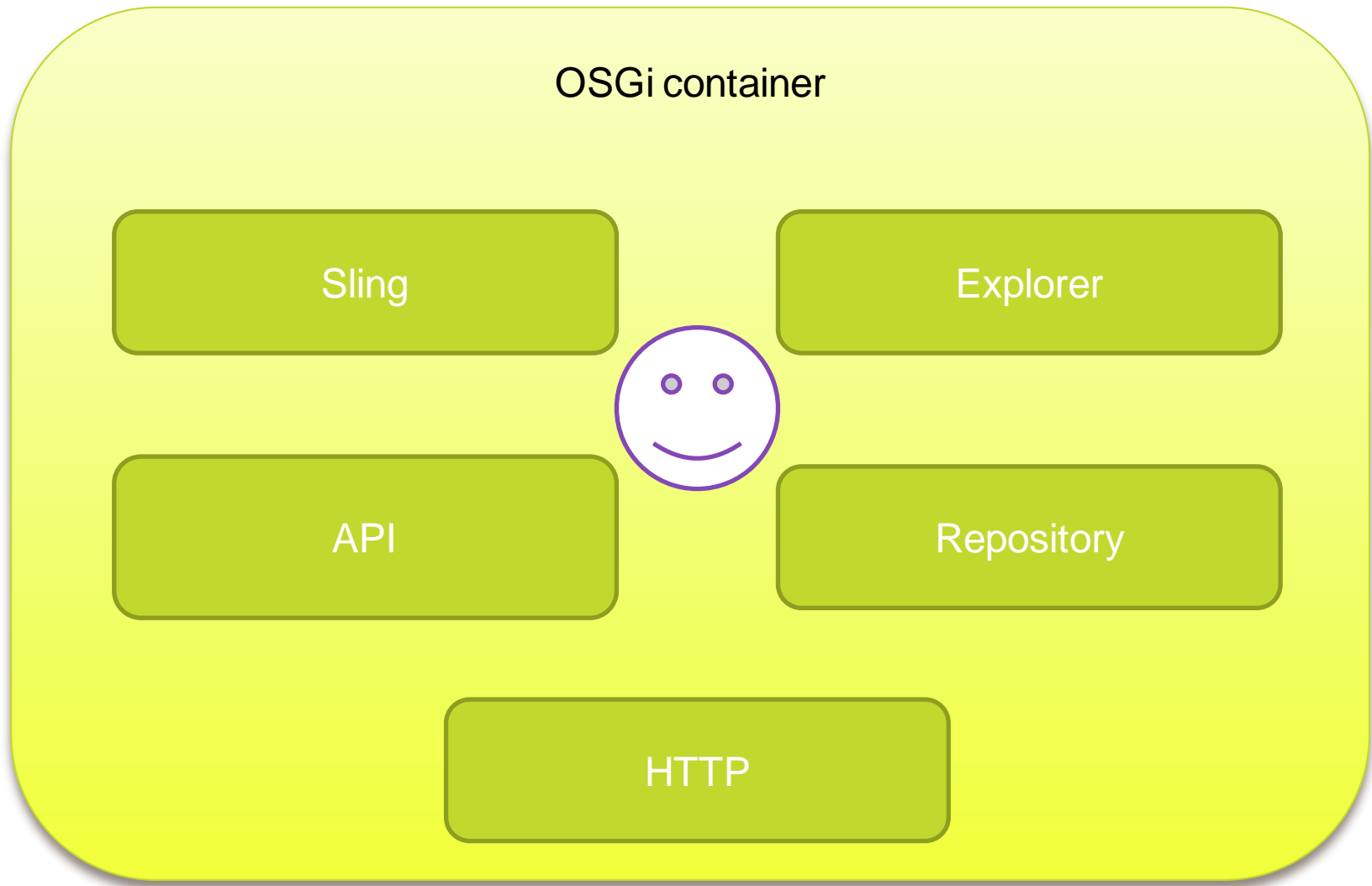
Agenda

- CRX platform architecture
- Granite platform architecture
- The repository bundle
- JMX support
- Ideas for the future

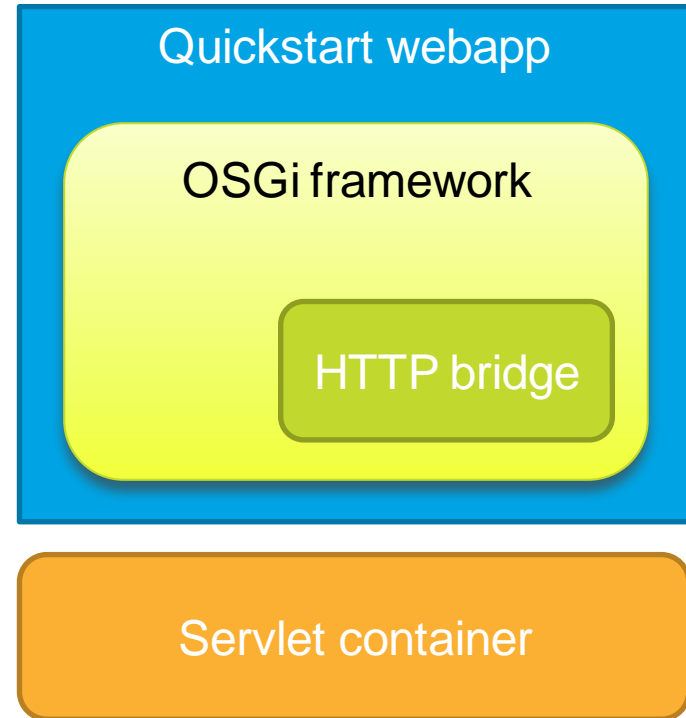
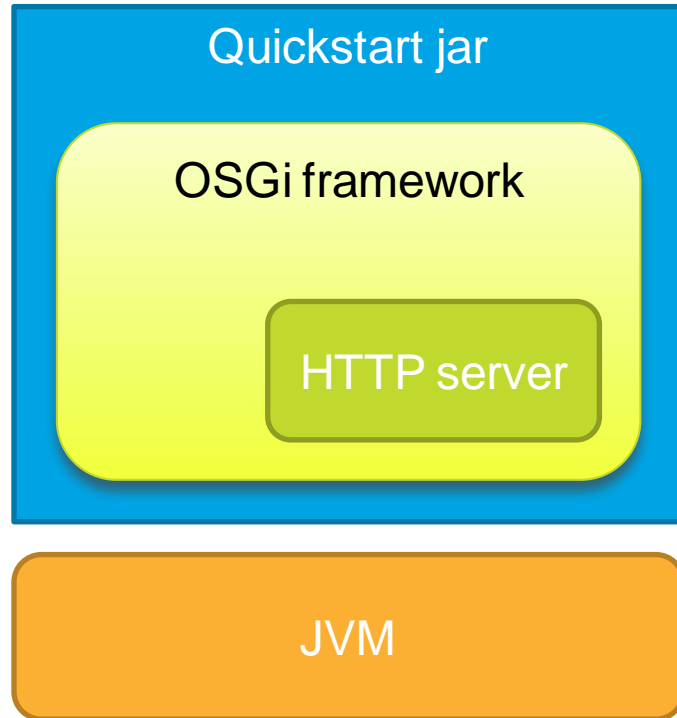
CRX Platform Architecture



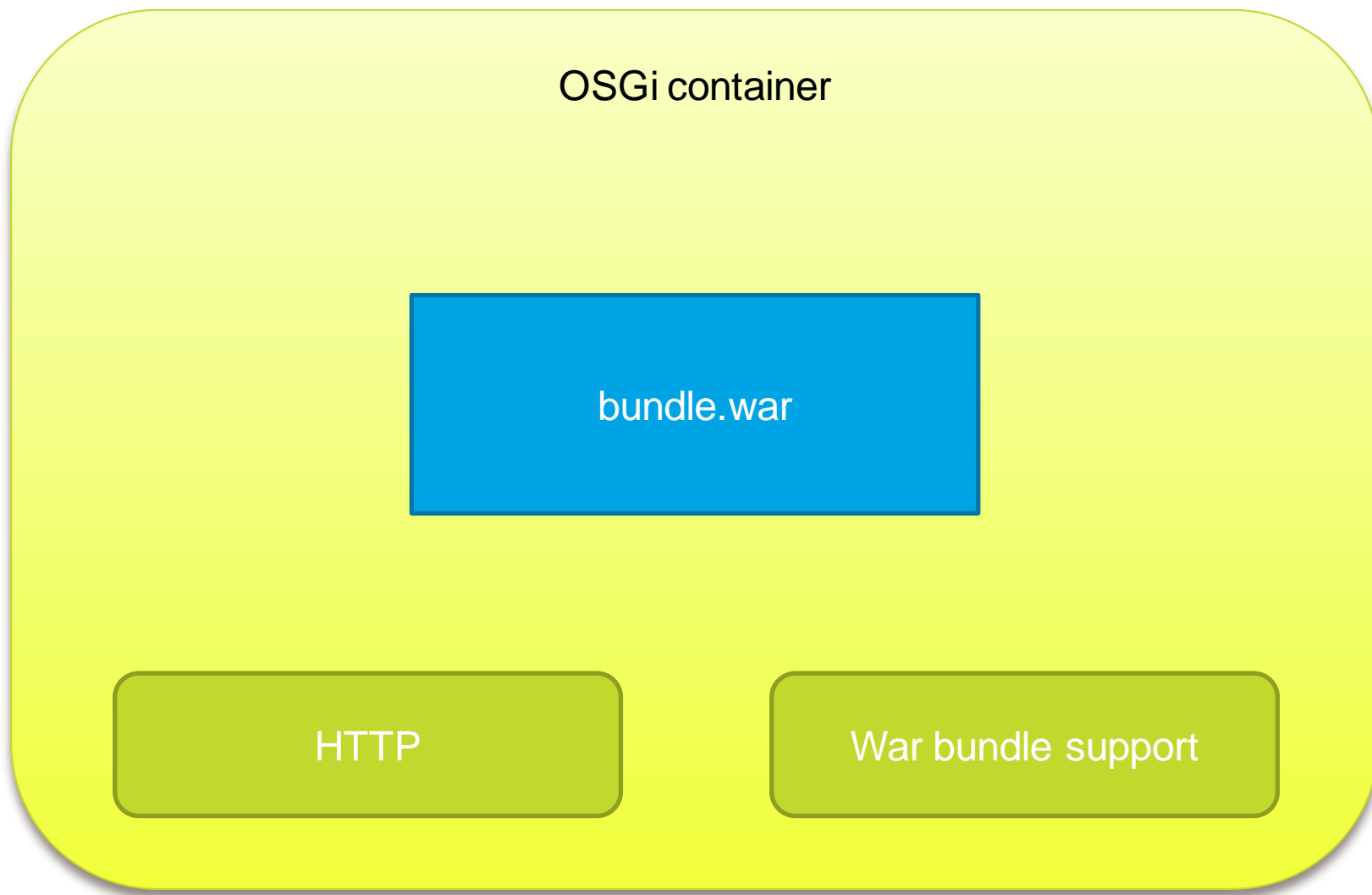
Granite Platform Architecture



Deploy anywhere



What about client applications in separate webapps?



The repository bundle

- Bundle that implementation of the JCR API
 - Plus the Jackrabbit and CRX API extensions
- Deploy to an OSGi container and you have a working repository
 - The running repository exposed as an OSGi service
- Just another packaging format from the repository perspective
 - Jackrabbit comes as a standalone jar, a webapp, a JCR archive, and now an OSGi bundle
 - In CRX we're dropping the Explorer webapp and only shipping the repository bundle as a part of the Granite platform
- Backwards compatibility with existing repositories
 - OSGi configuration just points to the repository directory and an existing repository.xml configuration file
 - Custom repository extensions may need to be slightly adjusted

- JMX: Java Management Extensions
- A bit of history
 - Repository diagnostics and maintenance tools as a more or less ad-hoc collection JSP pages in the CRX Explorer webapp
 - No documented API nor even a consistent look and feel
 - Should we simply port the JSPs to web console plugins?
- Better idea: implement equivalent functionality as JMX Mbeans
 - Java API for accessing diagnostics and maintenance features
 - Tooling support for pretty much any network management/monitoring client
- Also implemented a generic JMX web console plugin to expose these (and other!) features over the web
 - You can even use curl as a JMX client

JMX repository diagnostics

- JMX is ideal for exposing various repository performance metrics
- Coming up:
 - Number of open sessions
 - Read/Write/Query operations per second, average time per operations
 - Number of bundles read/written per second, average time per operation
 - Cache utilization metrics
 - Analysis of worst performing queries

Ideas for the future

- Extension points in the repository
 - Query & indexing
 - Authentication & access control
 - Triggers & stored procedures
- Breaking up the repository to smaller modules
 - Easier upgrades/customizations
 - More room for innovation
- Hot upgrade



Adobe