Rapid JCR applications development with Apache Sling

Bertrand Delacrétaz, Senior R&D Developer, Day Software
bdelacretaz@apache.org - grep.codeconsult.ch

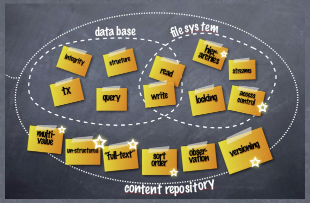
ApacheCon US 2008

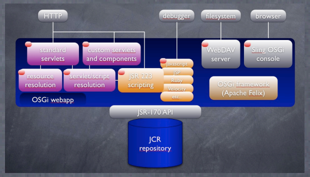
code is there

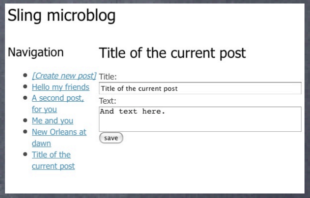
Don't miss Jukka on JCR, Thursday!

Slides revision: 2008-11-05
Slides theme design: David Nuescheler

is this slide busy enough?






Sling microblog

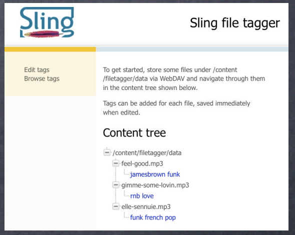
Navigation

- [Create new post]
- Hello my friends
- A second post, for you
- Me and you
- New Orleans at dawn
- Title of the current post

Title of the current post

Title: Title of the current post

Text: And text here.


Sling file tagger

To get started, store some files under /content/filetagger/data via WebDAV and navigate through them in the content tree shown below.

Tags can be added for each file, saved immediately when edited.

Content tree

```
/content/filetagger/data
  feel-good.mp3
    jamesbrown funk
  gimme-some-lovin.mp3
    rnb love
  elle-sennuie.mp3
    funk french pop
```

Intro to JCR

JCR = Java Content Repository API

JSR-170 / JSR-283

Everything Is Content - and JCR manages it as trees of Nodes and Properties, using rich data types.

Don't miss Jukka on JCR, Thursday!

What's JCR?

JSR-170

Content Repository for
JavaTM technology API

Spec-Lead:
Day Software

Status:
Final Release 17-jun-2005

Expert Group:

What's JCR?

JSR-170 | **JSR-283**

Content Repository for JavaTM technology API v2.0

Spec-Lead:
Day Software

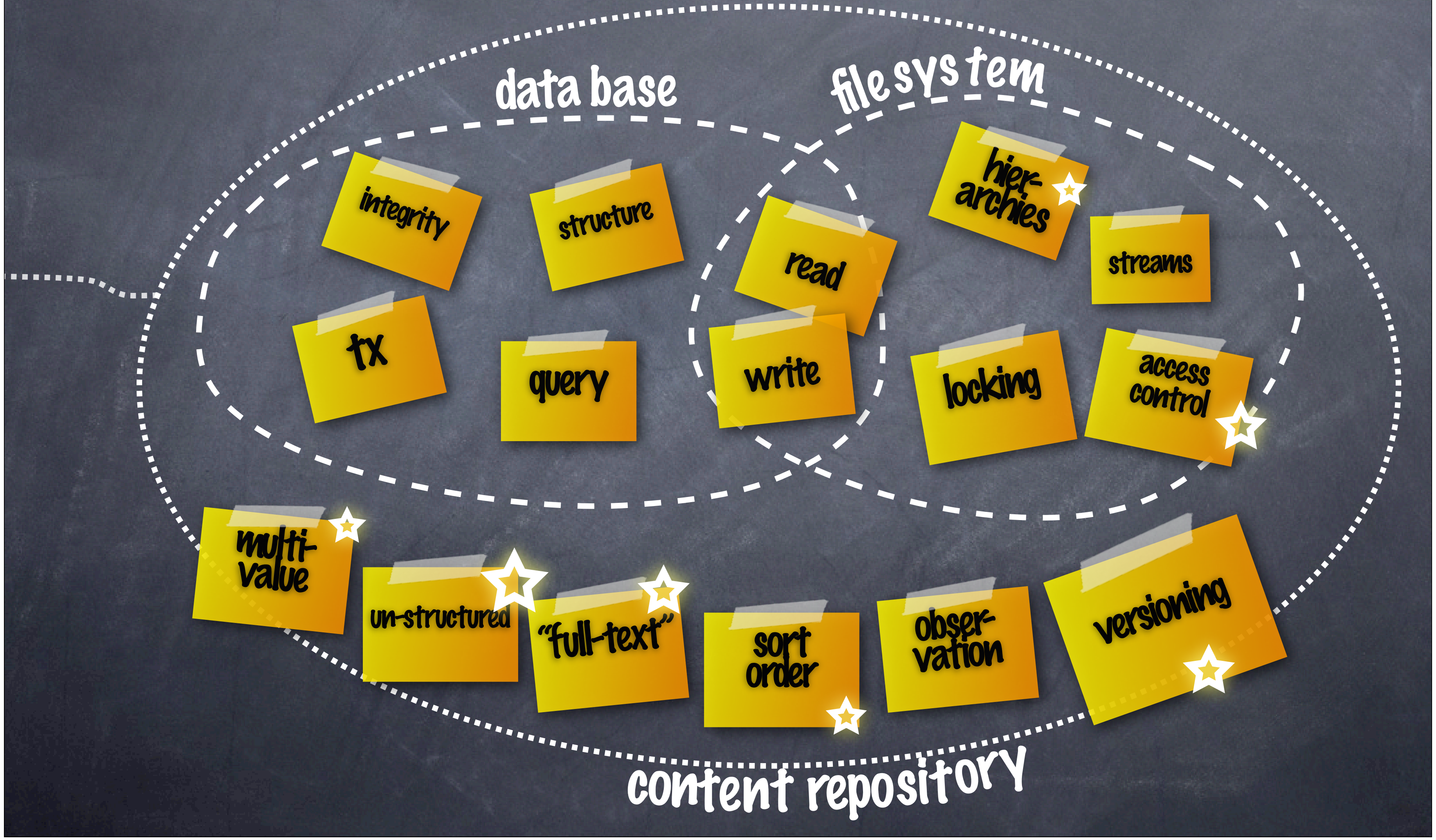
Status:
Public Review Closed sep-2007

Expert Group:

What's JCR?

"The **API** should be a standard, implementation independent, way to access content bi-directionally on a granular level to a content repository." ?

Best of both worlds.



Known compliant Repositories

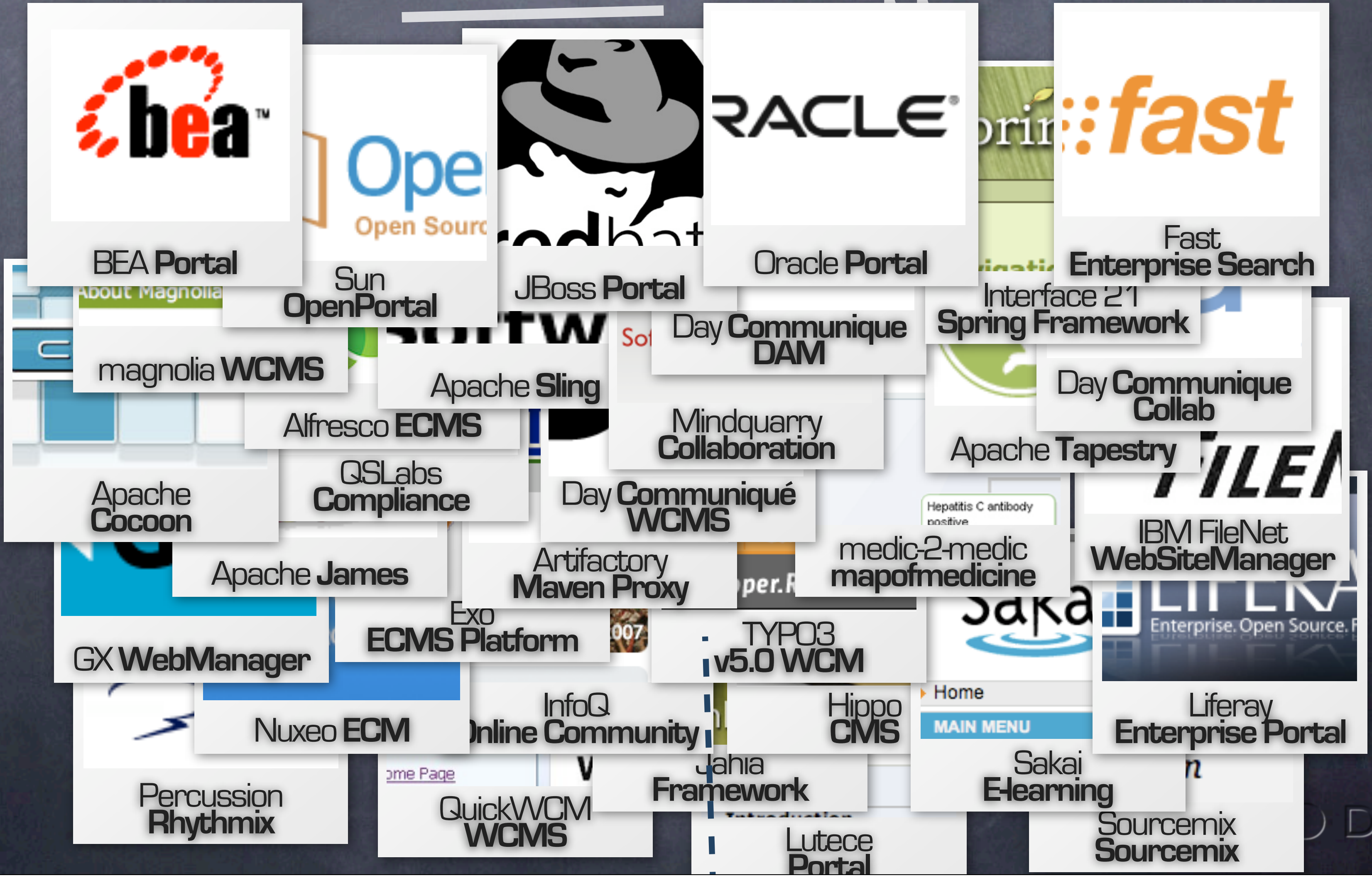
* using third party connector

Apache **Jackrabbit**	Oracle **XML DB**	Exo **ECMS Platform**	* Microsoft **Sharepoint**	* OpenText **Livelink**
Day **CRX**	IBM FileNet **P8**	Xythos **Repository**	Alfresco **ECM**	* Vignette **V7**
* Interwoven **Repository**	IBM **CM**	* EMC **Documentum**		

+hundreds of TCKs registered

How many RDBMS vendors do you need?

Some known JCR Applications



- BEA **Portal**
- Sun **OpenPortal**
- JBoss **Portal**
- Oracle **Portal**
- Fast **Enterprise Search**
- Interface 21 **Spring Framework**
- Day **Communique DAM**
- magnolia **WCMS**
- Apache **Sling**
- Day **Communique Collab**
- Alfresco **ECMS**
- Mindquarry **Collaboration**
- Apache **Tapestry**
- Apache **Cocoon**
- QSLabs **Compliance**
- Day **Communiqué WCMS**
- IBM FileNet **WebSiteManager**
- medic-2-medic **mapofmedicine**
- Apache **James**
- Artifactory **Maven Proxy**
- Exo **ECMS Platform**
- GX **WebManager**
- TYPO3 **v5.0 WCM**
- Liferay **Enterprise Portal**
- Nuxeo **ECM**
- InfoQ **Online Community**
- Hippo **CMS**
- Jahia **Framework**
- Sakai **E-learning**
- Percussion **Rhythmix**
- QuickWCM **WCMS**
- Sourcemix **Sourcemix**
- Lutece **Portal**

JCR code excerpt

```
Repository repository = new TransientRepository();
Session session = repository.login(...);

// Create content
Node root = session.getRootNode();
Node hello = root.addNode("hello");
Node world = hello.addNode("world");

world.setProperty("message", "Hello, World!");
session.save();

// Retrieve content
Node node = root.getNode("hello/world");
print(node.getPath());
print(node.getProperty("message").getString());
```

Sling builds on top of JCR

Scriptable applications layer on top of JCR

OSGi-based industrial-strength framework

APACHE felix

Simple, powerful, JCR inside

Runs on Apache Jackrabbit by default

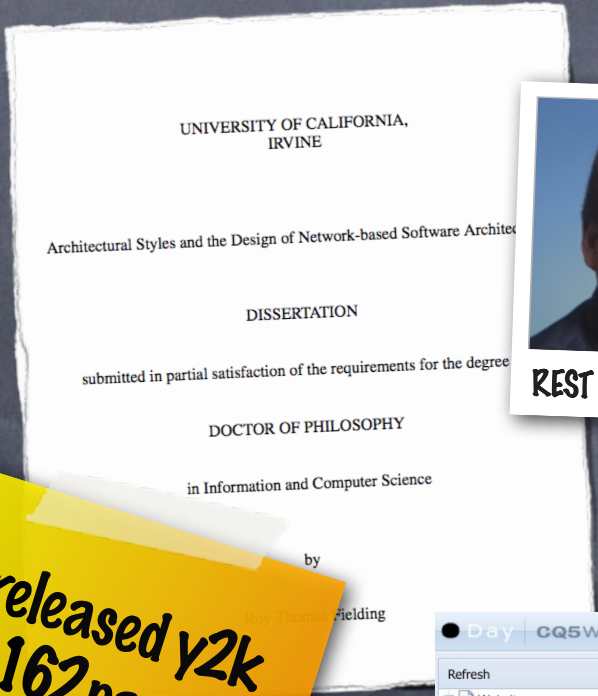
http://incubator.apache.org/sling

join the fun!

Day

Sling

REST over JCR



UNIVERSITY OF CALIFORNIA,
IRVINE

Architectural Styles and the Design of Network-based Software Architec

DISSERTATION

submitted in partial satisfaction of the requirements for the degree

DOCTOR OF PHILOSOPHY

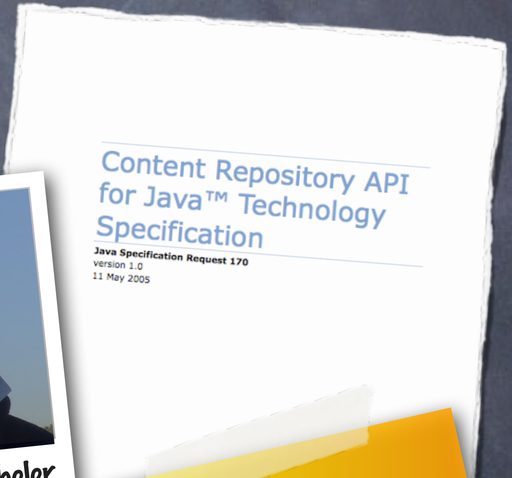
in Information and Computer Science

by

Roy Thomas Fielding

REST -> Roy T. Fielding

JCR: David Nuescheler

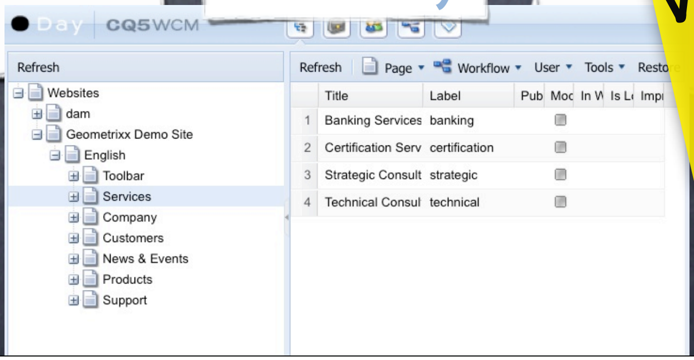


Content Repository API
for Java™ Technology
Specification

Java Specification Request 170
version 1.0
11 May 2005

released y2k
162 pages

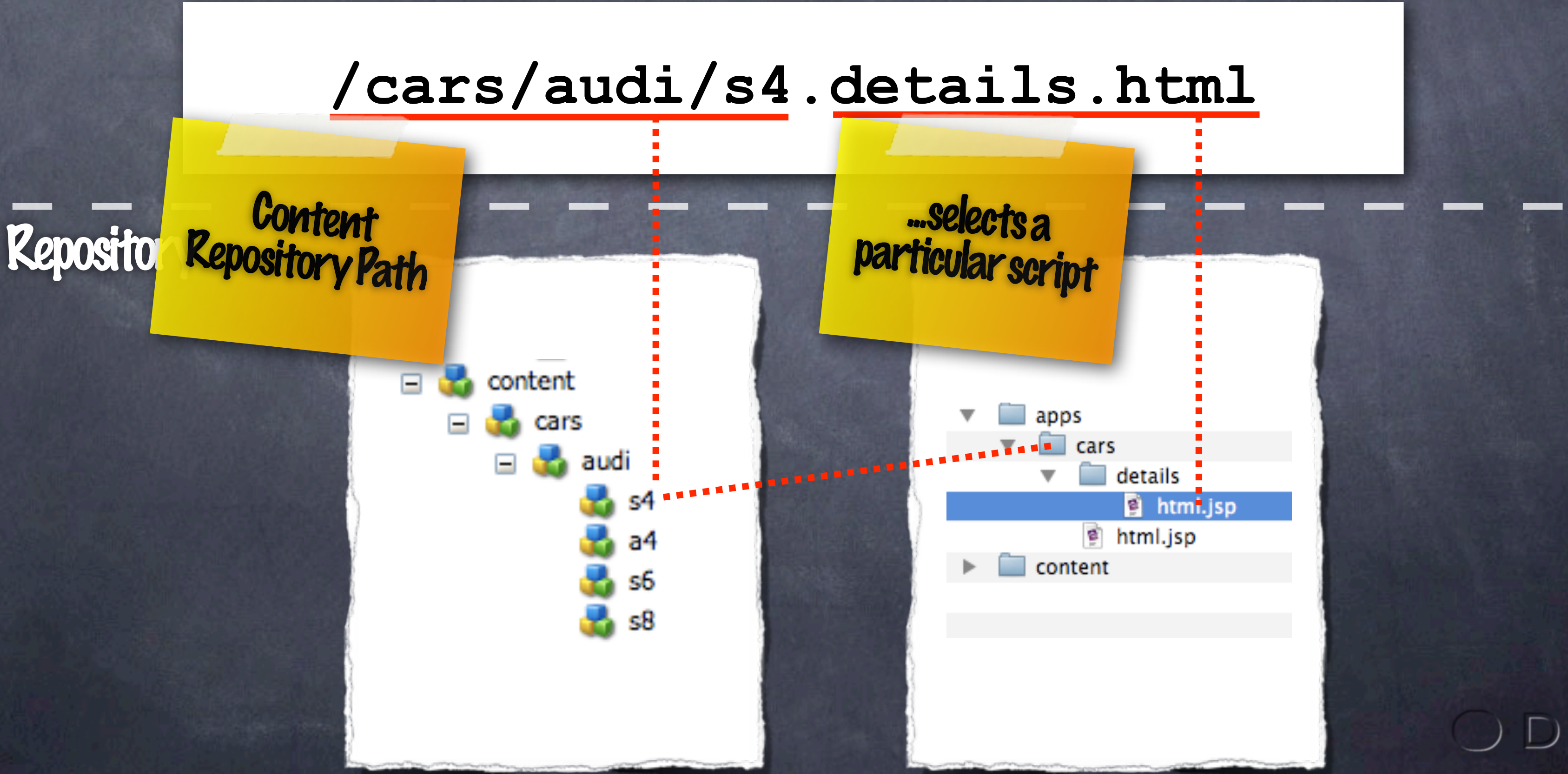
v1.0 released 2005
+300 pages



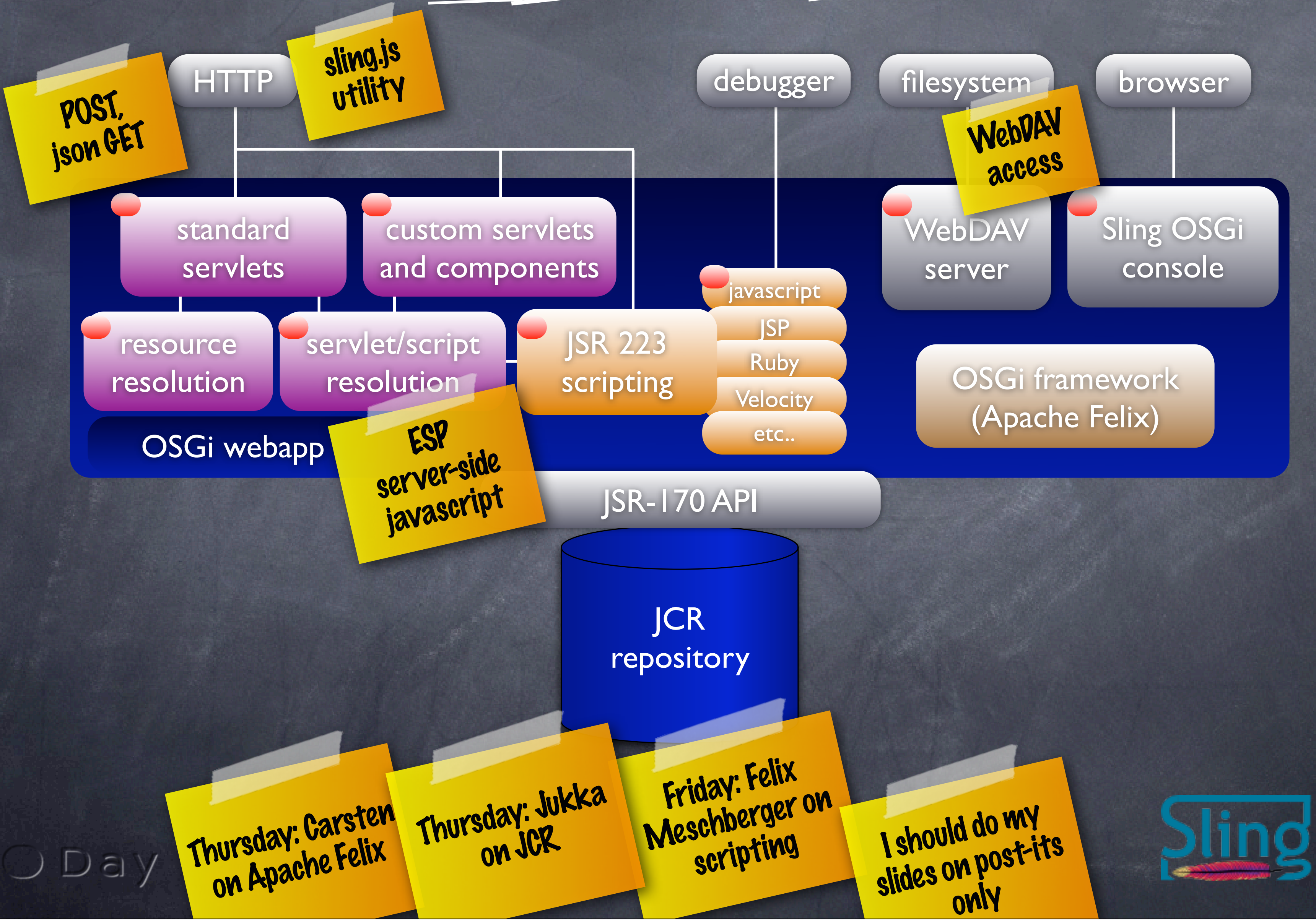
Websites
- dam
- Geometrixx Demo Site
  - English
    - Toolbar
    - Services
    - Company
    - Customers
    - News & Events
    - Products
    - Support

	Title	Label	Pub	Moc	In W	Is Lc	Impi
1	Banking Services	banking					
2	Certification Serv	certification					
3	Strategic Consult	strategic					
4	Technical Consul	technical					

Reclaiming the web.
Sling URL decomposition.



Sling architecture



A minimal Sling blog

Consisting of one .esp script

```
cat /Volumes/localhost/apps/blog/blog.esp | wc -l
54
```

get the code at
grep.codeconsult.ch

Sling POST servlet

```
# POST to Sling
curl -F title=hello http://localhost:8888/foo
-> 200 OK

# GET created node in json format
curl http://localhost:8888/foo.tidy.json
{
  "jcr:primaryType": "nt:unstructured",
  "title": "hello"
}
```

POST parameters set node properties

Day

blog step 1: create content

```
<form method="POST">
  Title:
  <input type="text" name="title" style="width:100%"/>

  Text:
  <textarea style="width:100%" name="text"></textarea>

  <input type="submit" value="save"/>

  <input type="hidden" name=":redirect" value="*"/>
</form>
```

Form fields drive the content model

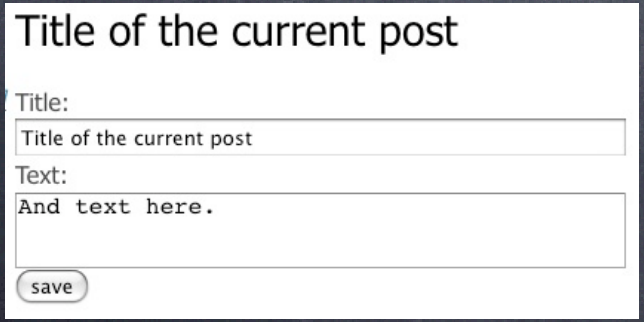
Day

Sling

blog step 2: retrieve content

```
<script src="/system/sling.js"></script>
<form method="POST">
...
</form>

<!-- initialize form fields from current node values -->
<script>Sling.wizard();</script>
```


Title of the current post

Title:

Title of the current post

Text:

And text here.

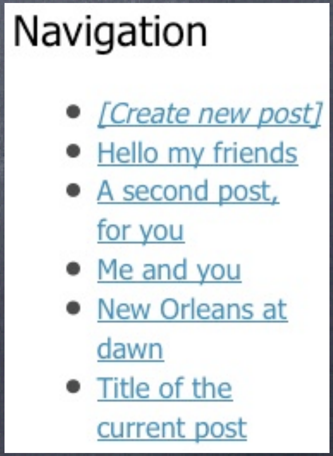
Instant CRU(D)

Day

Sling

blog step 3: navigation

```
<ul>
   <li>
     <a href="/content/blog/*">
     [Create new post]</a>
   </li>
  <script>
    var posts = Sling.getContent("/content/blog", 2);
    for(var post in posts) {
      document.write(
        "<li><a href='" + post + "'>"
        + posts[post].title + "</a></li>");
    }
  </script>
</ul>
```



Navigation

- [Create new post]
- Hello my friends
- A second post, for you
- Me and you
- New Orleans at dawn
- Title of the current post

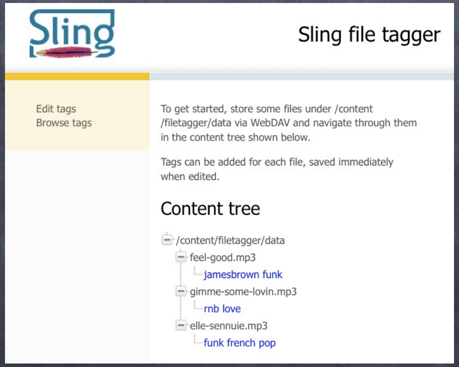
we got a blog!

html form + Sling.wizard() + Sling.getContent()

The filetagger sample

ESP scripting + java + javascript + YUI

OSGi bundle, initial content, WebDAV, observation, RAD



To get started, store some files under /content /filetagger/data via WebDAV and navigate through them in the content tree shown below.

Tags can be added for each file, saved immediately when edited.

Content tree

```
/content/filetagger/data
  feel-good.mp3
    jamesbrown funk
  gimme-some-lovin.mp3
    rnb love
  elle-sennuie.mp3
    funk french pop
```

Filetagger demo

buckle up

Filetagger source code files

pom.xml - defines and configures OSGi bundle, initial content path, etc.

TagParser.java, TagParserImpl.java - observe rawtags and parse into tags

/apps/filetagger/filetagger.esp - main app page, YUI tree
/apps/filetagger/browse.esp - browse tags page
/apps/filetagger/filetagger.js - YUI tree code

/apps/filetagger/tags/tags.esp - query for a given tag
/apps/filetagger/tags/siblings.esp - tag siblings navigation

/apps/filetagger/loadtags.esp - utility to display tags
/apps/filetagger/menu.esp - page menu

TagParserImpl - OSGi service

```
/** TagParser service, listens for changes to "rawtags"
 *    properties, and sets the multivalue "tags" property
 *    accordingly.
*
*  @scr.service
*  @scr.component
*      immediate="true"
*  @scr.property
*      name="service.description"
*      value="Sling TagParser sample service"
*/
public class TagParserImpl implements TagParser, EventListener
{
```

Apache Felix "bundle" and "SCR" plugins do the rest

Day

Sling

TagParserImpl - observe and process

```
// slightly simplified (pseudo-)code
protected void activate(ComponentContext context) {
  session = repository.login();
  final int events = PROPERTY_CHANGED |PROPERTY_REMOVED;
  session.getWorkspace().getObservationManager().addEventListener(
            this, events, DATA_PATH, ...);
  }

// Event callback
public void onEvent(EventIterator it) {
  while(it.hasNext()) {
    final Event e = it.nextEvent();
      if(e.getPath().endsWith("/" + RAWTAGS_PROPERTY)) {
        handlePropertyChange(e);
}}}
```

TagParserImpl - compute tags

```
// called with a JCR event e
// property rawtags contains for example "pop, slow; guitar"
Property rawtags = (Property)session.getItem(e.getPath());
Node parent = rawtags.getParent();
final String [] tagArray = parseTags(rawtags.getValue());

// add a node for each unique tag under our "tags" node
createTagNodes(tagArray);
parent.setProperty(TAGS_PROPERTY, tagArray);

parent.save();
```

Filetagger script mapping

/apps/filetagger/filetagger.esp :
  http:/.../content/filetagger (due to content/... path)

/apps/filetagger/browse.esp :
  http:/.../content/filetagger/tags.browse.html

/apps/filetagger/tags/tags.esp :
  http:/.../content/filetagger/tags/sometag
  due to sling:resourceType on the "sometag" node

/apps/filetagger/tags/siblings.esp :
/apps/filetagger/menu.esp :
  included by other scripts

Filetagger: saving tags from YUI

```
YAHOO.widget.TagsNode.prototype.saveEditorValue =
function(editorData) {
  super.call(this,editorData);
  var data = "rawtags=" + escape(this.label);

  // the YUI tree node knows its JCR path so
  // we just need to POST the property to it
  this.SlingPost(data, this.jcrPath);
};
```

```
POST /content/filetagger/
data/feel-good.mp3/
jcr:content

...
rawtags=jamesbrown
%20funk%20groove
```

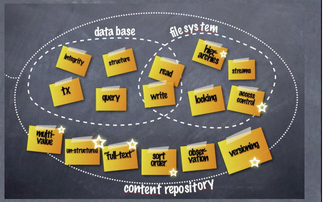
Day

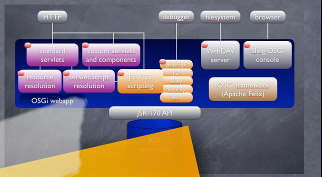
We got a typical Sling application!

JCR features: WebDAV, observation, multi-value properties, nt:unstructured

Sling goodies: simple script mappings (BYOL), POST servlet, RESTful, sling.js

OSGi bundle, code + initial content
Felix and Sling Maven plugins





"Sling is not a web applications framework, it's just a web framework"

join the fun!

http://incubator.apache.org/sling
http://dev.day.com
http://contentcentric.org/