

# Social Media API for Java

## JCP EC F2F Meeting

**Werner Keil**

**Antoine Sabot-Durand**

Redwood City, California

10 / 01 / 12

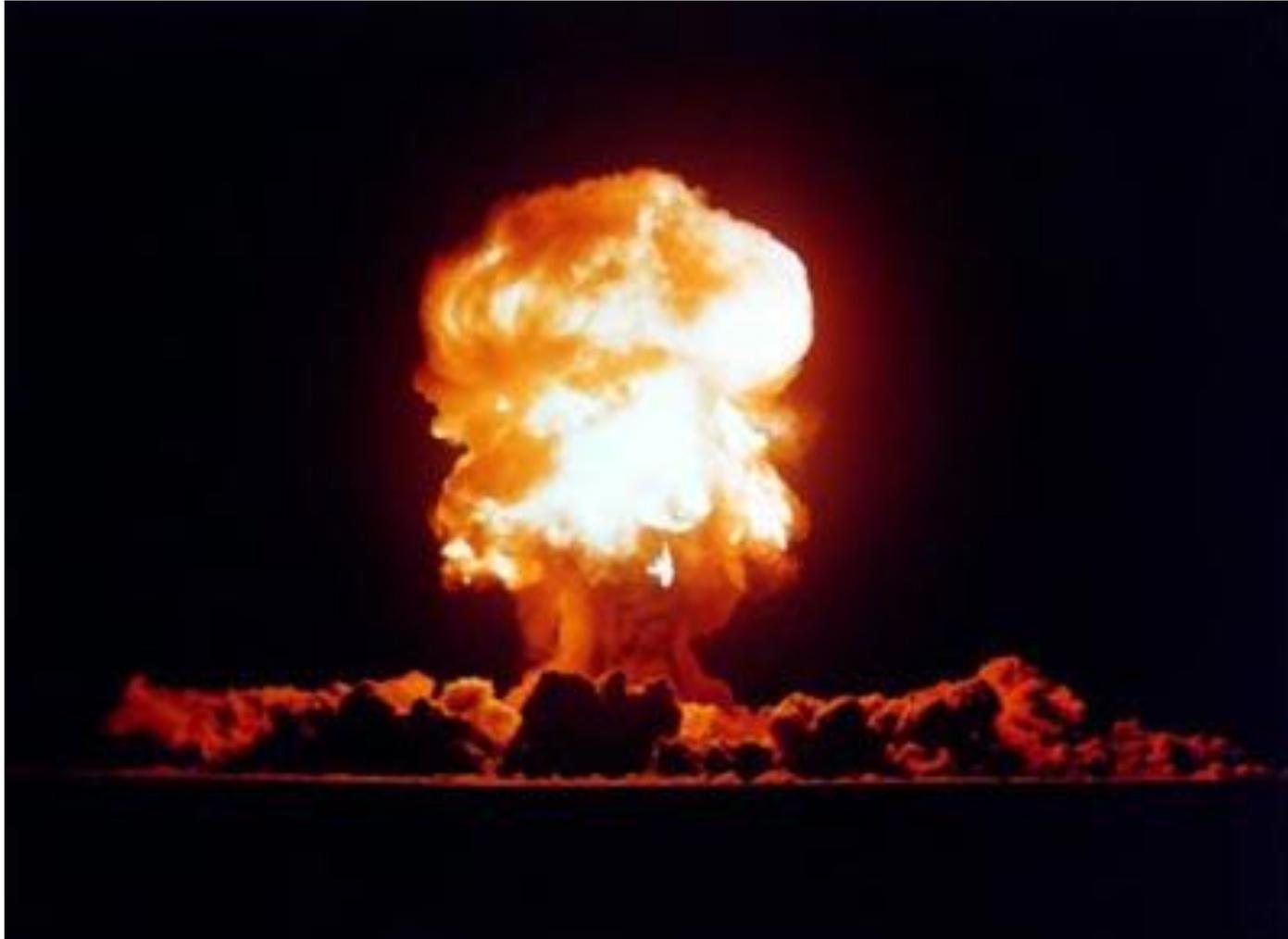


Our Goal

# A JAVA API FOR ACCESS TO SOCIAL NETWORKS



# Proliferation



# Proliferation of Social Media



©2010 SocialGrow



# Apache Shindig



- Open source implementation of OpenSocial & Gadgets specification
- An Apache Software project
- Available in Java & PHP
- <http://shindig.apache.org>

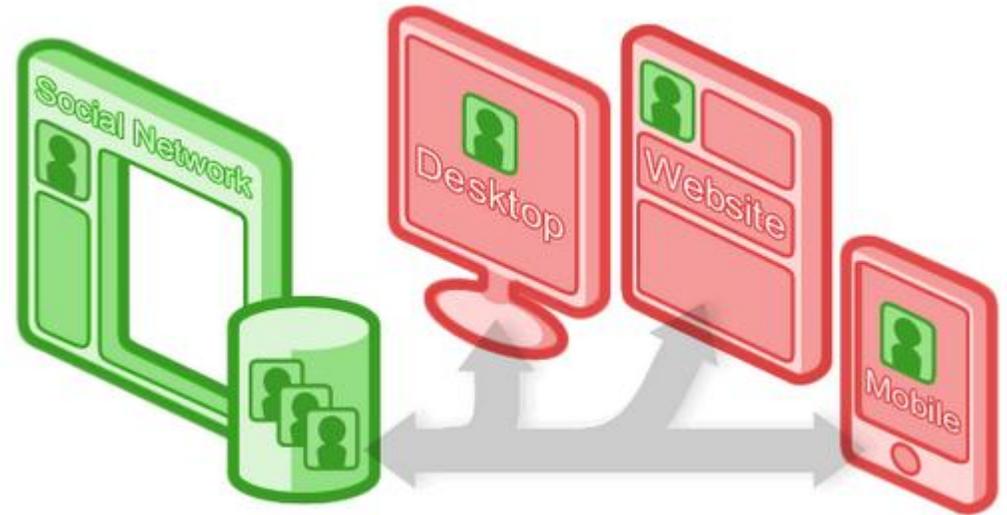
## It's Goal:

**“Shindig's goal was to allow new sites to start hosting social apps in under an hour's worth of work”**

# OpenSocial Container

- Become an OpenSocial Container
  - Get Shindig (PHP or Java) or the Google implementation
    - <http://shindig.apache.org>
  - Look at examples & documentation
    - <http://code.google.com/p/opensocial-resources/wiki/SampleApps>

- The “official” OpenSocial implementation has shifted – from Shindig to Google Code



# What Is a Gadget?

The screenshot shows a Google iGoogle page with a search bar and navigation links. Three gadgets are displayed:

- MyProxy Example:** A form for entering proxy credentials. It includes fields for Username (mpierce), Password, and MyProxy Server (myproxy.teragrid.org). Below the form is a table of user credentials.
- Job Submission Example:** A form for submitting a remote command. It includes fields for Hostname (gf1.ucs.indiana.edu), Executable (/bin/lis), Arguments (-l), Taskname (2eba2a9e:119aa5bc25c:-7), Grid Provider (CT2), and Standard Output (/tmp/out).
- My Gadgets:** A list of installed gadgets: MyProxyGadget.xml, JobSubmitGadget.xml, and developer.xml. It also includes an "Add a gadget" field with the URL http:// and an "Add" button.

Two blue arrows point from the text below to the "Submit" buttons of the MyProxy and Job Submission gadgets.

Simple gadgets for getting a Grid proxy credential and running remote commands. Both run on my own Web server.

# Portable Contacts

- REST API for Person and Contacts
  - Open source implementation
- Compatible with OpenSocial
  - <http://portablecontacts.net>

## Person info

Name

uid

photo

profile url

Age

URLs

Books

Movies

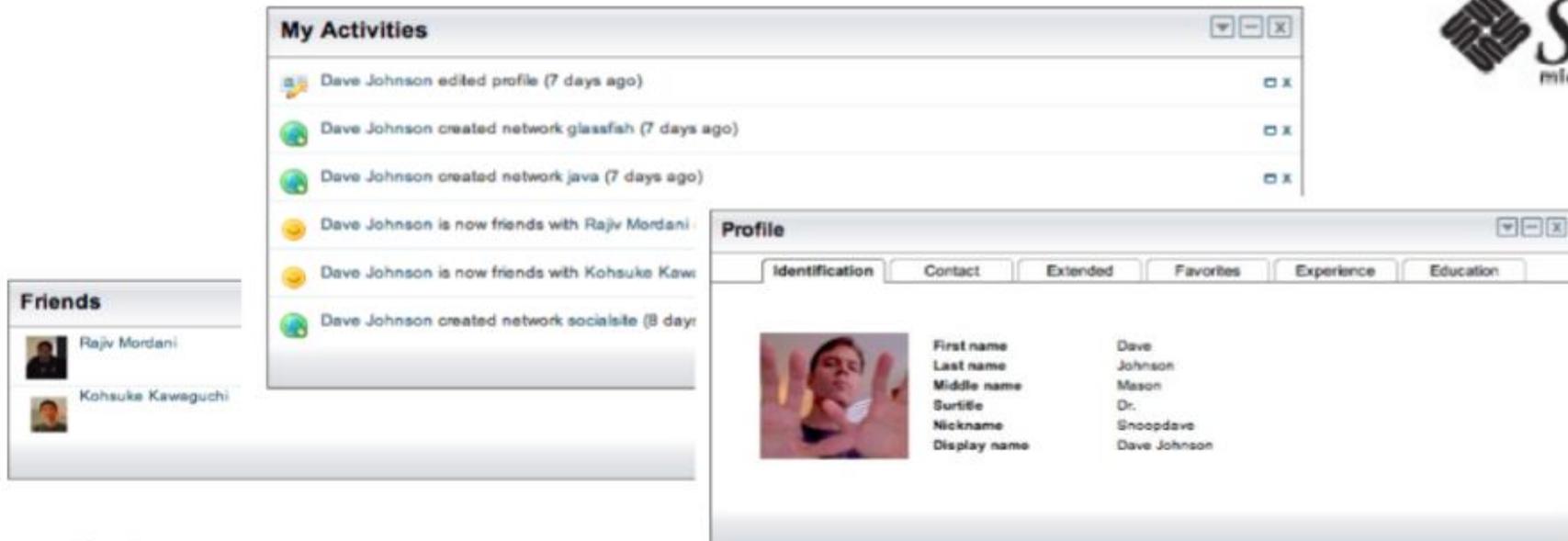
Smoker



# SocialSite – Sun’s approach to Social

## Sun Microsystems

- Socialsite: Shindig + gadget based UI written in Java
- Open Source <https://socialsite.dev.java.net/>



The screenshot displays a user interface for a social networking site. On the left, a 'Friends' list shows two users: Rajiv Mordani and Kohsuke Kawaguchi. The main area is divided into two panels. The top panel, titled 'My Activities', lists recent actions: 'Dave Johnson edited profile (7 days ago)', 'Dave Johnson created network glassfish (7 days ago)', 'Dave Johnson created network java (7 days ago)', 'Dave Johnson is now friends with Rajiv Mordani', 'Dave Johnson is now friends with Kohsuke Kawaguchi', and 'Dave Johnson created network socialsite (8 days ago)'. The bottom panel, titled 'Profile', shows a user's profile information under the 'Identification' tab. The profile includes a photo of Dave Johnson and the following details:

First name	Dave
Last name	Johnson
Middle name	Mason
Surname	Dr.
Nickname	Snoopydave
Display name	Dave Johnson



# Facebooks' Query Language

- All of the Facebook API is basically a set of wrappers around their SQL like FQL query language. So you can, if you prefer, make custom methods out of FQL query strings.

```
var myQuery='SELECT name FROM user WHERE  
uid='+api.get_session().uid;  
api.fql_query(myQuery, getFQLResponse);
```

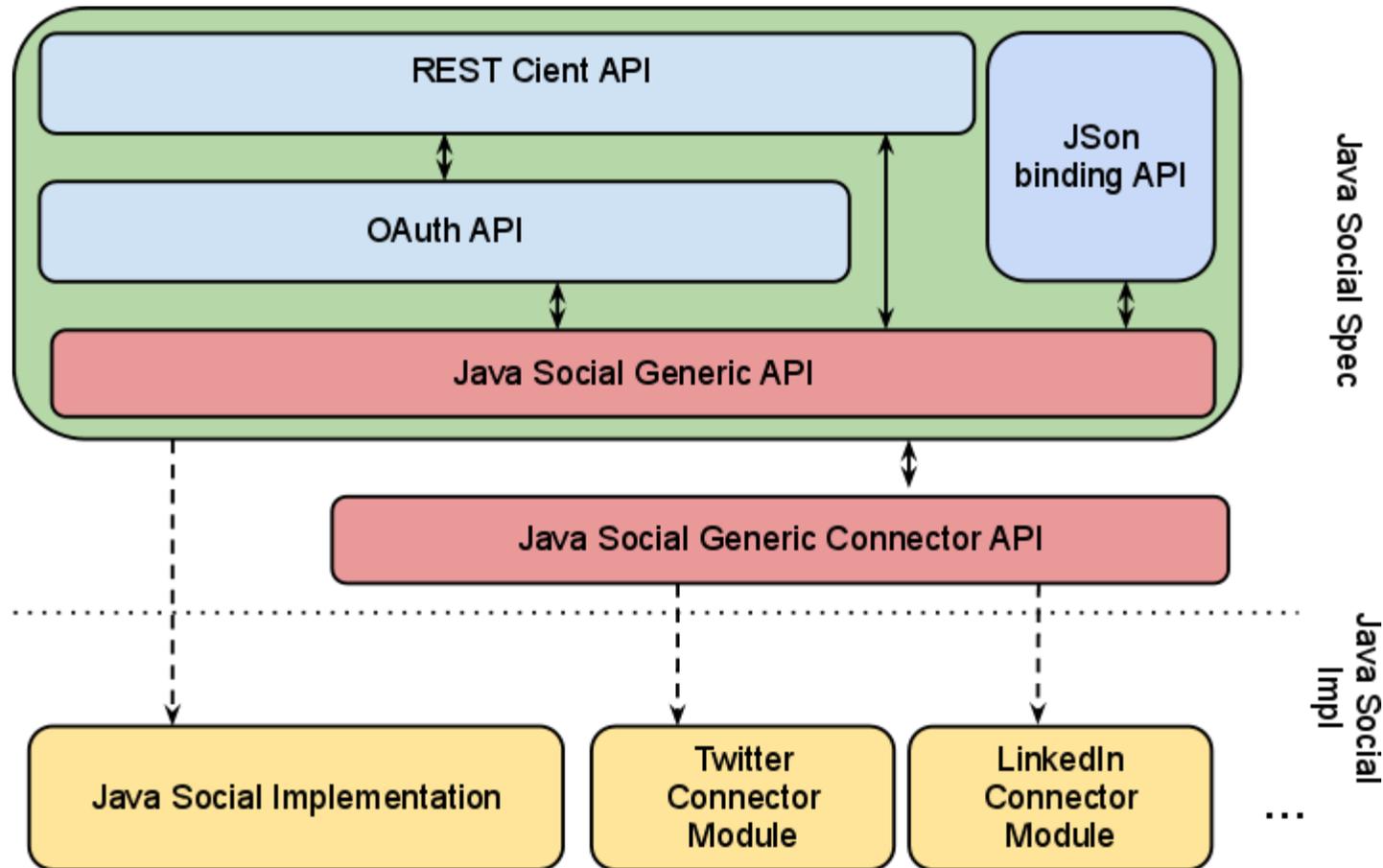
getFQLResponse() is a developer-written callback function

If it's SQL like, could we use something like JPA around it?

# Seam Social – a bootstrap for the API

- A full CDI API to interact with Social Networks
- Developed by Antoine Sabot-Durand (one of the proposed co-Spec Leads)
- Independent of CDI implementation
- A helper for:
  - Consuming REST services
  - Consuming OAuth 1 and 2 services
  - Identification via OAuth
  - JSON binding of Social Services API
- Ready to adopt JSR 351, 353 and JSR 339
- Will move to Apache Delta Spike initiative during 2012

# Java Social – a kind of JCA for Social



# DaliCore – History

- DaliCMS (2006): “Web 2.0 ready CMS”
- Open-source “CMS” with community features
- 2008: integration with Shindig, provides OpenSocial Implementation (JavaScript code from Shindig, Java code in DaliCMS)
- Developed as Glassfish incubator project

# DaliCore – CMS

- More than a CMS → DaliCore
- Adds functionality common to users, content and permissions on top of Java EE 6.
- Focus on Users and Permissions.
- In about every project that uses DaliCore, users should be able to login with existing credentials (Facebook, Twitter, Google Connect,...)
- Dali modules extend DaliCore

# DaliCore – Connect

```
public abstract class ExternalNetwork
    public abstract Response connect(String callback);
public abstract ExternalToken callback(String
requestToken, String verifier);
public abstract OnlineAccount
createOnlineAccount(ExternalToken accessToken) throws
DaliCoreException;
```

- **ExternalNetwork class is extended by**
  - FacebookExternalNetwork
  - TwitterExternalNetwork
  - GoogleExternalNetwork

# DaliCore – Integrate

- The User concept in DaliCore can have 0 or more instances of “OnlineAccount”
- In projects created with DaliCore, the administrator/configurator decides what social networks are supported.
- Users can access all functionality using their social network credentials.

# Twitter4J – History

- Twitter4J is an unofficial Java library for the Twitter API.  
With Twitter4J, you can easily integrate your Java application with Twitter.
- Its author, Yusuke Yamamoto now officially working at Twitter. Designated to represent Twitter in Social JSR EG.

Let's talk

**Q & A**

