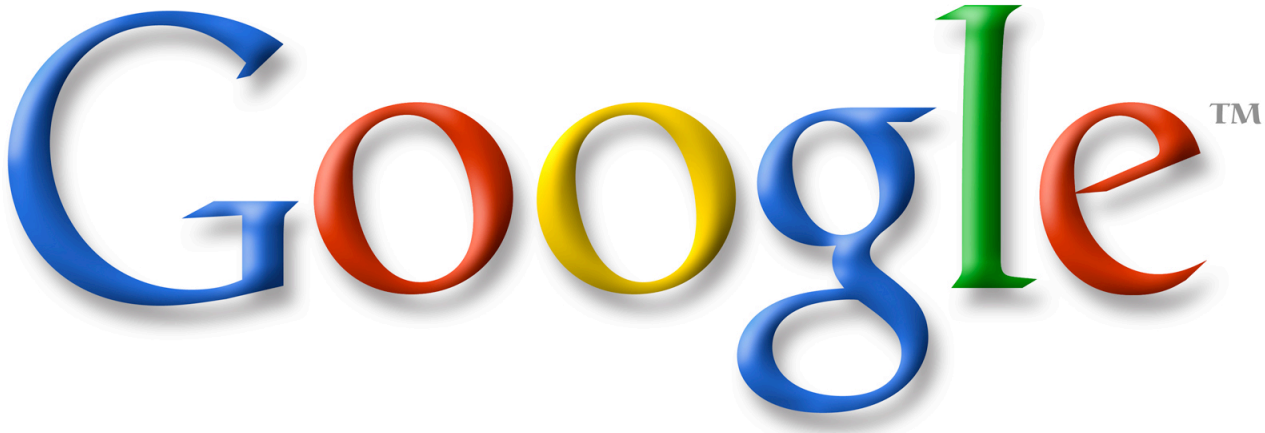

Google™

A Technical Overview

Chris Schalk, Developer Advocate

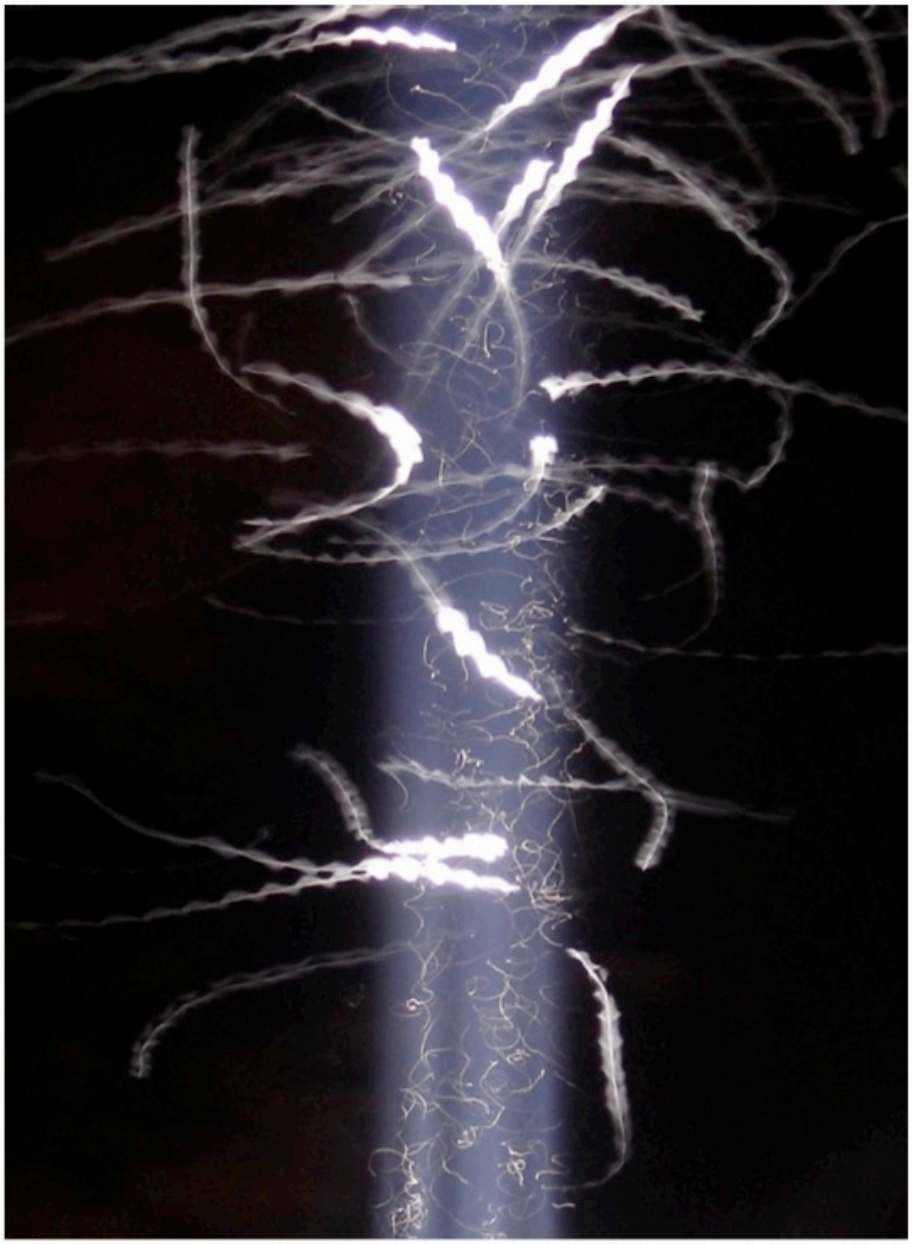
9/16/2008

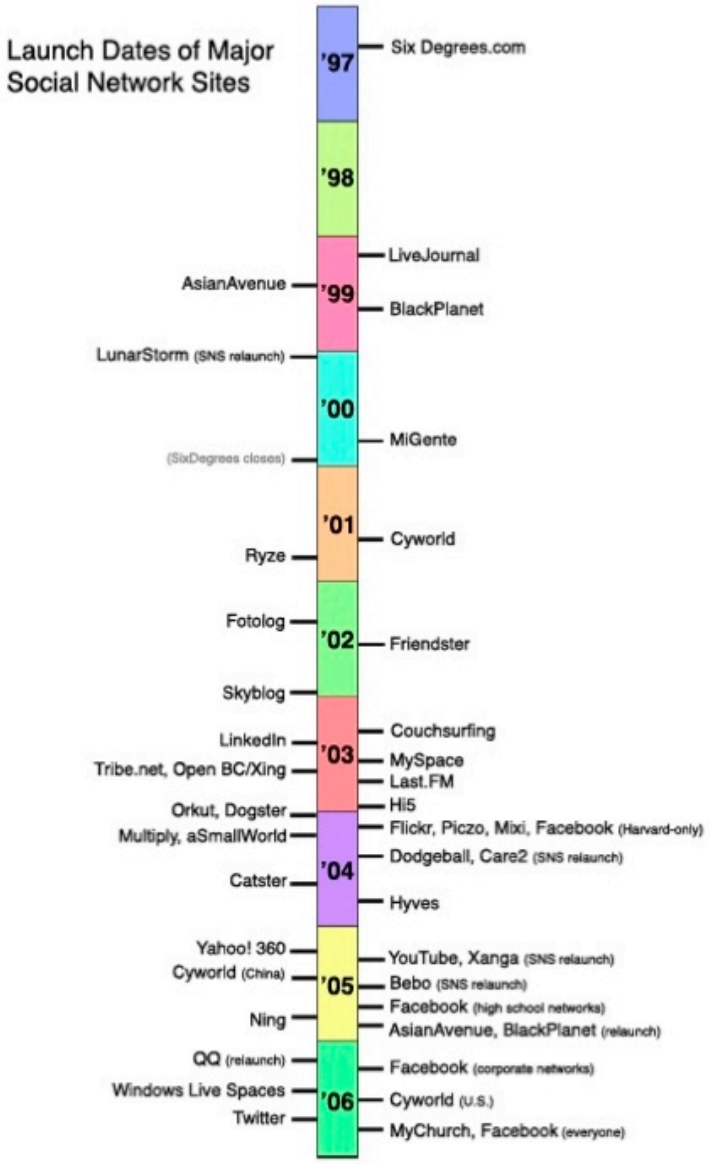
Agenda

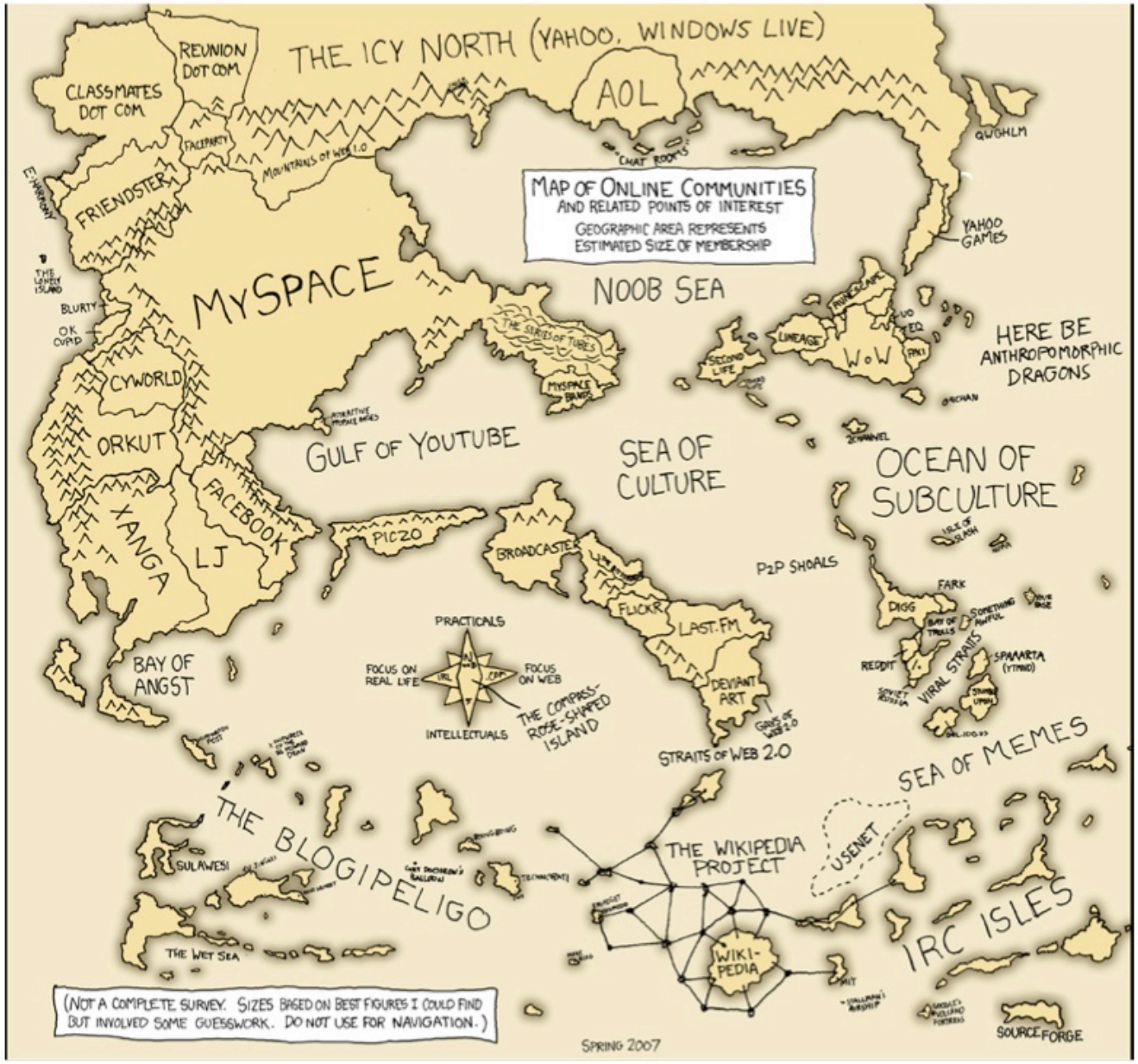
- OpenSocial Concepts
- Building OpenSocial Applications
- Hosting OpenSocial Applications
- Google Friend Connect

Agenda

- OpenSocial Concepts
- Building OpenSocial Applications
- Hosting OpenSocial Applications
- Google Friend Connect




Launch Dates of Major Social Network Sites
'97
Six Degrees.com
'98
'99
LiveJournal
AsianAvenue
BlackPlanet
LunarStorm (SNS relaunch)
'00
MiGente
(SixDegrees closes)
'01
Cyworld
Ryze
Fotolog
'02
Friendster
Skyblog
LinkedIn
'03
Couchsurfing
MySpace
Tribe.net, Open BC/Xing
Last.FM
Hi5
Orkut, Dogster
Flickr, Piczo, Mixi, Facebook (Harvard-only)
Multiply, aSmallWorld
Dodgeball, Care2 (SNS relaunch)
'04
Catster
Hyves
Yahoo! 360
YouTube, Xanga (SNS relaunch)
Cyworld (China)
'05
Bebo (SNS relaunch)
Facebook (high school networks)
Ning
AsianAvenue, BlackPlanet (relaunch)
QQ (relaunch)
Facebook (corporate networks)
Windows Live Spaces
'06
Cyworld (U.S.)
Twitter
MyChurch, Facebook (everyone)


MAP OF ONLINE COMMUNITIES
AND RELATED POINTS OF INTEREST
GEOGRAPHIC AREA REPRESENTS
ESTIMATED SIZE OF MEMBERSHIP
THE ICY NORTH (YAHOO, WINDOWS LIVE)
AOL
CHAT ROOMS
QWGHLM
YAHOO GAMES
REUNION DOT COM
CLASSMATES DOT COM
FACEPARTY
MOUNTAINS OF WEB 1.0
FRIENDSTER
E-HARMONY
THE LONELY ISLAND
BLURTY
OK CUPID
MYSPACE
NOOB SEA
THE SERIES OF TUBES
MYSPACE BANDS
SECOND LIFE
LINEAGE
WoW
HERE BE ANTHROPOMORPHIC DRAGONS
CYWORLD
ORKUT
GULF OF YOUTUBE
SEA OF CULTURE
OCEAN OF SUBCULTURE
XANGA
FACEBOOK
LJ
PICZO
BROADCASTER
P2P SHOALS
FLICKR
LAST.FM
DEVIANT ART
FARK
DIGG
REDDIT
VIRAL STRAITS
SPAAARTA (YTMND)
PRACTICALS
FOCUS ON REAL LIFE
FOCUS ON WEB
INTELLECTUALS
THE COMPASS-ROSE-SHAPED ISLAND
BAY OF ANGST
STRAITS OF WEB 2.0
SEA OF MEMES
THE BLOGIPELIGO
SULAWESI
THE WET SEA
THE WIKIPEDIA PROJECT
USENET
WIKI-PEDIA
IRC ISLES
SOURCEFORGE
(NOT A COMPLETE SURVEY. SIZES BASED ON BEST FIGURES I COULD FIND BUT INVOLVED SOME GUESSWORK. DO NOT USE FOR NAVIGATION.)
SPRING 2007

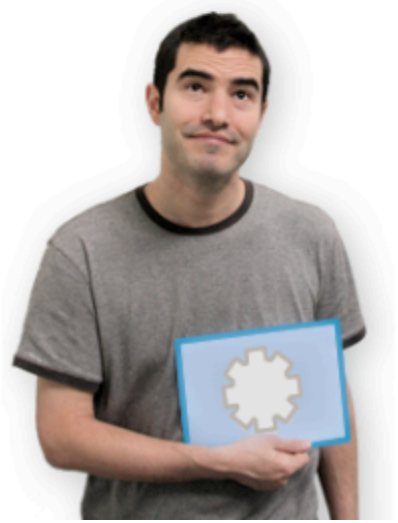
Too many platforms!

Standards based

OpenSocial

API

Friends
Profiles
Activities



Write once…

Deploy everywhere

orkut
LinkedIn
hi5
salesforce.com
OpenSocial
Use multiple times
Coooooool!

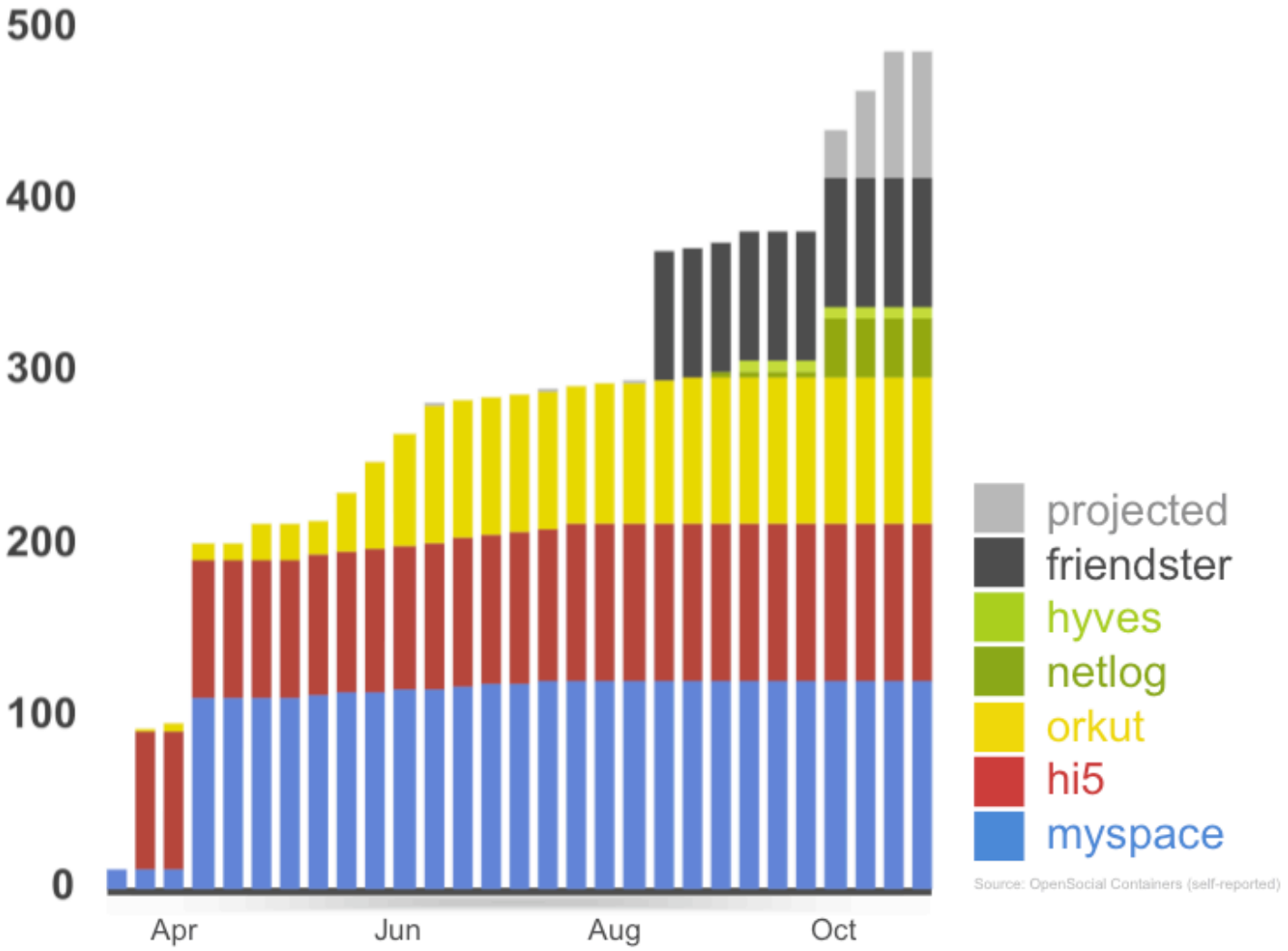
Not just Social Network Sites

- Social network sites - Profiles and home pages
  ex: MySpace, Hi5
- Personal dashboards – Widgets
  ex: iGoogle, My Yahoo
- Sites based around a Social Object
  ex: Flickr, YouTube
- Corporate CRM systems
  ex: Oracle, SalesForce.com
- Any web site

OpenSocial “Containers”

OpenSocial

User-Reach Growth (M)



Today: 375 Million User Reach

Where is OpenSocial live today?

Live to Users:

- MySpace
- orkut
- Hi5
- Freebar
- Friendster
- Webon from Lycos
- IDtail
- YiQi
- Netlog - New!
- Hyves - New!

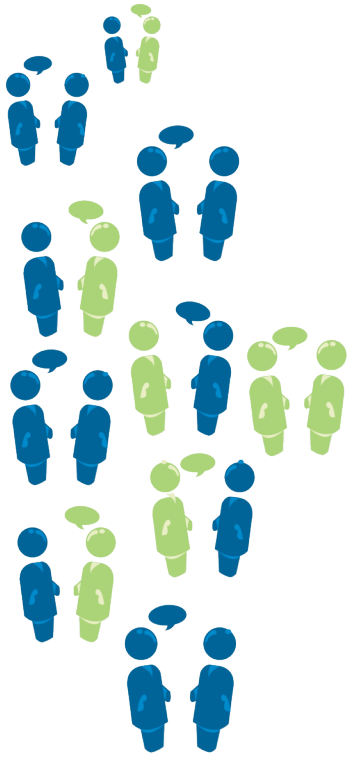
Live Developer Sandboxes:

- iGoogle
- imeem
- CityIN
- Tianya
- Ning
- Plaxo Pulse
- Mail.ru

Individual Developer Links:
http://code.google.com/apis/opensocial/gettingstared.html

OpenSocial today

- ✓ 375,000,000 users
- ✓ 150,000,000 installs
- ✓ 4,500+ apps
- ✓ Pipeline of 100+ containers worldwide
- ✓ 20,000+ developers
- ✓ 10,000,000 daily app users



OpenSocial Roadmap

- Version 0.5 was released in a “developer release” on Nov 1st.
  - First “sandbox” was made available on Orkut
- Version 0.6 was released in December
  - Initial version of Shindig server software was launched as Apache incubator project
  - Other sandboxes came live - Hi5, Ning, Plaxo …
- Version 0.7 (production) was released in January
  - MySpace, Hi5, Orkut currently running with 0.7

OpenSocial v0.8 is here!

- Latest evolution of OpenSocial as defined by the OpenSocial development community
  - Updated JavaScript API
  - Now contains a RESTful API

- Shindig supports v0.8 today
  - iGoogle now supports 0.8!
  - hi5 almost done with their support
  - orkut currently working on their implementation
  - Other containers also working on 0.8

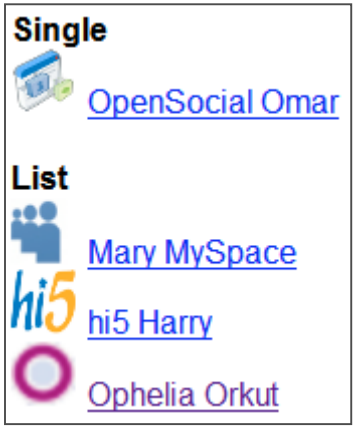
- Specification is here:
http://www.opensocial.org/Technical-Resources/opensocial-spec-v08

Beyond 0.8 - OpenSocial Templating Proposal

- Goal:
  - Enable a faster development experience that is both secure and efficient

Core principles:

- Client-side and server-side processing
- Standard set of tags with extensibility model
  - Example: <os:ShowPerson>



- http://groups.google.com/group/opensocial-and-gadgets-spec/web/opensocial-templates
  - Demo: http://ostemplates-demo.appspot.com/

To follow the progress of the Spec…

http://groups.google.com/group/opensocial-and-gadgets-spec

Agenda

- OpenSocial Concepts
- Building OpenSocial Applications
- Hosting OpenSocial Applications
- Google Friend Connect

OpenSocial Hi-level Details

- A set of open, standard APIs for building social applications

- APIs based on Web standards: HTML/CSS, JavaScript, REST+OAuth

- Reference Implementation: Shindig
  - Apache Open Source Project

- A global community of thousands of developers

OpenSocial Core Services

- People & Friends
  - Access friends information programmatically
- Activities
  - See what you're friends are up to
  - Share what you are doing
- Persistence
  - Provide state without a server
  - Share data with your friends

OpenSocial Client APIs

- JavaScript API - for browser based client development
  - Standard Web development technologies
    - HTML + Javascript, CSS, AJAX
  - Can integrate with 3rd party servers
    - PHP, Perl, Java, C/C++
- RESTful API - for server based client development
  - Based on Atom publishing protocol
  - Data transfer is Atom or JSON
  - More on RESTful API later…

Building JavaScript OpenSocial Applications

- If you know how to develop gadgets, you know how to develop JavaScript OpenSocial applications!
  - Follows same approach as gadgets
    - JavaScript/HTML/CSS embedded in an XML document
    - XML document containing gadget is hosted on the Internet
  - OpenSocial applications are “gadgets++”
    - Gadgets development, but with additional Social capabilities

OpenSocial People & Friends Example

An OpenSocial example to show the owner's friends

```
function getFriends(id) {
  var params = {};
  params[opensocial.IdSpec.Field.USER_ID] =
                opensocial.IdSpec.PersonId.OWNER;
  params[opensocial.IdSpec.Field.GROUP_ID] = "FRIENDS";

  var idspec = opensocial.newIdSpec(params);

  var req = opensocial.newDataRequest();
  req.add(req.newFetchPeopleRequest(idspec), "req");
  req.send(onLoadFriends);
};
```

OpenSocial People & Friends Example

An OpenSocial example to show the owner's friends

```
var html= "Listing friends:<br/>";

function printPerson(person) {
  html += "<br/>" + person.getDisplayName();
};

function onLoadFriends(dataResponse) {
  dataResponse.get("req").getData().each(printPerson);
  document.getElementById('message').innerHTML = html;
};
```

Activities Example

Posting an Activity

```
function postActivity(text) {
   var params = {};
   params[opensocial.Activity.Field.TITLE] = text;
   var activity = opensocial.newActivity(params);
   opensocial.requestCreateActivity(activity,
     opensocial.CreateActivityPriority.HIGH, callback);
}

postActivity("This is a sample activity, created at " +
  new Date().toString())
}
```

Persistence Example

Storing persisted data

```
function populateMyAppData() {
  var req = opensocial.newDataRequest();
  var data1 = Math.random() * 5;
  var data2 = Math.random() * 100;

  req.add(req.newUpdatePersonAppDataRequest("VIEWER",
                          "AppField1", data1));
  req.add(req.newUpdatePersonAppDataRequest("VIEWER",
                          "AppField2", data2));
  req.send(requestMyData);
}
```

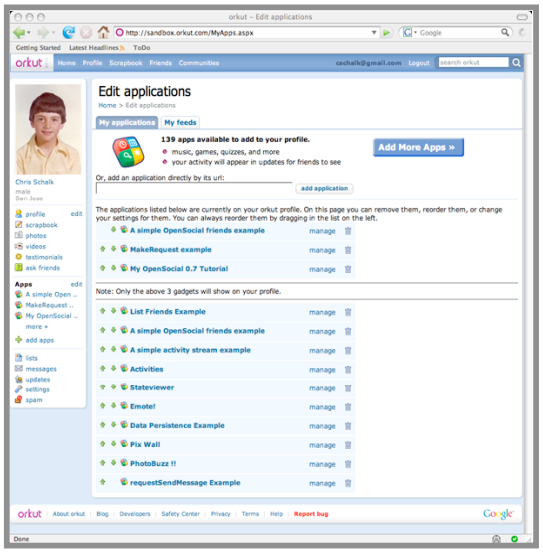
Persistence Example

Fetching persisted data

```
function requestMyData() {
  var req = opensocial.newDataRequest();
  var fields = ["AppField1", "AppField2"];

  req.add(req.newFetchPersonRequest(
    opensocial.DataRequest.PersonId.VIEWER), "viewer");
  req.add(req.newFetchPersonAppDataRequest("VIEWER",
    fields), "viewer_data");
  req.send(handleReturnedData);
}
```

OpenSocial JavaScript client application in action

Gadget using JavaScript Client API communicates with OpenSocial server



browser

OpenSocial Server

Gadget XML Source

Integrating OpenSocial applications with external servers

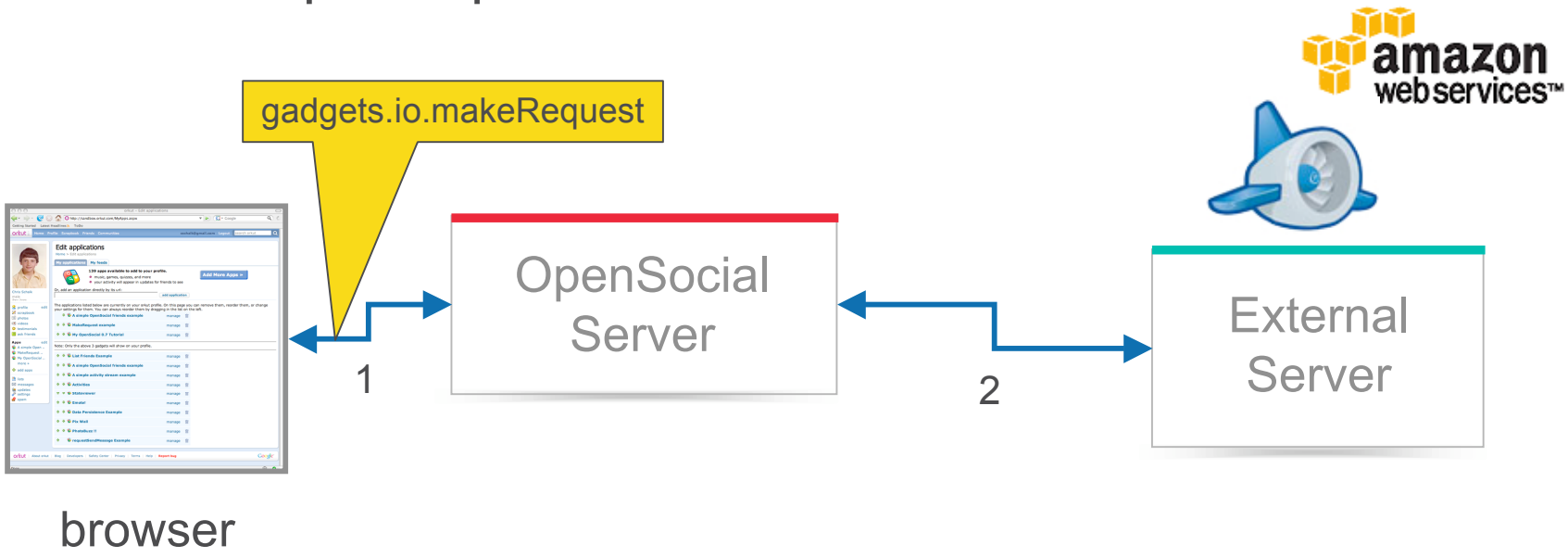
The OpenSocial persistence service provides a way to store/share small amounts of data on the OpenSocial server, but…

- What if you need to store more data than your container allows?

- Solution: You can make independent requests out to external servers.
  - Use: **gadgets.io.makeRequest**
  - Can also make authenticated requests using Oauth
  - Can use cloud services from: Joyent, Amazon or AppEngine!

Integrating OpenSocial applications with external servers

An example request to an external server

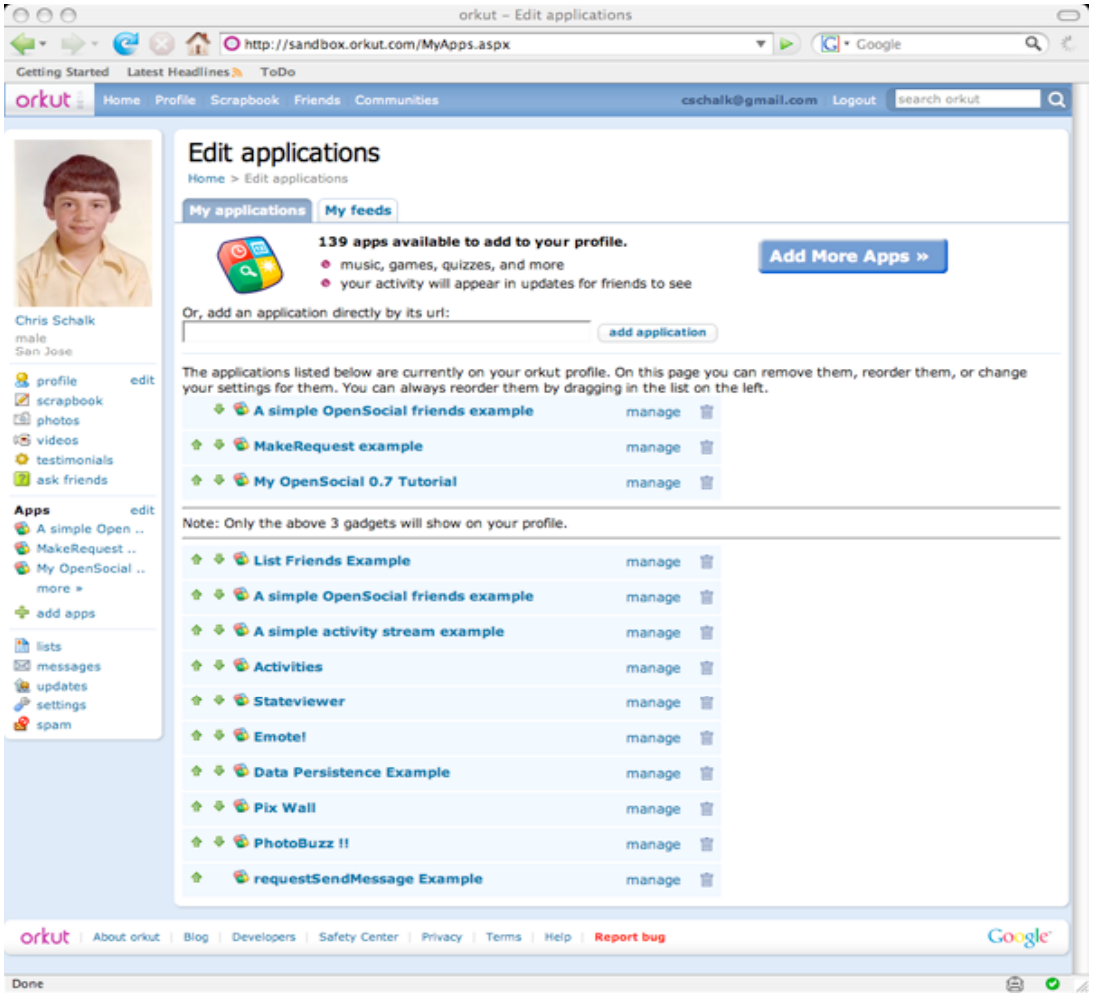


1. Initial request made from gadget
2. Server routes request to external server

Requests can be secured using OAuth!

Demonstration

- Building OpenSocial Client Applications



Agenda

- OpenSocial Concepts
- Building OpenSocial Applications
- Hosting OpenSocial Applications
- Google Friend Connect

How to host OpenSocial Applications

1. Can build your own server that implements OpenSocial specification…

2. Or can use “Shindig” - Reference implementation for OpenSocial

- Gadget Server
  - –Renders gadget XML as HTML/JS/CSS
- OpenSocial Data Server
  - –RESTful API server (in-progress)

Serverside OpenSocial

How to implement an OpenSocial server

Shindig

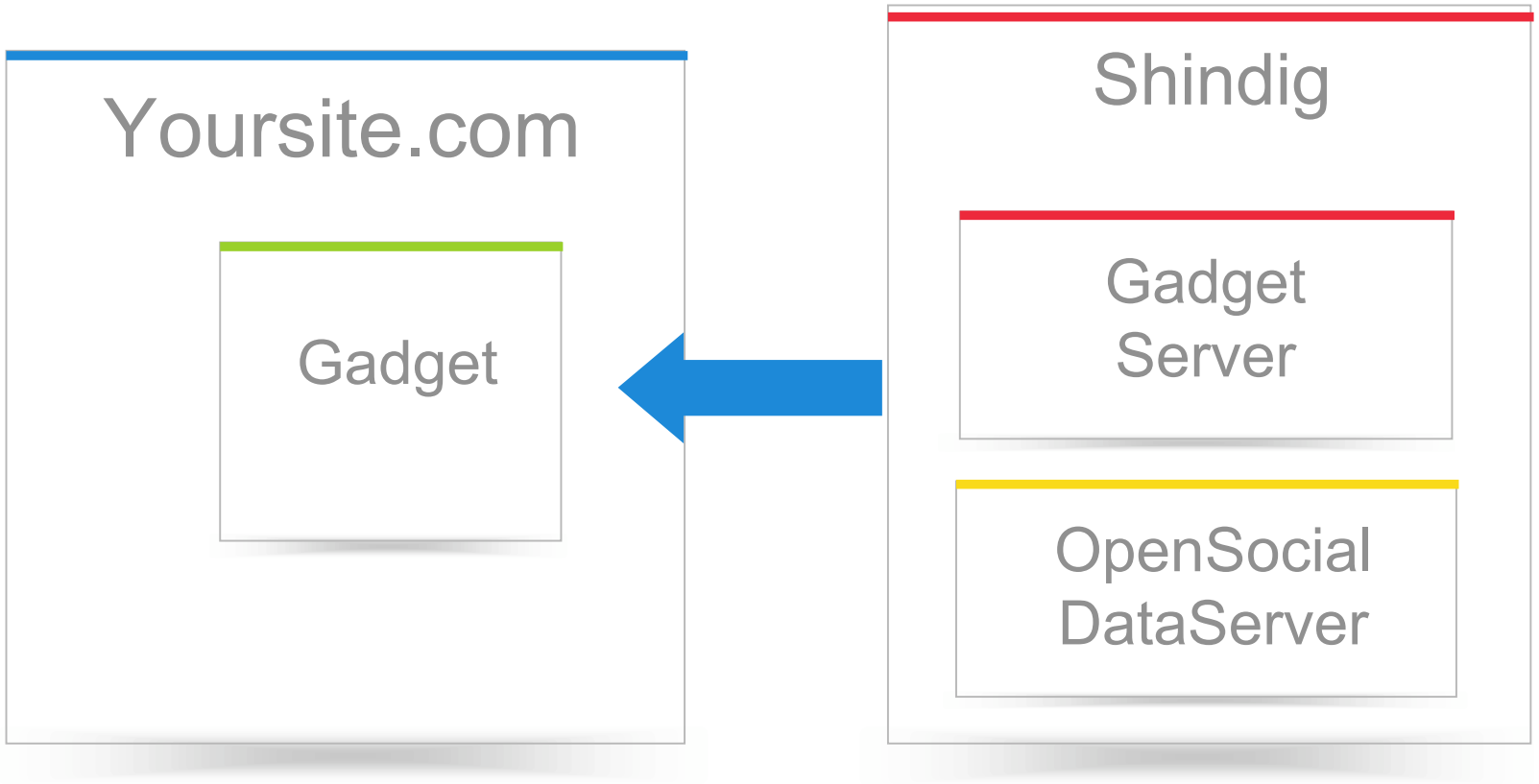
orkut, hi5, iGoogle, tianya.cn, …

- **Gadget Server**
  - –Renders gadget XML as HTML/JS/CSS
- **OpenSocial Data Server**
  - –RESTful API server
- **Gadget Container JavaScript**
  - –Core gadgets JavaScript environment

The Apache Software Foundation
http://www.apache.org/

How Shindig works

- Gadget Server
- OpenSocial Data Server



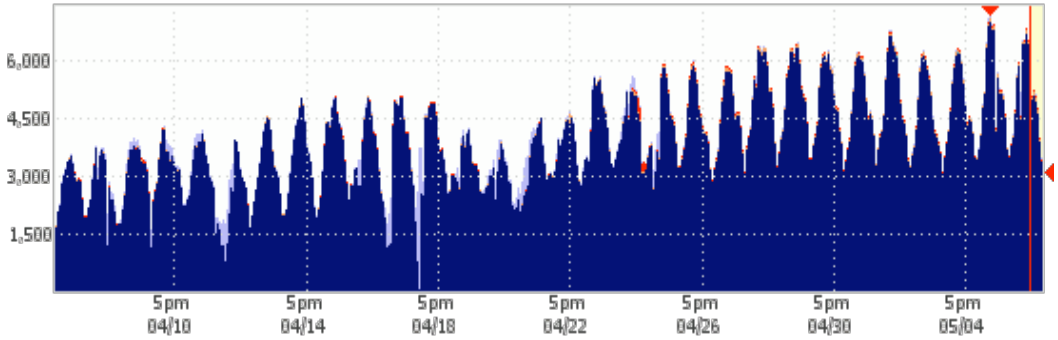
36

Why use Shindig?

- Strong open source community
- High quality production-ready code
- Update easily as OpenSocial evolves
- Fully compliant with OpenSocial v0.7, v0.8
- Shindig is language neutral (Java, PHP)

Shindig success at hi5

- Big Traffic
  - 10k req/sec Edge
  - 6k req/sec Origin

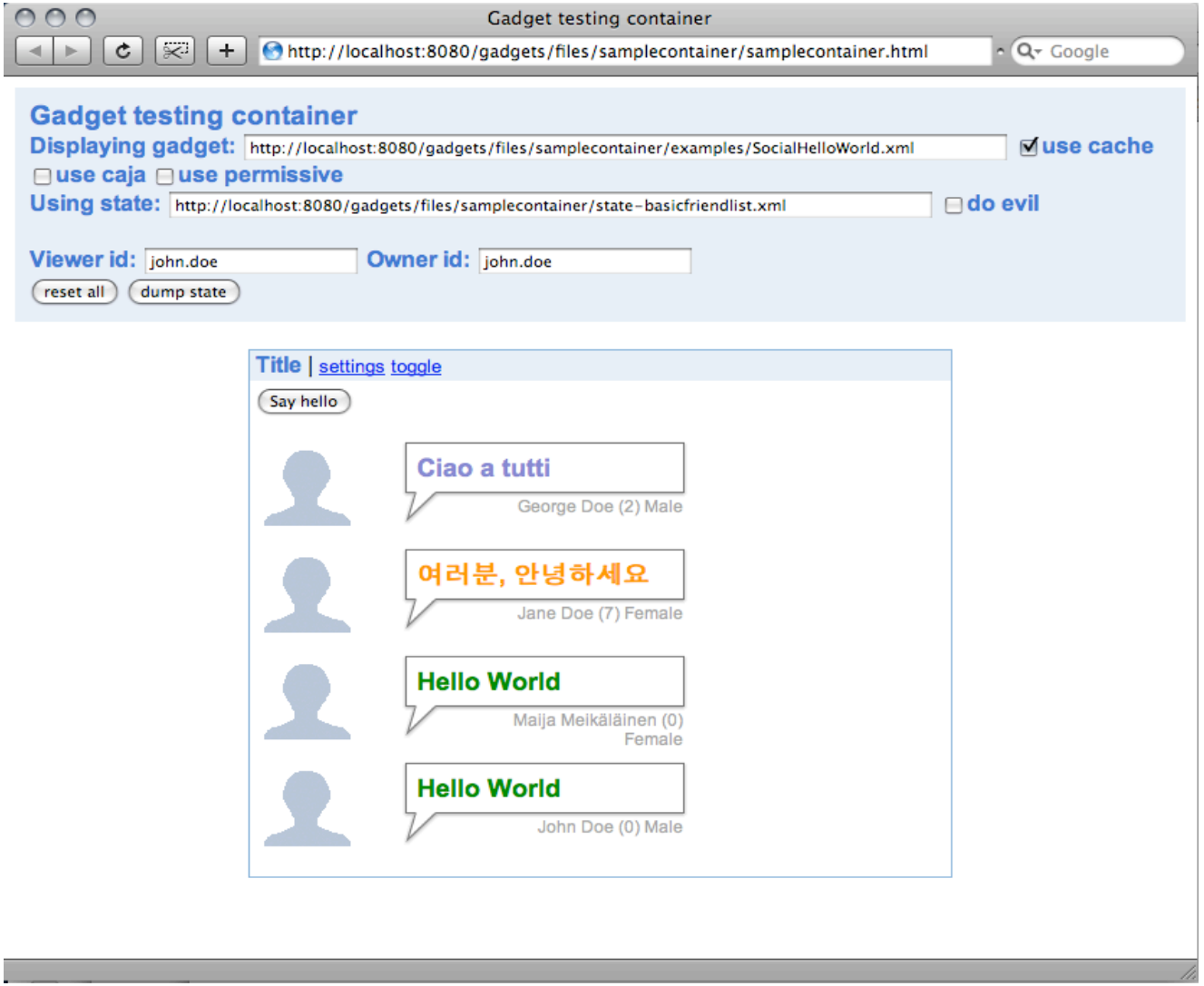


- Hundreds of Developers
- 1800+ Apps
- 1 Billion hits/day

… on 42 Shindig servers

38

Demonstration: Trying out Shindig



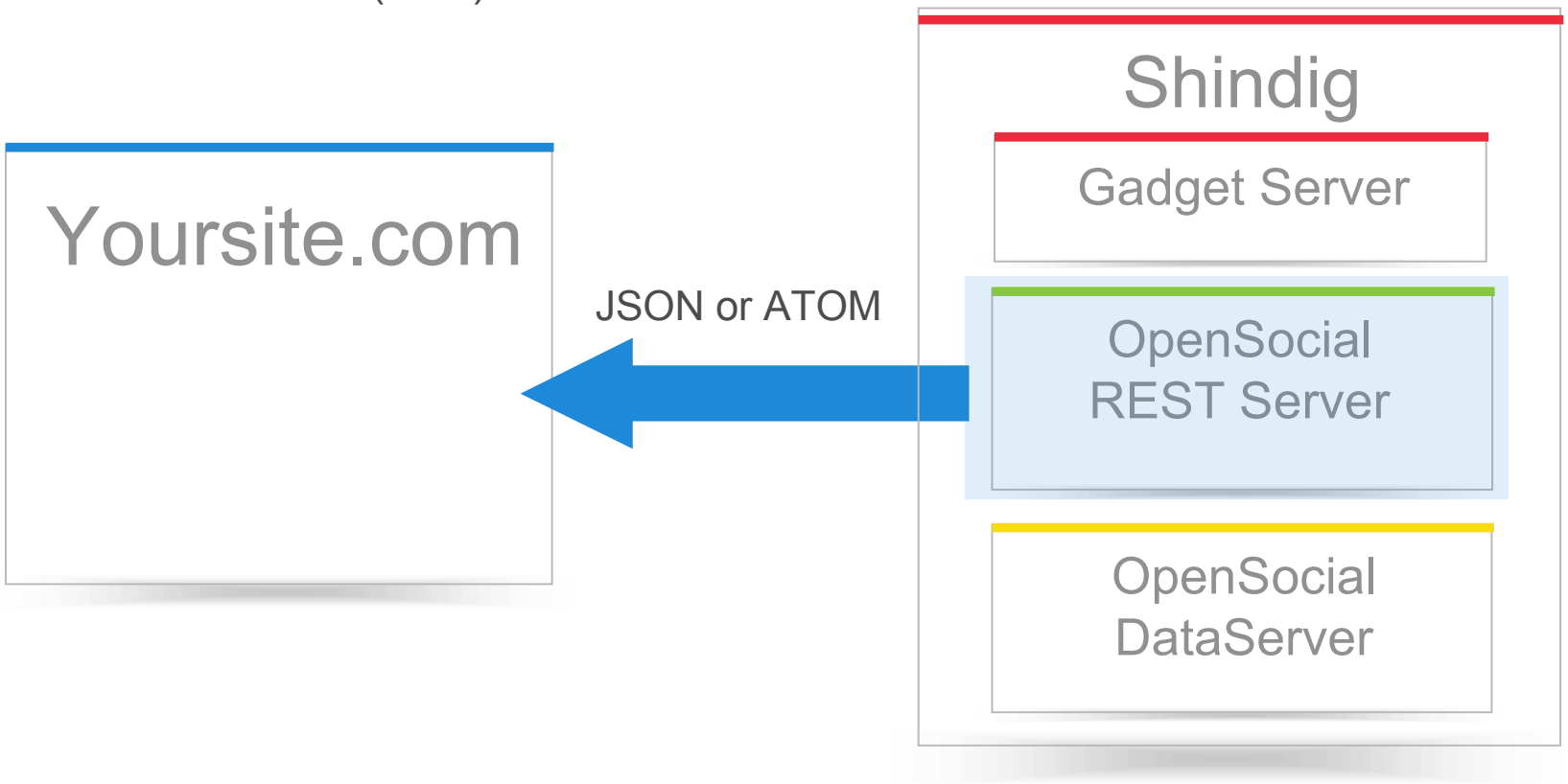
39

Interacting with Shindig's REST API

- Access OpenSocial data without JavaScript
  - Use simple REST calls instead
  - Does not require gadget container
  - Allows server-to-server communication
- Easily integrate with other server-side technologies: Java, PHP, Perl

Shindig's REST Server (new!)

- Gadget Server
- OpenSocial Data Server
- REST Server (v0.8)



41

REST API

Accessing People information

```
/people/{guid}/@all
  -- Collection of all people connected to user {guid}

/people/{guid}/@friends
  -- Collection of all friends of user {guid}
  -- subset of @all

/people/{guid}/@self
  -- Profile record for user {guid}

/people/@me/@self
  -- Profile record for requestor
```


42

REST API

Accessing Activities information

```
/activities/{guid}/@self
  -- Collection of activities generated by given user

/activities/{guid}/@friends
  -- Collection of activities for friends of the given
user {guid}
```


43

REST API

Accessing Persistent data

```
/appdata/{guid}/@self/{appid}
  -- All data for user {guid}, app {appid}

/appdata/{guid}/@friends/{appid}
  -- All data for friends of user {guid} and app
     {appid}; read-only
```


44

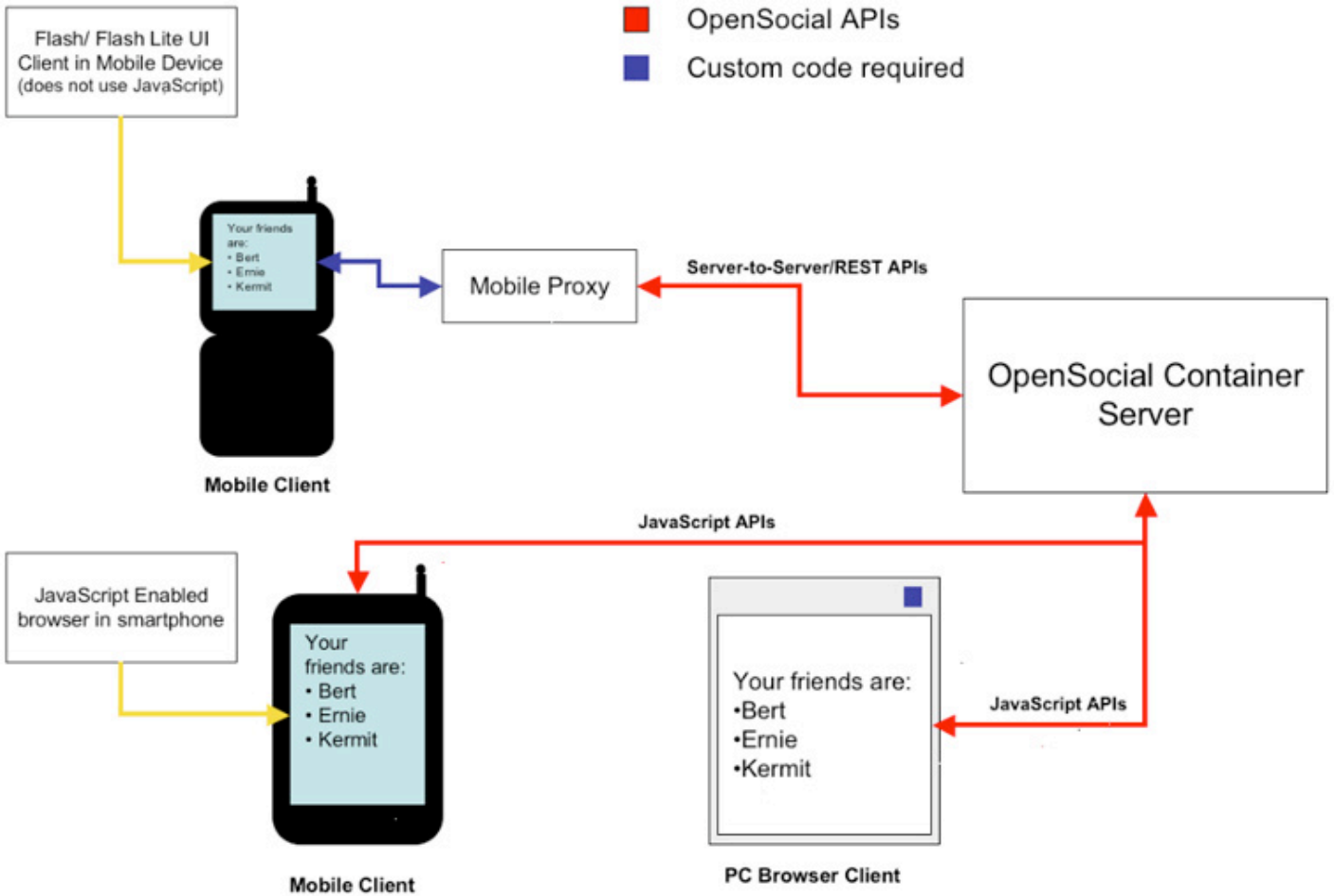
REST API

Additional query parameters

```
format={format}
  -- Format desired(atom, json). Default is json

fields={field+}
  -- List of fields to include in request

startPage={startPage}
  -- Index into a paged collection

count={count}
  -- Set page size for paged collection
```

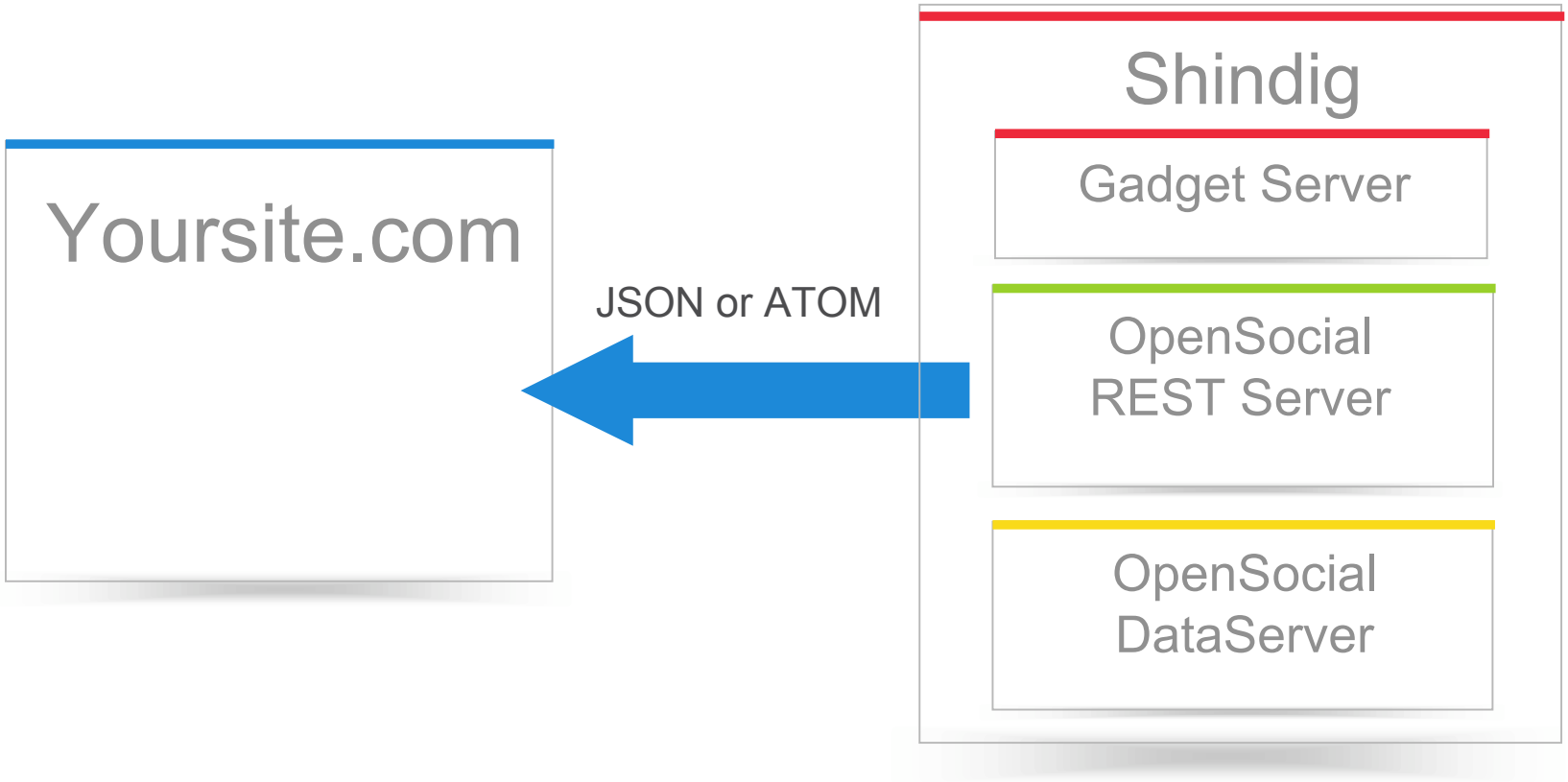

45

Opensocial RESTful API and Mobile



Demonstration

- Using the OpenSocial REST API



Agenda

- OpenSocial Concepts
- Building OpenSocial Applications
- Hosting OpenSocial Applications
- Google Friend Connect

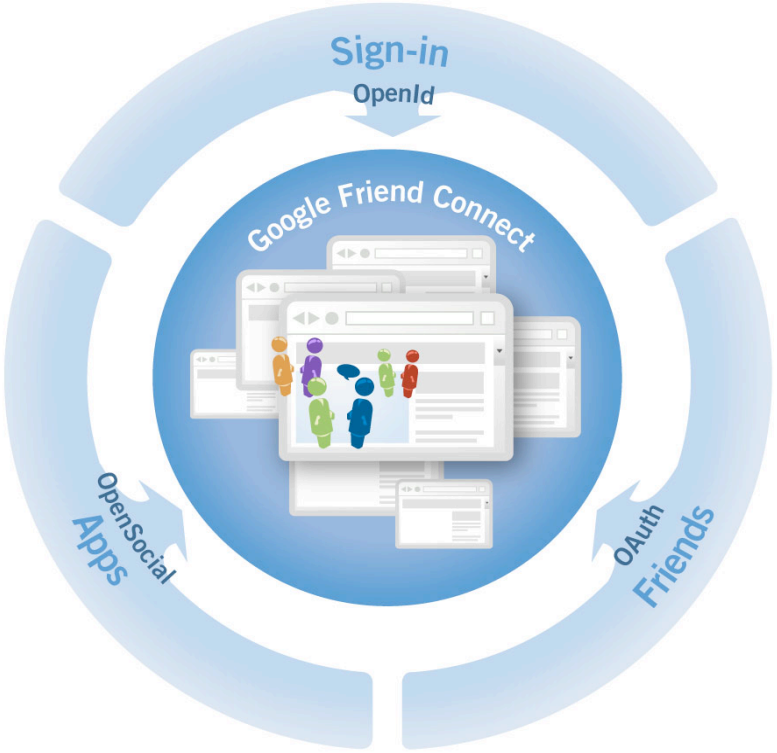
Google Friend Connect

- Personal Web sites (e.g.blogs) have readers, contacts, connections for a social graph
    - Can small personal websites host OpenSocial applications?

Problem: blogger's/Webmasters typically lack development resources!

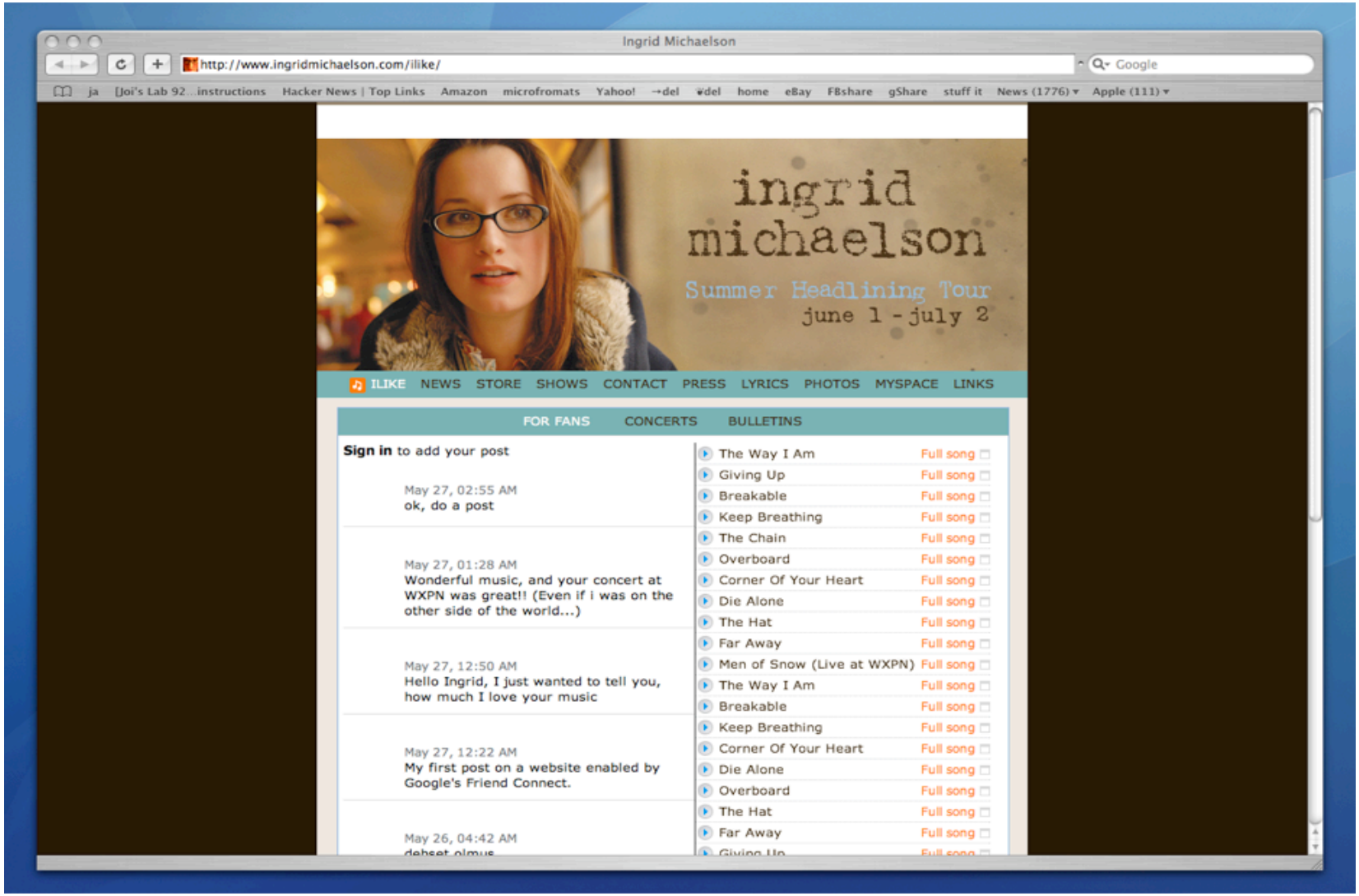
Solution: Google Friend Connect solves this problem!

Google Friend Connect

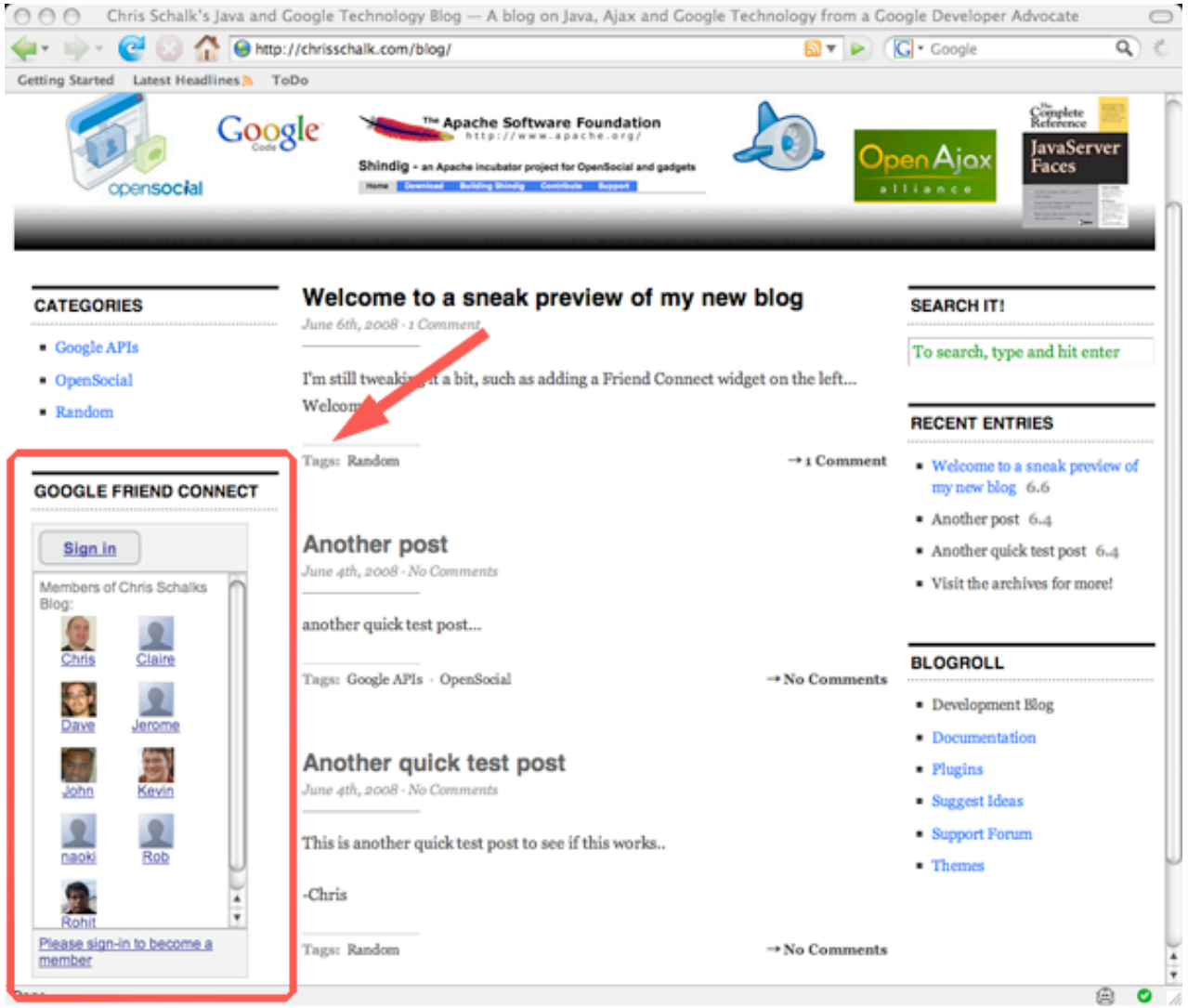


A *hosted* OpenSocial container solution

ingridmichaelson.com on Friend Connect



My new blog on Friend Connect



Google Friend Connect

- Users
  - … more ways to do more things with my friends
- Site owners
  - … more (and more engaged) traffic for my site
- App developers
  - ... more reach for my apps
- Sign up for the preview release!

  http://google.com/friendconnect/

Demonstration:

Google Friend Connect on my blog

Useful Links

Homepage & specification:

- http://www.opensocial.org

Get on the forums:

- http://groups.google.com/group/opensocial

Subscribe to the Shindig mailing list:

- shindig-dev-subscribe@incubator.apache.org

Help shape the specification:

- http://groups.google.com/group/opensocial-and-gadgets-spec/

Templates:

- http://groups.google.com/group/opensocial-and-gadgets-spec/web/opensocial-templates

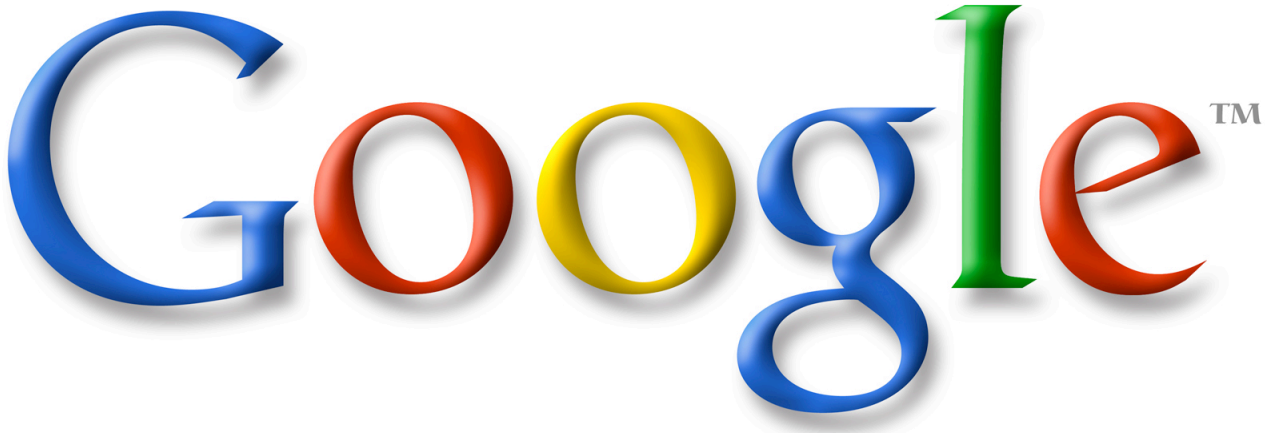
Check out Shindig:

- http://incubator.apache.org/shindig

Questions - Contact Info

Q&A

cschalk@google.com


Google™