



opensocial



Widget Summit

## OpenSocial Basics

Chris Schalk, Google Developer Advocate  
Paul Lindner, Engineering Manager, hi5

11/04/2008



# Agenda

---

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



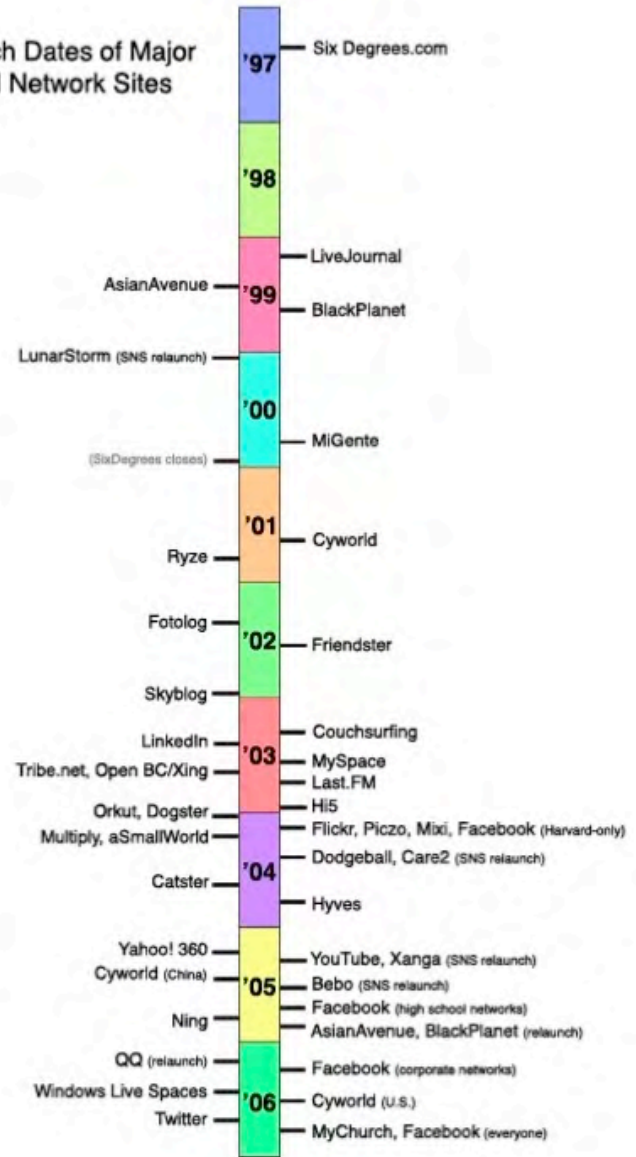
# Agenda

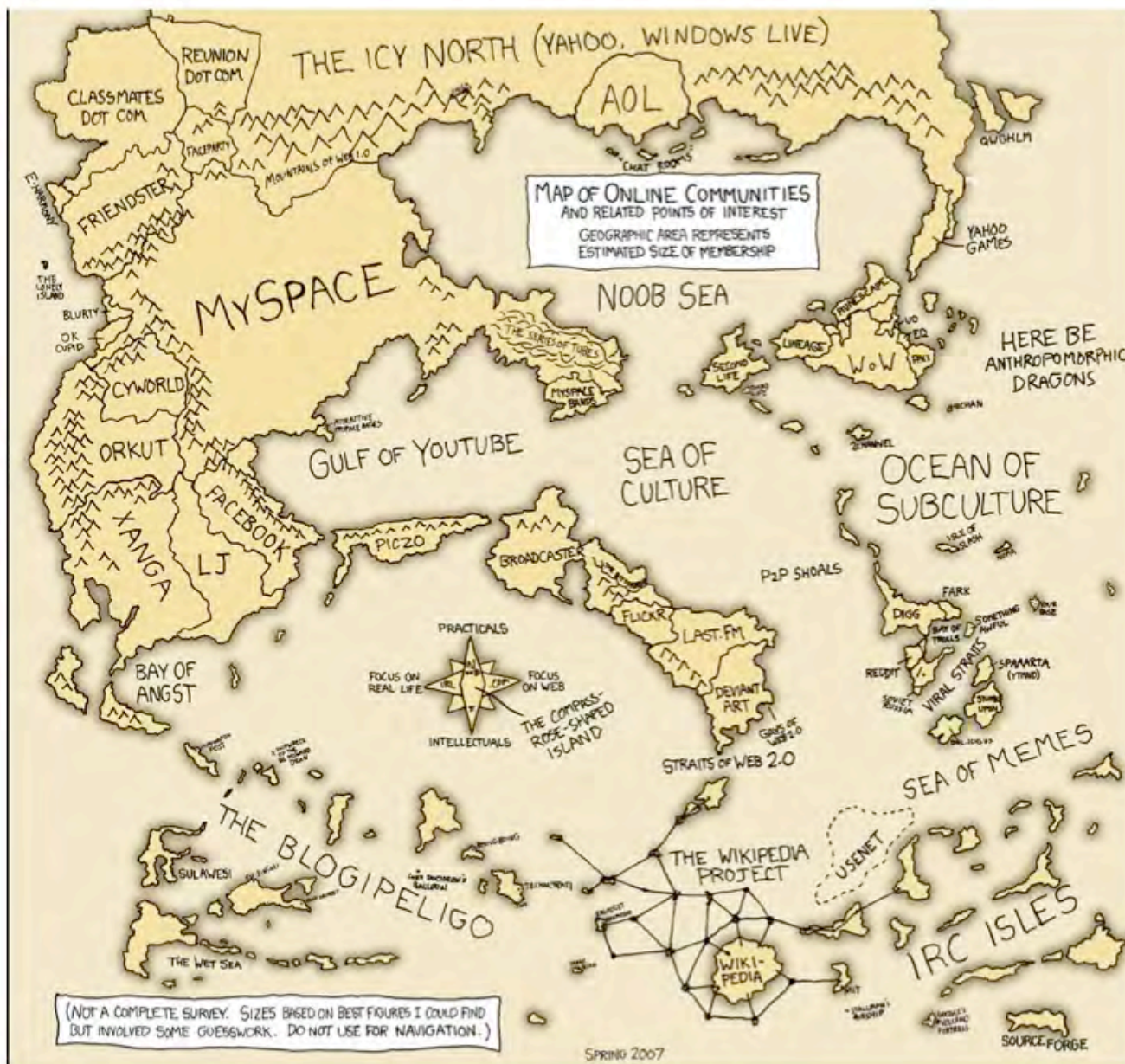
---

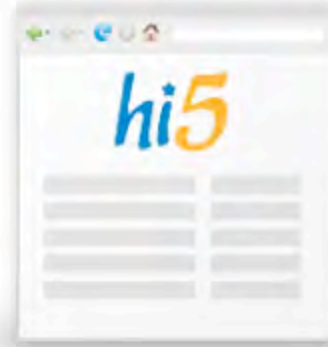
- **OpenSocial Concepts**
- Building OpenSocial Applications
- Hosting OpenSocial Applications
- Useful Updates
- Google Friend Connect



### Launch Dates of Major Social Network Sites







Too many platforms!



Standards based  
**OpenSocial**

API



Friends  
Profiles  
Activities



Write once...  
Deploy everywhere



Use multiple times  
Cooooool!



## Not just Social Network Sites

---

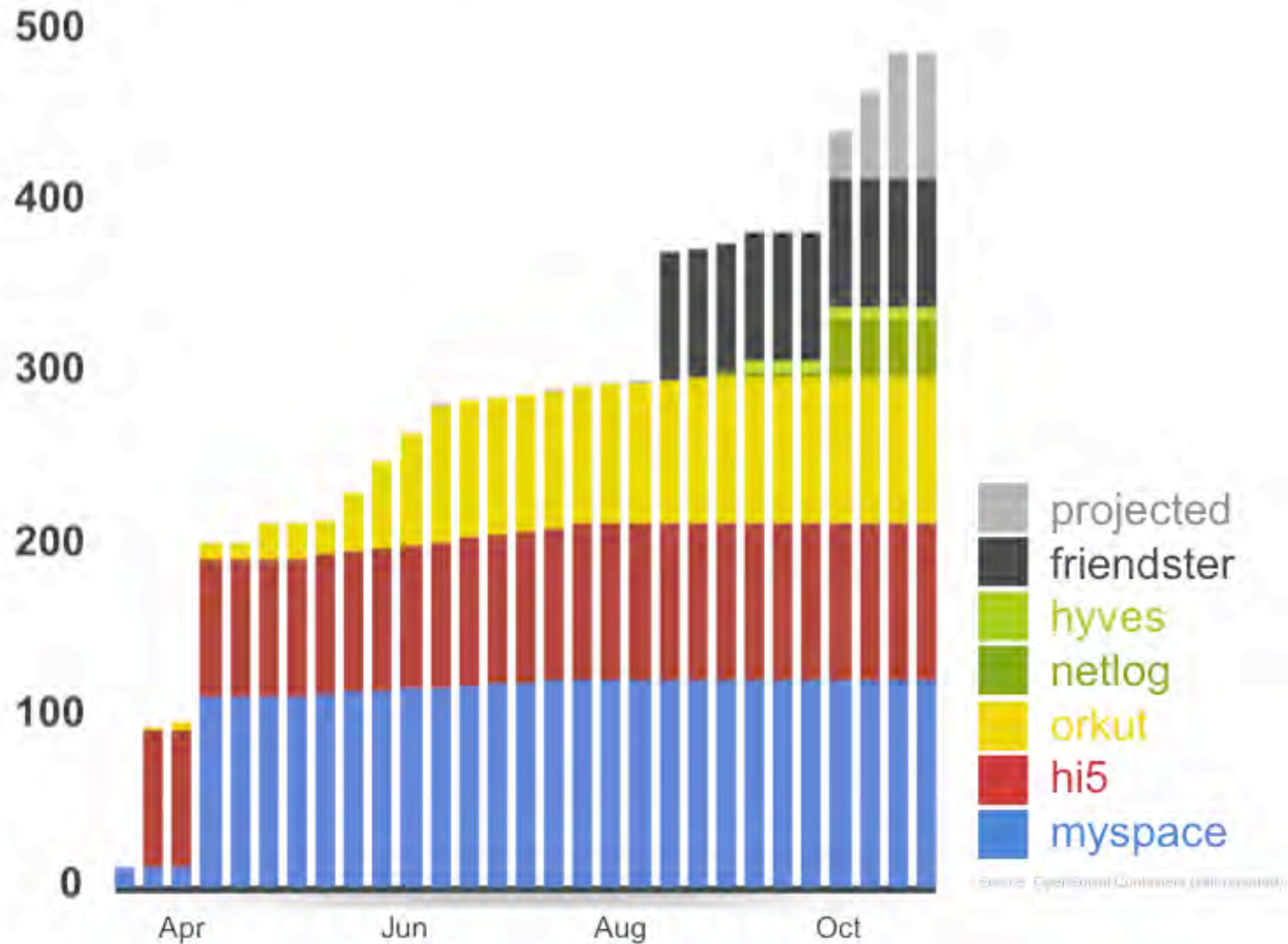
- Social network sites - Profiles and home pages  
ex: MySpace, Hi5
- Personal dashboards – Widgets  
ex: iGoogle, My Yahoo
- Sites based around a Social Object  
ex: Flickr, YouTube
- Enterprise, Professional  
ex: LinkedIn, Oracle, Salesforce.com
- Any web site

# OpenSocial Partners



# 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
- Hyves
- LinkedIn - **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 Roadmap

---



- Version v0.8 is current
- Latest evolution of OpenSocial as defined by the OpenSocial development community
  - Updated JavaScript API
  - Now contains a RESTful API
- hi5, MySpace, orkut, iGoogle now support 0.8
- Specification:  
<http://www.opensocial.org/Technical-Resources/opensocial-spec-v08>



## OpenSocial v0.8.1 spec is here

---

- Specification is now published
- Updates include:
  - RESTful Protocol updates
  - JSON RPC Protocol
  
- Specification:  
<http://sites.google.com/a/opensocial.org/opensocial/Technical-Resources/opensocial-spec-v081>



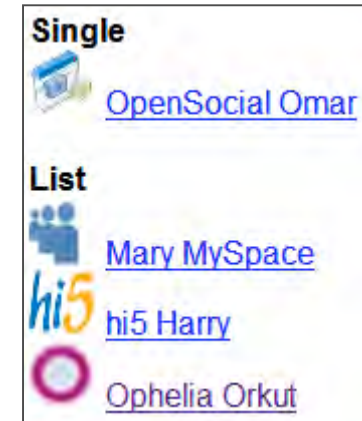
# OpenSocial 0.9 - Templating Proposal

---

- 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
- Useful Updates
- 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"];
  var p = {};

  p[opensocial.IdSpec.Field.USER_ID] = "VIEWER";
  var idSpec = opensocial.newIdSpec(p);

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





## Integrating OpenSocial applications with external servers

---

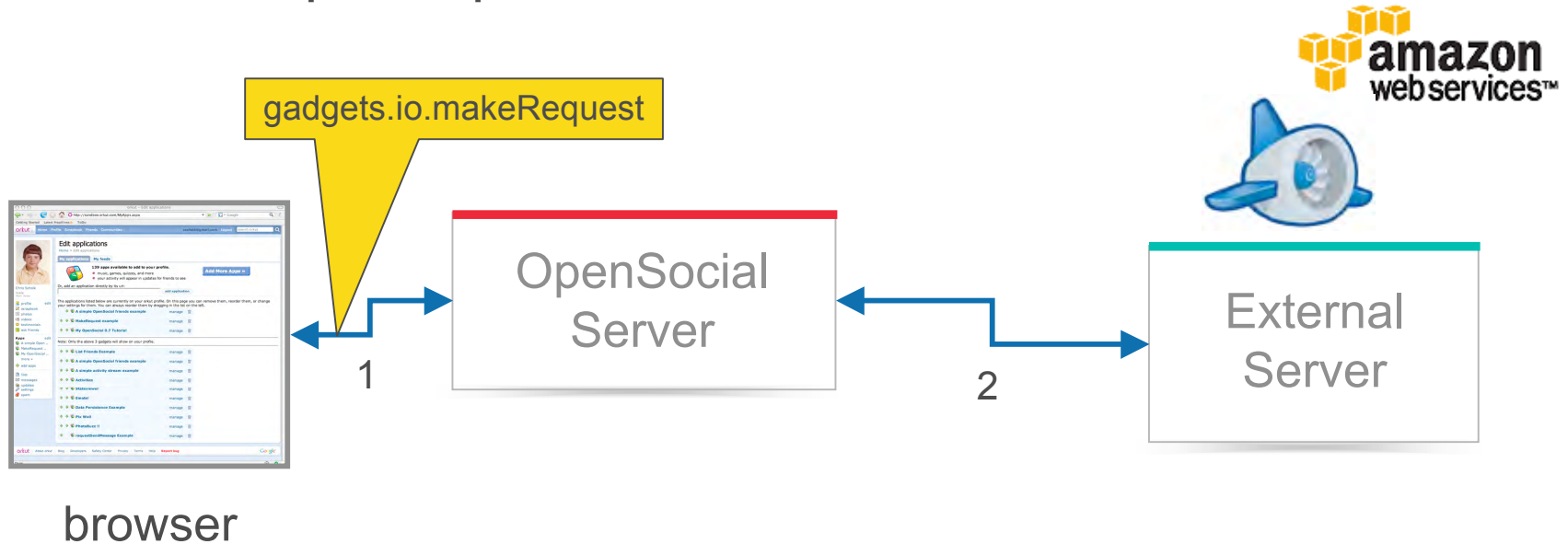
The OpenSocial **persistence** service provides a way to store/share small amounts of data on the OpenSocial server, but...

- 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

orkut - Edit applications

http://sandbox.orkut.com/MyApps.aspx


Getting Started Latest Headlines ToDo

orkut Home Profile Scrapbook Friends Communities zachalk@gmail.com Logout search orkut

## Edit applications

Home > Edit applications

My applications My feeds







 **139 apps available to add to your profile.**

- music, games, quizzes, and more
- your activity will appear in updates for friends to see




















[Add More Apps »](#)

Or, add an application directly by its url:  [add application](#)

The applications listed below are currently on your orkut profile. On this page you can remove them, reorder them, or change your settings for them. You can always reorder them by dragging in the list on the left.

-  **A simple OpenSocial friends example** manage 
-  **MakeRequest example** manage 
-  **My OpenSocial 0.7 Tutorial** manage 

Note: Only the above 3 gadgets will show on your profile.

-  **List Friends Example** manage 
-  **A simple OpenSocial friends example** manage 
-  **A simple activity stream example** manage 
-  **Activities** manage 
-  **Stateviewer** manage 
-  **Emote!** manage 
-  **Data Persistence Example** manage 
-  **Pix Wall** manage 
-  **PhotoBuzz !!** manage 
-  **requestSendMessage Example** manage 

orkut About orkut Blog Developers Safety Center Privacy Terms Help Report bug

Google

Done



# Agenda

---

- OpenSocial Concepts
- Building OpenSocial Applications
- **Hosting OpenSocial Applications**
- Useful Updates
- 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



# Serverside OpenSocial

---

How to implement an OpenSocial server

## Shindig

orkut, hi5, iGoogle, tianya.cn, ...

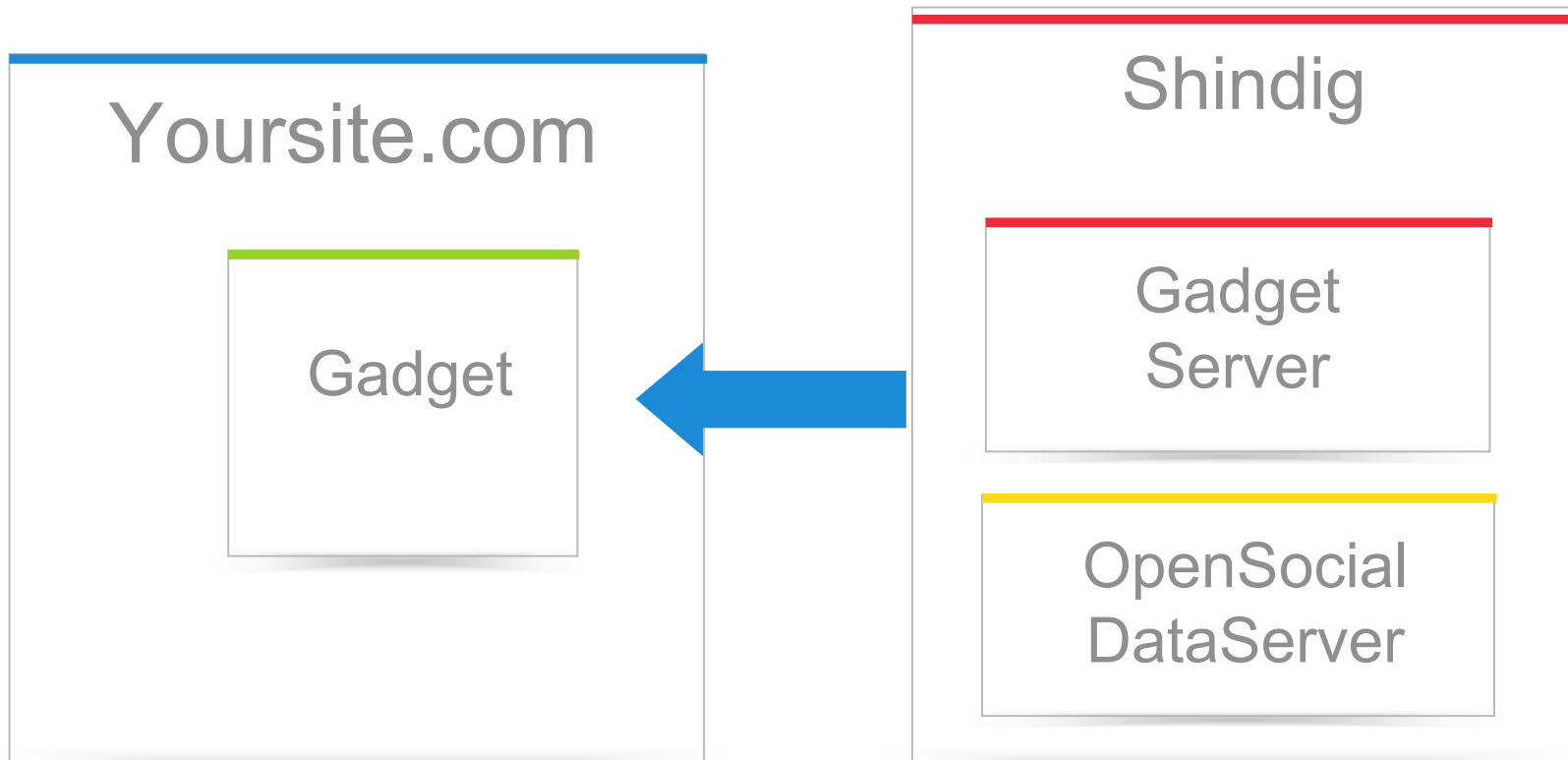
- **Gadget Server**
  - Renders gadget XML as HTML/JS/CSS
- **OpenSocial Data Server**
  - RESTful API server
- **Gadget Container JavaScript**
  - Core gadgets JavaScript environment



# How Shindig works

---

- Gadget Server
- OpenSocial Data Server

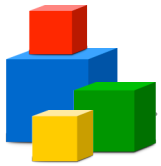




## Why use Shindig?

---

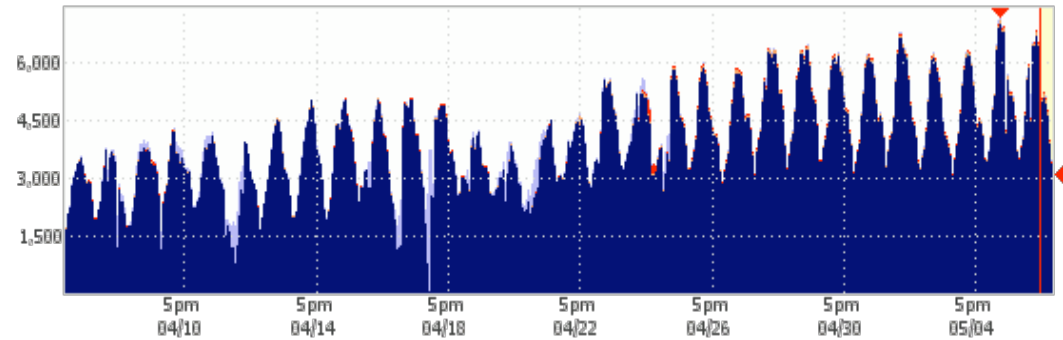
- Strong open source community
- High quality production-ready code
- Update easily as OpenSocial evolves
- Fully compliant with OpenSocial v0.7, v0.8
- Shindig is language neutral (Java, PHP)



## 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





# Demonstration: Trying out Shindig

Gadget testing container

http://localhost:8080/gadgets/files/samplecontainer/samplecontainer.html

**Gadget testing container**





Displaying gadget:   use cache

use caja  use permissive

Using state:   do evil

Viewer id:  Owner id:

**Title** | [settings toggle](#)

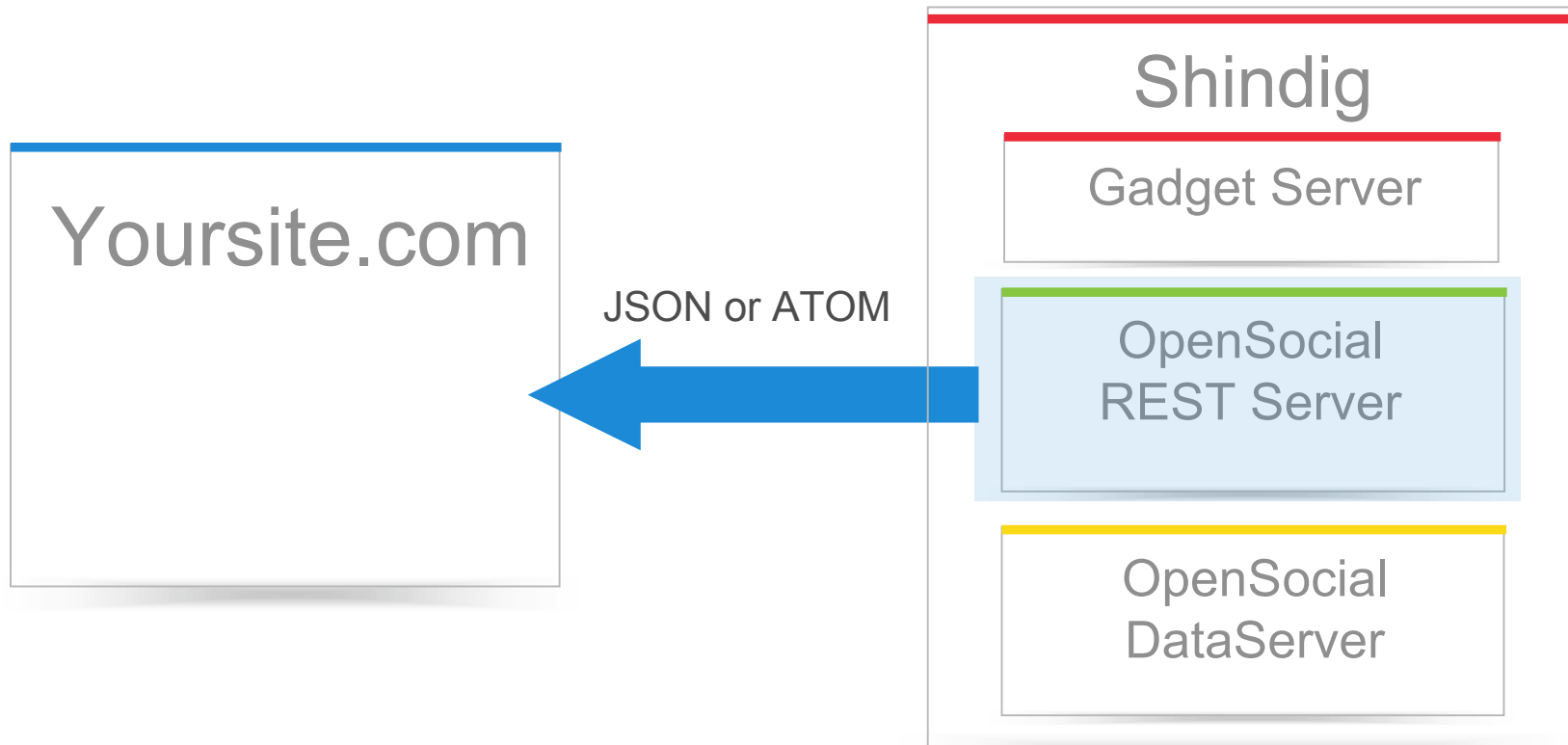
-  **Ciao a tutti**  
George Doe (2) Male
-  **여러분, 안녕하세요**  
Jane Doe (7) Female
-  **Hello World**  
Maija Meikäläinen (0) Female
-  **Hello World**  
John Doe (0) Male



# Shindig's REST Server (v0.8)

---

- Gadget Server
- OpenSocial Data Server
- REST Server





# 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
```



# 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}
```



# 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
```



# 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}**

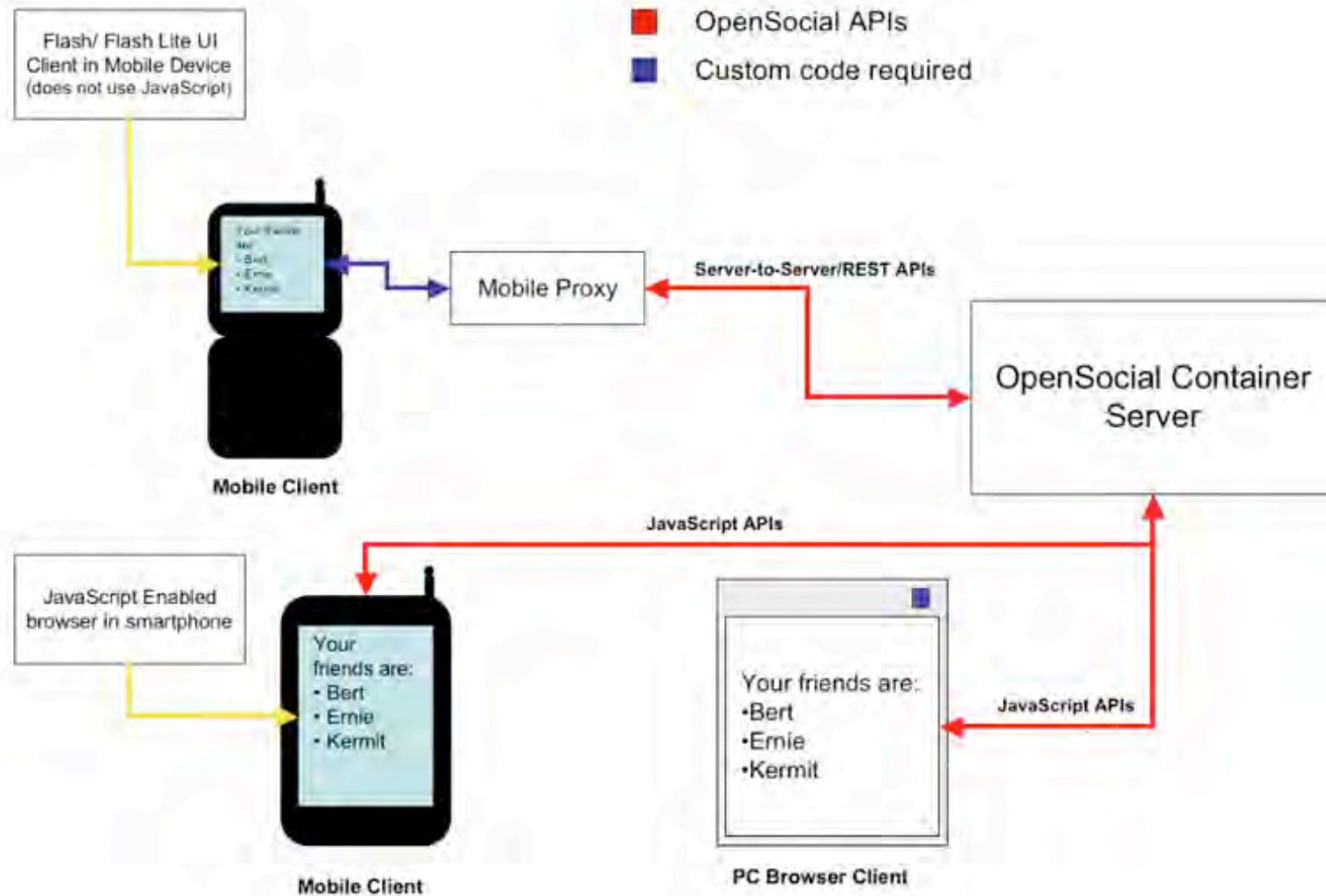
-- Index into a paged collection

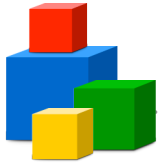
**count={count}**

-- Set page size for paged collection



# Opensocial RESTful API and Mobile

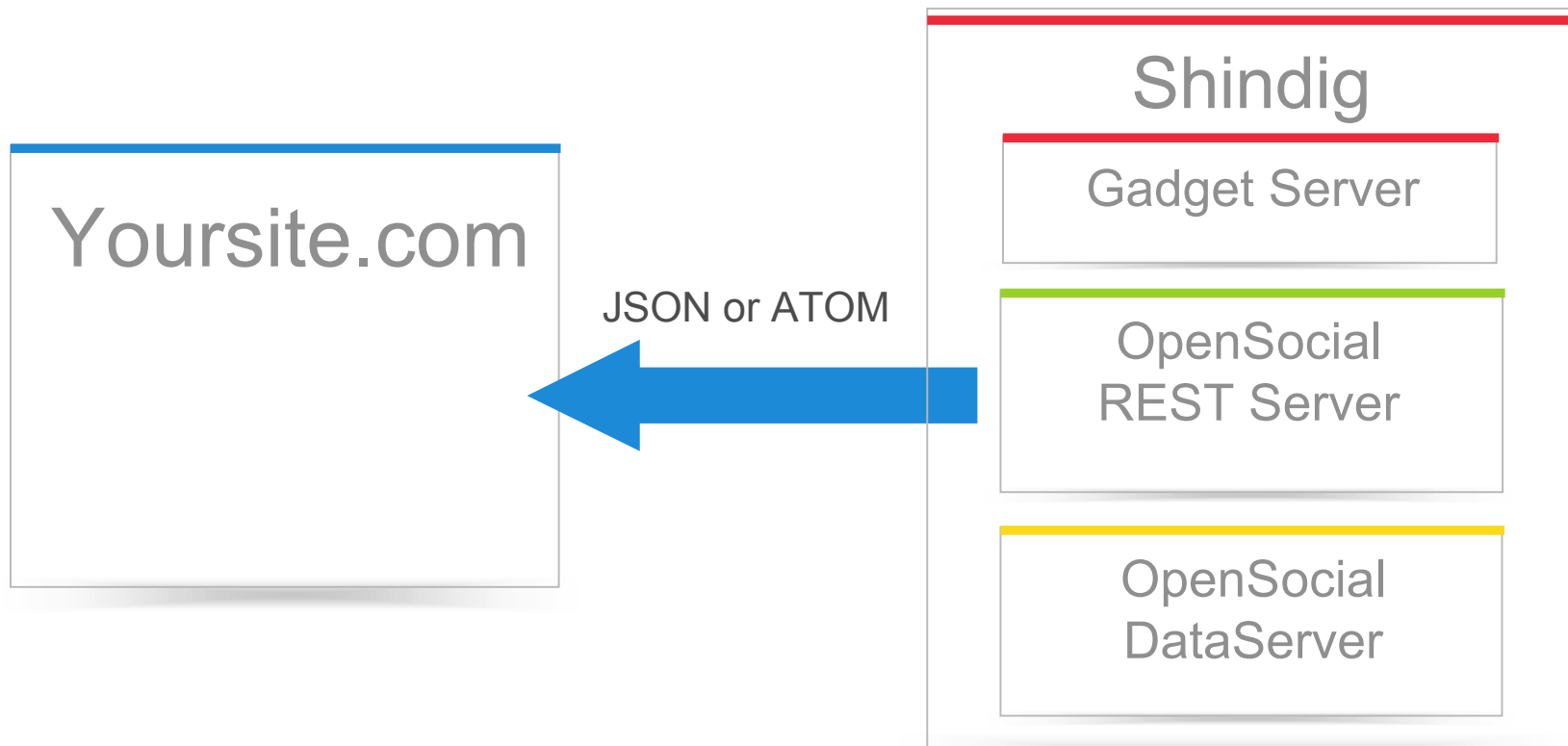




# Demonstration

---

- Using the OpenSocial REST API





# Agenda

---

- OpenSocial Concepts
- Building OpenSocial Applications
- Hosting OpenSocial Applications
- **Useful Updates**
- Google Friend Connect



## 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





# Useful Updates for OpenSocial Developers

- New OpenSocial Wiki Launched!
  - <http://wiki.opensocial.org>
  - Excellent articles & tutorials
  - Coverage on cross container development
    - Compliance Tests, Matrix

Total		Change [+/-]		
		myspace.com: 0.7 [x] 2008-9-9	hi5.com: 0.7 [x] 2008-9-26	orkut.com: 0.7 [x] 2008-10-11
	161 tests	57 tests 6 tests 77 tests 21 tests	114 tests 10 tests 14 tests 20 tests	122 tests 14 tests 5 tests 15 tests

Priority	People/Person Requests Test Suite	myspace.com	hi5.com	orkut.com
P0	opensocial.hasPermission(VIEWER)	Pass	Pass	Pass
P0	newFetchPeopleRequest - String ID	Unverified	Unverified	Unverified
P0	newFetchPeopleRequest - Array<String> IDs	Unverified	Unverified	Unverified
P0	newFetchPersonRequest - VIEWER/OWNER (all person fields)	Failed	Pass	Pass
P0	newFetchPersonRequest - VIEWER/OWNER (no parameters)	Failed	Pass	Pass
P0	newFetchPersonRequest - String ID	Failed	Pass	Pass
P0	newFetchPeopleRequest - VIEWER_FRIENDS (default - no params)	Failed	Pass	Pass
P0	newFetchPeopleRequest - VIEWER_FRIENDS (profile_details basic)	Failed	Pass	Pass
P0	newFetchPeopleRequest - VIEWER_FRIENDS (topFriends)	Failed	Pass	Pass
P0	newFetchPeopleRequest - VIEWER_FRIENDS (hasApp)	Unverified	Unverified	Pass
P0	newFetchPeopleRequest - OWNER_FRIENDS (default - no params)	Failed	Pass	Pass
P0	newFetchPeopleRequest - OWNER_FRIENDS (profileDetails all) ACL	Failed	Pass	Pass
P0	newFetchPeopleRequest - OWNER_FRIENDS (SortOrder TOP_FRIENDS)	Failed	Pass	Pass
P0	newFetchPeopleRequest - OWNER_FRIENDS (first, max)	Pass	Failed	Pass



# Useful Updates for OpenSocial Developers

---

- OpenSocial Foundation
  - <http://opensocial.org/opensocial-foundation>
  - Community Representative election Nov. 3rd
- OpenSocial Birthday
  - Nov. 13th, @ MySpace
  - Future of OpenSocial discussions
  - RSVP required - see OpenSocial blog



# 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

The screenshot shows a web browser window with the URL <http://www.ingridmichaelson.com/like/>. The page features a header with a photo of Ingrid Michaelson and the text "ingrid michaelson Summer Headlining Tour June 1 - July 2". Below the header is a navigation menu with links: ILIKE, NEWS, STORE, SHOWS, CONTACT, PRESS, LYRICS, PHOTOS, MYSPACE, LINKS. The main content area is divided into sections: "FOR FANS", "CONCERTS", and "BULLETINS". The "Sign in to add your post" section displays several user comments:

- May 27, 02:55 AM: ok, do a post
- May 27, 01:28 AM: Wonderful music, and your concert at WXPN was great!! (Even if I was on the other side of the world...)
- May 27, 12:50 AM: Hello Ingrid, I just wanted to tell you, how much I love your music
- May 27, 12:22 AM: My first post on a website enabled by Google's Friend Connect.
- May 26, 04:42 AM: debut album

To the right of the comments is a list of songs, each with a "Full song" link:

- The Way I Am
- Giving Up
- Breakable
- Keep Breathing
- The Chain
- Overboard
- Corner Of Your Heart
- Die Alone
- The Hat
- Far Away
- Men of Snow (Live at WXPN)
- The Way I Am
- Breakable
- Keep Breathing
- Corner Of Your Heart
- Die Alone
- Overboard
- The Hat
- Far Away
- Giving Up



# My new blog on Friend Connect

The screenshot shows a web browser window displaying a blog page. The browser's address bar shows the URL <http://chrischalk.com/blog/>. The page features a navigation menu with links for "Getting Started", "Latest Headlines", and "ToDo". Below the navigation menu is a banner with logos for "opencsocial", "Google Code 4", "The Apache Software Foundation", "Shindig", "Open Ajax", and "The Complete Reference JavaServer Faces".

The main content area is divided into three columns. The left column, titled "CATEGORIES", lists "Google APIs", "OpenSocial", and "Random". The middle column contains a post titled "Welcome to a sneak preview of my new blog" dated "June 6th, 2008" with "1 Comment". The text of the post reads: "I'm still tweaking it a bit, such as adding a Friend Connect widget on the left...". A red arrow points from this text to a "GOOGLE FRIEND CONNECT" widget on the left side of the page. The widget includes a "Sign in" button and a list of members: Chris, Claire, Dave, Jerome, John, Kevin, naoki, Rob, and Rohit. Below the list is a link: "Please sign-in to become a member".

The right column contains a "SEARCH IT!" section with the text "To search, type and hit enter", a "RECENT ENTRIES" section listing "Welcome to a sneak preview of my new blog" (6-6), "Another post" (6-4), "Another quick test post" (6-4), and "Visit the archives for more!", and a "BLOGROLL" section listing "Development Blog", "Documentation", "Plugins", "Suggest Ideas", "Support Forum", and "Themes".



# 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](mailto: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

[chrisschalk.com](http://chrisschalk.com)