

OpenSocial

A standard programming model for the Social Web



SOCIAL NETWORKING CONFERENCE

January 31 2008, Miami



Patrick Chanezon
Developer Advocate
chanezon@google.com

What does Social mean?



Eliette what do you do with your friends?



WE
LOOK
AT
EACH
OTHER.



TALKING



LAUGHING



WE
HELP
EACH
OTHER!



WE
READ

TOGETHER!



WE
DO PROJECTS
TOGETHER.

Jaiku's Jyri Engeström's 5 rules for social networks: social objects

1. What is your object?
2. What are your verbs?
3. How can people share the objects?
4. What is the gift in the invitation?
5. Are you charging the publishers or the spectators?

<http://tinyurl.com/yus8gw>



Making the web better
by making it social



facebook.

LinkedIn

hi5

viadeo

Ning



orkut^{beta}





API



Friends
Profiles
Activities







OpenSocial

A common open set of APIs for
building social applications across
multiple sites

This is NOT GoogleSocial.
It's about making the Web more
social, not just Google.





Developers

Distribution, distribution,
distribution

Learn once, write anywhere

Ning

plaxo

friendster

XING

hi5

orkut beta viadeo

six apart

salesforce.com
Success On Demand

LinkedIn

ORACLE

mixi mixi, Inc.

天際網
www.tianji.com

ENGAGE

ebo

imeem

Hyves.net
always in touch with your friends

OpenSocial

THEIKOS
Delivering Results On-Demand

RockYou!



FotoFlexer

virtual
Tourist
Real Travelers - Real Info

iLike

Flixster

slide

Standards-based

html+javascript

Websites

Large pool of developers

Consumers

More more more

Business is social too

XING  Linked  viadeo

salesforce.com[®]
 Success On Demand.™

ORACLE[®]

How does it work?

10 minutes to an OpenSocial app

One API, Many Websites

One API

- client-side JavaScript - version 0.7 ready for production
 - standard Web development tools: HTML + Javascript
 - server optional
- server-side REST (not specified yet)
 - Google proposal based on Atom Publishing Protocol
 - standard XML schema

Many Websites

- every OpenSocial website exposes the same API

==> more users for every app

==> more apps for every user

Core Services

- People ("who I am")
- Friends ("who I know")
- Activities ("what I'm doing")

- Persistence (state without a server)

Low-level Implementation

```
function getOpenSocialData(callback) {
  var req = opensocial.newDataRequest();
  req.add(req.newFetchPersonRequest('VIEWER'), 'v');
  req.add(req.newFetchPeopleRequest('VIEWER_FRIENDS'), 'vf');
  req.add(req.newFetchActivitiesRequest('VIEWER'), 'av');
  req.add(req.newFetchPersonAppDataRequest('VIEWER', keys), 'vd');
  req.send(callback)
}
```

Step 1: Hello World

```
function myInit() {  
    var html;  
    getOpenSocialData();  
    html = "Welcome, " + data.getViewer().getDisplayName();  
    byId("main").innerHTML = html;  
}
```

Caja Javascript sanitizer

- Open Source project from Google
- Optional but recommended for OpenSocial containers
- Gadgets can be a new vector for phishing, spam, malware
- Social spread of gadgets can spread bad gadgets too
- Caja reduces threats with a JavaScript sanitizer as an additional "sandbox" on top of iFrames protection
- The Caja sandbox/sanitizer will eventually be secure enough to run gadgets inline instead of in iframes to improve performance

Who should care about Caja?

- OpenSocial Containers can choose to require Caja's extra security, and decide how to handle gadgets that cannot be Cajoled. For example, you might show users an additional warning before they add a non-Cajoled gadget. Of course, you have the option to not use Caja at all.
- Developers writing OpenSocial gadgets may find that some large containers require Cajoled gadgets or warn users of uncajolable gadgets.
- Users, containers and developers will get benefits from Cajoled gadgets running inline (without iFrames) in the long term

Shindig: What is it?

- Apache Software Foundation project
 - Brian McCallister from Ning championed
- Open source reference implementation of OpenSocial + Gadgets stack
- Goal:
 - Launch a new (simple) container in under an hour's worth of work

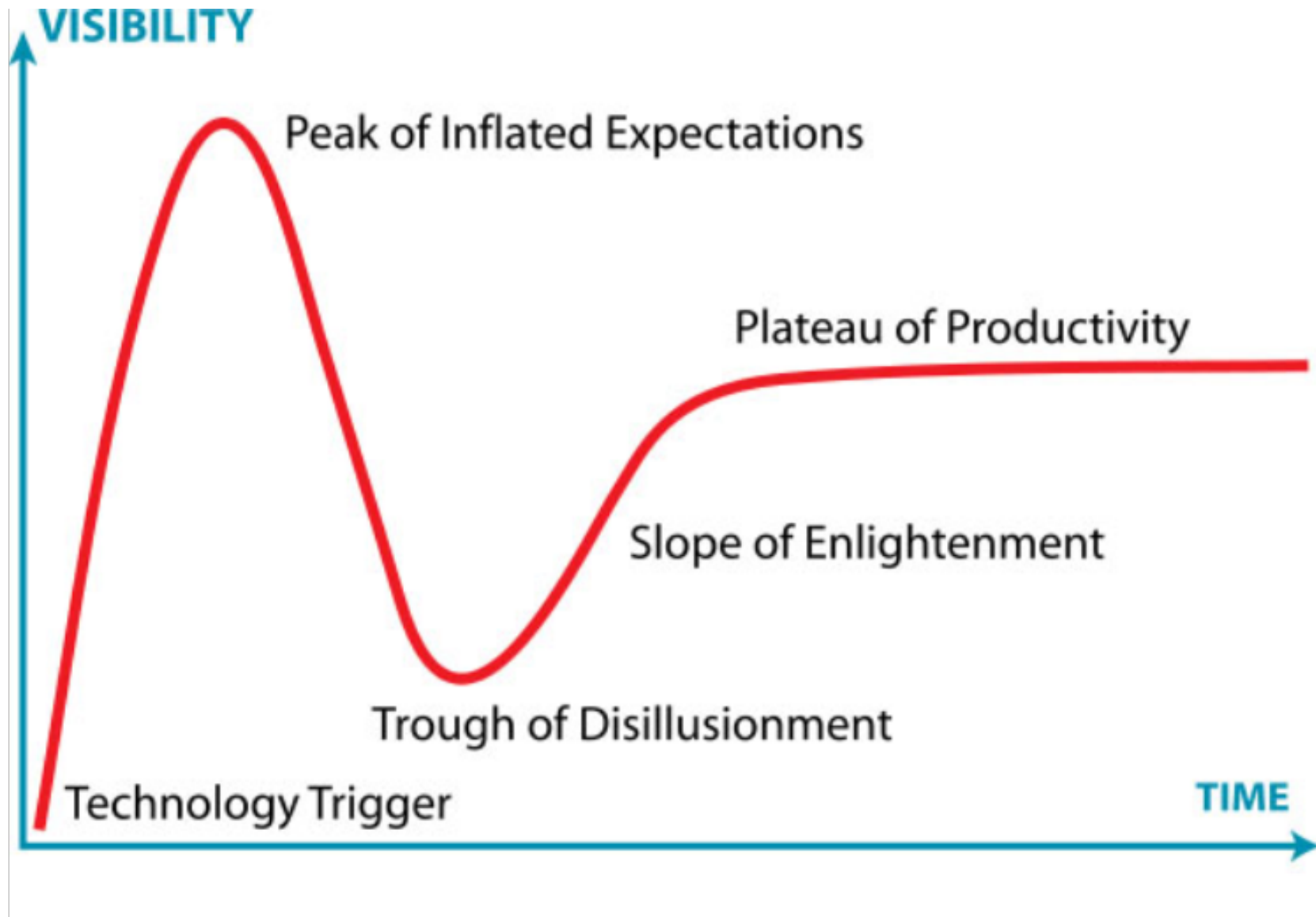
Shindig: Components

- Gadget Container JavaScript
 - Core gadgets JavaScript environment
- Gadget Server
 - Renders gadget XML (i.e. gmodules.com)
- OpenSocial Container JavaScript
 - JavaScript environment for people, activities, persistence
- OpenSocial Gateway Server
 - RESTful API server

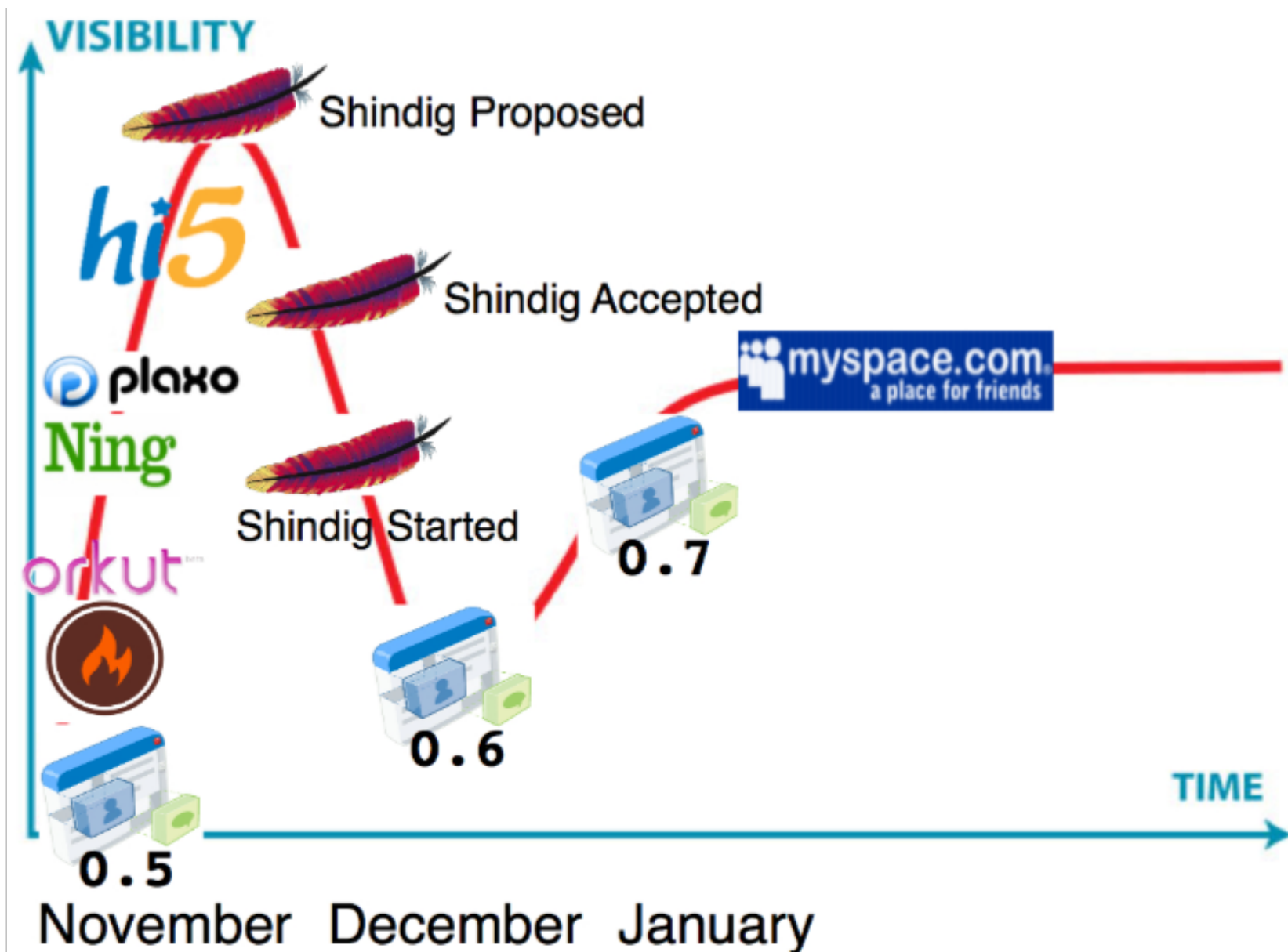
Code Status

- Not finished: javascript not plugged to java layer yet
- Googlers have contributed:
 - Java Gadget Server
 - Gadget Container JavaScript
 - OpenSocial Container JavaScript
- PHP Gadget Server contributed by Ning
- Rumors of Ruby, Python, Perl, C#, etc.

Gartner Technology Hype Cycle



OpenSocial Hype Cycle



Conclusion: it's time to get productive!

OpenSocial is a standard for social applications

The current version 0.7 is the version that will go in production

Developers can start creating social applications today

- Orkut sandbox, Hi5, Plaxo, Ning (and Feb 5 MySpace)

First containers to open for consumers in Q1

Social sites who want to implement OpenSocial should look at Shindig and start planning

Resources

Specification

<http://code.google.com/apis/opensocial/>

Sandboxes

<http://developer.myspace.com/>

<http://www.hi5networks.com/developer/>

<http://opensocial.ning.com/>

<http://pulse.plaxo.com/pulse/gadgets/>

<http://code.google.com/apis/orkut/>

For container developers

<http://opensocialapis.blogspot.com/2007/12/lets-get-this-shindig-started.html>

<http://incubator.apache.org/shindig/>

<http://code.google.com/p/google-caja>

My delicious feed: <http://del.icio.us/chanezon/opensocial>

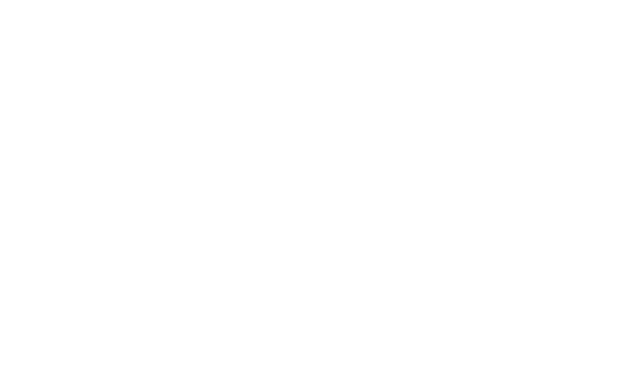
Creating OpenSocial was Social







TALKING





LAUGHING





WE
HELP
EACH
OTHER!





WE
READ
TOGETHER!



WE
DO PROJECTS
TOGETHER.





BUT
WHEN
WE ARE
MAD...



IT
NEVER
FINISH
WELL...



SO
WE
STOP
FIGHTING!

A standard for everyone



anyone
can be
friends!

Questions

