
Google™
10

Make your application real-time with PubSubHubbub

Brett Slatkin
May 19th, 2010

View live notes and ask questions about this session on Google Wave

http://tinyurl.com/push-io2010

Me
http://onebigfluke.com

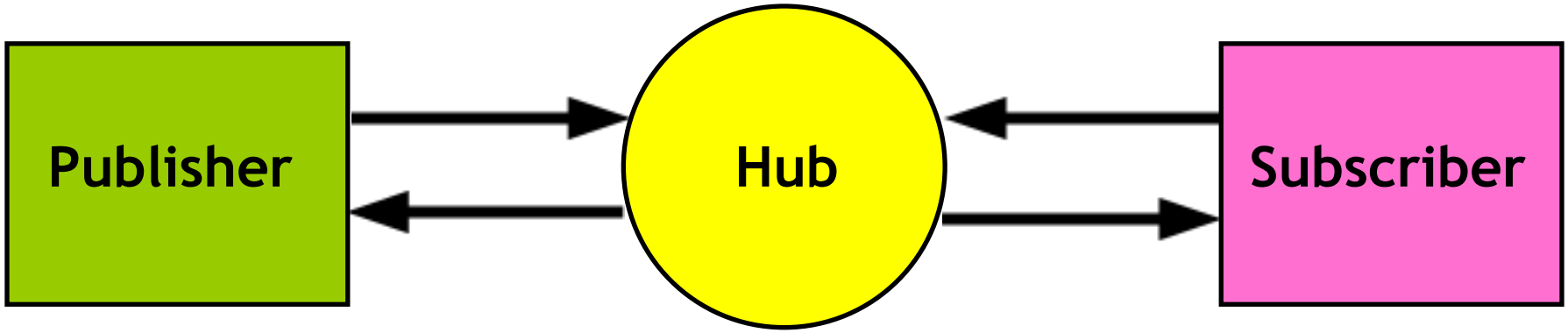
Agenda

- Intro
- Publishing
- Subscribing
- Hubs
- Special guest
- Progress & adoption
- Future work

Intro

What is PubSubHubbub?

- A simple, topic-based publish/subscribe protocol
- Turns Atom *and* RSS feeds into real-time streams
- **A single API** for web-scale, low-latency messaging
- Three participants: Publisher, Subscriber, Hubs



Design goals

- Decentralized: No one company in control
- Scale to the size of the whole web
- Publishing and subscribing as easy as possible
- Push any complexity towards the Hub
- Pragmatic (i.e., not theoretically perfect, but solve huge, known use cases with minimal effort)

Why another protocol?

- Almost every company already has an internal system
  - TIBCO, WebsphereMQ, ActiveMQ, RabbitMQ, ...
  - Proprietary message payloads, topics, networks

- Existing attempts at a standard haven't caught on
  - XMPP started in 1999, still isn't used for interop widely beyond IM (may change with OneSocialWeb.org?)
  - Overkill: XEP-0060, WS-*, AMQP, RestMS, new REST-*

How-to for publishers

1. Add a declaration in your feed with Hub(s) of choice

```
<link rel="hub"
 href="https://pubsubhubbub.appspot.com/"/>
```

2. Add something to your feed!

3. Send a ping to the Hub(s) with the feed URL

```
POST / HTTP/1.1
Content-Type: application/x-www-form-urlencoded
...

hub.mode=publish&hub.url=<your feed>
```

4. 204 = Success, 4xx = Bad request, 5xx = Try again

Publisher best practices

- Use URLs for server-side filtering
  - http://example.com/stuff?zip=94105
- Use URLs for authorization

How-to for subscribers

1. Detect the Hub(s') declaration in a feed

2. Send a subscribe request to the feed's Hub(s)

```
POST / HTTP/1.1
Content-Type: application/x-www-form-urlencoded
...
hub.mode=subscribe&hub.verify=sync&
hub.topic=<feed URL>&hub.callback=<callback URL>
```

3. Hub(s) will send a request to verify the subscription

```
GET /callback?hub.challenge=<random> HTTP/1.1

HTTP/1.1 200
...
<echo random>
```

How-to for subscribers

Receive new content from the Hub(s)

```
POST /callback HTTP/1.1
Content-Type: application/atom+xml
...
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Awesome feed</title>
  <link rel="hub" href="http://pubsubhubbub.
appspot.com"/>
  ...
  <entry>
    ...
  </entry>
</feed>
```

The role of a hub

- Functions
  - Accept and verify subscriptions to new topics
  - Receive pings from publishers, retrieve content
  - Extract new/updated items from feed
  - Send all subscribers the new content
  - DoS protections

The role of a hub

- Logical component
  - Publishers may be their own Hub
  - Combined Hub/Publisher has p2p speed-up
- Quality
  - Scalability
  - Reliability

Julien Genestoux, Superfeedr

PubSubHubbub at Superfeedr

Google IO, May 2010

Superfeedr

1. Default hub
2. Hosted hubs
3. PubSubHubbub + Benefits

Default Hub

- Historical Superfeedr
- Avoid polling
  - Smart scheduling
  - Protocol mapping : RSSCloud, SUP, XML-RPC ping...
- Push to subscribers (XMPP too :D)
- Schema mapping

Default Hub

Focus on what really makes a difference : your core business!

Use-cases

- iPhone Notification : Urban Airship, Boxcar
- Feed reader : Webwag, Feedingo
- Desktop Notification : Adobe Wave
- Semantic search : Guzzle.it, Twingly!
- Social Web : SixApart

Hosted Hubs

- Don’t re-invent the wheel
- Don’t run/maintain/debug the wheel
- Your hub, YOUR data
- Analytics, callbacks and more

References

- Blogging 

- Social Nets 

- Media 

Schema Mapping

- Tons of different formats : RSS X, Atom Y
- Tons of different namespaces : Digg vs. Mixx vs. Yahoo Buzz. Same semantics
- Tons of invalid stuff (missing tags, date, unique id..)
- Location : Geo-RSS
- Social

Activity Streams

Extensions

- Digest Notifications (Heartbeat + Digest)
- Feed status (querying superfeedr)
- Subscription callback
- Virtual feeds

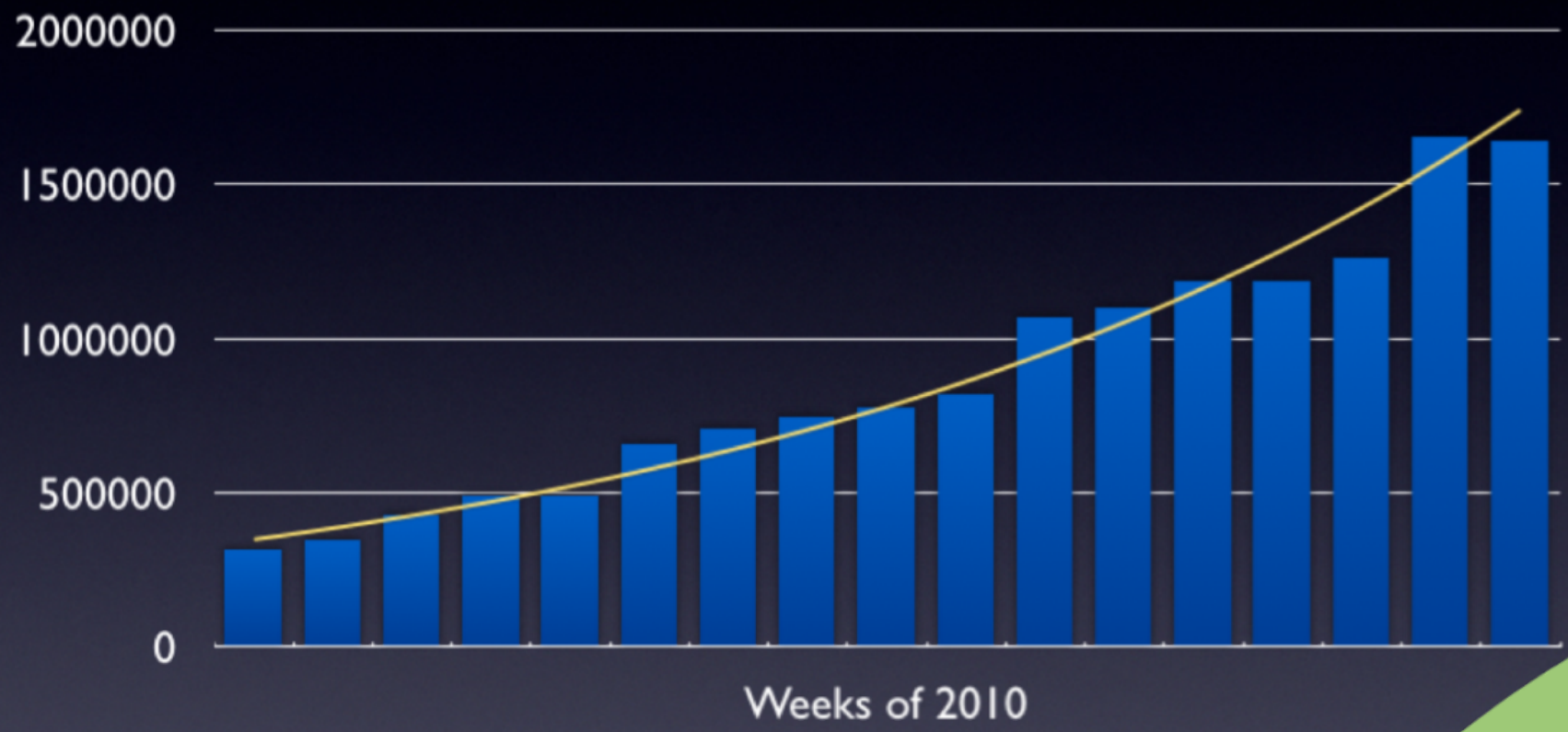
Infrastructure

- Botnet!
  - Independent XMPP workers with their own lifecycle.
  - Massive “Ring” for scheduling
  - Clustered cache for diff-ing

A few numbers

- Content pushed to 1.8M endpoints
- 20M+ of Atom entries pushed daily
- ~50 hosted hubs
- 45 “dispatchers”
- 80 “parsers”
- ~50 servers

A few numbers



TODO

- Make your feeds realtime
- Follow @superfeedr
- Say hello to @julien51
  - Pick Stickers
- Thanks

Progress

Adoption

- Over 100 Million feeds are PubSubHubbub-enabled
- Companies: Superfeedr (and friends), Google, Six Apart, LiveJournal, MySpace, TwitterFeed, Netvibes, Cliqset, Gnip, PostRank, ...
- Google products: Buzz, FeedBurner, Blogger, Reader shared items, Google Alerts, Fastflip, ...

Fun numbers from the reference Hub

- 200+ feed fetches per second (peak avg.)
- 250+ items delivered per second (peak avg.)
  - Includes item updates
- 70 million active subscriptions
- 1.2 billion items seen since July 2009

http://pubsubhubbub.googlecode.com

- Publisher clients: Perl, PHP, Python, Ruby, Java, Haskell, C#, MovableType, WordPress, Melody, Django, Zend, Drupal
- Subscriber clients/frameworks: PHP, .NET, Scala, Zend, Drupal, Django, Tornado, App Engine, NodeJS, Rails
- Hubs: App Engine, WordPress, Erlang, Twisted Python, Ruby, Perl, Django
- Active mailing list with 440+ members
- More publishers, subscribers, hubs on the way

Future work

In progress

- Arbitrary content types (JSON, HTML, XML)
  - Microformats folks want HTML push
  - Google wants XML Sitemaps updates
  - Plan to build on LRDD web linking
  - Facebook uses 1/2 of PuSH for their new APIs

In progress

- Private feeds
  - Fully encrypted, authorized, authenticated
  - Integration with OAuth, WebFinger
  - Apply business policies
  - Per-item privacy control

Further reading

OStatus
http://ostatus.org/

Buzz API
http://code.google.com/apis/buzz/

Facebook real-time API
http://developers.facebook.com/docs/api/realtime

View live notes and ask questions about this session on Google Wave

http://tinyurl.com/push-io2010

Me
http://onebigfluke.com

Superfeedr
http://superfeedr.com


Google™
10