Scalable Web Programming

CS193S - Jan Jannink - 3/04/10

Weekly Syllabus

1. Scalability: *(Jan.)*
2. Agile Practices
3. Ecology/Mashups
4. Browser/Client
5. Data/Server: *(Feb.)*
6. Security/Privacy
7. Analytics
8. Cloud/Map-Reduce
9. **Personalization & Published APIs: *(Mar.)**
10. Future

* assignment due

Assignment 2 Notes

- All six projects functional, if bare bones
  - we each installed, ran and tested all apps
- All students checked in (max 71, min 1)
  - we were able to wrangle a few last minute
- Average score, 34/50; standard deviation, 8
  - real world evaluation can be tough

Q & A

- Project concerns
  - scores vs. grades
  - level of project activity
- Questions about Friday requirements
- Project README for any notes
- Course process transparency

Topics

- Purpose of APIs
- Opening up to the world
- Programming APIs and scalability
- Best of REST and web services
  - REpresentational State Transfer
- Is open source an API?

APIs

- Create a language to bind components
  - implement an interaction interface
  - facilitate interaction with third parties
- Enable specialization of functionality
  - decouple separate concerns in code
  - reduce monolithic systems into components

Web API History

- eBay API in incipient dot com bust 2000
  - PayPal, Flickr
- Google maps reverse engineered late 2004
  - HousingMaps.com mashup
- MySpace embeds in 2005
  - YouTube, PhotoBucket, imeem


YouTube

API trends

- Infrastructure
  - AWS, App Engine, Yahoo Pipes
- Content syndication, feeds, timelines
- Identity, profile

ID APIs

- Microsoft Passport - Windows Live ID
- OpenID
- Stanford WebLogin
- Facebook Connect
- Twitter Credentials

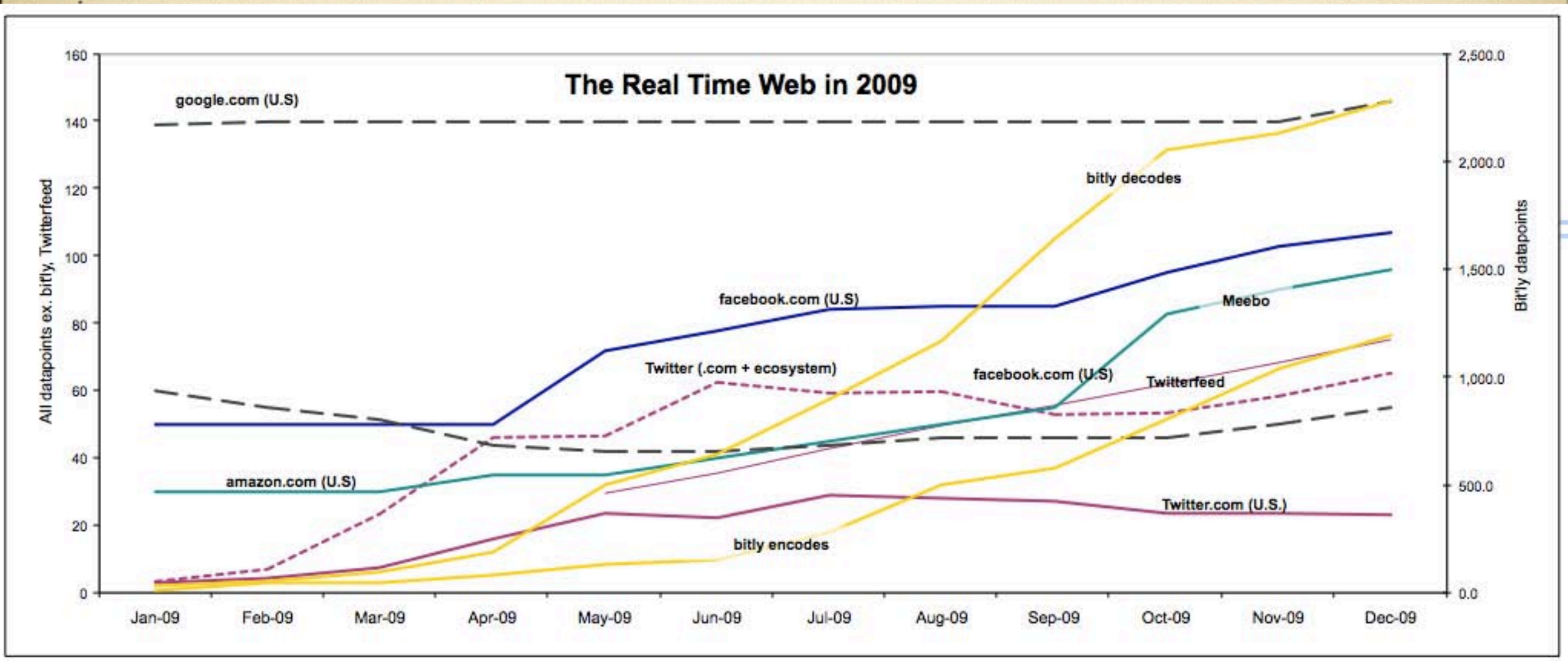
Languages as APIs

- RPC, RMI, CORBA
- HTTP + HTML, XML, RDF, JSON
- KML
- HTTP + RSS, microformats

Twitter Example

- First public launch July 2006
- API in use since January 2007
- ~1000 apps use API November 2008
- ~10000 apps use API May 2009
- Ecosystem estimated at 3x twitter traffic

A Lot of Growth



from TechCrunch

Best of REST

- Stateless
- Decoupled
  - scalable composition
  - MVC pattern
- Cacheable
  - efficient composition

Open Source as API

- Strictly speaking not an API, but POSIX is
- As a platform
  - highly composable, interoperable
  - completely configurable

Linux Ecosystem

- Prior to Linux
  - hundreds of freeware apps (think cygwin)
  - slowly developing, experts only
- Linux created a native home for open source
  - thousands of quickly developing apps
  - app repos, platforms, hundreds of distros

Critical Linux

- GCC, Linux kernel, POSIX compliance
- RedHat, VA Linux, IBM
- Debian, Ubuntu, Gentoo
- LAMP, live DVD/USB
- Android

Back to Software

- When to take the API plunge
  - infrastructure that grows the product
    - opportunity to build a market
- Internal APIs exposed
  - Amazon

Publish Internal APIs

- Be your own first client
- Develop API to support a critical new feature
- Imagine resources for many similar features
- Publish API, advertise it to coders
  - Simplified Wrapper and Interface Generator
- Foster developer community growth

Worth Checking Out

- Yahoo Pipes
  - http://pipes.yahoo.com/
- Bitly.tv
  - http://bitly.tv/
- Dipity
  - http://dipity.com/
- SWIG + PHP
  - http://swig.org/
- DistroWatch
  - http://distrowatch.com/

Scalable Web Programming

CS193S - Jan Jannink - 3/04/10