Building Social Applications with the Linked Data Platform

Sandro Hawke - Overview
sandro@w3.org, twitter: sandhawke

Joe Presbrey - Demo & Details
Andrei Sambra - Demo & Details
(MIT CSAIL)

معهد قطر لبحوث الحوسبة
Qatar Computing Research Institute
Member of Qatar Foundation عضو في مؤسسة قطر

CSAIL

11 March 2014, Cambridge Semantic Web Gathering
Slides at http://www.w3.org/2014/Talks/0311-sandro/

Overview

- The Social Lock-In Problem
- The “Crosscloud” Solution
- The Linked Data Platform
- An Example
- Research Questions

(then)

- Joe’s Demo
- Andrei’s Demo

Individual User Lock-In

- User does some work in some program
- Work is saved in proprietary format
- User is stuck using only that program until:
    - willing to lose that work
    - someone reverse-engineers the format (usually lossy)
    - export then import (usually lossy)
- Ongoing drag on innovation, productivity
- “TCO should include exit costs”
- Solutions: open source software, standard formats

Group Lock-In

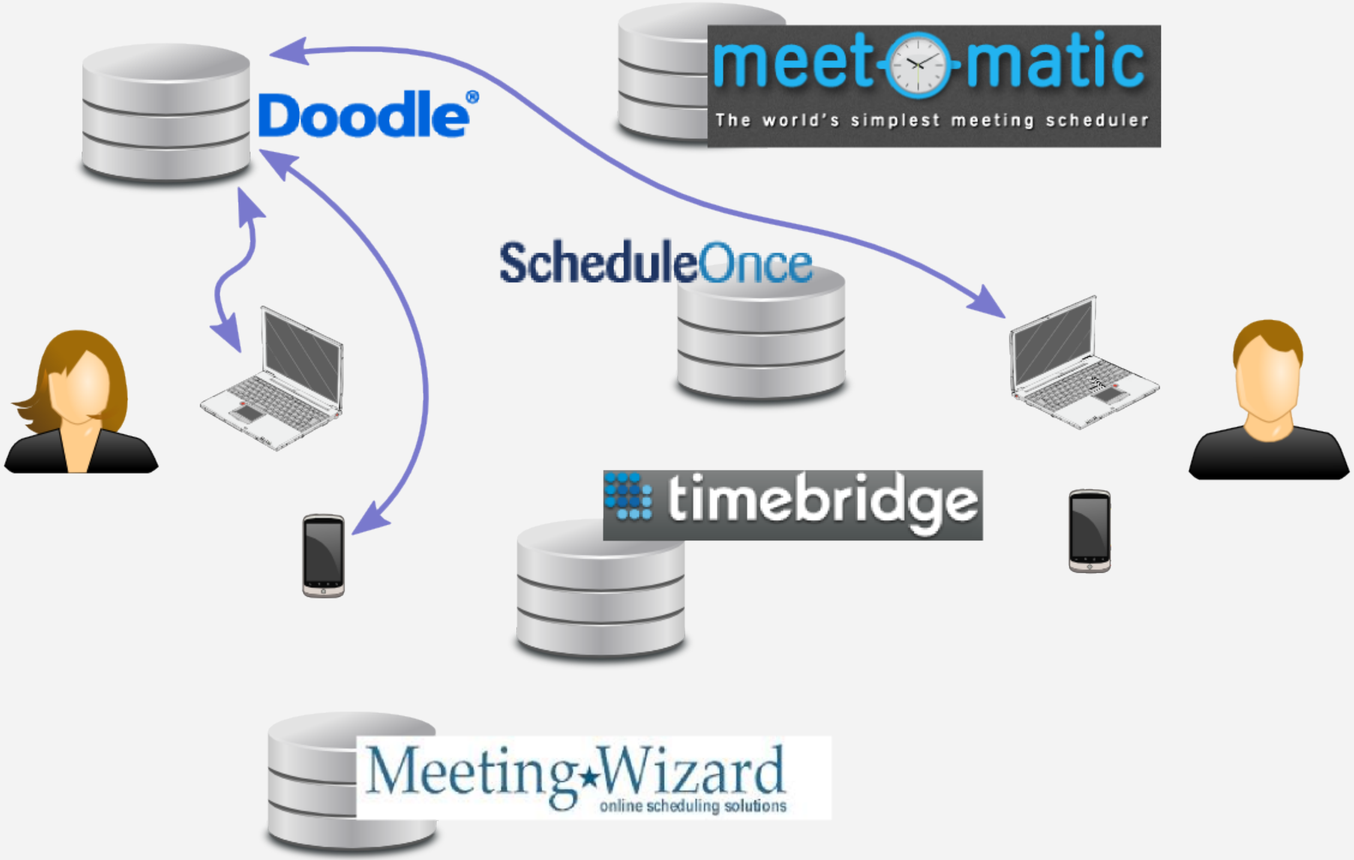
- Economics, “The Network Effect”: more users = more value
- If I want to share my photos, with access control, on flickr
  - then all my friends have to use flickr
- If Facebook is the only system everyone is using, then the only way to reach everyone is Facebook
- Thus Facebook’s 1.8e+11 USD market cap?
- If something better came along, could you get your high school friends and cousins to move it?
  - Open Source and Standard Formats don’t help here
- Also: not everyone uses Facebook, or … anything modern.
  - Postal service, Telephone, E-Mail

Developer Lock-Out

- If users are locked into a system, developers are unable to attract them elsewhere
- Imagine a better Facebook, but you're the only one there….
- You can build on the major platforms
    - But only subject to their terms
    - And subject to the limitations of their platform
    - And you still only get the users who want to use that platform

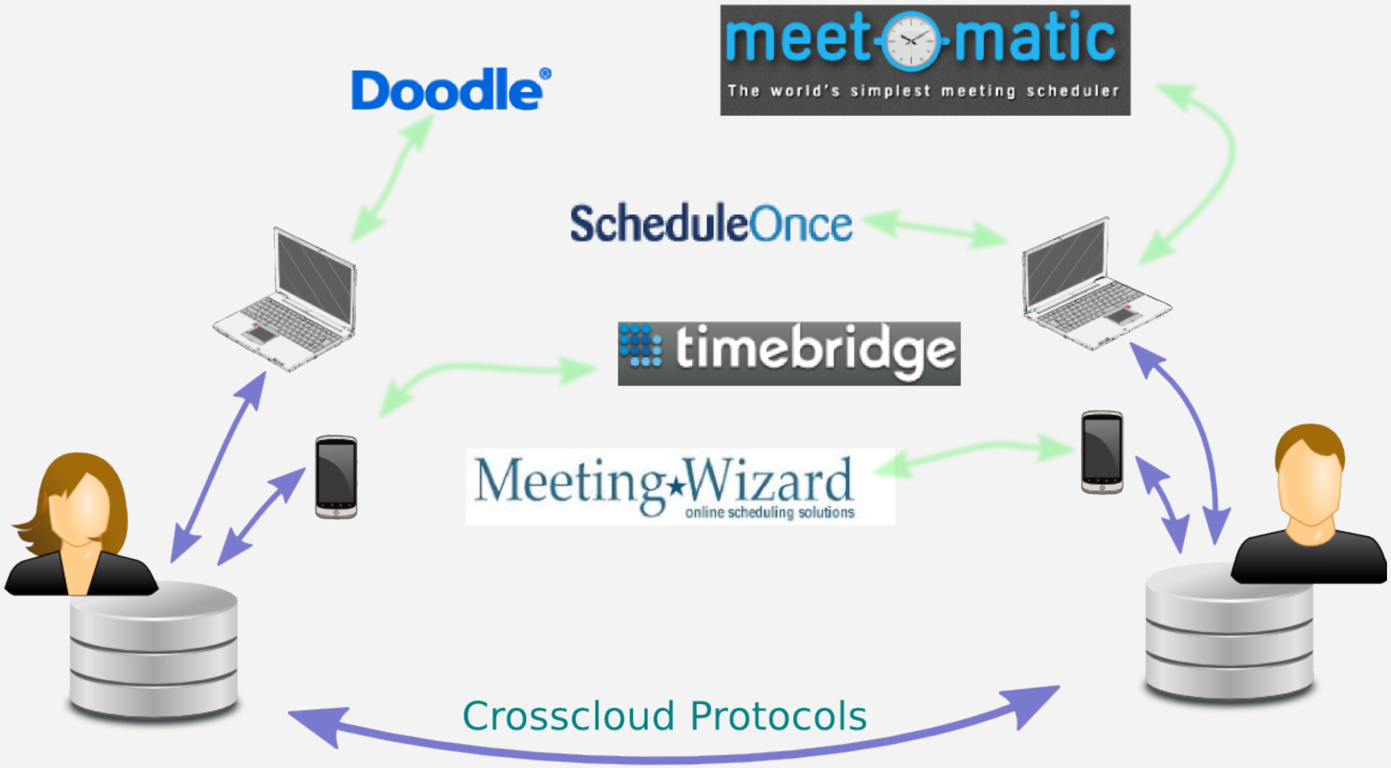
Is there a better way?

Scheduling a meeting today (without Crosscloud)


Doodle®
meet-o-matic
The world's simplest meeting scheduler
ScheduleOnce
timebridge
Meeting★Wizard
online scheduling solutions

Everyone invited has to use the same vendor!

Scheduling a meeting -- With Crosscloud



You remain in control of your data.
You are free to pick different software and service providers.

Crosscloud Vision

- User’s data remains on a general-purpose server, responsible to the user
- Users can expose it to any application they like
- Users can switch providers
- Applications (ideally) provide the same functionality, but with external data
- Data is in RDF (open standard format)
- Data is accessed via HTTP (open standard protocol)

Linked Data Platform

- Start with Linked Data ...
  - Use working HTTP IRIs to name things, provide data about things, provide links
- … and make it into a full data storage/access system
- W3C Submission Feb 2012
  - From: IBM, DERI, EMC, Oracle, Red Hat, SemanticWeb.com, and Tasktop
- LDP Working Group - Second Last Call published *Today*
  - http://www.w3.org/TR/ldp
- Very basic starting point

Linked Data Platform 1.0 Provides

- Containers
  - POST to create new Resource
  - GET provides triples linking to each member
- Paging (not in today's draft)
  - rel=first, last, next, prev
  - flow control
  - just the interesting stuff, if sorted

Extensibility

- HTTP
  - HTTP Link headers (with defined relations)
  - HTTP Prefer / Preference-Applied headers (with defined preferences)
  - Can’t really use media type extensibility, since RDF already uses that
- RDF
  - define new properties and classes, as usual

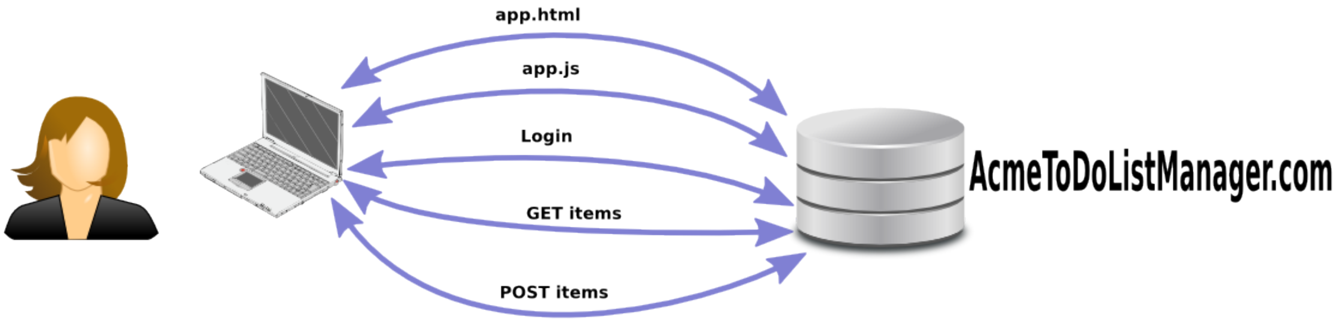
Extensions used in today's demos

- WebID -- Use IRIs to identify people, properties/classes in what you get back
- Workspaces -- Link from WebID to Containers of stuff
- Change Propagation -- Real-time notify client of changed data
- Access control -- server grants read/write access based on user's WebID (using client certificates)

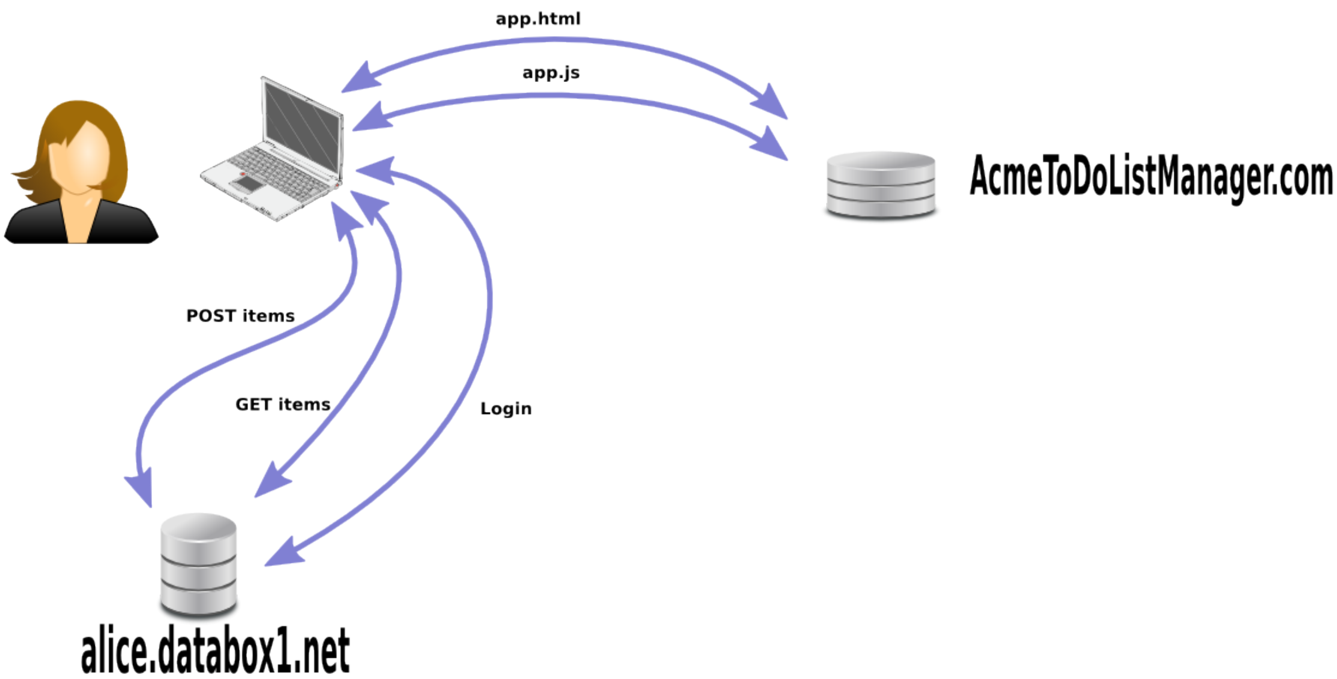
Example: To-Do List WebApp

- User logs in with their WebID
- App can read RDF triples describing to-do list items
- App can modify/delete RDF triples describing to-do list items
- It’s just doing HTTP GET, POST, DELETE, PUT with small Turtle documents

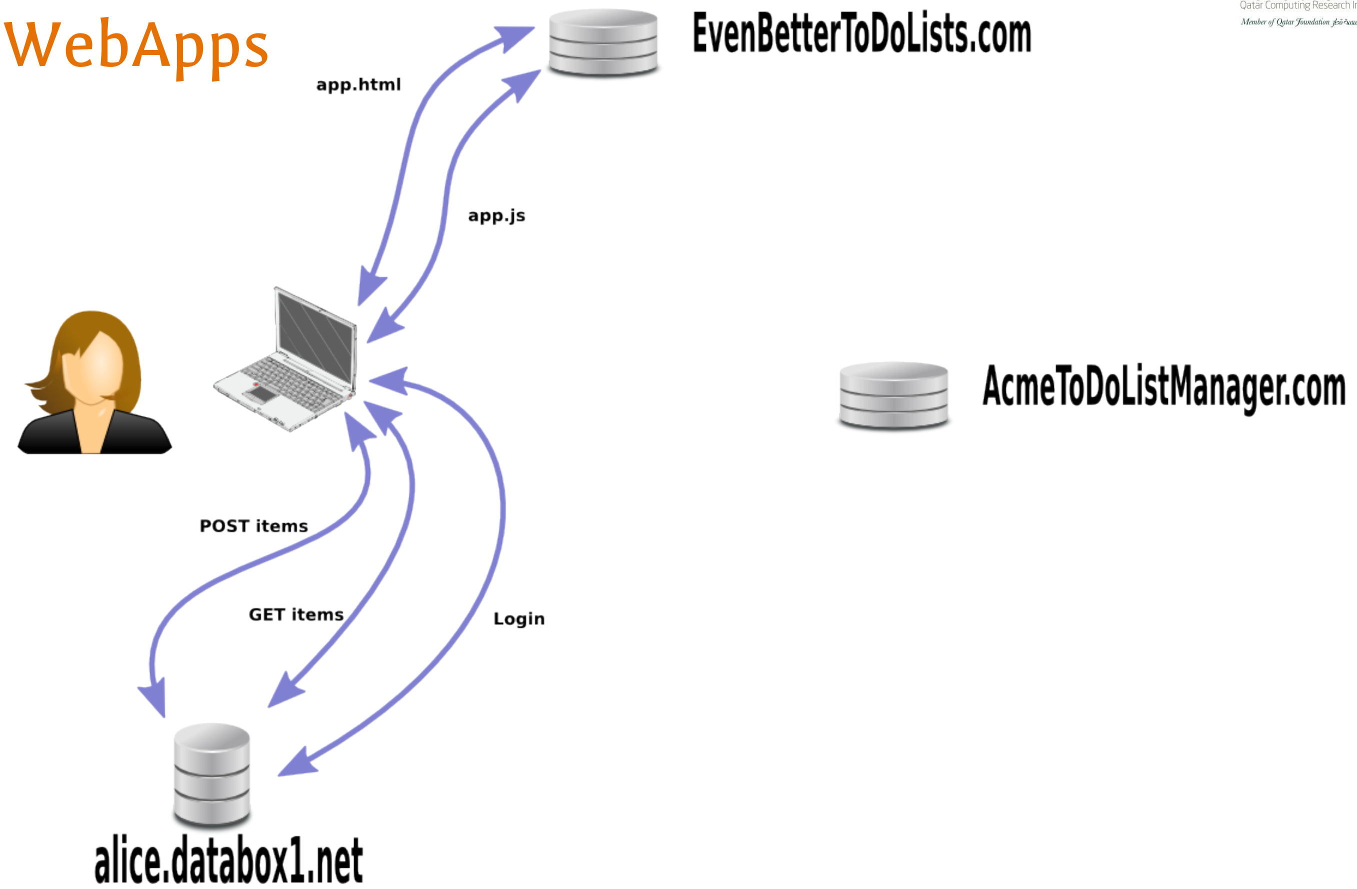
Pre-Crosscloud WebApp



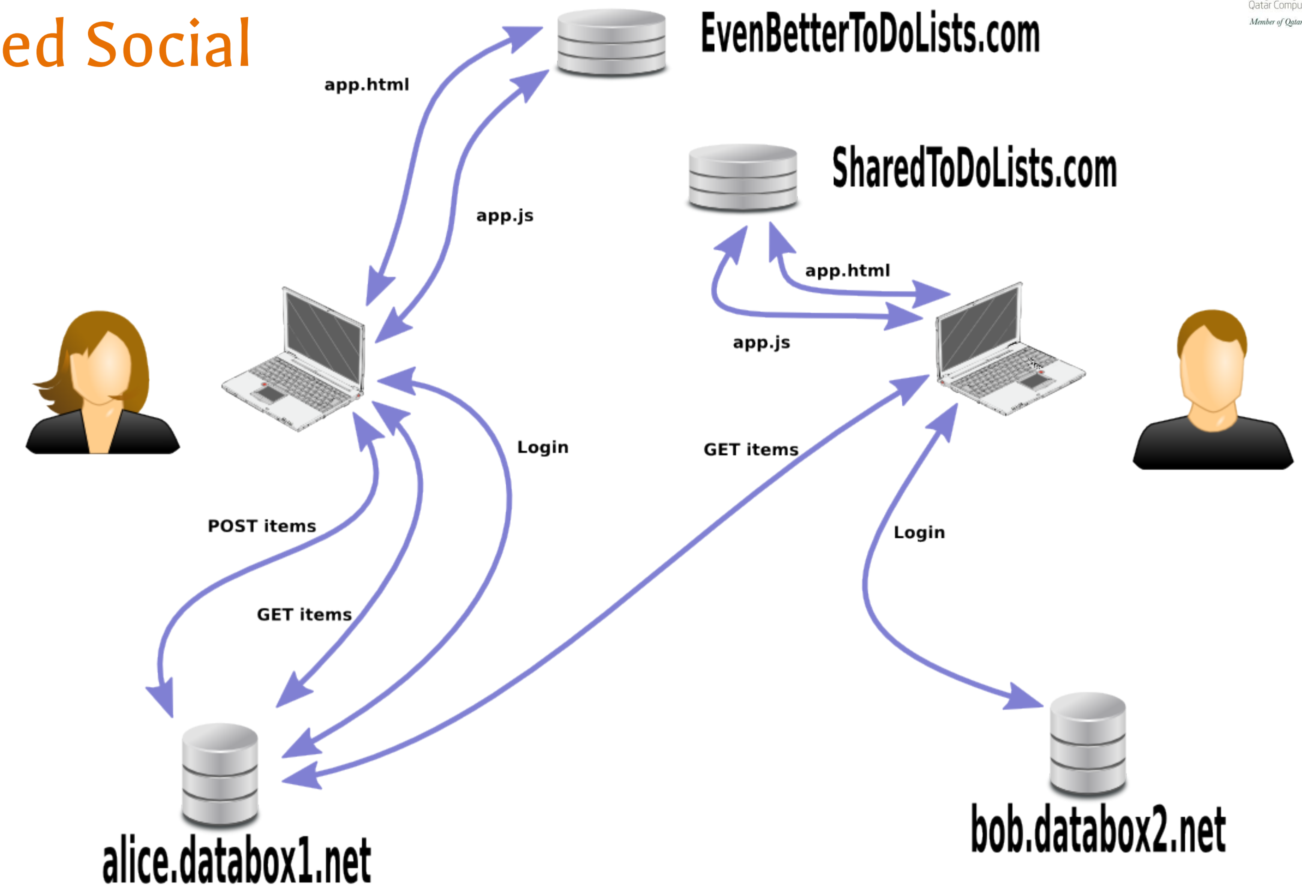
Move the Data to Alice's Server



Switch WebApps



Unlocked Social



Why is that so great?

Same shared state, but…

- Open Standard Protocols: anyone can build another UI (“EvenBetterToDoLists”)
- UI can be forked, improved, with *the same user base*
    - **We’ve escaped group lock-in**
- Standard, commodity backend service (like web hosting)
- User control, optional additional privacy
- No Password Proliferation

Other Applications

- Microblogging (cimba) [twitter, facebook]
- Office Apps
- Media Sharing
- Product Reviews/Feedback
- Gaming
- ...

Lots of Research To Do

- This work is supported by and in collaboration with QCRI (which is hiring)
- Related work in our group supported by NSF & Knight Foundation
- The “Social WebArch” Group
  - Tim Berners-Lee, PI (MIT)
  - Ashraf Aboulnaga, Co-PI (QCRI)
  - Essam Mansour, Ahmed El-Roby (QCRI)
  - Sandro Hawke, Andrei Sambra, Joe Presbrey (MIT)
  - Lalana Kagal, Project Lead PI (MIT)

Efficient Change Propagation

- Fast (notify, patch)
    - (while running as a WebApp)
- Inbound (following 1000s of people)
- Outbound (1000s of followers)
- Massive scale (Obama Campaign has 40M followers, follows 0.65M)

Access Control

- Who can see my data?   (read access)
  - Individuals, Groups, Logical Groups
- Who can write to my space? (write access)
- Who do I pay attention to, when they write to their own space?   (notify access)
- Semi-Trusted Software (like current OAuth WebApps)
  - give them access to *some* of your data
  - limit the harm they can do by writing
  - be able to identify and undo any harm they do

Designing for Reliability

- Where can we build redundancy/failover into the protocols?
  - A smarter client allows much simpler server architecture
- Load balancing

Vocabulary Translation

- Interoperability Requires using the same Vocabulary, or Translation
- Multiple vocabularies for one domain, at some point in time
- Multiple vocabularies over time (migration)
- Conversion Rules are more data (indexed, trusted, crowdsourced, purchased)
- Conversion via OWL? N3?   RIF? JavaScript?   Java?   Something New?

Technology Adoption

This can and should be research, too

- What do developers need?
  - Quick results, simple model, open source, open standards
- What do end-users need?
  - Quick results, great apps
- What do business users need?
  - Stability, proven utility, trained workforce
- What does the industry need (for standardization) ?
  - consensus among a critical mass

Next Steps

- W3C Participation
  - WebID Community Group
  - Federated Social Web Community Group
  - Linked Data Platform Working Group
- Open Source Contributions
- Academic Collaboration
- Business Collaboration