The Web of Things and context-aware services

Open Source Junction
Oxford, July 2011

Dave Raggett <dsr@w3.org>

World Wide Web Consortium
& webinos project researcher

Background

- More and more devices are being networked
  - Declining costs for network hardware
    - Moore's law applying to digital and RF integrated circuits
  - Micro-controllers are now ubiquitous
    - In all kinds of consumer devices, from kettles to cars
  - Research projects on the Internet of Things
    - Focus on hardware and transport protocols
- Now time to focus on applications
  - But, low level languages, complex protocols, security
  - How to make it easier to realize the possibilities?

The Web of Things

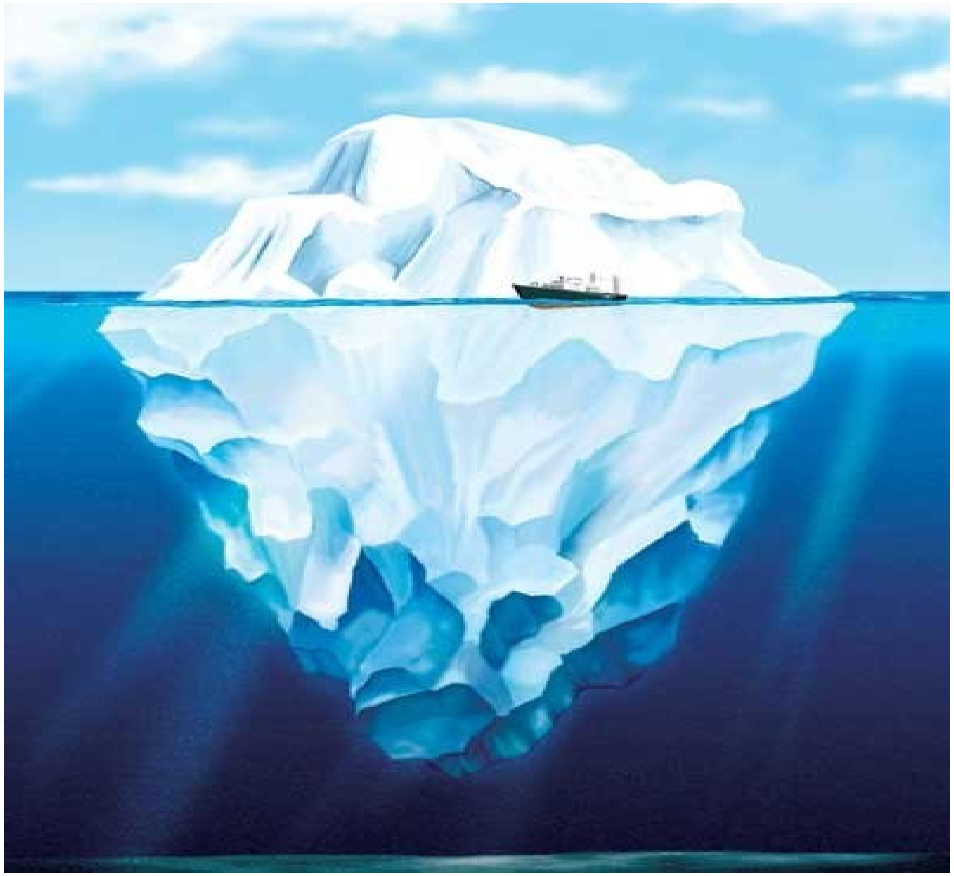
- The Web has been hugely successful
  - Vast number of developers
  - Comparatively easy to learn with markup, style sheets and scripting
  - Plenty of materials to leverage on the Web
- Let's apply Web technologies to distributed applications
  - Scriptable objects as proxies for remote services
  - Set event listeners or call backs to handle input
  - Set properties, call methods or target events for output

The Web of Things

- Making it easy for Web developers to create applications that span devices and firewalls
  - Logical communication paths based on trust relationships, and decoupled from underlying interconnect technologies
  - Simple discovery of devices/services
  - Simple access to local and remote services
  - Trust based on social relationships between people
- Simplicity provided by 3rd party components that layer on top of lower level standards
  - Establishing a market for toolkit providers

Simplicity is hard work!

Shielding Web Developers from unnecessary details



- Web oriented models of world as basis for easier development of apps
- Hidden infrastructure and associated models, supported by 3rd party libraries

Architecture

- Your devices & services form a ***Personal Zone***
  - Single sign-on
    - You authenticate to a device, and the device to the Zone
    - Distributed architecture that is robust to being offline
      - e.g. when Internet is inaccessible and devices are isolated
  - Synchronization across the Zone
    - Identifying what needs to be synchronized to fulfil use cases
  - Discovery and access to services and the context
    - Local and remote
  - Trust relations based upon social graphs
    - Context dependent access control rules

Discovery

- What can be found locally
  - Internet Protocol: mDNS, SSDP (UPnP), SLP
  - Bluetooth, USB, Firewire, Zigbee, etc.
- What can be found via your social graph
  - What devices does my friend Mathew have?
    - Which he chooses to expose to me
- Wide area discovery
  - By name, email address, phone number or URI
    - Name via collective search across zones
    - Email address via domain name and directory service
    - URI via HTTP for someone's zone

Local Discovery Plugin

- Webinos browser plugin for Linux
  - mDNS, SSDP, SLP, Bluetooth, USB
  - Developer passes object to plugin
  - Plugin calls back to named method
- What we learned
  - Some things are fast, others are slow
  - Widely varying vocabularies
  - Complications for search by service type
  - Split standards into high and low level APIs

Wide Area Discovery

- Map email address to URL for social agent
  - Web Finger or DNS-SD for pointer to query service
    - Value added service by email provider
- Distributed search based upon social relevance
  - Search by name or pseudonym
  - Collectively supported by web of social agents
    - Distributed Hash Tables
  - Rank results based on metrics for social relevance
    - The Henrietta I went to school with
  - Challenge is to preserve privacy
    - Search agents trusted with private data
    - Need for common vocabulary, e.g. profile data

Case study: Robot Toy

Replacing proprietary protocols by open standards

- Meccano Spykee
  - Embedded Linux with ARM processor (by WaveStorm)
    - Open source firmware
  - Motors, LEDs, Microphone, Speaker, Videocam
    - Streaming video/audio
    - Proprietary protocol
  - Can be controlled locally or remotely over WiFi
    - NAT traversal
  - Controllers developed by community for iPhone, Android

Case Study: sharing a movie

- Sue is visiting her friend Bill's house, and is keen to share with him a movie, which another of her friends, Janet has just sent her a message with an enthusiastic recommendation and a pointer to a review.
- Sue is looking at Janet's message on her smart phone, and now wants to start playing the movie on Bob's surround sound big screen entertainment system in his den.
- Sue clicks on the share button in the web application and selects "find local devices" from the pop-up menu.
- She see's Bob's home entertainment system and selects it.
- A confirmation message appears on Bob's big screen, and he clicks ok on his remote controller. The movie starts and Sue sits down and moves closer to Bob, the lights fade as the music starts, …

How it works

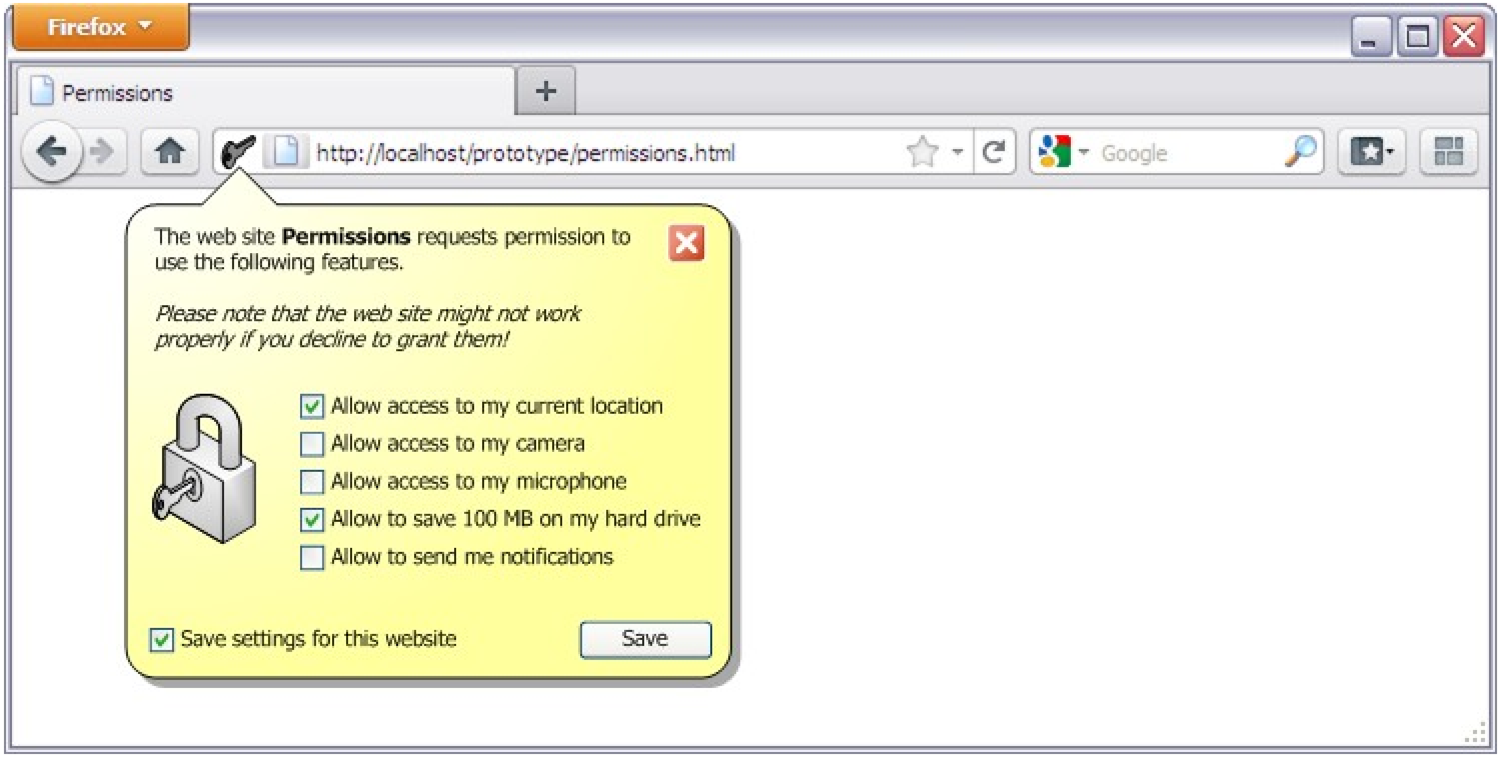
- Discovery
  - Use of local discovery protocols
    - If Phone and TV both have Bluetooth, or WiFi
  - Or of context and social proximity
    - Bob's social agent exposes his TV to his friends
    - Sue's phone app sends discovery request via her personal zone, which in turn passes it to Bob's personal zone which responds
    - Bob's agent sends subsequent play request to his TV
      - Tunnel request, or set up P2P connection via zone hubs
- Bob has to authorize requests
  - He can set a policy when responding to the request
    - e.g. always allow Sue access when she is in his home

Keeping in touch

- Bob's devices could keep long lived web socket connections open to his personal zone
  - Personal zone hub as agent running 24x7 in the Cloud
- But this could be done indirectly, via a message hub, e.g. as a service of a home gateway which manages presence info
- For cellular connections, further optimizations
  - Use SMS wake up messages to re-establish web socket connection, avoiding need for long lived connections
- Tunnel messages via shared connection
  - Avoiding need to setting up multiple connections
  - Reduced latency especially for cellular networks

Requesting Permissions (1)

- Web App requests multiple permissions in same call to avoid plaguing users with a sequence of individual requests
  - See http://www.simonheckmann.de/proposal/



The web site **Permissions** requests permission to use the following features.

Please note that the web site might not work properly if you decline to grant them!

☑ Allow access to my current location

□ Allow access to my camera

□ Allow access to my microphone

☑ Allow to save 100 MB on my hard drive

□ Allow to send me notifications

☑ Save settings for this website

Save

What's wrong with the above?

Requesting Permssions (2)

- User needs sufficient information for an informed decision
  - Identifying the app – the URI isn't sufficient
    - Text description + link to more information
  - Why the application needs these permissions
    - Textual explanation + standard icons?
  - What its privacy policy is
    - Notice & Consent model for data handling commitments
    - Machine interpretable policy + link to full policy
  - Whether other people think this app is trustworthy
    - White/black lists
    - Trust delegation (I don't know, what do you think?)
- Giving users an effective means to review and revoke earlier decisions
  - Browsers fail to do this for gelocation, but we can do better …

Securing the Browser

- Today it is easy to introduce security holes in web apps in both client and server-side code
- How can we make it harder for attacker to gain access within firewall/zone?
- Run browser in secure mode that locks down known weaknesses, e.g. eval and innerHTML
- Apps have to ask for elevated privileges
- Browser can check integrity of app components against signed application manifest
  - For both installed and server-hosted applications

Who or What?

- Authentication on the Web is deeply flawed
  - Say no to typing credentials into web page forms!
  - Don't send passwords to servers!
  - Do use mutual authentication
- Two step model
  - First authenticate yourself to your zone
    - 2 factor, e.g. biometric techniques and fall-backs
  - Zone then authenticates you to each service
    - Based on info provided by that service
    - Need for transparency, see previous slide
- Sometimes it is what we are, not who we are
  - Providing proof of membership of group or other properties
    - Zero knowledge proof as privacy friendly solution

Trust relationships across Zones

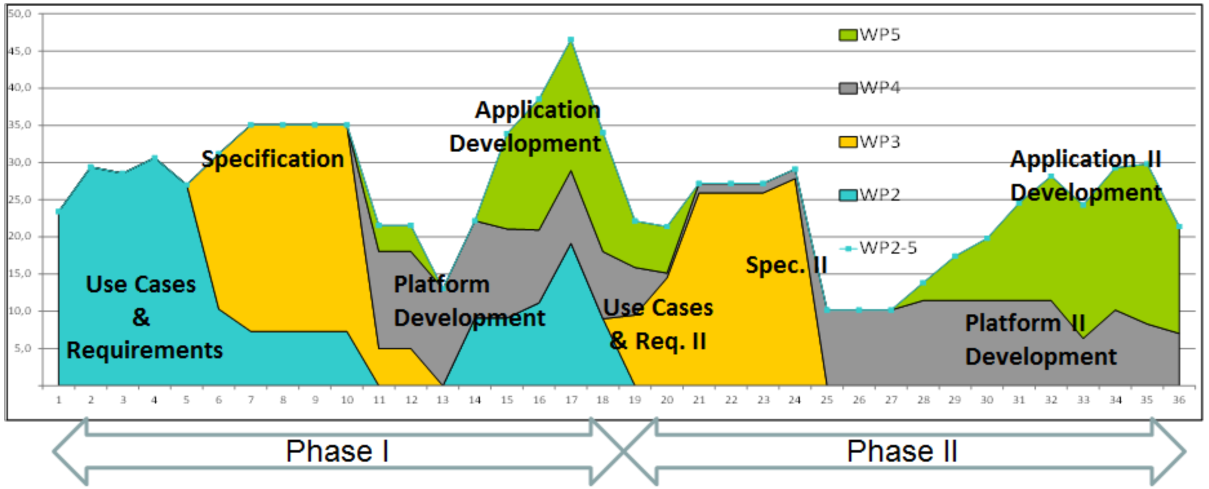
- Some devices are shared
  - Open access to anyone on local network
  - But web app on TV could be restricted
- Consider unions of zones
  - Family zone + personal zone
  - Trusted networks (WiFi + WPA + Firewall)
- You may have multiple zones for different contexts
  - Your mobile device when you are working vs at home
- APIs and access control rules for access across Zones
  - Which “face” I present and to whom
    - – Face determines what you expose
      - Face is a kind of identity you decide to take on in different contexts
      - You can present different faces to the same people in different contexts
    - – Social context: personal, family, friends, acquaintances, colleagues, everyone

See http://www.tilburguniversity.edu/topic/innovation/socialnetworkclique/

http://webinos.org

Webinos

- EU project to deliver open source platform based upon Web technologies that enable secure and consistent use of applications across the domains:
  - PC – Mobile – Home media (TV) – Automotive



September 2010 – August 2013

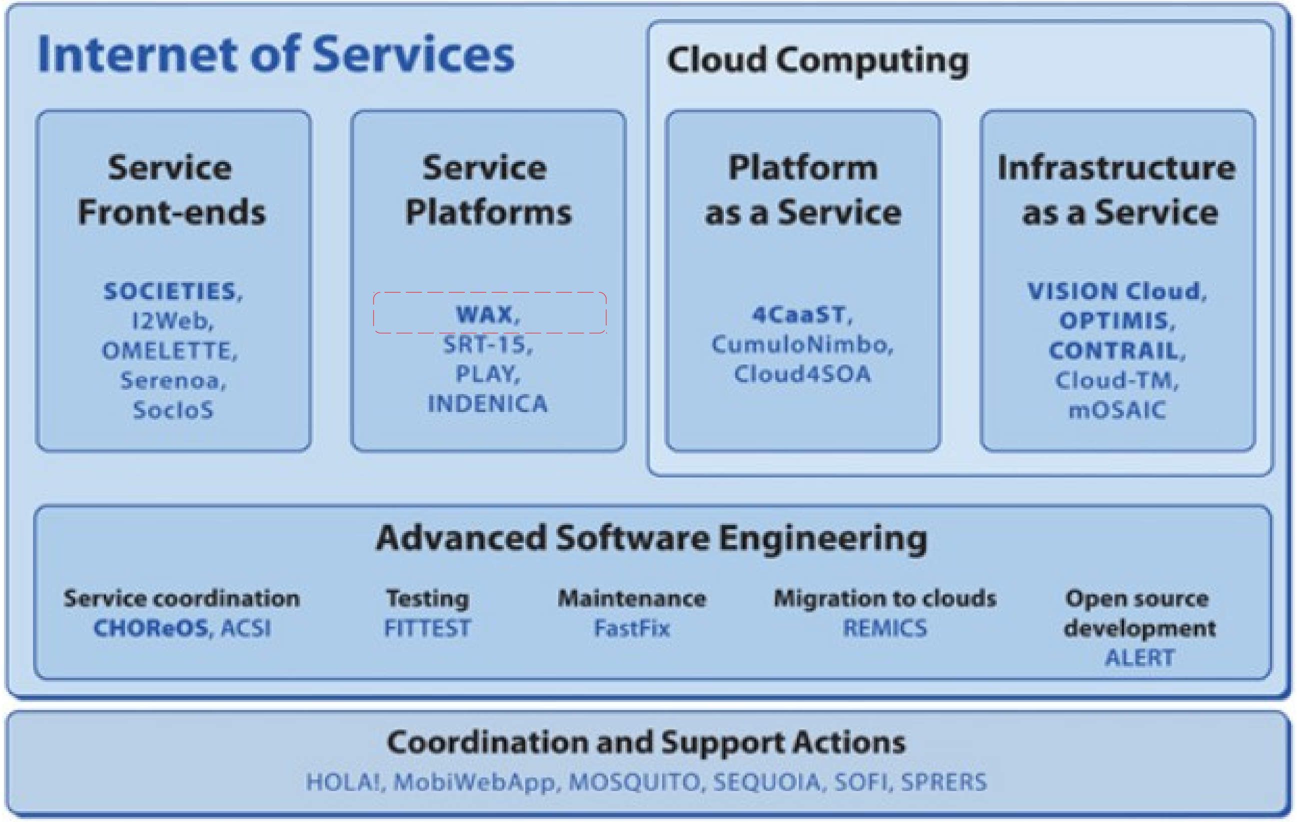
Webinos

http://webinos.org

Previously known as “WAX”

- EU FP7 Call 5 Project with around 20 partners

http://cordis.europa.eu/fp7/ict/ssai/projects-call5_en.html



Internet of Services

Service Front-ends
SOCIETIES, I2Web, OMELETTE, Serenoa, SocIoS

Service Platforms
WAX, SRT-15, PLAY, INDENICA

Cloud Computing

Platform as a Service
4CaaST, CumuloNimbo, Cloud4SOA

Infrastructure as a Service
VISION Cloud, OPTIMIS, CONTRAIL, Cloud-TM, mOSAIC

Advanced Software Engineering

Service coordination
CHOReOS, ACSI

Testing
FITTEST

Maintenance
FastFix

Migration to clouds
REMICS

Open source development
ALERT

Coordination and Support Actions
HOLA!, MobiWebApp, MOSQUITO, SEQUOIA, SOFI, SPRERS

The Web of Things and context-aware services

Thank you for listening