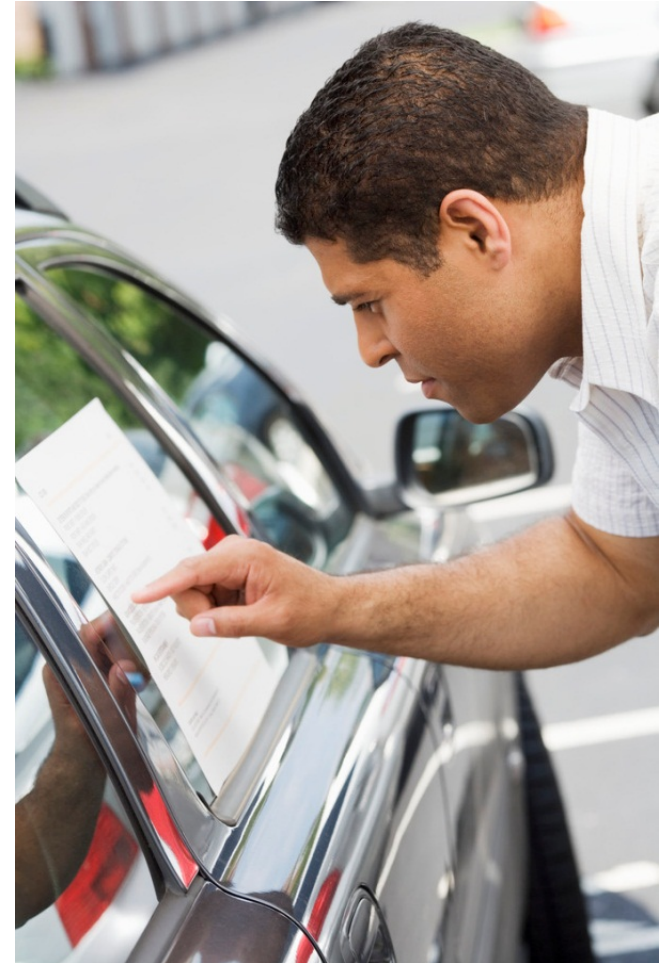


[web and automotive]

... "Shopping List"

Workshop
Rome 2012-11-14 .. 15

Magnus Olsson
Ericsson AB



Authors Profile



› About Magnus Olsson

Standards Management at Ericsson Group Function (cross business unit)

–Primary role in standards domain

- › Application and service layer standards
- › APIs, Frameworks and Security aspects

–Related relevant engagements (some)

- › LIF, OMA, JCP, WAC, W3C and Bluetooth SIG

› About Ericsson

At Ericsson we use innovation to empower people, business and society. We envisage a Networked Society that is sustainable, and where everything that can benefit from a connection will have one. Our mobile and fixed networks, multimedia solutions and telecom services make a real difference to people's lives, and the world we live in.

Web and Automotive



› Introduction

- Bare bone architecture, *separation of concern*
- Leverage Open Web Platform, *re-cycle it's free*

› Challenges

- Sensors want to be heard
- Keep the user in the drivers seat
- Ensure a compelling developer proposition

› What else...

- W3C Innovative ideas, worth exploring

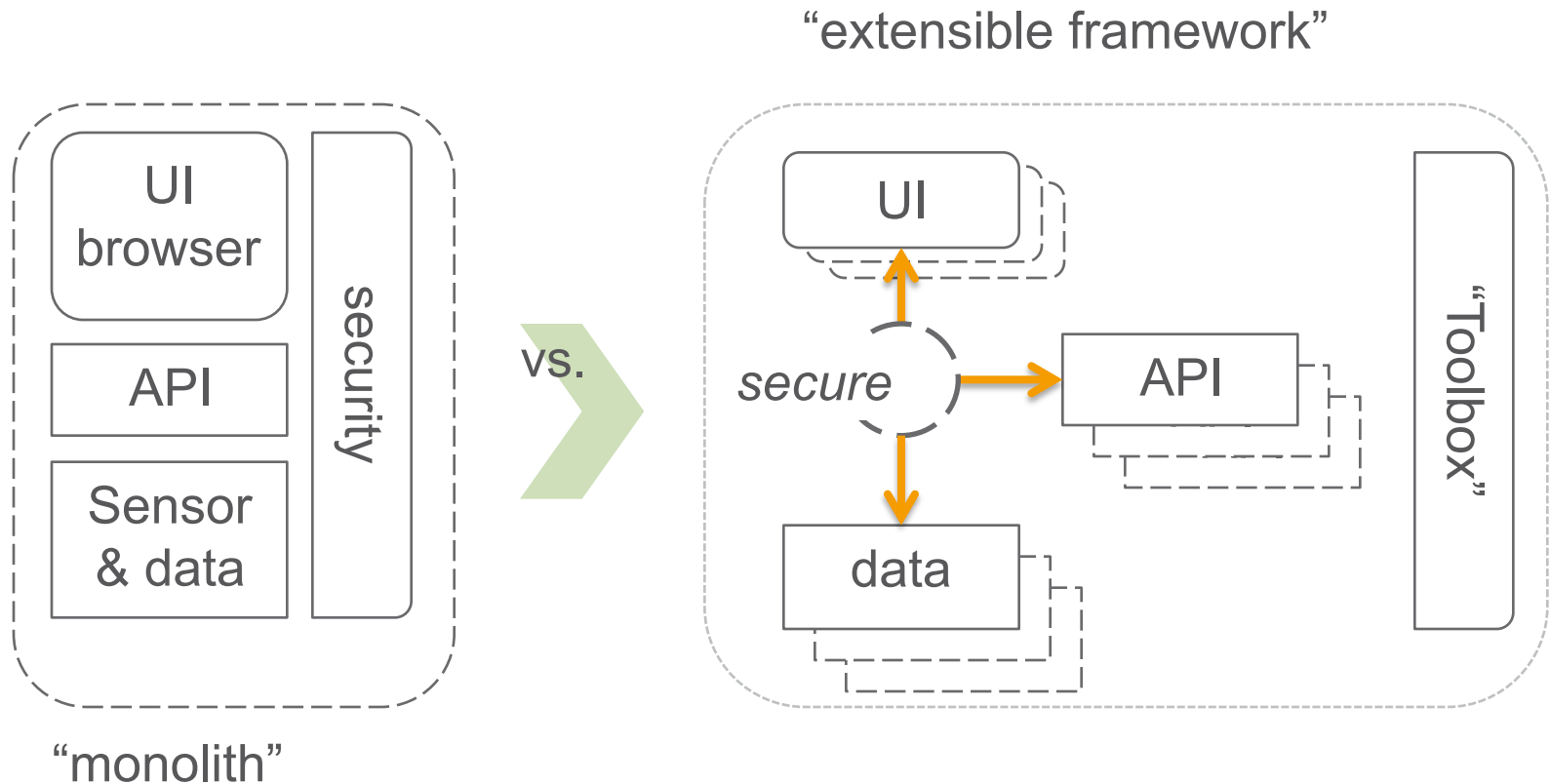
› Rounding Up

- Web and Automotive, cool use case
- Conclusions + Q&A



Bare bone architecture

...separation of concern



Leverage the Open Web Platform

- Requirements on Framework



- › Administrative Features – “abilities”
 - Configurability, Discoverability, Capability, Upgradeability ...

- › Connectivity “drivers”
 - Fundamental to enable physical separation when desired
 - Protocol: HTTP, Web sockets (also peer to peer)
 - Media: Wire, Wireless (BT, Wifi, 3G/4G ...)

- › Responsive User Interface (*Context based)
 - Multi screen, Remote/Local, Markup, Video, Audio
 - Input: Abstract Events, Physical Controls
 - Intuitive Security and non intrusive (incl. safety measures)

- › Future Proof
 - Keep the framework itself a level field eco system (*plug-able business model(s))

Challenges

- A Multiverse of ECO Systems
- Sensors want to be heard
- User in the Driver's seat
- Developer Proposition

Connected Eco Systems



Walled Garden
"Business to Consumer"



Connected
Service
Booking



App Stores
"Developer to Consumer"



Open
Innovation



Bundled Service
"Publisher/Writer to Consumer"



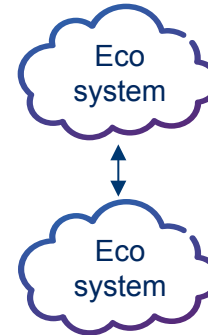
Navigation
Map Update



Cloud Services
"N-Screen"



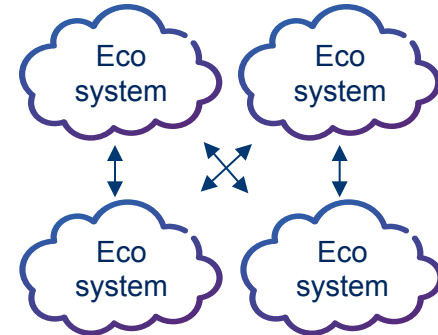
Internet
Radio



Connected Industries
"Business to Business"



ELVIIS
Electrical Vehicle Charging



Sensors want to be heard



- › Automotive is an untapped source of interesting sensors and data
 - Industry players need to agree on the principles how this data can be
 - › Shared, who's the owner, where is the cloud(s)?
 - › Consumed using JSON, HTTP, Javascript, Open Web Platform
- › Challenges (Safe guarding data/devices)
 - Differentiate between readable (safe) and writeable (unsafe) data sources
 - Scope or lifetime of the rights to use that information
 - › Primarily to protect end user personal integrity
- › Opportunity
 - Developers would most likely be thrilled with the opportunity to build applications based on web and automotive sensors and integrated UIs
 - E.g. “crowd sourced” automotive speed sensors, *temperature readings...

User in the driver's seat



- › Privacy (Integrity of user data)
 - Should be a primary concern from the start
 - Threats like fingerprinting, leak of user data (contacts)
- › Avoid intrusive behavior
 - E.g. minimizing how application notifications (pushed content) is presented based on user context and preferences
- › Enable end-user to audit information
 - Settings, Log files and enable ability to “reverse” a decision on application install and revoke privileges
- › Keep cost of services at an “affordable” level
 - Enable use of Push Aggregation (APIs on top of “native” push framework)
 - Enable mechanism to Share and Manage Bandwidth (in car zone)

Developer Proposition



- › Enable as many APIs / Sensors as possible with good quality
 - Consider offering utility Javascript libraries (or Shims) to alleviate developer effort
 - Consider HTTP/REST before other RPC models (simpler) and JSON before XML (leaner)
- › Consider safe mash-up techniques
 - Such as Web intents/activities before UI prompting schemes
- › Keep cost and overhead (time) down
 - For developer to engage and “register” a new application
- › Separate roles of application provider, security provider and payment provider
- › Avoid discrimination of OS platforms (reality check)
 - (HTTP clients) could be based on HTML5, Android, IOS or other OS

W3C Innovative ideas



› Indie UI and Pointer Events

- Potential to realize a web enabled dashboard, responsive input
 - › Automotive apps need to be well integrated and accessible
 - › Human senses compromised (busy with driving, conversations ...).

› Web Driver

- Enable repetitive and extensive automated test of web framework applications (based on browser)
 - › Web Apps in Automotive are not limited to “monolith” but exert dynamic behavior in terms of APIs, UI controls and modes

› Web Intents / Activities

- Mash-up with providers of map data, social info, data mining events and automotive based local data/conditions
 - › Data and services are distributed and need to be re-combined on the client side in a safe and predictable manner

Ample power, comfy seats, what else...

The Future of Web and Automotive

- Perhaps, a web based HUD?
- Social Fleet Management
- Zombie Drive (a game seriously?)

Cool use case, Web HUD



- › *Collect relevant ambient sensor data, mash up with maps, social graph and serve to multiple screens (Dashboard, Smartphone, Remote etc.). Associate with physical “remote controls” and map to events/message ports*

- › Involved APIs, Protocols
 - Service Discovery (ZeroConf, mDNS), Web Intents
 - HTTP + Media Source Streaming over...
 - › HDMI, Airplay, WiDi (Wifi Display) or
 - › Bluetooth Video Distribution Profile (VDP)

- › Challenges (what to work with W3C on)
 - Need to understand what is a “Dashboard” style sheet?
 - › Or possibly a dedicated Media Query?
 - Consider creating an abstraction of remote controls vs. web input controls/events

- › Real world examples
 - Google Glasses and [Oakley Airwave Snow](#)

Use case – Social Fleet



- › *Collect sensor data and share (anonymously) with a cloud service provider able to make sense of the data. The data in turn can be “mined” to discover facts about the traffic or related activities such as recreational stops, shopping, maintenance or re-fueling / repairs etc. Finally Web Application front-ends or social apps can provide suitable views into the same data cloud.*

- › Involved APIs, Protocols
 - HTTP (Legacy and historical data), Web sockets (for real time data/video/audio)
 - Sensor Discovery and Capability detection service (upload profile as JSON /SensML structs?)
 - Security Features (may need to be activated to anonymize and secure data streams)

- › Challenges (what to work with W3C, IETF on)
 - Sensor Discovery (mDNS mapped to automotive sensors)
 - Sensor ontology (SensML, other formats?)
 - Architecture able to securely monitor and stream data to a cloud provider)
 - Make end-user aware about the background scan being activated

- › Real world examples
 - Waze (Crowd sourced driving data),
 - Social Web of things (Ericsson)

Use case, Zombie drive



- › *Collect relevant ambient sensor data, mash up with maps, social gaming. Combine with In vehicle entertainment system to create an immersive experience (3D audio, Augmented reality Video). Upload sensor data to other players.*

- › Involved APIs, Protocols
 - Sensor APIs, HTML5 media, Web GL
 - Parking sensors, Surround Audio, Light Sensors

- › Challenges (what to work with W3C on)
 - Need to define external sensors using a discoverable (JSON/XML) profile
 - Need to discover new user interface in car audio system with 3D capability

- › Real world examples
 - Zombie Run, Waze...most Smartphone Fitness apps

Conclusions and Q&A

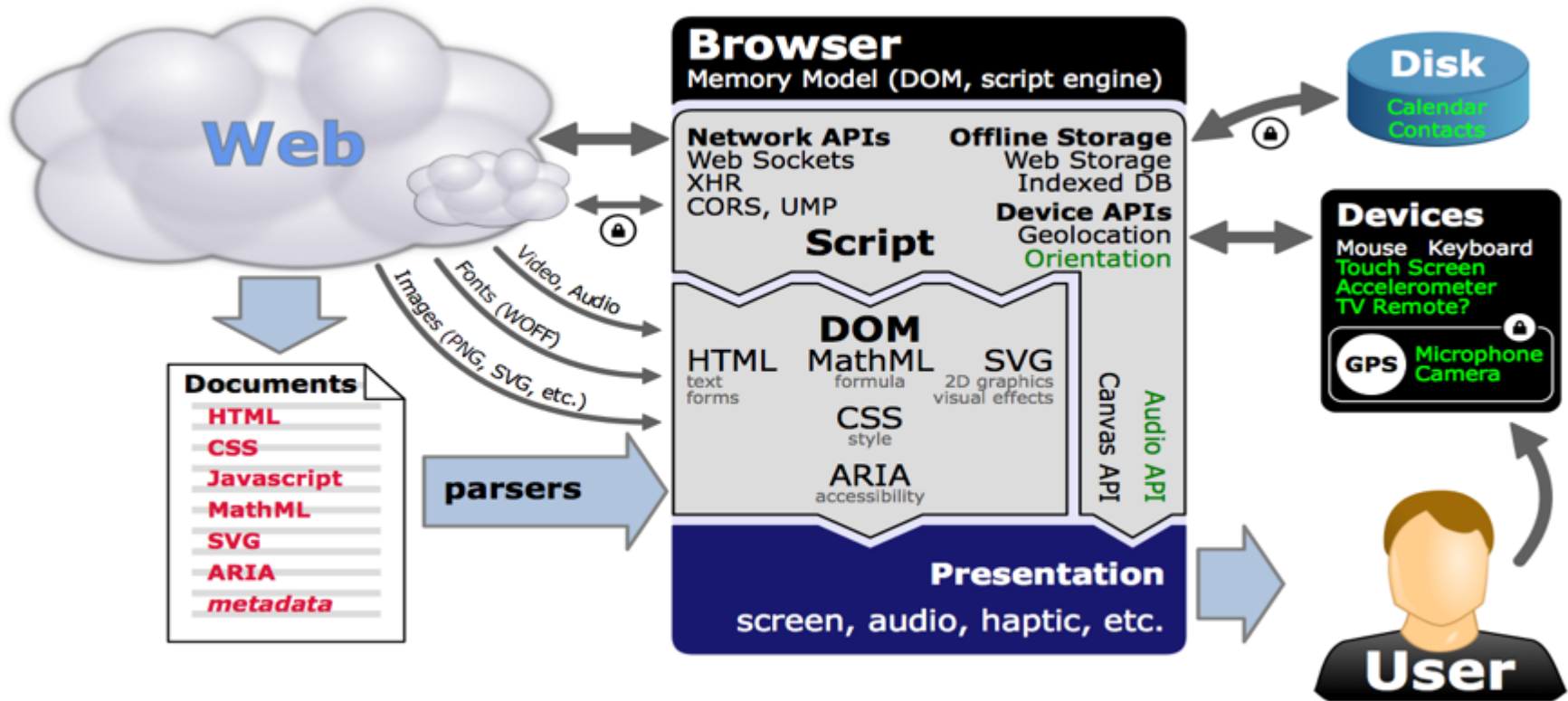


- › Automotive an extra sensory power house
 - May need a leap of faith to become a sharing community
- › Personal Integrity must not become an afterthought
- › Auto Motivate your self to Re-cycle, re-cycle the OWP
 - The Open Web Platform and other W3C innovative ideas
- › Think outside the glove compartment
 - Architecture need to be enable distributed roles /concerns
- › Don't forget to “Connect” with other Eco Systems

Keep the family happy, drive responsibly

Background Slides

W3C Open Web Platform



Courtesy of W3C

W3C Indie UI



› Indie UI: Events 1.0

an abstraction between device-specific user interaction events and inferred user intent such as scroll, activate, etc. This provides an intermediate layer between device- and modality-specific user interaction events, and the basic user interface functionality used by Web applications

› Indie UI: User Context 1.0

a set of properties and methods related to the environmental context of a specific user, and a vehicle to access them to facilitate a Web application's ability to adapt to the user's needs. This is meant to provide information about whether a user is using a screen reader, screen magnifier, or other Assistive Technology, and to expose relevant user settings, allowing optimal adaptation of the Web application's user interface

W3C WebDriver



› WebDriver

– a platform- and language-neutral interface that allows programs or scripts to introspect into, and control the behavior of, a web browser. The WebDriver API is primarily intended to allow developers to write tests that automate a browser from a separate controlling process, but may also be implemented in such a way as to allow in-browser scripts to control a browser. The WebDriver API is defined by a set of interfaces to discover and manipulate DOM elements on a page, and to control the behavior of the containing browser. This specification also includes a non-normative reference serialization (to JSON) of the interface's invocations and responses that may be useful for browser vendors.



ERICSSON