Collaborative Open Market to Place Objects at your Service

D9.1.2 Intermediate Standardization Report

Project Acronym	COMPOSE	
Project Title	Collaborative Open Market to Place Objects at your Service	
Project Number	317862	
Work Package	9	Standardization and exploitation
Lead Beneficiary	9	
Editor	Dave Raggett	W3C/ERCIM
Reviewer	Juan David Parra Rodriguez	University of Passau
Reviewer	Lukasz Radziwonowicz	Fraunhofer FOKUS
Dissemination Level	PU	
Contractual Delivery Date	31/10/2014	
Actual Delivery Date	31/10/2014	
Version	final	

This publication has been produced with the assistance of the European Union. The content of this publication is the sole responsibility of the COMPOSE consortium and can in no way be taken to reflect the views of the European Union.

The European Union is established in accordance with the Treaty on European Union (Maastricht). There are currently 27 Member States of the Union. It is based on the European Communities and the member states cooperation in the fields of Common Foreign and Security Policy and Justice and Home Affairs. The five main institutions of the European Union are the European Parliament, the Council of Ministers, the European Commission, the Court of Justice and the Court of Auditors. (http://europa.eu/index_en.html)

HTML5Apps is a project funded in part by the European Union.

TABLE OF CONTENTS

TABLE OF FIGURES

ACRONYMS

Acronym	Meaning
COMPOSE	Collaborative Open Market to Place Objects at your Service
W3C	World Wide Web Consortium
HTTP	Hypertext Transfer Protocol
API	Application Programming Interface
IoT	Internet of Things
JSON	JavaScript Object Notation
EXI	Efficient XML Interchange
MQTT	Message Queue Telemetry Transport
SDK	Software Development Kit

1. INTRODUCTION

This is an intermediate report on standardisation for the COMPOSE project. It reviews the COMPOSE project and then describes the standardisation workshop organised by W3C with the support of the COMPOSE project. The report then describes the vision of the Web of Things and the plans for a W3C Web of Things Interest Group that will work on use cases and requirements, best practices and identification of specific work items for standardisation. These are expected to cover descriptions of services and their dependencies, scripting APIs, data models, privacy and end to end security. W3C will work on standards for browsers (e.g. direct access to wearables and home automation from smart phones and tablets), service platforms in the cloud, or at the network edge (e.g. home hubs or embedded on phones/tablets) and gateways that bridge IoT and Web protocols.

2. COMPOSE PROJECT

2.1. PERSPECTIVE ON THE INTERNET OF THINGS

The COMPOSE Project is developing a predominantly open source cloud-based platform for services primarily using HTTP for data transfer and JSON as the data representation. COMPOSE supports sensors and actuators with discrete updates. Streamed media like audio and video are not supported. With the use of HTTP for conveying sensor readings and actuator updates, the COMPOSE platform is not intended for applications with tight real-time requirements. Likewise, COMPOSE is not intended for cyber physical system applications that need tightly coordinated control over actuators with precise synchronization.

Sensors and actuators are mirrored by virtual equivalents known as *service objects*. These are updated either periodically or whenever the sensor value changes. Each sensor reading is associated with a timestamp, and the service object may hold a history of readings. Sensors can be controlled through *subscriptions*. These determine when sensor readings are notified to service objects, and which URL to use for delivering notifications to the service object. Sensor specific fields can be used to adjust how the readings are taken. Service objects are associated with rich metadata that is represented in JSON. The COMPOSE platform can act as both an HTTP client and an HTTP server.

COMPOSE allows for a range of deployment scenarios:

The sensor/actuator is directly reachable on the Internet
In this case, the sensor/actuator implements an HTTP server that can be invoked by the COMPOSE platform, e.g. to access the metdata, to subscribe for sensor readings, or to update the actuator.

The sensor/actuator is behind a firewall
In this case, the COMPOSE platform would be unable to open connections to the device, as these will be blocked by the firewall unless

it has been specifically customized to pass such requests through. A work around is for the device to open a Web Socket connection to the COMPOSE platform, as this then allows for asynchronous message transfer in both directions. The COMPOSE Project decided against the alternative of using HTTP to poll the COMPOSE platform for requests.

The sensor/actuator is connected via a gateway

In this case, the gateway uses IoT protocols to access the sensor/actuator, and HTTP to access the COMPOSE platform. The gateway may itself be behind a firewall (see previous case).

Use of MQTT

The MQTT pub-sub protocol can be used as an alternative to HTTP and Web Sockets for pushing data to service objects, and for service objects to push commands to actuators.

COMPOSE applications are regular web applications implemented in HTML and JavaScript. These can access services exposed by the COMPOSE platform. *COMPOSE services* are essentially transformations of data from service objects (see above) and other COMPOSE services, and in turn, expose transformed data for use by other COMPOSE services or COMPOSE applications. COMPOSE supports simple transformations based upon functional expressions. For greater flexibility, you can express transformations in the JavaScript programming language. Applications can be run within the COMPOSE platform or externally. By deploying applications within COMPOSE, developers can take advantage of the platform's support for enforced compliance of COMPOSE services with security policies based upon information flow analysis.

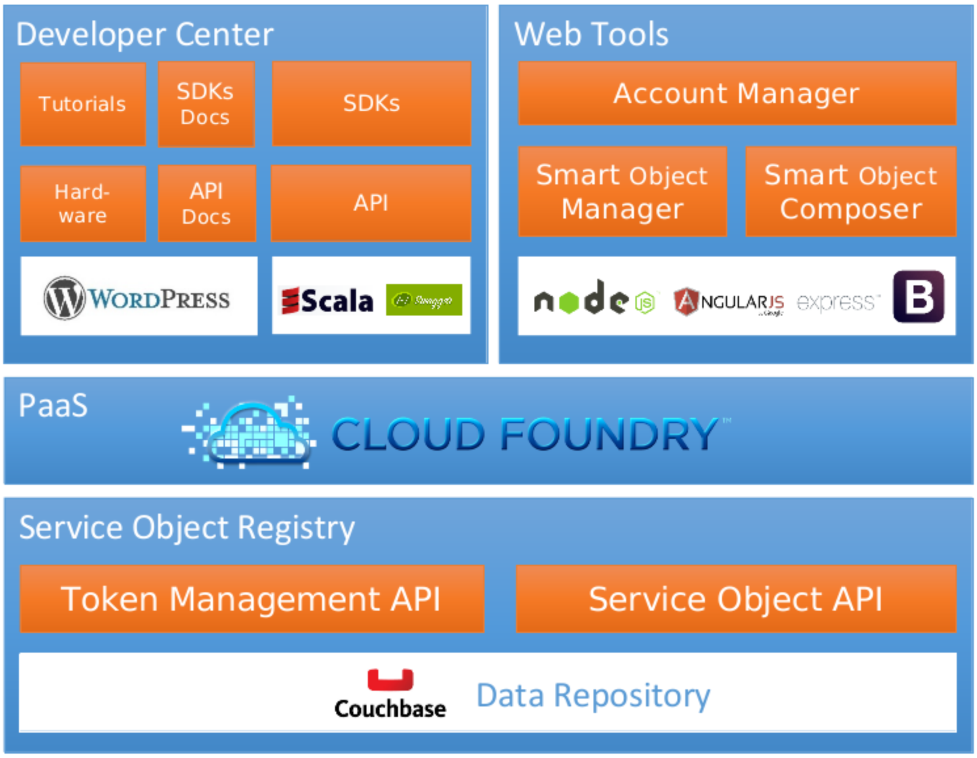

Figure: 1 COMPOSE architecture, with the IoT Dashboard above and the COMPOSE run-time platform below.

COMPOSE provides an IoT Dashboard for developers, and the COMPOSE run-time platform for deploying services. Both are implemented with Cloud Foundry as a basis for scaling to many developers and services. The COMPOSE IoT Dashboard provides developers with a suite of tools for developing COMPOSE services and applications, including a browser based graphical tool for composing services that is based upon NodeRED which itself relies on NodeJS.

The COMPOSE run-time platform is designed for scalability, and implemented on top of Cloud Foundry and Couchbase. Support for JavaScript is provided by NodeJS. Internal platform notifications between COMPOSE components such as service objects and services is currently performed via the peer to peer overlay network developed by IBM. This uses an efficient based topic based pub/sub mechanism.

2.2. PROJECT ADVANCES ON THE STATE OF THE ART

The following draws upon the details in D1.1.0 (Progress beyond the state of the art) and more recent project reports, and seeks to identify potential oppportunities for standardization. It is unlikely that the specific approach taken by the COMPOSE project can be standardized as such since standardization involves a mix of stakeholders with different backgrounds and priorities. However, the experience gained by COMPOSE and related projects will be extremely valuable when it comes to the discussions that precede standardization. Several COMPOSE project partners were involved in the W3C standardization Workshop in June 2014. An important challenge for the final year of the COMPOSE project is for project participants to engage in the pre-standardization activities in the proposed W3C Web of Things Interest Group.

2.2.1. Service-oriented Software Engineering

This is about the role of services for transforming data, along with the means to compose pipelines of services as needed. For COMPOSE, services can be connected to:

- Other COMPOSE services
- COMPOSE service objects that proxy sensors and actuators
- Web applications running on browsers and servers
- Native applications running on Android and other such platforms

Traditional object oriented design includes the distinction between classes and instances of things. You could thus have a class of sensors, and an object corresponding to a particular instance of a sensor. Objects inherit software interfaces defined by the classes that they belong to. The same applies to services and compositions of services.

An early design choice for the COMPOSE run-time platform was to avoid classes. Instead of registering a service object as an instance of a class, you need to create a new service object either from scratch or by copying and modifying an existing one. Likewise for services and service compositions.

More recently, COMPOSE has adopted Node-RED as the basis for designing service compositions. The glue.things mashup platform includes tools for device management, service composition, and service deployment. Here is a screen shot of the service composer:

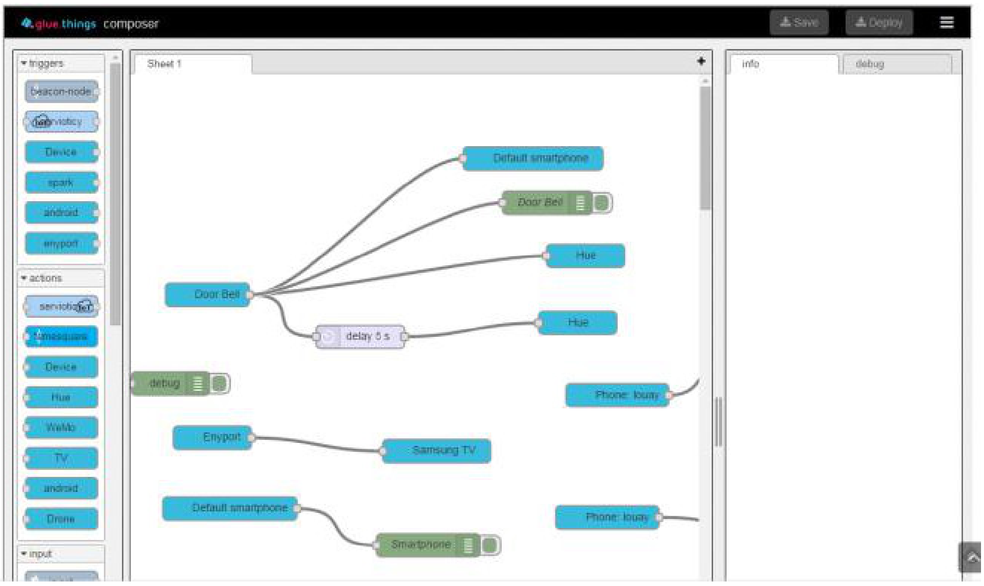

Figure: 2 Glue.Things service composer

This provides a catalog of nodes that can be dragged onto the canvas, customized and wired together. Nodes include service objects, APIs for REST services, for JavaScript code and so forth.

Developers develop and register services, which can then be composed with other services and service objects as needed, and deployed on the COMPOSE run-time platform. This presents a number of opportunities where standardization could be valuable:

- Discovery of services and service objects, either for direct use, or for adaptation for slightly different purposes. This relates to standards for data and metadata formats, service descriptions, and software licenses. Further possibilities include support for trust (e.g. ratings, reviews and recommendations).
- Registration of service objects, services and compositions of these.
- Security policies, see below.
- Deployment on the run-time platform — programming languages, APIs, protocols, distributed storage, service orchestration, scaling policies, software updates etc.
- Monitoring and analytics — high and low level interfaces to support this.
- Monetization - a standard means for paying for services.

2.2.2. Scalable Communication Infrastructure

The COMPOSE platform is designed to support very large numbers of services and service objects with the compute and storage involved spread across many servers. One of the challenges is to efficiently support the flow of notifications between dependent services and service objects. For point to point flows, regular TCP/IP protcols are adequate. When notifications have many recipients, one approach is to make a separate TCP connection for each recipient. Greater efficiency can be obtained by routing notifications through an overlay network of nodes that are interested in a given subset of notifications.

The COMPOSE platform communication infrastructure has been designed to support membership and various kinds of communication patterns for the entities hosted by COMPOSE. The approach is based upon establishing a peer to peer overlay network in which each node is connected to a small number of other nodes, and used as a basis for efficient implementation of the publish - subscribe (pub-sub) communication paradigm. The overlay network automatically adapts as nodes join or leave, whether intentionally or through faults. More details are given in D4.2.2 (Initial Prototype of the COMPOSE communication infrastructure).

Pub-sub communication can be used where entities are interested in a particular "topic", and otherwise do not need to be aware of each other. This can include notifications that enable subscribers to identify when operations have finished and to obtain the corresponding results. For open markets of services, hosted on separate platforms, there will be a need for standard based protocols that can connect services both within and across platforms.

2.2.3. Security

Compose work on security supports data-centric policy enforcement, and data provenance collection. This entails effective means to identify and authenticate users, service objects and their data sources. Static analysis of COMPOSE entities, e.g. services and service compositions, is used as a basis for guiding dynamic analysis of information flows. Data can be associated with data provenance information and with sticky policies that also apply to derived data. Sticky policies and provenance information support the policy decision points in the COMPOSE platform to enforce security policies defined by users. Secure storage is required to protect data and metadata subject to privacy concerns. Many of these aspects would benefit from open standards.

Reputation mechanisms are provided to help developers and end-users decide which services and applications to use, respectively. This is based upon popularity ratings and reviews provided by users, and potentially also by automatic monitoring tools which could use heuristics to determine when a given app/service should be subject to further scrutiny, e.g. by the markeplace operator, or trusted third parties. It is unlikely that reputation mechanisms are suitable for standardization, however, open standards would be valuable for access to metadata concerning reputation and endorsement by third parties. W3C held a meeting on next steps for trust and permissions for web applications in Paris on 3-4 September, and a Community Group is planned to work on best practices and emerging techniques. For more information, see:

- http://www.w3.org/2014/07/permissions/

2.2.4. Marketplace and App Stores

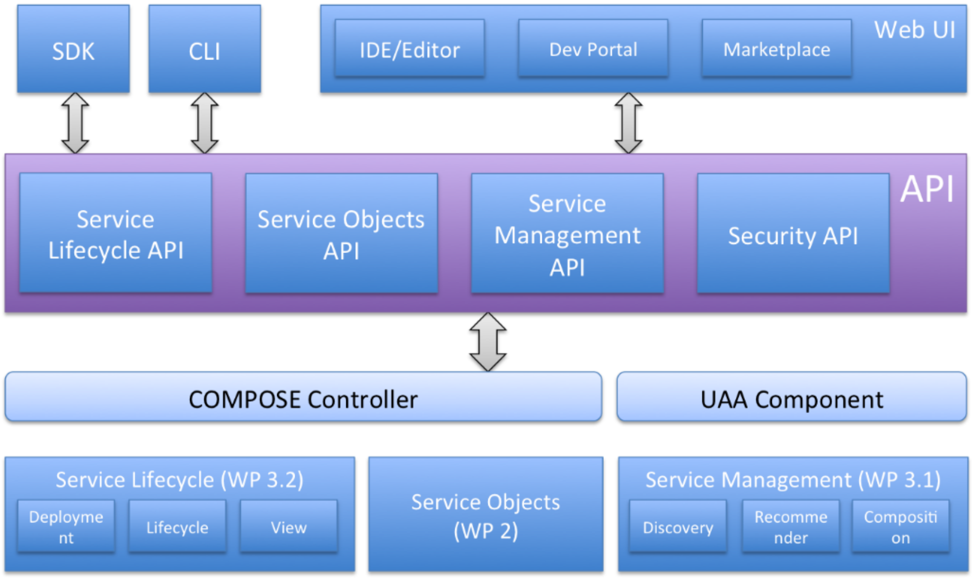

Figure: 3 Functional components for the COMPOSE marketplace.

The COMPOSE marketplace provides the externally visible interfaces for the various stakeholders, e.g. service object providers, service and application developers, and end users. Object providers can register and manage their objects, including security policies. Developers can make use of the SDK and associated APIs for creating sophisticated services and applications. A browser based graphical user interface is available for discovering and composing services. Applications can be deployed within the Compose run-time platform or externally to it. End users can search for applications that match their needs.

The various APIs provide opportunities for standardization. The motivation is to enable developers to deploy applications and services to multiple platforms with minimal effort for tailoring to vendor specific APIs.

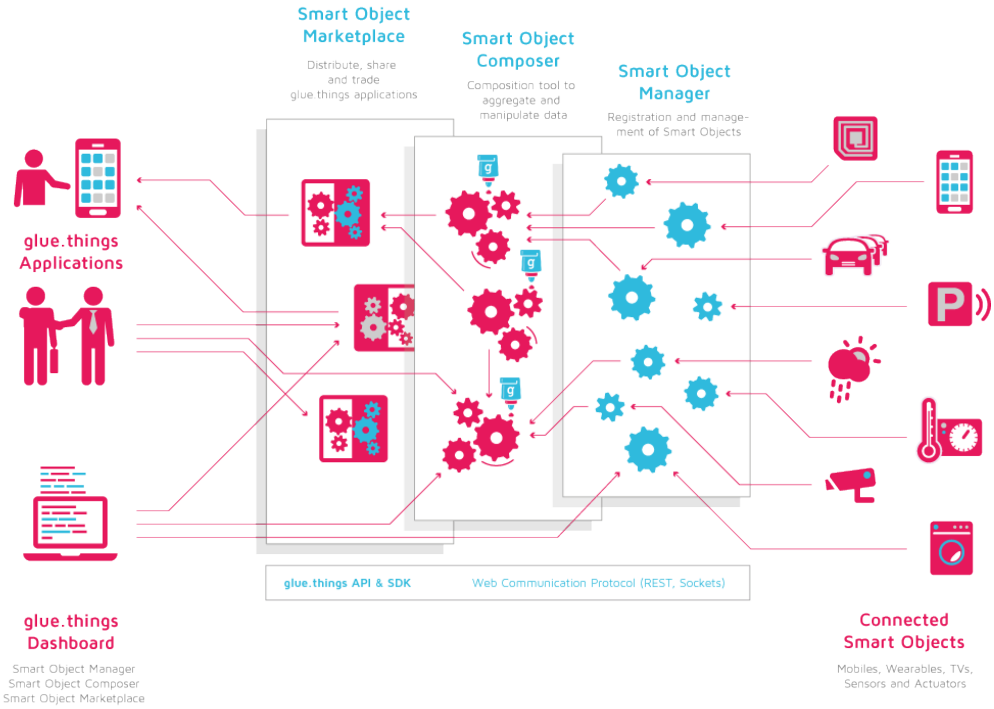

Figure: 4 Designer's perspective of COMPOSE

3. W3C WORKSHOP ON THE WEB OF THINGS

W3C has traditionally launched new standards work following a workshop that was held to assess the level of interest and timeliness for work in a new area.

With the support of the COMPOSE project, W3C held a two day workshop on the Web of Things in Berlin on 25-26 June 2014, hosted by Siemens. The workshop was attended by 120 people (the maximum allowed by the venue). Here are links to the workshop materials:

- Call for papers
- Program Committee
- Workshop Submissions
- Agenda
- Workshop Report

3.1. INTRODUCTION

The following extract is from the workshop introduction.

It is common to think about the Internet of Things from the perspective of sensors and transport protocols, but you can also think about it from the point of view of services, which is where most of the money is expected to be made:

> *Services, Not Sensors: Gartner expects Internet of Things vendors to top $309 billion in direct revenue by 2020, with most of that money deriving from services.*

This of course will depend on open standards to get us out of the current fragmentation where companies are working in isolated silos:

> *The trick will be whether hardware companies will push hard enough for standardization so they can capitalize on services revenue. Companies that see themselves as pure hardware manufacturers are likely doomed, but those that see beyond the "things" to instead focus on the services built on the "Internet," the future is very bright. Matt Asay, MongoDB*

Web technologies are expected to be very important, e.g. JavaScript and open standards for data formats, interface definitions, access control and so forth:

> *Eventually, something like HTML, the language of the web, will be required to make the internet of things realize its potential.*

"Interoperability is critical," says Mike Bell, head of wearables at Intel.

Continuing advances in electronics have dramatically reduced the cost for devices functioning as tags, sensors and actuators for the physical environment, i.e. the Internet of Things (IoT). The market potential for the IoT is currently held back by fragmentation due to a plethora of communication technologies and the lack of a common approach to enabling services.

This workshop will examine the potential for open standards as a basis for services, either between devices, at the network edge, e.g. in home hubs, or in the cloud. It will discuss the use of web protocols and scripting languages for implementing services, the need for APIs for implementing drivers for specific IoT technologies, a shared approach to describing services as a basis for interoperability, and the underlying use of HTTP/COAP, Web Sockets, and EXI/JSON for RESTful services.

Figure: 5 Web of Things application domains

There is a very wide range of application domains, and the following is just a sample:

- Home automation and extended warranties
- Security for homes, businesses and public settings
- Healthcare at home and in hospitals

- Manufacturing, construction and retail
- Transportation: cars, buses, metro, trains
- Utilities: electricity, water, gas, drainage
- Energy efficiency, smart appliances and the smart grid
- Managing emergencies: floods, fires, earthquakes and civil disturbances

---

Extending the Open Web Platform as the basis for Open Markets of Services that combine the Web of Devices with the Web of Data to form the Web of Things.

The Web of Things is expected to have broad and sweeping economic and societal impact. Open standards will be critical to enabling exponential growth of the kind we experienced with the early days of the Web, that saw it growing from a handful of enthusiasts in the early nineties to a global phenomenon in just a few years.

The proposed topics for the workshop included:

- Core Technologies
- Domain Challenges
- From Things to the Web of Things
- The Role of Semantics
- Security, Trust and Privacy
- User Interfaces
- Scalability
- Open Markets

Workshop Chairs

- Joerg Heuer, Siemens
- Dave Raggett, W3C

Program Committee

- Steve Bratt, GS1
- Meng Chee, Samsung
- Bill Curtis, ARM
- Dietmar Dengler, DFKI
- Wolfgang Dorst, Bitkom
- Donald Ferguson, DELL
- Dominique Guinard, Evrythng
- Joerg Heuer, Siemens
- Ajit Jaokar, futuretext
- Jonathan Jeon, ETRI
- Scott Jenson, Google
- Kazuo Kajimoto, Panasonic
- Thomas Knebel, German Federal Ministry for Economic Affairs and Energy (BMWi)
- Ryuichi Matsukura, Fujitsu
- Milan Milenkovic, Intel's Internet of Things Group
- Claes Nilsson, Sony

- Milan Patel, Huawei
- Carlos Pedrinaci, The Open University
- Dave Raggett, W3C
- Mitko Vasilev, Cisco
- Laurent Walter Goix, Telecom Italia

Would be participants were invited to submit expressions of interest, and longer position papers, for publication on the workshop website. These were reviewed by the program committee to identify people to invite to give presentations, appear on panel sessions, etc. We also asked for people interested in poviding practical demonstrations.

3.2. THE WORKSHOP REPORT

The two days of the workshop were split into a variety of sessions: talks, panels, demonstrations and parallel break-out sessions. The following is a brief overview, to see more details, please consult the workshop report.

3.2.1. Talks

We planned for 20 talks, each with 15 minutes including questions and answers. These were split into the following sessions:

- Sharing experiences, use cases and requirements
- From things to the Web of Things
- Security, trust, privacy, provenance, access control and policies
- Semantics, linked data, vocabularies and best practices

3.2.2. Panel Sessions

We had three panel sessions:

- Core technologies
    - Moderator: Jörg Heuer, Siemens
    - Markus Isomäki, Nokia [slides]
    - Charalampos Doukas, CREATENET
    - Charles Eckel, Cisco [slides]
    - Milan Patel, Huawei
    - Matthias Kovatsch, ETH Zurich
- Who is doing what (IoT projects and standardization bodies)
    - Moderator: Claes Nilsson, Sony
    - Richard Soley, OMG [slides]
    - Eric Kauz, GS1 [slides]
    - Mike Bergman, CEA [slides-short, slides-long]
    - Istvan Lajtos, GSMA [slides]
    - Ingo Friese, DT [slides]
    - Nick Allott, UbiApps [slides]
- Conclusions and next steps
    - Moderator: Dave Raggett, W3C
    - Philipp Hoschka, W3C [slides]
    - Milan Milenkovic, Intel [slides]
    - Ryuichi Matsukura, Fujitsu [slides]

  - Milan Patel, Huawei
  - Dominique Guinard, Evrythng
  - Laurent Walter Goix, Telecom Italia, [slides]

3.2.3. Break-out sessions

We invited proposals for break-out sessions spread across both days. Here is the list of the sessions that took place. You can see the session proposals and minutes in the workshop report.

- Service Descriptions for the Web of Things, Dave Raggett
- Interaction Models for the Web of Things, Erik Wilde
- Business Models for the Web of Things, Alessio Gugliotta
- Digital Object Memories for the Web of Things, Jens Haupert
- The Web of Things Friendly Label, Dominique Guinard
- Privacy in the Web of Things, John Mattsson
- Scripting in the Web of Things, Andreas Harth and Ricardo Morin
- The Web of Data We Want, Phil Archer
- Application Layer Protocols and Data Encoding for Constrained Devices, Hauke Petersen
- Think Robot, the next smart object, Redouane Boumghar

3.2.4. Demonstrations

We arranged for demonstations during the lunch time, morning and afternoon breaks. Here is a list of the demonstrators:

- David Conway-Jones (IBM) Node-RED - a wiring tool for the web of things.
- Kosuke Nagano (ACCESS Company) Connect devices to the Web
- Ricardo Morin (Intel) - Using remote Web Workers to dynamically off-load computationally intensive tasks
- Mitko Vasilev (Cisco)
- Benoit Herard (Orange) sensoNet - A low cost, arduino-based open source sensor network enabler
- Nicholas Allott (NquiringMinds) - secure IoT using web friendly protocols
- Dominique Guinard (EVRYTHNG) EVRYTHNG Web of Things API
- Andreas Harth (Karlsruhe Institute of Technology) On-the-fly Integration of Static and Dynamic Linked Data
- Robert Kleinfeld (Fraunhofer FOKUS) glue.things Platform as a Service (PaaS) for integrating the Internet of Things with the Internet of Services
- Scott Walsh (Plantronics) Web Connected Contextually Aware Wearable Device Streams via JavaScript APIs on a Chromebook
- Jens Schmutzler (Technische Universität Dortmund) WoT-based Vehicle to Grid Communication and Conformance Testing
- Thomas Amberg (Yaler) Web-enabled Bluetooth Low Energy Watcher
- Sebastian Käbisch (Siemens) µRDF Store with Efficient RDF
- Charalampos Doukas (CREATE-NET) Bridging HTTP REST with M2M protocols like MQTT

3.2.5. Next Steps

The Web of Things is expected to develop into a huge market of services, and W3C is well positioned to develop open standards around service platforms, security, privacy, and integration with the Web of data for a Web of services. This positions W3C to define standards at the service and application layer, complementing the role of other standards development organizations and industry fora that are focusing on the device layer and the array of communication technologies for accessing sensors and actuators, that form the Internet of Things.

The workshop gave a strong message of support for W3C to initiate work on standardization for the Web of Things. The foundations include RESTful HTTP and pub-sub protocols, but the detailed requirements vary across the use cases, e.g. for latency, throughput, transactional robustness and so forth. Building upon these foundations, the workshop identified the need for standards for Web APIs that abstract away from the protocols, including the wide range of IoT technologies used at the network edge to connect to sensors and actuators. Through standardization, we can encourage re-use of APIs and data models.

The closing session of the workshop raised a proposal to launch a W3C Interest Group. The existing W3C Web and TV Interest Group provides a model for what we can expect for the Web of Things Interest Group. It has established requirements and handed work off to existing Working Groups or launched new Working Groups where appropriate. The Interest Group consists of several tasks forces for different topics with different leads for each. W3C Interest Groups work on gathering use cases and requirements, surveying existing work, and identifying gaps, general introductions and best practices. There are no intellectual property commitments as a precondition of participation. A consequence of this is that Interest Groups can't work on specifications. However, Interest Groups can produce charters for specification work in W3C Working Groups.

The proposed Web of Things Interest Group will require:

- A formal charter
- Chair(s)
- Task force leaders
- Document editors
- Contributors

The W3C staff will drive the process of chartering the Interest Group through reaching out to a wide range of interested stakeholders. This consultation will seek to clarify the mission for the Interest Group, the target topics and industry priorities, and opportunities for liaisons with other related standards development organizations.

3.2.6. Participation

There were 119 registered participants. We planned for 120 people, but one person was unable to come at the last minute.

The following shows the different categories of organizations who provided submissions to the workshop.

3.2.6.1. Companies

- ACCESS
- Beijer Electronics
- Bosch Rexroth AG
- Bosch Software Innovations
- Canon
- CA Technologies
- Cisco
- Dacteev
- EDF
- Ericsson
- Fujitsu
- Hewlett-Packard
- Huawei
- IBM
- INNOVA
- Intel
- KDDI Corporation
- kieback&peter
- LG Electronics
- mobiLead
- Nokia
- NTT
- Oracle
- Orange
- Panasonic
- Plantronics
- Samsung Electronics
- Siemens
- Software AG
- Sony Mobile
- Telecom Italia
- Toshiba
- Yaler Gmbh

3.2.6.2. Startups

- Algebraix Data
- EVRYTHNG
- JOLOCOM
- Monohm
- nquiringminds
- Oberon microsystems
- sensefields
- soixant-dix
- The Grid

3.2.6.3. Universities

- Aston University
- Department of Engineering, University of Modena & Reggio Emilia
- Dortmund University of Technology (CNI)
- ETH Zurich
- École polytechnique fédérale de Lausanne (EPFL)
- Freie Universität Berlin
- Kanagawa institute of Technology
- Karlsruhe Institute of Technology
- Ontology Engineering Group, Universidad Politécnica de Madrid
- University of California, Berkeley
- Université de Lyon
- WU (Vienna University of Economics and Business)

3.2.6.4. Research Institutes

- Center For Development of Advanced Technologies (CDTA))
- Centre National de la Recherche Scientifique
- CREATE-NET
- DFKI
- ETRI
- fortiss
- Fraunhofer FOKUS
- Fraunhofer Institute for Applied Information Technology (FIT)
- Insight Centre for Data Analytics

- INRIA

3.2.6.5. Standards Organizations, Industry Associations, and Government Institutions etc.

- BITKOM e.V.
- CEA
- European Commission
- Federal Ministry for Economic Affairs and Energy, Germany
- Free Software Foundation Europe
- GS1
- GSMA
- Japan Smart Community Alliance
- Industrial Internet Consortium
- Linux Foundation
- OMG
- W3C

4. THE WEB OF THINGS

This section considers the vision for the Web of Things in relation to opportunities for standardization, and builds upon what was learned from the W3C June Workshop, and the experience gained during the first two years of the COMPOSE project.

4.1. INTERNET OF THINGS CONNECTIVITY TECHNOLOGIES

There is a large number of technologies for connection to IoT devices, and these are continuing to evolve at a rapid pace. The following is intended to illustrate the variety of technologies and isn't meant to be comprehensive.

- **Hypertext Transfer Protocol (HTTP)** may be used for powered devices with a wired network connection or support for WiFi. HTTP is often used in conjunction with the Representational state transfer (REST) design pattern. HTTP is a client-server protocol, but can be used in a polling mode to handle requests pushed to the device by a server.
- **Web Sockets** is similar to HTTP, but allows for asynchronous message transfer in either direction. Web Sockets is often used with JSON for remote method invocation and event notification.
- **Constrained Application Protocol (CoAP)** is designed as an IP protocol for embedded or constrained devices. It translates easily to HTTP for integration with the Web and RESTful APIs. It also supports notifications pushed from a server to the device. CoAP is often used together with 6LoWPAN for short range wireless connections
- **6LoWPAN** is short forIPv6 over Low power Wireless Personal Area Networks. It is layered on top of the IEEE 802.15.4 standard for the physical layer and media access control for personal area networks, and may be used in conjunction with CoAP.
- **ZigBee** is a low power wireless communications technology optimized for devices requiring a very long battery life. ZigBee is layered on top of the IEEE 802.15.4 standard for the physical layer and media access control for personal area networks.
- **Near Field Communications (NFC)** is a very short range wireless technology and can be used to access sensor readings, and operate door locks, or to open the browser in a smart phone to a URL for a web page relating to the tagged object.
- **Bluetooth** is a short range technology with a suite of profiles for different categories of applications. Bluetooth Low Energy (BLE) offers extended battery life. It can be used for exchange of small amounts of data, either in a broadcast mode or for bidirectional connections. This is expected to be of increasing importance for applications running on smart phones or tablets. The W3C Bluetooth Community Group is drafting an API based upon the GATT profile for BLE, and Google have proposed the use of BLE for broadcasting URLs as part of their vision for the Physical Web.

- **ANT** is a proprietary sensor network technology operating in the 2.4 GHz band. It can be used to transfer small amounts of data across networks with hundreds of sensors.
- **DASH7** is designed for long lived battery operated sensor networks, it works in the 433 MHz unlicensed band. The range is up to 1000m depending on power levels and data rates. Like ZigBee and BLE, DASH7 is aimed at transferring small amounts of data, and unsuitable for audio or video.
- **KNX for buildings** s a standardized (EN 50090, ISO/IEC 14543), OSI-based network communications protocol for intelligent buildings. KNX is the successor to, and convergence of, three previous standards: the European Home Systems Protocol (EHS), BatiBUS, and the European Installation Bus (EIB or Instabus). The KNX standard is administered by the KNX Association. KNX can be realized over a mix of networking technologies, e.g. twisted pair cable, powerline networking, radio (KNX-RF), infrared and conventional ethernet.
- **EnOcean** is a similar protocol to KNX for sensors that are self powered, e.g. harvesting energy when you push a switch that is sufficient for sending 2 or 3 packets. The sensors are quite expensive (e.g. 60 CHF) but available for motion sensors (light and thermal IR), beds, seats, window handles and so forth.
- **Infrared** is widely used for remote control of TVs, air conditioners etc. Infrared was popular for PDAs and laptops in the late 90's and early 2000's, but lost ground to RF technologies such as WiFi and Bluetooth. Infrared is making a comeback for fast transmission of photos from phones to printers etc.
- **Universal Serial Bus (USB)** s an industry standard defining cables, connectors and protocols. It is widely used for connecting devices to computers, e.g. keyboards, mouse pointers, hard drives for storage, game controllers, and also for connecting to printers, scanners, digital cameras, smart phones and tablets. USB is designed to power devices and is commonly used for charging device batteries, replacing the need for a separate cable.
- **Wireless USB (WUSB)** is a standard for connecting devices using a wide band protocol in the 3.1 GHz to 10.6 GHz region. The range is 3 to 10m.
- **IEEE 1394 (Firewire)** is a serial connection designed for high speed transfers, and similar in some ways to USB. IEEE 1394 has lost ground to USB as the latter has increased in speed, and due to the need for a separate power connection for Firewire devices.
- **WiFi (IEEE 802.11)** is a popular local area network technology for managed or ad hoc networks in 2.4 GHz or 5 GHz bands. WiFi access is increasingly available in public locations and even on bus's trains and aeroplanes.
- **Machine to Machine (M2M)** is a generic term for wired or wireless communication technologies between devices. Mobile network operators are promoting cellular M2M, e.g. based upon GSM data modules, for applications such as smart meters.
- **Low Throughput Network (LTN)** is a wide area wireless technology defined by ETSI, and offers long range and minimal battery consumption.

- **Weightless** is a protocol for using white space spectrum for exchanging data between a base station and thousands of client devices. Base stations are directly connected to the Internet. Clients are allocated a schedule of times and frequencies to communicate with their base station. A database is used to avoid interference with local terrestrial TV broadcasts.
- **MQTT** is a lightweight publish-subscribe protocol based upon TCP/IP connections. It is intended for embedded/constrained devices, and needs to be used in conjunction with a message broker.
- **XMPP** is an XML based protocol used for presence, instant messaging, and real-time communication and collaboration.
- **Efficient XML Interchange (EXI)** is a binary format for structured data that is suitable for embedded/constrained devices and offers further compression when used with a specific XML schema. It may be used in conjunction with CoAP.
- **JavaScript Object Notation (JSON)** is a textbased representation for structured data that is increasingly popular with Web developers. JSON-LD is a set of conventions for using JSON for linked data.

The large number of these technologies and the rapid rate of change presents challenges to application and service developers. This points to the need for an abstraction layer that decouples app/service developers from lower level details, thereby increasing robustness and resistance to change.

4.2. POSITIONING OF THE WEB OF THINGS IN RELATION TO THE IOT

The W3C Berlin Workshop in June 2014 gave strong suppport for the role of web technologies at the application/service layer for the Internet of Things. This was reiterated at the Internet of Things 2014 conference held at the MIT Media Lab on 6-8 October 2014. The term *Web of Things* is rapidly gaining in popularity for this role, with widespread acceptance that open standards are critical to enabling open markets of apps/services. In particular, we need standards for identification, discovery and interoperation across vendors, and close attention to security and resilience. There is a bright future for scripting in browsers, service platforms and gateways that bridge IoT protocols and the Web.

The "Things" in the Web of Things are not limited to connected devices, but can also include things that are not and cannot be connected such as people and places, and abstract ideas, such as concerts, organizations, and time periods (e.g. the 70s). Each thing can have one or more virtual representations — **avatars**. Avatars are discoverable and have identities, rich descriptions and services. Avatars have URIs and are accessible via web technologies. Avatars make it easier to build applications and services that combine information from different sources and different levels of abstraction.

4.3. BUILDING BRIDGES BETWEEN DIFFERENT COMMUNITIES

The Internet of Things covers many application domains, e.g. consumer electronics, retail, manufacturing, construction & building management, transport, major utilities (water, gas, electric) and healthcare. Each of these will have its own distinct way of working and this creates challenges for bringing people together to share use cases and requirements and forge a common vocabulary. This also applies to people from different professional backgrounds, e.g. hardware engineers, system administrators, application developers and business managers, each of whom will have different ways of talking about problems. This stresses the need for effective outreach and clear explanations if standardization is to be successful and lead to wide adoption..

4.4. A BROADER SCOPE THAN THE COMPOSE PROJECT

The COMPOSE project focuses on a **scalable cloud based platform**. However, stakeholders have an even broader set of needs:

There is a need for work on standards for **direct access to IoT devices**, e.g. from smart phones and tablets to wearables, healthcare devices, home automation and services advertised by devices in your proximity (Google's Physical Web). W3C expects to play a role in developing related standards for browser APIs to support these.

There is interest in open standards for **service platforms at the network edge**, e.g. in home hubs or smart phones/tablets, which allow users to install apps and services for devices in the home. Standards are also needed for **gateways that bridge IoT devices and the Web**. Such gateways need to be updatable to enhance functionality, to fix bugs and security flaws. Gateways also enable the use of stronger security than may be possible on resource constrained sensors. Open standards would allow devices to be identified as a basis for selecting the appropriate drivers. Scripting languages such as JavaScript and Lua could be used within gateways along with standards for APIs. Scripting languages could be precompiled into byte codes for efficient execution. For really resource constrained devices, the Forth scripting language is a possibility due to its very low demands on memory.

Cyberphysical systems involve a control loop that bridges sensors and actuators. In some use cases, there is a need for tightly synchronized control over multiple actuators. The control loop could be expressed at multiple levels of abstraction with lower level control delegated to controllers near to the network edge. The communication protocols need to be matched to the latency and jitter requirements. This calls for standards for real-time scripts that can be loaded into controllers. In some cases latency is less of an issue, and transactional robustness is much more important.

4.5. OPEN MARKETS OF APPLICATIONS AND SERVICES

What kinds of standards are likely to be important for open markets of applications and services? A starting point is the means to discover apps/services and to assess their trustworthiness via third party endorsements, reputation scores and reviews. The means to support this should be independent of the company hosting an app store or service platform. This implies the need for standards for describing apps/services and for accessing endorsements etc.

The existence, and the descriptions of apps/services could be subject to access control. This creates challenges for scalable discovery, and hints at the need for distributed discovery mechanisms that can enforce access control policies.

Rather than searching for a specific service, users may want to express their intent and leave this to the discovery system to identify service compositions that match that intent. A number of projects have been looking into this besides the COMPOSE project itself (see D2.3.2). As an example, see the presentation by Simon Mayer (ETH Zurich) at IoT2104.

- Configuration of Smart Environments Made Simple

This confirms the practicality of automatic service composition based upon semantic descriptions of services covering both what a service does and the details of the service API. Simon Mayer reported scaling tests in which it took 300 mS to chain 2000 services. Carlos Pedrinaci (a COMPOSE team member) reports (personal communication) that he has achieved compositions involving 8000 services in one second. This includes an exhaustive examination of alternative compositions satisfying the user's request and the selection of the "shortest". The approach combines service discovery and service composition.

There is an analogy with Linux packages (Debian deb and Redhat rpm) which define a unique name and version number for the package, and the version number constraints for the packages it depends on. For Web of Things services, we need to describe the software interface the service exposes, the interfaces it depends upon, and the function of the service. This requires semantic descriptions couched in terms of standard vocabularies.

For greater flexibility, the discovery service could delegate service composition to third parties. A search engine would recognise the user intent from search strings, and invoke a intent specific interface to third parties that have registered for that intent. These third parties would need to return results within short time limits for the search engine to assemble for presentation to the user. Third parties could provide a valuable role in bridging different vocabularies, something that is inevitable as disparate communities evolve their own vocabularies to suit their specific requirements.

Standards are also needed for discovery and registration of devices, something that needs to be simple and robust. This often involves some form of association that sets up a security context for future communications.

Users may have to scan or type a code, or use NFC for touch based association.

The term *avatar* is a little more general than "service", and can be used for more than sensors and actuators. Avatars could represent a synthesis of sensor data, and other information and services. This allows developers to combine information from the physical environment with information about people, places, and abstract things like events (e.g. concerts and football matches), organisations, time periods (e.g. the 70's) and so forth. Standards are needed for identification of avatars, machine and human interpretable descriptions, the services that avatars expose, and for privacy and confidentiality policies.

End to end security is essential for individual privacy, to protect confidential information and to prevent harm from cyber attacks. Special attention is needed for the connection to sensors and actuators, as these may be too constrained to support the level of security expected across the Internet. This points to the need for standards for web facing gateways, and requirements relating to identity and authentication. Security also needs to be considered in terms of different levels of abstraction in distributed system. This goes along with the principles for defence in depth, so that a breach at a given level/interface can be contained rather than spreading contagion to the whole system.

Resilience for the Web of Things will involve a number of factors:

- Adapting to changes in demand — to address denial of service attacks or to cope with sudden increases in popularity
- Recovering from faults — which will be inevitable in systems with very large number of devices, whether from hardware faults, or software faults, e.g. failed upgrades
- Accommodating a heterogeneous mix of devices — e.g. from different vendors, and different ages and versions of equipment
- Containing and recovering from cyber attacks — whether from criminal organizations, angry individuals or hostile nations

This calls for work on best practices and an analysis of what standards are needed to support the associated mechanisms, for example relating to monitoring services.

5. W3C INTEREST GROUP FOR THE WEB OF THINGS

An outcome of the W3C Workshop in June 2014 was a proposal to launch a W3C Interest Group following the successful precedents of the Web & Mobile and Web & TV Interest Groups. A draft charter was circulated as a basis for discussion in September 2014, targeting the people who attended the June workshop, and the members of the Web of Things Community Group. The following description of the proposed Interest Group is extracted from the draft charter.

W3C Interest Groups are open to W3C Member organizations and Invited Experts, and others who have observer status. The Chairs and document Editors are expected to contribute one to two days per week towards the group. There is no minimum requirement for other Participants.

In order to make rapid progress, the group MAY form several task forces, each working on a separate topic. The group members may participate in one or more task forces.

5.1. GOALS

The mission of the Web of Things Interest Group is to accelerate the development of open markets of applications and services based upon the role of Web technologies for a combination of the Internet of Things (IoT) with the Web of data.

Open markets encourage innovation through free competition. For the Web of Things, this requires open standards for identification, discovery and interoperation of services across platforms from different vendors, and will involve the need for rich descriptions and shared data models, as well as close attention to security and privacy. Wide support for open standards will stimulate growth by lifting the burden on developers for tailoring their products to vendor specific platforms.

This includes the potential for using scripting languages like JavaScript, data encodings such as JSON and EXI, formats for data and metadata, including Linked Data, and protocols such as HTTP and WebSockets, to name just a few examples. JavaScript could be used for direct access to IoT devices from the browser, in service platforms in the cloud or at the network edge, and for device drivers in gateways that use IoT protocols to access embeded / constrained devices, and web protocols to expose them to service platforms.

Applications and services often need data at a higher level than the raw data provided by sensors. Moreover, data needs to be interpreted in the context of other sources of information. The same applies to control systems whose actions need to be translated in context into actions on lower level entities. The Web of Things needs to be able to model the real world at different levels of abstraction, and to enable open markets with free competition of services across these levels. The things in the Web of Things can be considered as virtual representations of objects.

A consequence of this is that the "Things" in the Web of Things are not limited to connected devices, but can also include things that are not and cannot be connected such as people and places, and abstract ideas, such as events (e.g. a concert), organizations, and time periods (e.g. the 70s). Each thing can have one or more virtual representations (avatars). Avatars have identities, rich descriptions, services, access control and data handling policies. Avatars have URIs and are accessible via web technologies. Avatars make it easier to build applications and services that combine information from different sources and different levels of abstraction.

The aim is to launch the Interest Group by the end of 2014. To have initial reports by October 2015 and proposals for initiating standards work by the end of 2015. These are expected to include support for discovery, rich descriptions of services, including the interfaces they expose and are dependent on, scripting APIs, privacy and end to end security, and data models for industry specific application domains.

5.2. SCOPE

The Web of Things Interest Group is a forum for discussion of requirements of Web technologies that enable the development of open markets of products and services based upon tags, sensors and actuators (the Internet of Things) and the Web of data. Such services can be hosted at the network edge, e.g. in home hubs, or in the cloud for increased scalability. By focusing on the services layer that overlays the devices and communication protocols used to access them, W3C's work will complement work by other standards development organizations that focus on the Internet of Things. The aim is to identify opportunities for standards that will enable services to break out of product silos.

The forum is intended to include organizations that commission such products and services, designers, developers, equipment manufacturers, tool and platform vendors, operators and other relevant participants in the value chain that creates and operates such products and services.

It is intended that the forum provides a focus for participants from a wide range of sectors including consumer electronics, retail, manufacturing, construction, transport, major utilties, healthcare, technology, network operators, service creation companies and marketplace hosts.

The first step for the Interest Group will be to collect and publish Web of Things use cases from these different sectors/domains and to identify common elementary use cases and requirements. This is essential for ensuring a shared understanding and will be a prerequisite for splitting work up into task forces that can then proceed in parallel. Some potential candidates for task forces include:

- Survey of existing practices and standards relevant to the Web of Things

- Identifying requirements for open markets of services for the Web of Things
    - Data modeling
    - Unique identification of things, for example using URIs
    - Registration of things and the services they provide
    - Service descriptions and dependencies
    - Coordination and synchronization
    - Real-time control and cyber-physical systems
    - Discovery and trust management
    - Scripting: browsers, service platforms and gateways
    - Bridging the gap between the Web of Things and the IoT
    - Relationship between app/service layer and network layer
    - Monetization
    - Provisioning and lifecycle management
- Gathering requirements and techniques relating to security and privacy for the Web of Things, e.g. strong authentication and data handling policies
- Requirements and best practices for integrating the Web of data
- A study of scalability — how to support billions of users, and what is needed to enable services to dynamically adapt to changes in demand — what are the implications for standards?
- Investigating the potential role of social relationships between people, places and things
- Analysis of requirements for industry initiatives like Industrie 4.0 and the Industrial Internet
- Coordination and outreach for different sectors/domains, and an identification of opportunities for shared data models, e.g. searchable repositories like schema.org
- Collaborations with other standards development organizations

5.3. DELIVERABLES

The primary deliverables of the Web of Things Interest Group are IG notes that identify requirements for existing and/or new technical specifications, that would advance the group's mission.

In addition, the group will review and comment on documents generated by the other W3C groups and may review documents coming from external organizations.

Use Cases and Requirements for the Web of Things

This document will collect use cases from a range of application domains, and identify requirements that are shared across domains, and those that are specific to each domain.

Survey of Existing Practices and Standards Relevant to the Web of Things

This document will look at existing practices and standards, and identify opportunities for new work.

Requirements for Open Markets of Products and Services for the Web of Things
This document will examine what is needed to enable open markets of applications and services for the Web of Things. See Section 1.0 for some of the topics that will need to be covered.

High level architecture for the Web of Things
This document will provide a high level overview of the architectural components of the Web of Things.

End to End Security for the Web of Things
This document will look at what is needed to ensure end to end security for applications and services for the Web of Things. For instance techniques for identifying and authenticating users, devices, applications and services; techniques for ensuring confidentiality and privacy, including encryption, access control and privacy policies. This should include an examination of the potential role and security impact of social relationships between people, places and things.

Resilience for the Web of Things
This document will look at what is needed to ensure resilience of applications and services in the face of varying demand, heterogenous versions of hardware and software, hardware and software faults, and cyber attacks.

Additional deliverables may be added by the Interest Group as appropriate to explore specific topics in greater depth.

5.4. W3C GROUPS

The charter identifies the following W3C groups:

Technical Architecture Group
The breadth of the Web of Things is likely to raise issues that effect the Web as a whole, and when this occurs, the Web of Things Interest Group will seek the help of the Technical Architecture Group with these issues.

Data Activity Coordination Group
The Web of Things Interest Group will liaise with the Data Activity Coordination Group to ensure that the use cases and requirements for the Web of Things are understood by W3C working groups developing technologies that support the automation, integration and reuse of data across applications.

Privacy Interest Group
The Web of Things Interest Group will collaborate with the Privacy Interest Group in respect to requirements and techniques for ensuring privacy for applications and services for the Web of Things.

Web Payments Interest Group
The Web of Things Interest Group will collaborate with the Web Payments Interest Group in respect to requirements for monetizing products, applications and services for the Web of Things.

Web Security Interest Group

The Web of Things Interest Group will collaborate with the Web Security Interest Group to enable end to end security of applications and services for the Web of Things.

Web Crypto Working Group

The specifications of the Web Crypto Working Group are very pertinent to the aims of the Web of Things Interest Group in respect to confidentiality and authentication.

Web App Sec Working Group

For techniques for constraining services to a minimum set of capabilities as a means to increase security through a reduction in the attack surface.

WAI Protocol and Formats Working Group

Review of accessibility perspectives for Web of Things technologies

Data on the Web Best Practices Working Group

Focuses on developing the open data ecosystem, guidance to publishers and fostering trust in data.

Other relevant Working Groups

The Web of Things Interest Group will review a broad range of W3C Working Groups for APIs that are applicable to applications and services for the Web of Things.

W3C Community and Business Groups

The Web of Things Interest Group will watch for and coordinate with Community Groups whose work are likely to have an impact on the the Web of Things.

5.5. EXTERNAL GROUPS

CEA

The Consumer Electronics Association (CEA) is a standards and trade organization for the consumer electronics industry in the United States, and as such will be important for integrating the Web of Things with consumer electronics devices.

GS1

GS1 produces global standards and solutions to improve the efficiency and visibility of supply and demand chains globally and across sectors. This is especially relevant to the integration of the Web of Things in the retail and manufacturing sectors.

IIC

The Industrial Internet Consortium (IIC) seeks to further development, adoption and wide-spread use of interconnected machines, intelligent analytics and people at work. This is complementary to W3C's aims to create standards that enable open markets of services for the Web of Things.

Industry 4.0

Industry 4.0 is a long term project supported by the German government to realize the potential for smart manufacturing with richer integration along the manufacturing value chain, faster time to market, and an increased focus on bespoke products tailored to each customer's need. The Web of Things Interest Group should coordinate with Industry 4.0 in respect to use cases and requirements for manufacturing.

Bluetooth SIG

The Web of Things Interest Group will liaise with the Bluetooth Special Interest Group to understand requirements for services based upon Bluetooth connections.

OASIS

OASIS is a standards development organization, and is working on standards such as MQTT that offers an efficient messaging protocol for the Web of Things.

OMG

OMG is a standards development organization, and is working on standards such as the Data Distribution Service (DDS) that offers an efficient messaging protocol for the Web of Things.

ETSI

The European Telecommunications Standards Institute (ETSI) produces globally-applicable standards for Information and Communications Technologies, which are likely to be an important enabler for the Web of Things.

ISO/EIC JTC 1/WG 7

This is a standardization working group that focuses on sensor networks, and coordination would help build a shared understanding, harmonization of terminology and interoperability with the Web of Things.

GSMA

The GSM Association (GSMA) represents mobile operators and develops proposals that can bring important value to the usage of the Web of Things on mobile networks.

IETF

The Internet Engineering Task Force develops the protocols that clients use to connect to the Web; ensuring these protocols match the needs of the Web of Things is an important part of achieving the group's mission. One example would be JavaScript APIs for accessing resourced constrained devices via CoAP.

6. CONCLUSIONS

The amount of hyper around the Internet of Things has been growing rapidly (see e.g. Gartner's hype cycles) with news articles abounding on wearables, healthcare, smart cities and the future of manufacturing. The deployment of applications and services is lagging behind, and there is general acceptance of the risks of product silos and the importance of standards as a basis for enabling open markets with interoperability across vendors, and a vigorous ecosystem of developers.

Most of the standardization activities relating to the Internet of Things has so far focused on the devices and the technologies needed to communicate with them. There is a growing urgency for creating standards as a basis for enabling open markets of applications and services. The W3C Berlin workshop, arranged with the support of the COMPOSE project, brought together a wide range of stakeholders, resulting in a strong consensus that W3C should follow through on the vision of the Web of Things.

A draft charter has now been prepared for a W3C Web of Things Interest Group following the precedents of the W3C Web & Mobile and Web & TV Interest Groups. The Web of Things Interest Group would start by gathering use cases and requirements across a broad range of sectors, and proceed to develop best practices and specific suggestions for work items for standardization by W3C Working Groups. These are expected to include support for discovery, rich descriptions of services, including the interfaces they expose and are dependent on, scripting APIs, privacy and end to end security, and data models for industry specific application domains.

My thanks to my colleagues for their help in preparing this report.

7. INDUSTRY INPUT FOR THE WEB OF THINGS INTEREST GROUP CHARTER

The German government has launched the Industrie 4.0 initiative on next generation manufacturing which heavily features the Internet of Things. The following is consolidated input from members of the Industrie 4.0 Standards Task Force as of 20 October 2014.

Companies involved:

- N+P Informationssysteme GmbH
- Deutsche Telekom AG/T-Systems
- Fortiss
- Siemens
- Emeres GmbH
- SAP AG
- BITKOM e.V.

7.1. N+P INFORMATIONSSYSTEME GMBH

- The safety of communication between individual users and / or devices must be ensured
- There must be a clear distinction between Public and Private
- A prioritization in communication (eg defined delivery times of information) is important
- The protection against hacker attacks must receive special attention and appropriate countermeasures must be developed.

7.2. DEUTSCHE TELEKOM AG/T-SYSTEMS

1. Scope: Web of Things = Internet of Things + rich content
    - Object Identifier and Discovery / Authentication / Authorization
    - Find communication end points, How will Authentication, Authorization and Trust-Building look like in the Web of Things? User name / Password / biometrics are not possible in the Web of Things yet.
    - Privacy: Things can generate a lot of data, and their combination allows third parties to draw conclusions about people. How do we deal with this problem?
    - Interoperability
    - All solutions to be standardized should also match those of others, e.g. oneM2M, IEEE 2413 IoT WG. Therefore, in the Kantara Initiative, we have established a discussion group that will address the theme "Identity of Things" across parties.
    - Web technologies for co-modeling of real and virtual world
    - Augmented Reality features, z.B. Teil einer Produktionsanlage + virtuelles Modell
    - WoT service platforms, smart services
    - Standards interoperability

2. Differentiation to IoT, M2M
3. Deliverables
    - Collection of use cases and best practices
    - Including APIs for backend integration and re-planning of applications in the production environment
    - Web technologies for WoT object identification and discovery / authentication / authorization
    - Web technologies for co-modeling of real and virtual world
    - Definition of semantics: IT and production technology
    - Recommendation for WoT service platforms
    - Standards interoperability analysis / cookbook
    - Realtime communication requirements

Attachment: Excerpt from a whitepaper of the Industrie4.0 – platform (Integration of real and virtual world)

7.2.1. What do we mean by this?

The interaction of real and virtual / digital world is moving more and more into focus in the Industry 4.0. All objects have a digital image, the model. The real world in this context usually is characterized by problems to be solved and decision-making processes. Key elements of the virtual / digital world are simulations, planning and descriptive models. The Co-modeling considers, moreover, the interfaces between the two worlds at different scales.

What is needed is a scientific foundation in terms of production engineering modeling theory for mechanical engineering and plant construction. Proven theories, description means and methods including related basic technologies from computer science are to get in trim by suitable adaptation, extension and combination in terms of widespread use in the engineering sciences. Here, the addressee correct integration into known domain-specific working approaches and software tools plays a key role.

Based on a defined common semantic, concepts must be integrated consistently. In addition to the costs for the creation of models, their useful application throughout the entire life cycle must be considered.

7.2.2. What do we want to achieve?

The necessary basis is a common understanding of models in mechanical engineering, electrical engineering and computer science in the field of production. The long term goal is to enable manufacturing companies to create economical, useful and bidirectional models. This is how elements of virtual worlds can be linked interdisciplinary with the real world at a high semantic level in order to increase the efficiency of internal order processing and the security of decisions significantly.

We expect the following results:

- Modeling theory including any derivative work, requirements for tools and data or information flows (at all levels of the automation pyramid)

- Method for the proof of efficiency as well as case studies
- practicable modeling rules
- general tool-assisted meta-model

7.3. FORTISS

- Safety, especially for cyber-physical systems that perform control tasks in the real world
- Monetization and value chain models
- Tool chains (for development, operation and testing)
- Scalability

7.4. SIEMENS

The main point is, in our opinion, to avoid too early fragmentation into separate interest groups, but as the first goal to gather a large collection of use cases from all application domains and break them down to elementary use cases. These elementary, domain-independent use cases (such as finding and describing "Things") ultimately lead to clear, reasoned requirements in the collection and analysis of technologies and a cross-domain understanding.

7.4.1. Further points are:

- Another deliverable that seems to be crucial for success are guidelines or a best practices document identifying relevant technologies and approaches to guarantee best interoperability across all use cases.
- For the mission statement, we would also like to point to semantic WoT: e.g. "Facing challenges in WoT such as identification and discovery (e.g., of services and devices), the device plug & play functionality, and access to data in a growing heterogeneous service and device landscape, semantics will be a key enabler to tackle these challenges."
- Could/Should the mission statement cover also the seamless adaption and enablement of web technologies and principles on embedded/ constrained devices similar as e.g. EXI enables using XML
- Additionally to the gateway scenarios I would like to have mentioned the scenario of using web technologies down to the field resp. on the device itself (which motivates he point above)
- To the Liaisons, we would like to suggest addition these groups:
    - Add EXI working group to the existing W3C group list given that it is already identified in the charter prose as one possible encoding format
    - W3C RDF Stream Processing Community Group: The well known approaches such as the W3C RDF model shall be also applicable to constrained devices to meet interoperability and to operate on common ontologies. There is a need of a standardized efficient RDF serialization.
    - As external group, the XMPP Standards Foundation (XSF) @ xmpp.org: In terms of adding XMPP to the transport protocols alongside MQTT and CoAP

- The term “open market of services” should maybe be defined or referenced – as its relation and/or differentiation to terms like WoT or web platform is not very clearly stated.

7.5. EMERES GMBH

The critical point is the transition from "IoT" to "WoT". For this transition, strong and precise typing is needed.

In the IoT world, we, (at Emeres) define objects and their types. In order to connect objects or link objects to types, we define references. The references may also be typed in order to gain different semantics. With these tools we model industrial equipment.

For the transition from IoT to WoT, we link Object Types in the IoT world to previously defined Types in the WoT world. These are just HTML or SVG-Fragments.

Thus in order to map the IoT to WoT, it is helpful if HTML, SVG and Javascript DOM

stick to the XML standard as closely as possible.

Some examples from deviations from this are:

- Closing tags are not allowed (input tag, SVG Filter and gradients)
- setAttribute and getAttribute work in SVG very well, in HTML5 this is a problem (e.g. "input.value").
- There should be a way for input in SVG namespace for all browsers: foreignObjects should be supported in all browsers or the two input elements (“text” and “select”), that are really necessary, should be implemented natively by SVG.

7.6. SAP AG

Semantic/ data model

- There's a need a for a domain specific language (DSL) or a framework to define domain specific languages per industry or application area.
- Build a ‘Enterprise data model’ that defines attributes of relevant things in the world of sensors, actuators etc,
- The need of a standardized semantic layer would be essential for industry 4.0 applications. A description of object types (e.g. what is a drilling machine, what can it do, what is the relation to other objects) and a semantic of available tags (sensors and their values) are important to build self-coordinating and self-organizing systems
- Build a standard ontology to describe ‘things’ like initiative before (DAML + OIL / OWL) for the Internet of Things context or a high-rack storage is a storage and a storage is a location, etc

Communication/ Protocols

- As bandwidth may be limited at the edge, the data format should compact, i.e. not XML, even JSON may be too verbose.
- Is Push-communication (the big web problem) the way to go, or do we need other technologies to connect all devices (IoT endpoints).

Interoperability and Architecture

- Mass Data-Capturing in Real-Time during Process by
- Embedded Controller
- Big Data Client
- Interfacing Embedded Systems with Cloud-Platforms

Software deployment and life-cycle

- SW-Upgradability of Embedded Systems
- Upgradability Embedded System
- Upgradability by Cloud-Platform

Security

- Security will be BIG. How should authentication (e.g. X.59 certificates) work? Which communication channels need to be secured with all the usual caveats.
- Not all IoT endpoints are likely to run IP (for power reasons), so there will be a need seamless integration between IP and other transport protocols.
- Cyber Security Embedded Control Systems

I personally think the W3C should soon have a clear mission and vision statement for the companies which standards they want to drive and what will be their role in the IoT topic. The standard organization has to show that they’re faster than the de-facto standard.

Furthermore, I would expect a clear positioning and focus of the different SDOs (IIC, IEEE, etc.).

7.7. BITKOM E.V.

As we have in image in real time from the real world, we have the possibility to perform a higher-level optimization in real-time. This higher-level digital optimization is not only the planning of, for example, production processes, but also to resolve conflictual situations in real time.

But we have no form of description, that provide content and timing as a real-time image consistent interpretable by anyone in order to control the company through factories and across companies.

What distinguishes Industrie 4.0? What sets the requirements for the Internet? It is the real-time image of the real world to the virtual world. This is the crucial space for concept proposals and standardization in the W3C "Web of Things".

There is no new description language sought that comprehensively describes the whole world, but it lacks the modern languages that describe the content of the real world, only by time reference. This is to be added. Furthermore, it's to describe what we want to do with the time information. This includes the way in which we introduce time as possible as an extension of an existing language.