

WoT Virtual F2F Architecture + Profiles

Michael Lagally 26 Oct 2021

Architecture + Profiles



Agenda:

- Architecture Task Force
 - People and Deliverables
 - Current work
 - Next steps
 - Publication schedule

Profiles

- Introduction to WoT profiles
- Use Cases and Core Requirements
- Current work
- Open issues
- Proposal for a "Push" event model
- Next steps
- Publication schedule



WoT Architecture Task Force People

The WoT Architecture task force is responsible for the abstract architecture and interoperability profiles for the Web of Things.

People:

WoT Architecture Co-Editors:

Michael Lagally (Oracle Corp.), Ryuichi Matsukura (Fujitsu Ltd.), Toru Kawaguchi (Panasonic Corp.), and Kunihiko Toumura (Hitachi, Ltd.)

WoT Profile Co-Editors:

Michael Lagally (Oracle Corp.), Michael McCool (Intel Corp.), Ryuichi Matsukura (Fujitsu Ltd.), Sebastian Kaebisch (Siemens AG), and Tomoaki Mizushima (Internet Research Institute, Inc.)



WoT Architecture Task Force Deliverables

WoT Architecture (W3C Recommendation)

- The WoT Architecture specification describes the abstract architecture for the W3C Web of Things. It defines terminology that is used by all other WoT building blocks
- Conceptual framework that can be mapped onto a variety of concrete deployment scenarios.

WoT Profile (W3C Recommendation)

- Profiling Mechanism and a WoT Core Profile which enables out of the box interoperability among things and devices.
- Out of the box interoperability implies that devices can be integrated into various application scenarios without deep level adaptations. Integration can be done by anyone without specific training.

WoT Architecture 1.1



27/10/2021, 11:57	Web of Things (WoT) Architecture 1.1	27/10/2021, 11:57	Web of Things (WoT) Architecture 1.1	27/10/2021, 11:57	Web of Things (WoT) Architecture 1.1
		Table of Contents		8.7.3 8.7.4	Standard Set of Methods Media Types
	Web of Things (WoT) Architecture 1.1	Abstract		8.7.4	WoT System Components and their Interconnectivity
	Web of Things (WoT) Architecture 1.1 W3C Editor's Draft 27 October 2021	America		88.1	Direct Communication
	W.S. Editor & Draft 27 October 2021	Status of TI	is Document	8.8.2	Indirect Communication
	▼ More details about this document	1. Introduction		9.	WoT Building Blocks
	This version:	2. Conforman		9.1	WoT Thing Description
	https://w3c.github.io/wos-architecture/	2. Conforman	•	9.2	Thing Model
	Latest published version:	3. Terminolog		9.3 93.1	Profiles Profiling Methodology
	https://www.w3.org/TR/wot-architecture11/			9.3.1	WoT Discovery
	Latest editor's draft:	4. Device Cate	ories	94.1	Introduction Mechanisms
	https://w/3c.github.io/wot-architecture/ History:	5. Application	Domains (Verticals)	94.2	Exploration Mechanisms
	Publication history	5.1 Consumer		9.5	WoT Binding Templates
	Commit history	5.2 Industrial		9.6	WoT Scripting API
	Editors:		Smart Factory	9.7	WoT Security and Privacy Guidelines
	Michael Lagally (Oncie Corp.)	5.3 Transportation 5.4 Utilities	n & Logistics	10.	Abstract Servient Architecture
	Ryuichi Matsukura (<u>Fujitsu Lad.</u>)	5.4 Utilities 5.5 Oil and Gas		10.1	Behavior Implementation
	Toru Kawaguchi (Panasonic Corp.)	5.6 Insurance		10.2	WoT Runtime
	Kunihiko Toumura (Hinchi, Ltd.)		and Construction	10.3	WoT Scripting API
	Kazuo Kajimoto (Former Editor, when at Panasonic) Foodback:	5.8 Agriculture		10.4	Exposed Thing and Consumed Thing Abstractions
	Feedback: GitHub-w2c/wot-architecture (pull requests, new issue, open issues)	5.9 Healthcare		10.5	Private Security Data
	public-wor-westwal.org with subject line (wot-architecture11) _ message topic _(archives)	5.10 Environmen	Monitoring	10.6	Protocol Stack Implementation System API
	Contributors	5.11 Smart Cities 5.12 Smart Build		10.7	System API Alternative Servient and WoT Implementations
	In the GitHub repository	5.12 Smart Build 5.13 Connected C		10.8.1	Native WoT API
	Coparight © 2017-2021 Wile (MIT, ERCIM, Keir, Beilmag). WSC liability, trademark and permissive document license rules apply.		ed Car Example	10.8.2	Thing Description for Existing Devices
				11.	Example WoT Deployments
		6. System Top 6.1 Device Cont	logies (Horizontals)	11.1	Example WoT Deployments Thing and Consumer Roles
	Abstract	6.1 Device Cont 6.2 Thing-to-Th		11.2	Topology of WoT Systems and Deployment Scenarios
		63 Remote Acc		11.2.1	Consumer and Thing on the Same Network
	The W3C Web of Things (WoT) is intended to enable interoperability across IoT platforms and application domains.	6.4 Smart Home	Gateways	11.2.2	Consumer and Thing Connected via Intermediaries
	Overall, the goal of the WoT is to preserve and complement existing IoT standards and solutions. In general, the W3C WoT architecture is designed to describe what exists rather than to prescribe what to implement.	6.5 Edge Device		11.22.1	Intermediary Acting as a Proxy Intermediary Acting as a Digital Twin
		6.6 Digital Twin		11 2 3	Intermediary Acting as a Digital Iwan Devices in a Local Network Controlled from a Cloud Service
	This WoT Architecture specification describes the abstract architecture for the W3C Web of Things. This abstract	6.6.1 Cloud- 6.6.2 Legacy	ady Devices	1124	Discovery Using a Thing Description Directory
	architecture is based on a set of requirements that were derived from use cases for multiple application domains as described in [WOT-USE-CASES-REQUIREMENTS]. A set of modular building blocks were identified whose detailed specifications	6.0.2 Legacy 6.7 Multi-Cloud	ACTACS .	11.2.5	Service-to-Service Connections Across Multiple Domains
	in [WOT-USE-CASES-REQUIREMENTS]. A set of modular building blocks were identified whose defailed specifications are given in other documents. This document describes how these building blocks are related and work together. The WoT	6.8 Cross-domai	Collaboration	11.2.5.1	Connection Through Thing Description Directory Synchronization
	abstract architecture defines a basic conceptual framework that can be mapped onto a variety of concrete deployment			11.2.5.2	Connection Through Proxy Synchronization
	scenarios, several examples of which are given. However, the abstract architecture described in this specification does not	7. System Inte	ration	12.	Security and Privacy Considerations
	itself define concrete mechanisms or prescribe any concrete implementation.	8. Abstract W	T System Architecture	12.1	WoT Thing Description Risks
		8.1 System Com		12.1.1	Thing Description Private Security Data Risk
	Status of This Document		nd Consumers	12.1.2	Thing Description Personally Identifiable Information Risk Thing Description Communication Metadata Risk
	Status of This Document	8.1.2 Metada		12.1.3	Thing Description Communication Metadata Risk WoT Scripting API Security and Privacy Risks
	This section describes the status of this document at the time of its publication. Other documents may supersede this		ng Descriptions or Medels	12.2	Cross-Script Security and Privacy Risk
	document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C	8.13 Links	No. of the contract of the con	12.2.2	Physical Device Direct Access Security and Privacy Risk
	technical reports index at https://www.w3.org/TR/.	8.1.4 Interme	iories	12.3	WoT Runtime Security and Privacy Risks
	This document describes an abstract architecture design. However, there is an <u>Implementation Report</u> that describes a set of	8.2 Affordances		12.3.1	Provisioning and Update Security Risk
	concrete implementations based on the associated WoTThing Description specification. These are implementations	8.3 Web Thing		12.3.2	Security Credentials Storage Security and Privacy Risk
	following the W3C Web of Things architecture.	8.4 Lifecycle 8.4.1 System	Stanoola	A.	Recent Specification Changes
	This document was published by the Web of Things Working Group as an Editor's Draft.		atecycle ple System Lifecycle		
	GitHub Issues are preferred for discussion of this specification. Alternatively, you can send comments to our mailing list.		em Lifecycle with Registration	В.	Acknowledgments
	<u>Cutturb issues</u> are preferred for discussion of this specification. Atternatively, you can send comments to our maining list. Please send them to public-wort-wg@tw3.org (subscribe, archives).	8.4.2 Thing L		c.	References
		8.4.3 Informa	ion Lifecycle	C.1	Normative references
	Publication as an Editor's Draft does not imply endorsement by the W3C Membership.	8.5 Interaction b 8.5.1 Properti		C.2	Informative references
	This is a draft document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to	8.5.1 Properts 8.5.2 Actions		1	
	cite this document as other than work in progress.	8.5.3 Events		5 1 Int	roduction
	This document was produced by a group operating under the 1 August 2017 W3C Patent Policy. W3C maintains a gublic	8.6 Hypermedia	Controls		
	list of any patent disclosures made in connection with the deliverables of the group; that page also includes instructions for	8.6.1 Links			Is of the Web of Things (WoT) are to improve the interoperability and usability of the Internet of Things (IoT).
	disclosing a patent. An individual who has actual knowledge of a patent which the individual believes contains Essential	8.6.2 Forms			a collaboration involving many stakeholders over many years, several building blocks have been identified that
	Claim(s) must disclose the information in accordance with section 6 of the W3C Patent Policy.	8.7 Protocol Bin 8.7.1 Hypern	lings dia-driven	help ad	fress these challenges.
	This document is governed by the 15 September 2020 W3C Process Document.	8.7.2 URIs			
			and the second s	https://w3c.github.io/wot-architecture	/#sec-interaction-model 3/45
https://w3c.github.io/wot-ar	rchitecture/#sec-interaction-model 1/45	https://w3c.github.io/wot-architecture/#sec-interaction	s-model 2/45		

2021-10-27 W3C Web of Things (WoT) WG/IG 5

WoT Profile



27/10/2021, 11:58	Web of Things (WoT) Profile	27/10/2021, 11:58	Web of Things (WoT) Profile	27/10/2021, 11:58 Web of Things (WoT) Profile				
				5.15.2 Additional Constraints				
			EDITOR'S NOTE	5.1.5.3 Recommended Practice				
	Web of Things (WoT) Profile		The name WoT Core Profile is still under discussion in the group and is used as a working title. It is subject to change	5.1.6 Event Affordance				
			after the profile specification has reached a certain level of maturity.	5.1.6.1 Mandatory fields				
	W3C Editor's Draft 23 September 2021		•	5.1.6.2 Additional Constraints				
				5.1.7 Forms				
	▼ More details about this document		Motivation for a Profile	5.1.7.1 Mandatory fields 5.1.7.2 Additional Constraints				
	This version:		The W3C WoT Thing Architecture [wot-architecture] and WoT Thing Description [wot-thing-description] define a powerful	5.1.2 AGENTALIS CONTRIBES 5.1.8 Links				
	https://w3c.github.io/wot-profile/		the W.M. Wo I Imag Arentecture [wot-arentecture] and Wo I Imag Description [wot-mang-description] define a powerful description mechanism and a format to describe myriads of very different devices, which may be connected over various	5.1.9 Security				
	Latest published version:		protocols. The format is very flexible and open and puts very few normative requirements on devices that implement it.	5.1.9.1 Recommended Practice				
	https://www.w3.org/TR/wot-profile/			5.2 Protocol Binding				
	Latest editor's draft:		However, this flexibility de-facto prevents interoperability, since, without additional rules, it allows implementers to make	5.2.1 Properties				
	https://w3e.github.io/wot-profile/		many choices that do not provide guarantees of common behavior between implementations.	5211 readproperty				
	History:			5212 writeproperty 5213 readalloroserties				
	<u>Publication history</u>		Status of This Document	5213 readaltproperties 5214 writemaltpleproperties				
	Editors:		Status of This Document	3.21A WITCHULLIPUPPOPPETERS 5.22 Artions				
	Michael Lagally (<u>Oracle Corp.</u>) Michael McCool (Intel Corp.)		This section describes the status of this document at the time of its publication. Other documents may supersede this	5.221 invokeaction				
	Nichael McCool (<u>intel Corp.</u>) Ryuichi Matsukura (<u>Puitsu Ltd.</u>)		document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C	5.22.2 queryaction				
	Sebastian Knebisch (Siemens AG)		technical reports index at https://www.w3.arg/TR/.	s223 cancelaction				
	Tomoski Mizushima (Internet Research Institute, Inc.)		This document was published by the Web of Things Working Group as an Editor's Draft.	5224 queryallactions				
	Feedback:			5.2.3 Events				
	public-wot-wg@w3.org with subject line [wot-profile] _ message topic _(archives)		Comments regarding this document are welcome. Please send them to gublic-worw.wg@w3.org (subscribe, archives).	52XI subscriberent				
1	Contributors		Publication as an Editor's Draft does not imply endorsement by the W3C Membership.	5232 unsubscriberent 5233 subscribealleuerts				
	In the GitHub repository			5233 SUBSCTIBULITY THE TOTAL STATE OF THE ST				
	Repository		This is a draft document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to	5.2.4 Error Responses				
1	We are on GitHub		cite this document as other than work in progress.	5.3 External TD representations				
	File a bug		This document was produced by a group operating under the 1 August 2017 W3C Patent Policy, W3C maintains a public	5.3.1 Canonical TD representation				
	Contribute		list of any patent disclosures made in connection with the deliverables of the group; that page also includes instructions for	5.4 Open Issues				
	Copyright © 2020-2021 W3c [®] (MIT. ERCIM, Keio, Beitung). W3C liability, trademark and permissive document license rules apply.		disclosing a patent. An individual who has actual knowledge of a patent which the individual believes contains Essential	5.5 JSON Schema of the Core Profile				
1			Claim(s) must disclose the information in accordance with section 6 of the W3C Patent Policy.	A. References				
1			This document is governed by the 15 September 2020 W3C Process Document.	A. References A.1 Normative meterances				
1	Abstract			A.1 Informative references A.2 Informative references				
				·····				
	The WoT Profile Specification defines a Profiling Mechanism and a WoT Core Profile, which enables out of the box		Table of Contents					
	interoperability among things and devices. Out of the box interoperability implies, that devices can be integrated into			§ 1. Introduction				
various application scenarios without deep level adaptations. Typically only minor configuration operations are necessary			Abstract	The W.R. WoT Architecture [wot-architecture] and the WoT Thing Description [wot-thing-description] have been developed as a versatile format, that allows describing the interactions between multiple devices and protocols.				
	(such as centuring a servowle key, or IP address) were the elevies as certain scenario. These actions can be due by anyone without specific craming. The WITCOM: Profile defines a set of constraints and rules, which compliant thing descriptions have to adopt to guarantee interoperability.		Motivation for a Profile					
			Status of This Document	developed as a versame format, mai anows describing the interactions between managed developes and protocols.				
				This flexibility permits an easy integration of new device types and protocols, however it risks interoperability, since there are no guarantees that two devices which are formally spec-compliant, will be able to communicate.				
			1. Introduction					
	These rules are prescriptive, to ensure that compliant implementations satisfy the semantic guarantees implied by them. We		1.1 Deployment Scenarios	To increase adoption of the WoT specifications, interoperability between on premise devices, edge devices and the cloud is				
	call this set of rules a Profile.		1.2 Why a Core Profile? 1.3 Out-of-the-box interoperability	essential. Even if every manufacturer is implementing the current Thing Description specification in full flexibility, there is				
	To W.T Built, Co. 18 and a state of a fair of a fair		1.3 Out-ot-tne-tox meroperatuity 1.4 Structure of this document	no interoperability guarantee; many choices are still left to the implementations and there are very few normative				
	The WoT Profile Specification as defined in this document serves two purposes:		· · · · · · · · · · · · · · · · · · ·	requirements that a device has to fulfill.				
	 It defines a generic Profiling Mechanism which provides a mechanism to describe a profile in an unambiguous way. 		2. Conformance					
	This mechanism can be used to define additional profiles.			The same of the sa				
	 In addition, it defines a WoT Core Profile of the Thing Description, which consists of a core data model and protocol 		3. Terminology	i 1.1 Deployment Scenarios				
	binding rules. The WoT Core Profile formalizes the results of several PlugFests that were conducted by the WoT		4. Profiling Mechanism	A Thing Description can be used in two fundamentally different deployment scenarios:				
	Interest Group and of tests that were conducted as part of the development.		-					
	This document incudes a binding of the core data model to HTTP(S) and selected notification sub-protocols. The core		5. WoT Core Profile	 a "brown-field" scenario, where it is created to describe the interactions with existing systems. 				
	data model can be bound to other protocols - it is expected that bindings to other protocols (e.g. MQTT, CoAP) will be		5.1 WoT Core Data Model	 a "green-field" scenario, where a device model and a thing description are developed together. 				
	defined in the near future.		5.1.1 General S.1.1 Mandatory fields	For green field deployments, where the implementations are being carried out and corresponding thing descriptions are				
	A Marie Control of the Control of th		5.1.1.1 Mandatory fields 5.1.1.2 Date format	For given next deployments, where the implementations are being carriero out corresponding timing descriptions are being created, it is causer to achieve full interoperatibility by using a small, extensible Core Profile.				
	A TD that is compliant to the <u>core profile</u> MUST adhere to both the constraints on the data model and the protocol binding.		5.1.2 Length and Value Limits					
			5.1.2 Thing	In the brown field area, due to the nature of existing deployments and protocols, a broad spectrum of variations and				
	Devices that constrain their use of the Thing Description to the WoT Core Profile can interoperate with each other out-of-		5.12.1 Mandatory fields	potentially high complexity of thing descriptions inhibits interoperability and will most likely lead to additional profiles of				
	the-box.		5.12.2 Recommended practice	the WoT Thing Description and domain-specific thing consumer implementations.				
	Note that the core profile is not exclusive. Device implementers are free to adopt other features of the thing description that		5.1.3 Data Schema	The WoT Core Profile can be used by green field deployments and gives guidance to new implementers of the WoT				
	go beyond the constraints of the core profile, however the interoperability guarantees of the core profile hold only for the		5.1.4 Property Affordance 5.1.4 Mandatory fields	specifications. It has already proved in brown-field scenarios in the PlugFests, where existing devices, that already existed as				
	WoT Core Profile subset.		5.1.4.2 Additional Constraints	products, prototypes or demonstrators, were described with Thing Descriptions that are constrained to the Core Profile.				
			5.1.4.3 Recommended Practice					
			5.1.5 Action Affordances	§ 1.2 Why a Core Profile?				
			5.1.5.1 Mandatory fields	s 1.2 wny a Core Frome:				
i								
1		1/22 https://w3c.github.io/	vot-profile/ 2/22	https://w3c.github.jo/wot-profile/ 3/22				
https://w3c.github.io/wot-	prome/	1/22 https://wsc.github.io/	ou-promer 2/22	тірьл мәсізіншәле мосі-рісіне				
		1		1				



WoT Architecture



Current work: Terminology Alignment

The WoT Architecture REC defines the common terminology for all WoT specifications.

Other WoT specifications should contain a reference this terminology.

Some specifications have their own terminology section, need to align and move common definitions to architecture.

https://github.com/w3c/wot-architecture/labels/spec%20alignment

```
# 614 - Additional terminology entries for binding spec
```

613 - Move terminology section from discovery to architecture spec

612 - Move terminology from TD to architecture spec

611 - Update Terminology regarding Binding Templates

To avoid confusion and potential ambiguities, the terminology section should be normative.



Current work: Protocol bindings

Recent restructuring of the WoT Binding Templates requires updates of the corresponding introduction section to this guideline.

Restructuring of the binding document goes on, some synchronisation is needed.

Initial discussions have started in the WoT editors call, will be continued in the Architecture call.

Next Steps



- Implementation Report
- Explainer Document

Help and contributions are very welcome.





Tentative, based on the planned 6 months charter extension.

Architecture Feature Freeze: Jan 31, 2022

CR transition: mid-March

PR transition: mid-April

REC transition: before end of extended charter end of July



WoT Profiles

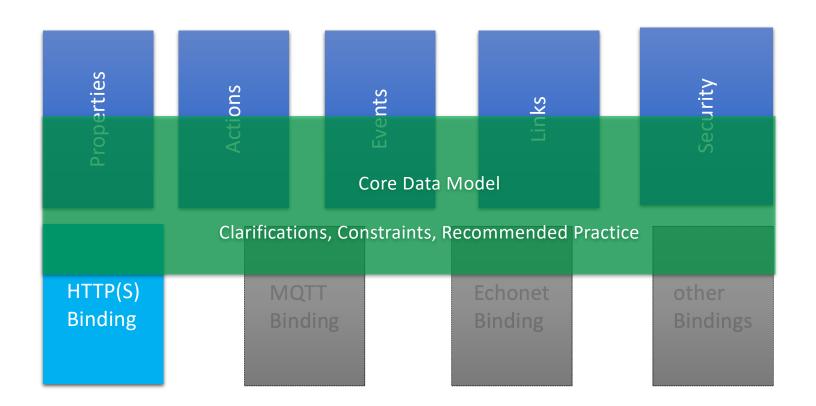




- A **WoT Profile** is a <u>normative subset</u> of a *WoT Thing Description* with a <u>normative binding</u> to a selected protocol.
- Profiles guarantee **interoperability** between compliant implementations, multiple profiles are possible.
- The WoT Profile Specification defines a normative set of constraints and rules on the data model, representation format and protocol binding.
- These constraints and rules set limitations and make decisions that reduce the complexity for implementers of the WoT standard.
- The rules are prescriptive, to ensure that compliant implementations satisfy the semantic guarantees implied by them.







Use Cases for Profiles



Agriculture	Smart Cities	Building Technologies	Manufacturing	Retail	Retail	Health	Energy	Transportation	Automotive	Smart Home	Education		
Security										Require-	Liaisons		
Discovery									ments	LIGISOTIS			
Multi-Vendor System Integration													
Out of the box interoperability													
Digital Twins													
Cross Protocol Interworking													
Multimodal System Integration													
	Accessibility												

2021-10-26



Use Case Cross protocol interworking

In smart city, home and industrial scenarios various devices are connected to a common network. These devices implement different protocols.

To enable interoperability, an "agent" needs to communicate across different protocols. Platforms for this agent can be edge devices, gateways or cloud services.

Interoperability across protocols is a must for all user scenarios that integrate devices from more than one protocol.

Core requirements: Common data model across protocols,



Use Case Out of the box interoperability

As a device owner, I want to know whether a device will work with my system before I purchase it to avoid wasting money.

As a developer, I want to be able to validate that a Thing will be compatible with a Consumer without having to test against every possible consumer.

As a cloud provider I want to onboard, manage and communicate with as many devices as possible out of the box. This should be possible without device specific customization.





Data Model:

- common metadata
- common unit/metrics system
- Common time formats
- Constrained data model across protocols
- Naming conventions

Protocol binding:

- Unambiguous protocol binding
- Clarifications and constraints
- Fully specified interaction semantics for properties, actions and events
- Fully specified error behaviour

Current work



- Defining a core/baseline profile with a HTTP binding.
- Identifying constraints and rules on the data model.
- Unambiguous interaction semantics for properties, actions and events.
- Constraints on payload formats.
- Protocol binding semantics, e.g. headers, response codes.
- Best practice security requirements.
- Compliance TD Validation

Open issues



Core data model section

- Ongoing discussion in several PRs
- Current constraints need to be reworked
- One TF member proposed removal of the entire section, this would break cross-protocol interoperability use cases and impact OOTB interoperability

Event model limitations

- Current model is a "Consumer Pull model" using SSE
- Requires an open socket connection for each subscription
- Does not support firewalls
- Does not scale to cloud scenarios
- Works only in closed networks



Proposal for a "Push" event model (1)

In typical cloud scenarios devices are not reachable from the internet

- Devices send messages and events via firewalls
- Only on-demand network traffic, no open connection if no messages

Pull model wastes significant resources

- open connections to thousands of devices even if no traffic
- Network traffic overhead due to timeouts and reconnects

Webhooks over HTTP are industry standard for scalable event systems, see for example:

- Github: https://docs.github.com/en/developers/webhooks-and-events
- Paypal: https://developer.paypal.com/docs/api-basics/notifications/webhooks/
- Atlassian: https://developer.atlassian.com/server/jira/platform/webhooks/
- Wordpress: https://wordpress.com/support/webhooks/



Proposal for a "Push" event model (2)

In typical cloud scenarios devices are not reachable from the internet

- Devices send messages and events via firewalls
- Only on-demand network traffic, no open connection if no messages

Pull model wastes significant resources

- open connections to thousands of devices even if no traffic
- Network traffic overhead due to timeouts and reconnects

Webhooks over HTTP(S) are commonly used for scalable event systems, see for example:

- Github: https://docs.github.com/en/developers/webhooks-and-events
- Paypal: https://developer.paypal.com/docs/api-basics/notifications/webhooks/
- Atlassian: https://developer.atlassian.com/server/jira/platform/webhooks/
- Wordpress: https://wordpress.com/support/webhooks/

Push event model



Follow Industry best practice:

- Use Webhooks over HTTP(S) for push events
- Use cloud events as message payload format (https://github.com/cloudevents/spec/blob/v1.0.1/spec.md)
- Provide additional clarifications:
 - Retry and reconnect behaviour
 - Error behavior
 - Standard attributes

What are Cloud Events?



A specification for describing event data in a common way.

Spec has been developed by several companies, including Alibaba, Amazon, Google, Huawei, IBM, Intel, Microsoft, Oracle, PayPal, Red Hat, SAP, Serverless Inc, SolarWinds, VMWare.

It defines a type system, context attributes and a payload format.

Open source implementations for various languages, including Java, go, C#, javascript, python are available.

https://cloudevents.io



Cloud Event example (JSON)

```
"specversion": "1.0",
"type" : "com.github.pull_request.opened",
"source": "https://github.com/cloudevents/spec/pull",
"subject" : "123",
"id": "A234-1234-1234",
"time": "2018-04-05T17:31:00Z",
"comexampleextension1" : "value",
"comexampleothervalue" : 5,
"datacontenttype" : "text/xml",
"data" : "<much wow=\"xml\"/>"
}
```

Next steps



- "Push" event mechanism
- Rework data model section
- Implementation Report
- Compliance Section JSON Schema for profile
- Authentication, Security
- Incorporate Plug Fest findings



Future Work – next charter period

Support for other protocol bindings Candidates include:

- MQTT
- Echonet Lite
- OPC-UA

Include TD 1.1 features

• E.g. Thing Models

Include TD 2.0 features





Tentative, based on the planned 6 months charter extension.

Profile Feature Freeze: Jan 31, 2022

CR transition: mid-March

PR transition: mid-April

REC transition: before end of extended charter end of July

References



WoT Architecture task force

https://www.w3.org/WoT/activities/tf-architecture/

WoT Architecture repository / working draft

https://github.com/w3c/wot-architecture

WoT Profile repository / working draft

- https://github.com/w3c/wot-profile
- Wot Use Cases editors draft:
- http://w3c.github.io/wot-usecases



Backup



PRs

PRs



RFC 2119 markup: add css mark into index.html

https://github.com/w3c/wot-profile/pull/79

Discussion:

We noticed that some span markups are in editors notes – these need to be cleaned up in a further iteration.

We merged the PR, Mizushima-san will provide additional comments.

McCool will do another pass using the test tooling.

References



WoT Use Cases repository

• https://github.com/w3c/wot-usecases