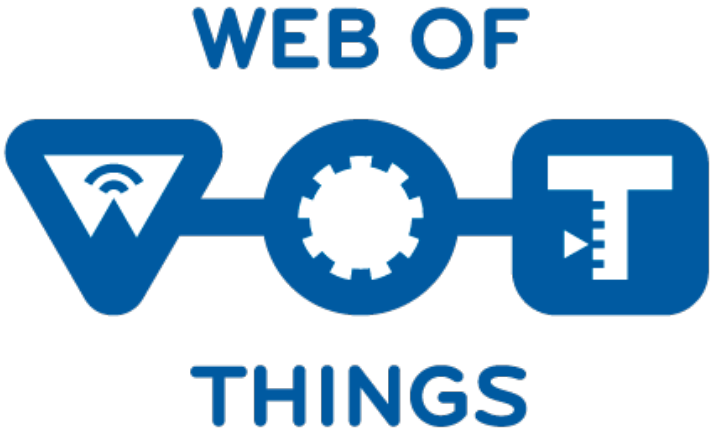


WoT Profiles

Michael Lagally

16 Dec 2021

Out of the box interoperability definition

- https://github.com/w3c/wot-profile/issues/155

Ben Francis

- "Any Consumer which conforms with a given profile can interact with any Thing which conforms with the same profile, without additional customization."

➢ Can you please clarify which consumers you think about? Clients, Servers, Servients, Intermediaries, Humans?

- For profiles in general Consumers can theoretically be a client or server or both. Consumer-Producers can be intermediaries for other Consumers. Humans are a separate issue because they don't generally consume Thing Descriptions at the API layer, they consume a human-computer interface at the UI layer, which has lots of other considerations!
- For the initial HTTP+JSON profile, I anticipate an assumption that Consumers are clients and Producers are servers.

Out of the box interoperability definition

https://european-iot-pilots.eu/wp-content/uploads/2018/11/D06_02_WP06_H2020_CREATE-IoT_Final.pdf

- Technical Interoperability: usually associated with communication protocols and the infrastructure needed for those protocols to operate;
- Syntactic Interoperability: usually associated with data formats and encodings along with techniques for compressing them;
- Semantic Interoperability: associated with shared understanding of the meaning of the exchanged content (information);
- Organisational Interoperability: associated with the ability of organisations to effectively communicate and transfer information even across different information systems, infrastructures or geographic regions and cultures.

Out of the box Interoperability

Cristiano:

- Interoperability in Internet of Things: Taxonomies and Open Challenges. There the authors define a slightly different classification:
- Device interoperability: Mostly related to hardware interoperability and the availability of dedicated hardware for communication.
- Network interoperability: Protocol related
- Syntactic interoperability: interoperation of the format as well as the data structure used in any exchange
  information or service between heterogeneous IoT system entities
- Semantic interoperability: The document refers to an old definition of semantic integration by W3C, but in general it nails down to a shared understanding of the data model.
- Platform interoperability: the ability to interoperate between different IoT technology stacks.

Revisiting Requirements

- https://github.com/w3c/wot-usecases/blob/main/REQUIREMENTS/profile-requirements.md

Use Cases:

- As an end user, I want to know whether a device will work with my system before I purchase it to avoid wasting money.
  - Installers of IoT devices want to be able to determine if a given device will be compatible with the rest of their installed systems and whether they will have access to its data and affordances.
- As a developer, I want TDs to be as simple as possible so that I can efficiently develop them.
  - Here "simple" should relate to the end goal, "efficiently develop"; that is, TDs should be straightforward for the average developer to complete and validate.
- 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.

Revisiting Requirements

Interoperability

- Supporters: Oracle, Intel, Siemens, Fujitsu, Ben (*)

Limit and reduce complexity

- Supporters: Oracle, Siemens(*), Ben(*)

Ambiguities

- Supporters: Oracle, Fujitsu, Intel(*), Ben (*)

Human readability

- Supporters: Oracle, Ben (-), IRI, Fujitsu

Developer guidance

- Supporters: Fujitsu, Siemens, Intel, Ben

Multiple profiles

- Supporters: Intel, Siemens, Ben(*), Hitachi
- **Composable profiles**

Proposer: Intel, Ben(*), Hitachi, Siemens

Legend:
* means partially agree, need to refine / rework the requirement
- means oppose

Revisiting Requirements

Validatable TDs

- Supporters: Intel, Oracle, Ben

Identification of profiles

- Supporters: Intel, Siemens, Fujitsu, Ben, Hitachi

Profile should define a finite set of features and capabilities to implement by the consumer.

- Supporters: Intel, Oracle, Fujitsu, Ben(*), IRI

Limit resource consumption

- Supporters: Intel, Oracle, Siemens, Fujitsu, Ben (-)

Follow Security and Privacy Best Practices

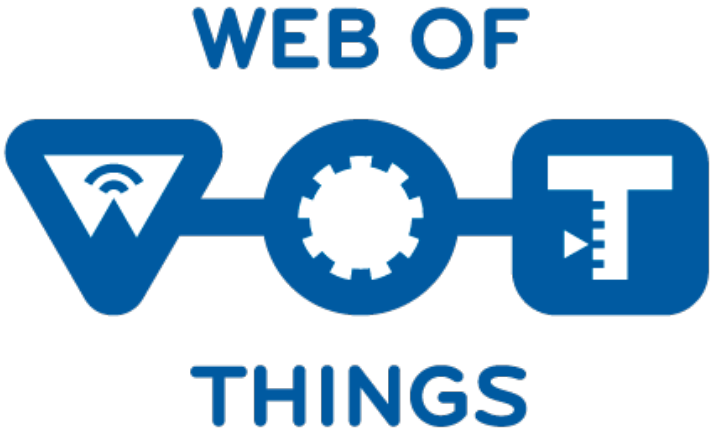
- Proposers: Intel, Ben (*)

Developer Mode

- Proposers: Intel, Ben (-)

Deployment Scenarios

- Not discussed



WoT Profiles

Michael Lagally

9 Dec 2021

Scope

- Profile Use Cases:

https://w3c.github.io/wot-usecases/

3 use cases: OOTBI, X-protocol interworking, digital twin

Ben: We can solve OOTBI use case, X-protocol interop cannot be solved by a profile alone, you need a gateway or a protocol adapter that does the translation – They cannot resolve interop directly

McCool: You could have a translator to a common protocol that does that (EdgeX does that), we may have some constraints and could define something that's appropriate

Scope

Lagally: I think we can do X-protocol to a certain extent, the Oracle cloud service binds to different protocols

Cris: We do have implementation experience with MQTT bindings, OPC-UA and Modbus in node-WoT

Sometimes there are differences between protocols, a profile can help to mitigate that

Kaz: Main scope and goal was OOTBI, we should clarify what is meant by that?

McCool: We agreed on OOTBI, digital twin may have significant impact, should be careful about our scope. Who are consumers, producers?
TD purpose is to describe limitations of interfaces, constraints are already described, we should not overdo them

HTTP+JSON is appropriate for Internet connectivity, who are the agents?
Cloud services, Edge, Digital Twins, make sure they interoperate.

Scope / Interop

Ben:
TD is protocol agnostic

- Protocol bindings are protocol specific, we should not make them protocol agnostic, cleanest mapping to that protocol
- We agreed on interop, OOTBI, not resource constrained devices
- The conclusion of focus on the scope was not recorded in the last call
- We should do a resolution in this call, if possible

Cristiano:

- Like that we narrow down the scope, OOTBI can have different interpretations, what Ben said is a good start, let's clarify what we want to do
- A specific protocol similar to what WebThings is doing would be good

Kaz: Would like to get opinions from other participants

Scope / Interop

Toumura: I have implemented a consumer, narrow down the specification is very useful, we have effort for consumer implementation – if we can lower spec our implementation would be easier to implement

Lagally: For Oracle it is essential to have limits and constraints that match for database based server implementations. No problem of reading large TDs, event streams. Generic UI is very important to us.

Mccool: support a resolution on OOTBI , suggest to define the scope and entities that are interacting

Ben: Generic UI should be resolved in the TD, title and description constraints also

Scope / Interop

Matsukura-san:
I understand everybody's concern wrt. Interop.
Fujitsu has product aligned with WoT in Japanese market, now we focus on maintenance, how to connect to a device. We can get TD identifier, but information is not connected with actual devices, so we have to describe on TD when we install in the field, when and where we installed it. This is important to the data, therefore human readable information is important to us. TD is abstract description, profile document is very suitable for these practical use cases. Limitations, max length of description, depth of nesting, some implementations describe nesting, gateway needs to have that.
Expect that from the profile document.

Scope / Interop

Sebastian:
It would be very useful if we consider a specific protocol such as HTTP, it would be very helpful to describe how the protocol to be interpreted. This is a very important aspect. Our commercial implementation is flexible on the data model, not a strict representation needed, specifically for sayWoT, can be adapted to any kind of customer implementation, we do not need a restriction on the data model. There are existing deployments which are used there. It is hard to narrow that down to fit all application domains.
We can make a clear guideline on protocol binding, but not restrict the data model - titles and descriptions everywhere should not be mandatory.

Scope / Interop

Ege: 2 points: definition or use of interop: a single profile and multiple implementations can interoperate, we cannot solve OOTBI with a single profile, this is the TD, to solve the wide interop we can use the TD
2: reduce implementation effort, allowing consumers to have more assumptions, for UI generation if you always know that there's a description, a "not identified" fallback could be used. Reduce new consumers implementation effort, e.g. for a UI, on a number there should be min and max, all could be done with the current TDs, but if you have these guarantees you could build Uis easier, more beautiful, more robust

We can motivate some of the requirements to reduce implementation effort for consumers. Different profiles will create silos.

Scope / Interop

Expectation: reduced implementation effort for consumers

Mizushima:

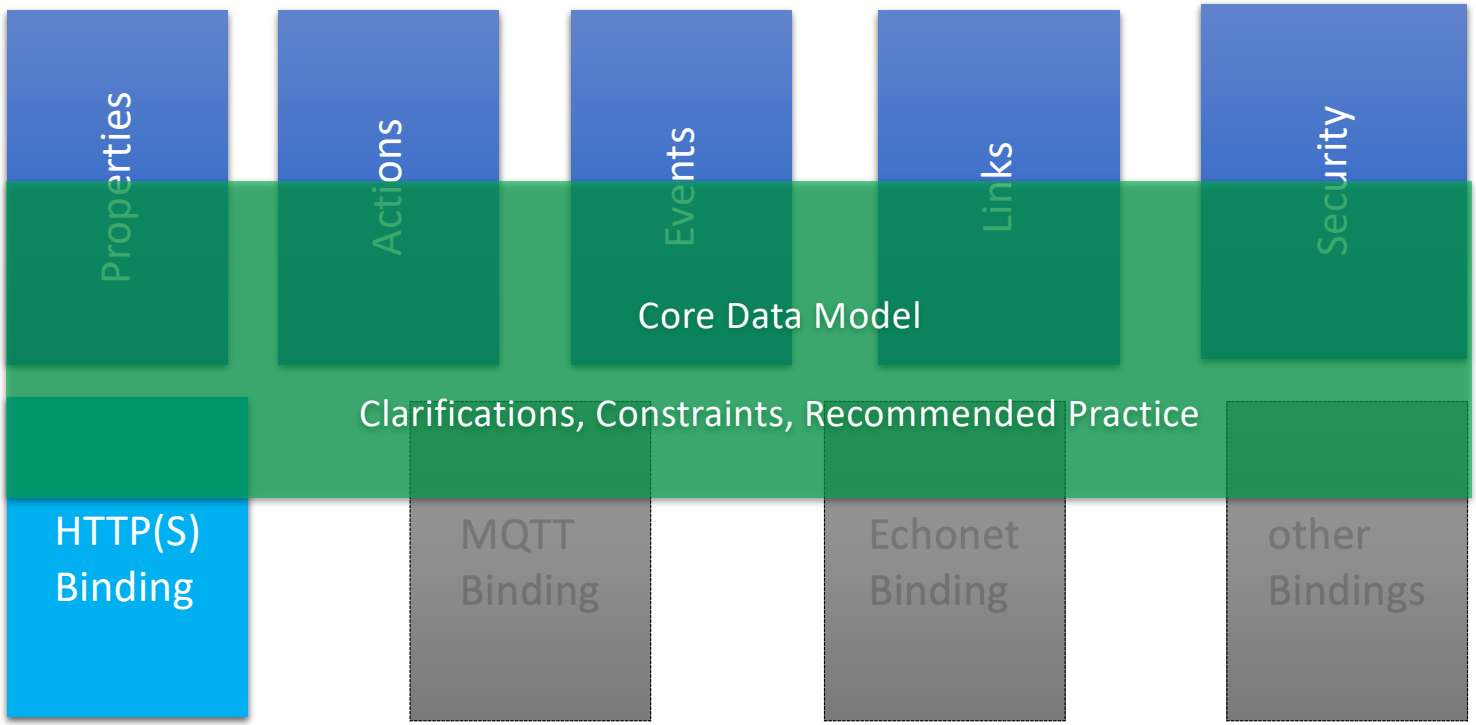
Expectation: Japanese CG members said they can't understand TDs, because very complicated. How to use a TD in a use case. Need a guideline how to sync use case and TD structure, how to create TD files easily.

Ben:
Expectation: OOTBI: Any conforming consumer can interact with anything that conforms to the profile without customisation. AD-Hoc interoperability.

Expectation: We have defined building blocks: TD, discovery, … Profile should be a guideline for HTTP based interaction to create a TD without any big requirements, detailed information on what to do when HTTP based interaction.

Backup – last Week’s slides

Profile Concept



Example Consumer Scenario

A worldwide climate monitoring system obtains data from sensors and gateways around the world to indicate the current weather conditions and be able to predict critical conditions.

The system displays a world map with all sensors, where the user can zoom in to individual regions.

Temperature, humidity and other sensor readings etc. are provided to a common server, which aggregates the data and uses configurable rules to trigger alerts based on sensor data.

This example motivates to consider the following aspects:

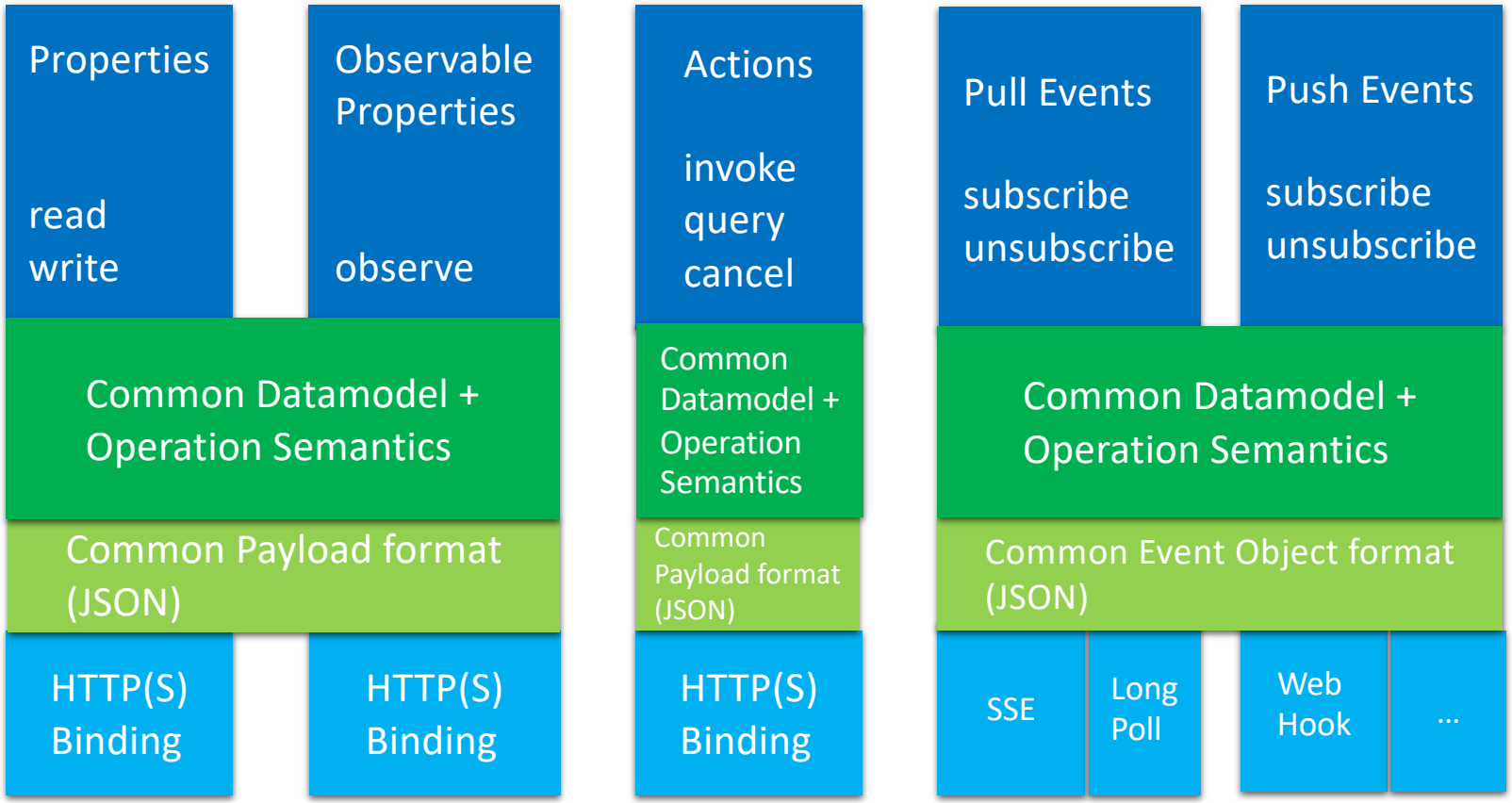
- All sensors and gateways must use unit schemes that are known and can be interpreted by the consumer.
- All sensors and gateways must use an unambiguous time and date format.
- All sensors and gateways must provide a human readable name that can be displayed on a map.
- All sensors and gateways must provide their location in a format that is known to the consumer.
- If a sensor and gateways provides interactions, these must be displayed in a UI in a human readable form.

Sensor readings will be displayed in a UI, the names must be displayed in a UI in a human readable form.

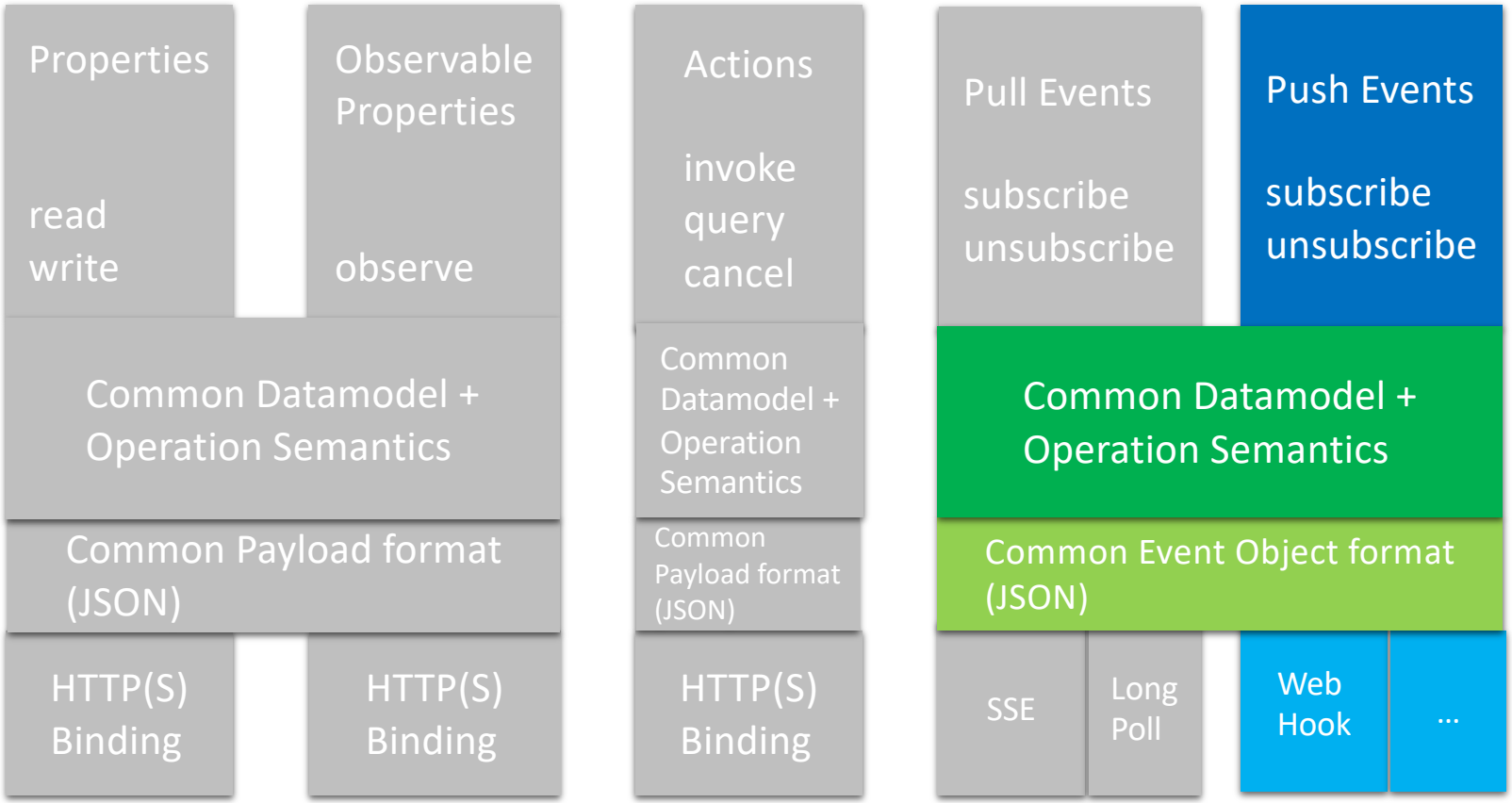
Some Implications

- Consumer must handle sensors and gateways
- Some gateways are aggregating/indirectly providing sensors data
- From a consumer's perspective the implementation (sensor or gateway) should not make a difference.

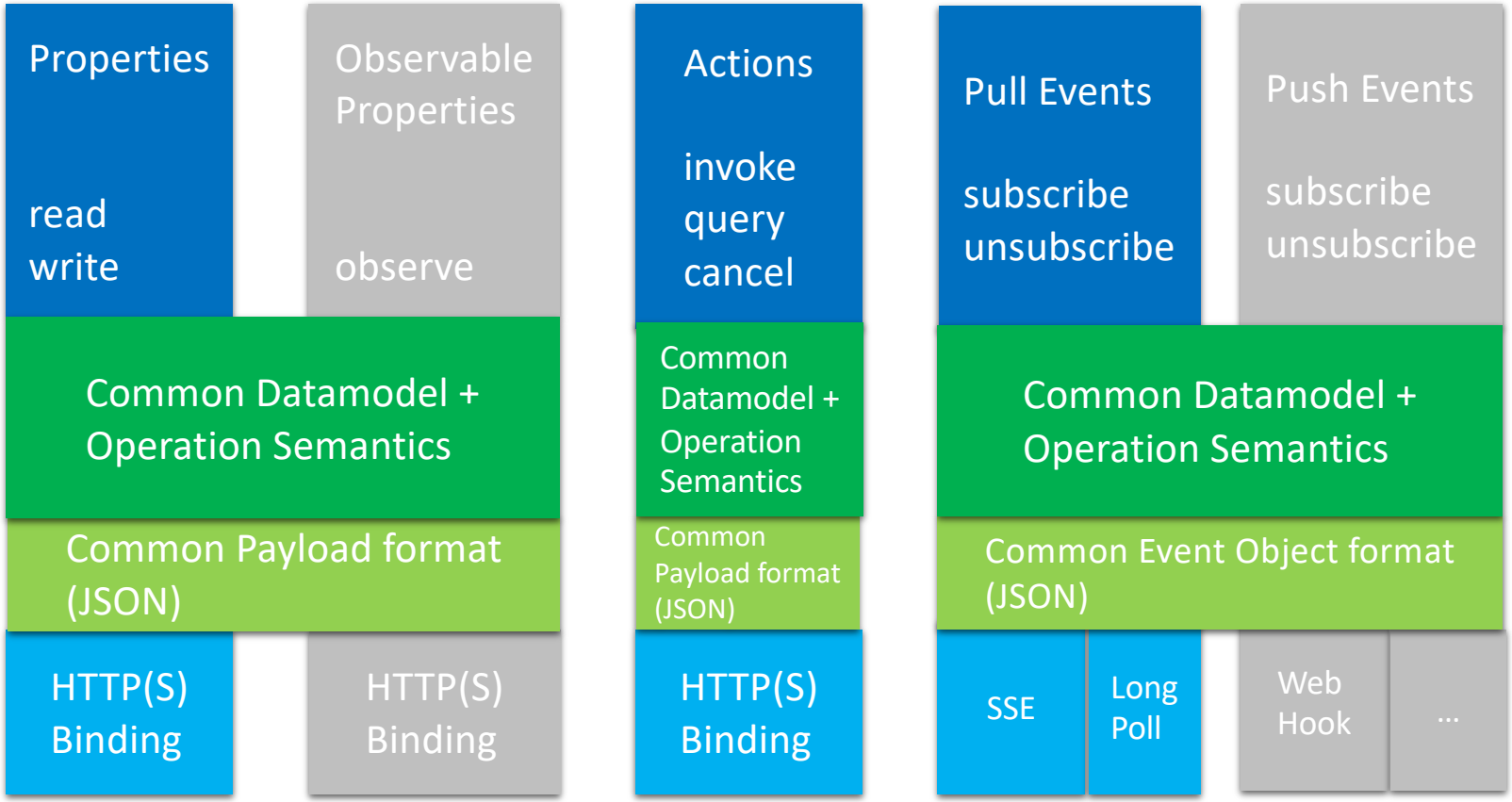
Common Datamodel + Operation Semantics



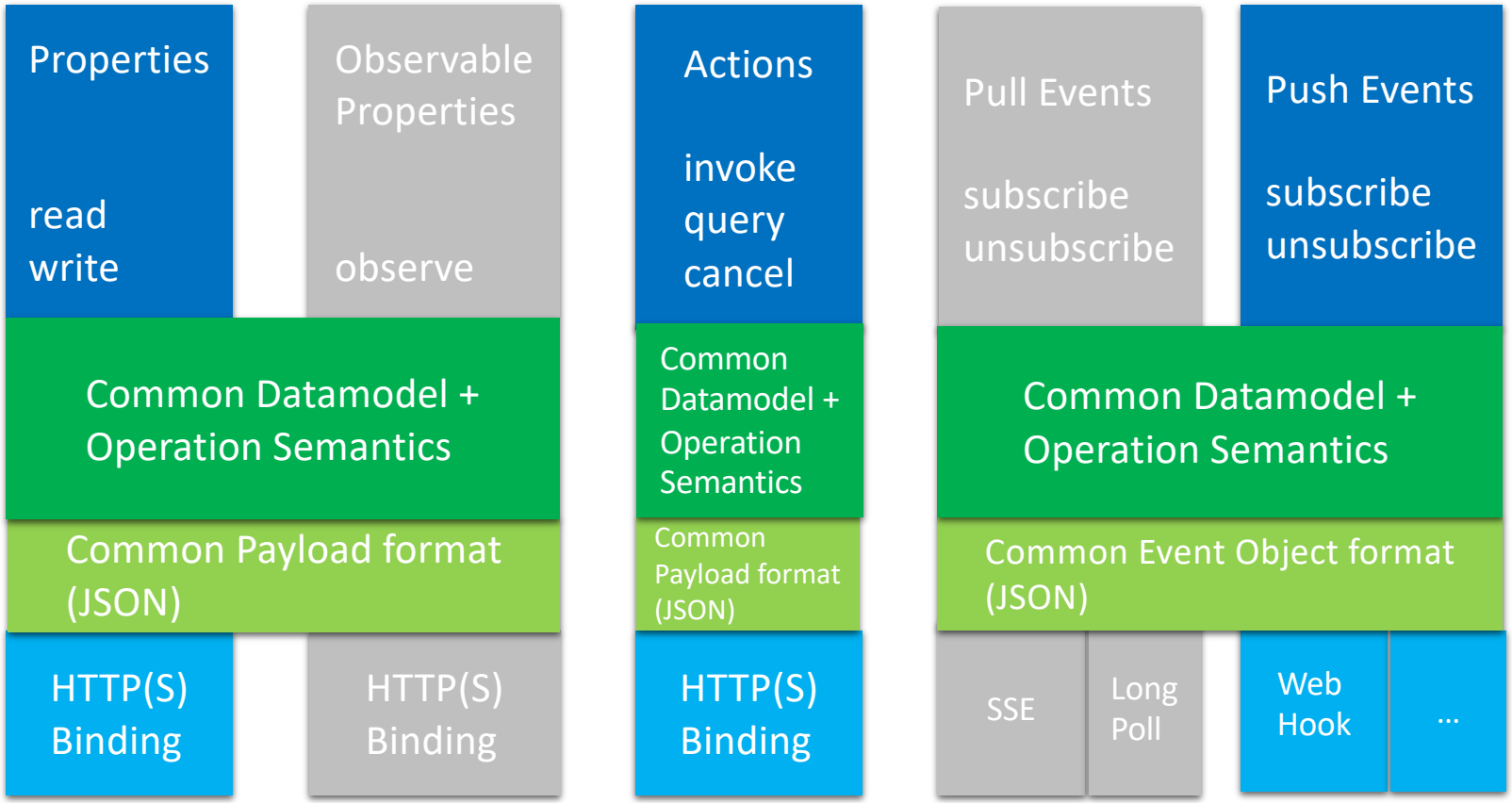
What does a simple Sensor need?



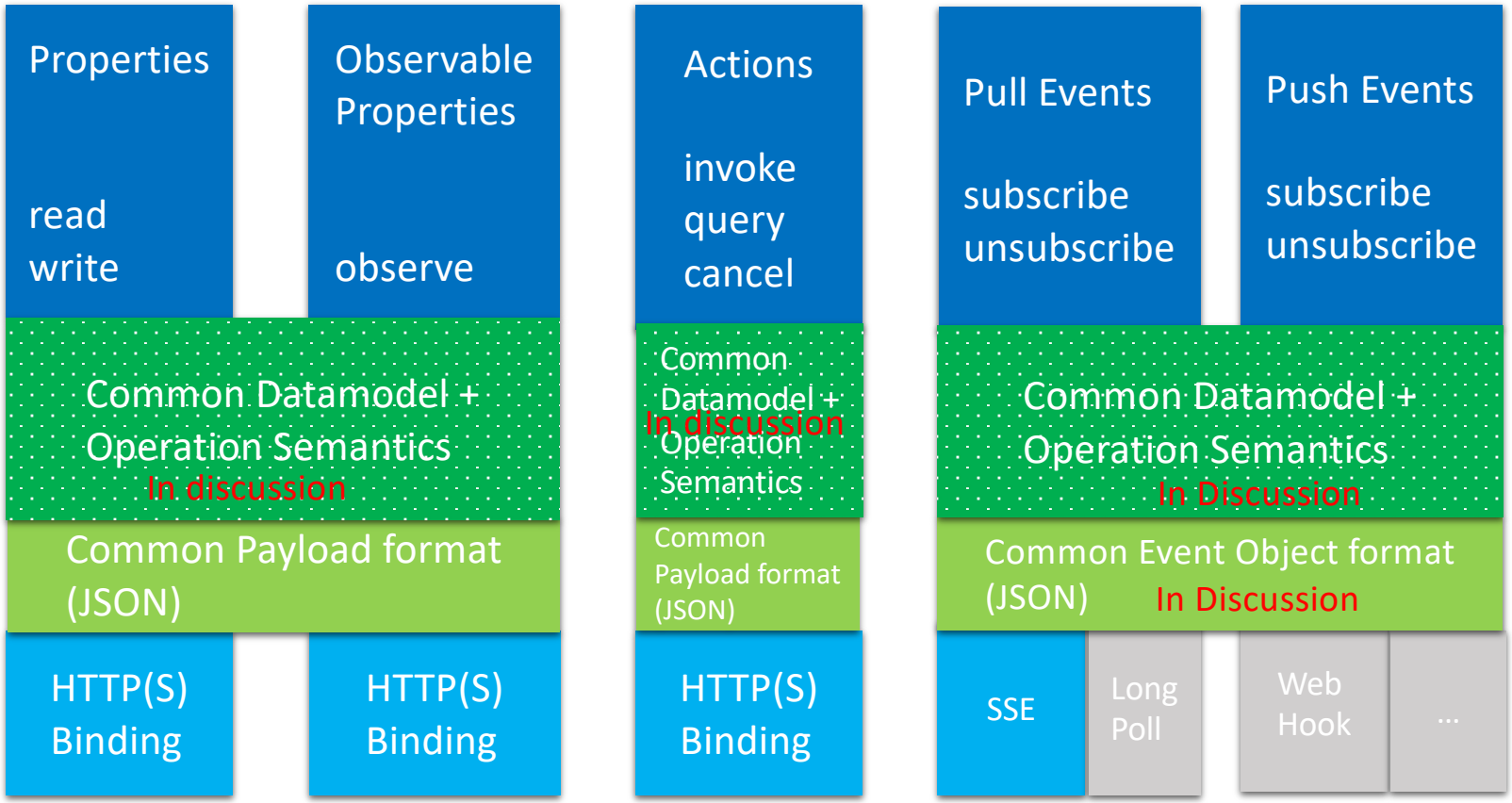
What does a Home Gateway need?



What does an industrial gateway need?



Where are we today?



Profile Names

The current HTTP binding of the core profile is not sufficient to address industrial use cases, since it does not contain a push event model.

It is more like a smart home gateway profile, the name core profile is misleading.

To ensure interoperability for things with HTTP (s) additional profiles have to be defined for:

- Sensors
- Industrial Gateways
- Digital Twins

However the core profile could include the missing pieces and be suitable for these deployments. A common datamodel and operation semantics is required.

Discussion (1)

McCool:

- Talking about core / home gateway
- – common constraints, such as proposed names
- We should work bottom up, define profiles for narrow scenarios, home, industrial, …
- Derive common constraints based on that experience
- Start with home and industrial, consider others later
- Look for overlaps
- Avoid to have different event models in different profiles
- Common constraints could be defined based on experience

Discussion (2)

Sebastian:

- Common data model: Usage of the TD information model
- Data model is application specific – semantics depend on application domain
- Structure of the TD information model
- We may have for 1.1 data model aspects for smart city, …

Discussion (3)

Kaz:

- Which part to be described as the core profile?
- We should look into industry based IoT standards, that might be quicker
- Such as OPC-UA, Echonet, oneM2M, SG20
- This is much related to the binding discussion

- Echonet has 2 levels of interfaces – WebApi and binary level API

Discussion (4)

Ben:

- Agree with Sebastian and McCool
- Difficulties to define common constraints on all devices that are defined on the TD information model
- Units would be useful, but difficult to chose
- Bottom up approach
- Disagree that current draft targets only home gateways
- Application domain specific profiles
- Events discussion is misleading – HTTP is not suitable for events

Discussion (5)

Scope of the spec:

- Interop
- Human readability was excluded
- We should revisit each assertion individually

Lagally:

- Building a UI is part of interoperability, otherwise you cannot interoperate between devices and humans