iot.schema.org

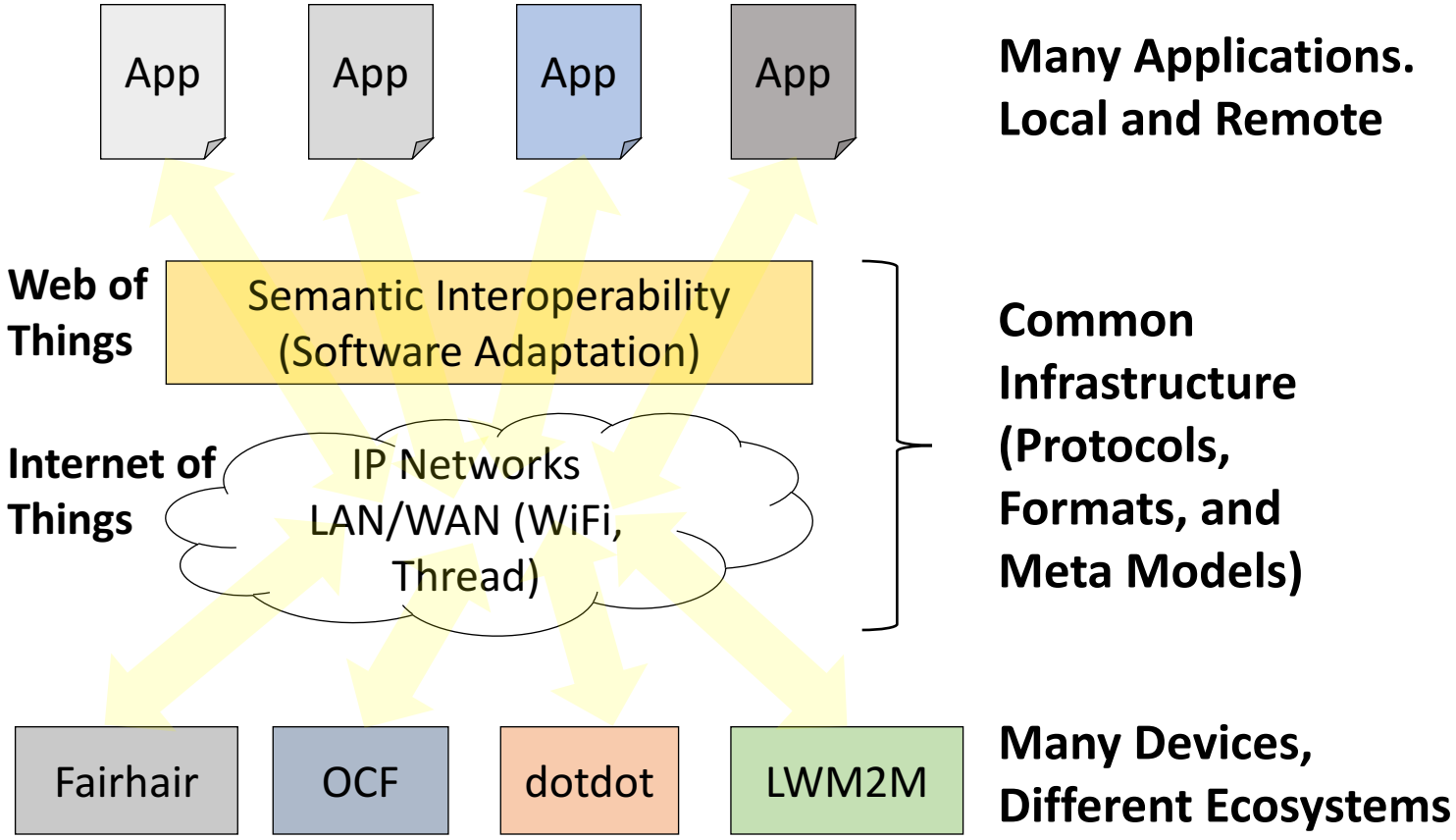
Overview and Update

July 1, 2018

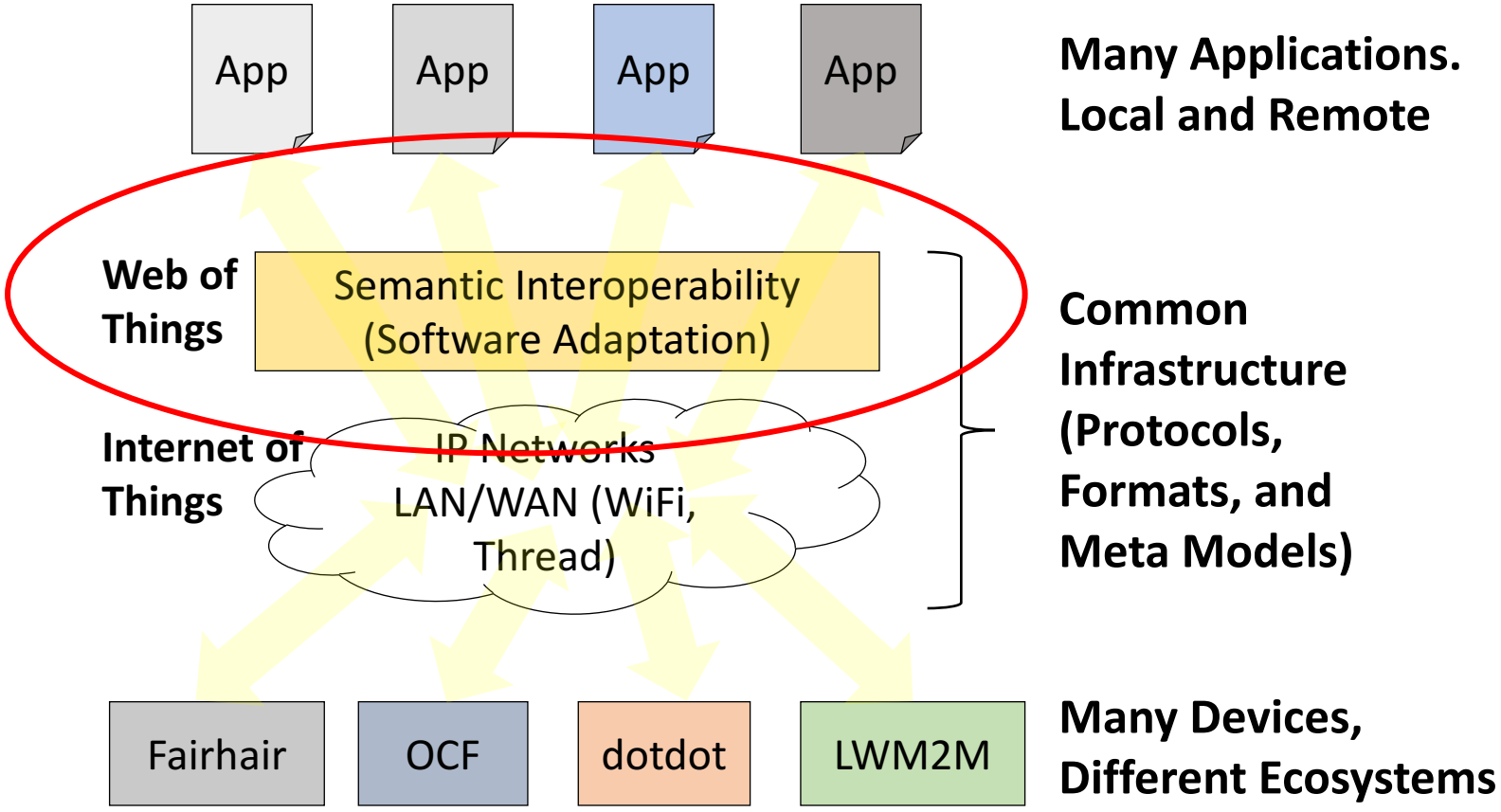
Semantic Interoperability

- What?
  - Common ways to describe interactions with the physical world – Abstract Interactions – e.g. Temperature
- How?
  - Detailed instructions for protocol translation and adaptation – Protocol Bindings – e.g. REST API
- iot.schema.org provides the domain-specific vocabulary that describes "What to do"
- Used to annotate Thing Descriptions

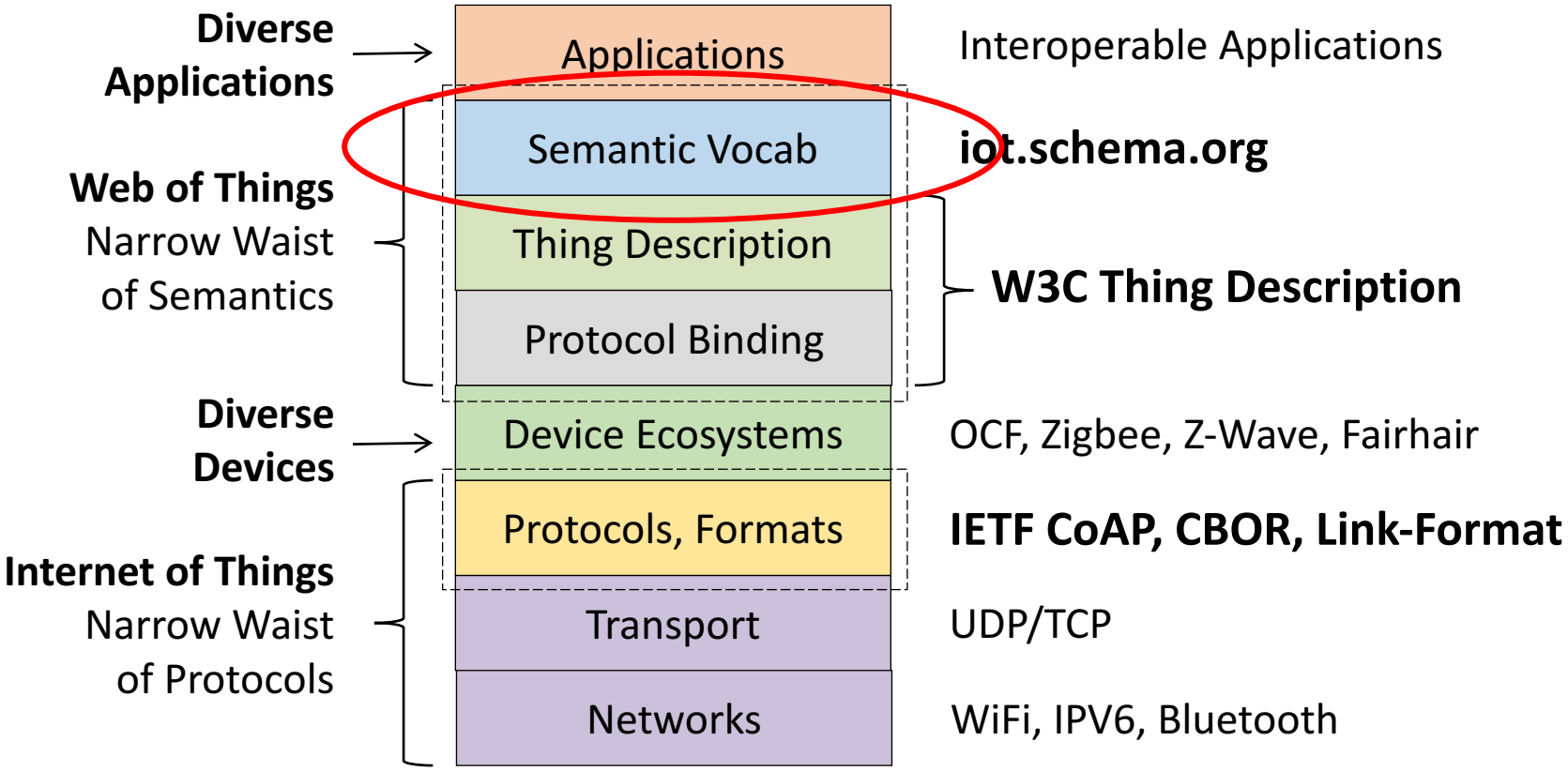
Narrow Waist in System Design



Narrow Waist in System Design



Diverse Devices and Applications, Common Protocols and Semantics



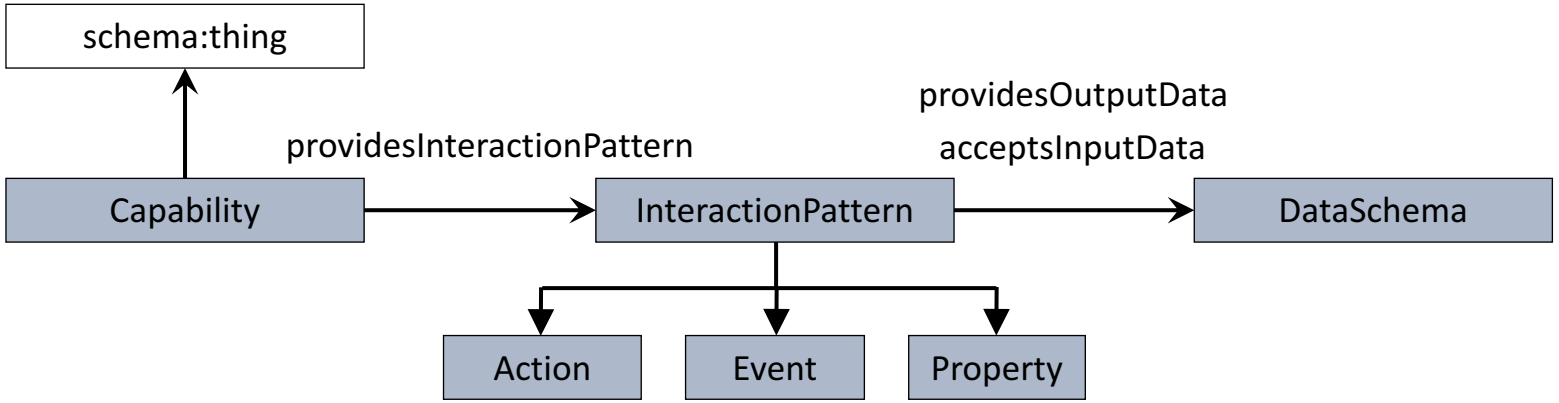
iot.schema.org

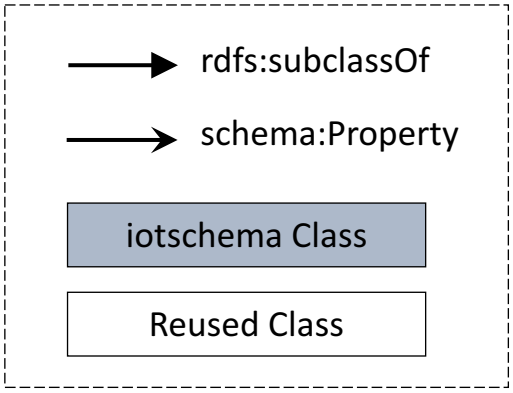
- What is it?
  - An extension to schema.org and a meta-model for the semantics of interacting with connected things
  - Open, public definitions for connected things and their context, in a reusable and protocol-agnostic format
- What problems does it solve?
  - The difficulty for domain experts and system developers to create and use formal semantic descriptions
  - The lack of common semantic vocabularies and conceptual models
  - Interoperability across diverse application domains

iot.schema.org

- How does it solve these problems?
  - Common meta-model with categories of Capability, Interaction, and DataSchemas
  - Interaction types of Event, Action, and Property align with existing devices
  - Associated with Features of interest to describe relationship to the physical world
  - Enable multiple domain-specific vocabularies that can re-use the common categories and a core set of common definitions
- What is the benefit of this approach?
  - Domain experts can focus on the high value semantic definitions that are important to them, without needing to become semantic web experts

iot.schema.org Common Pattern

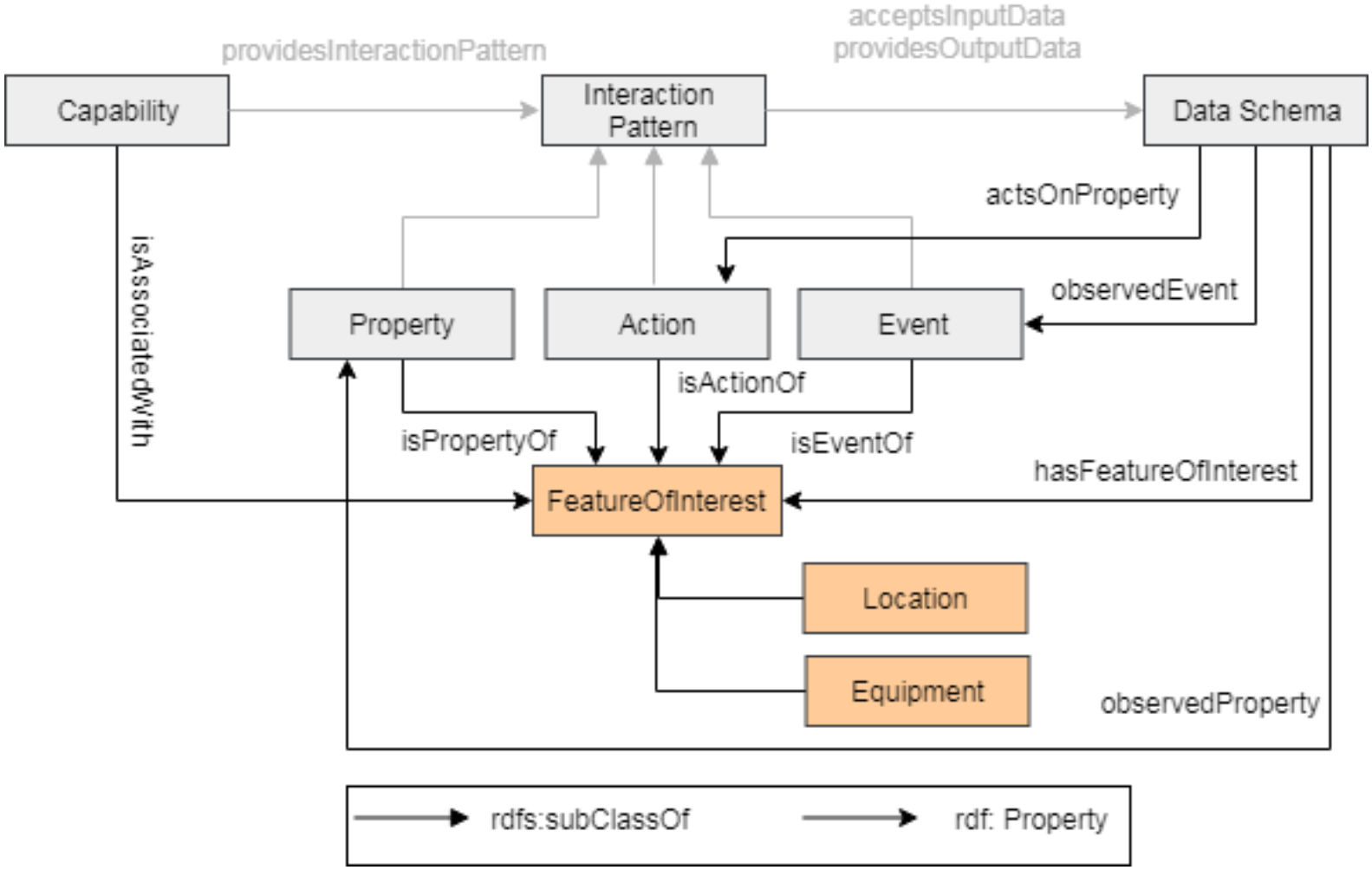




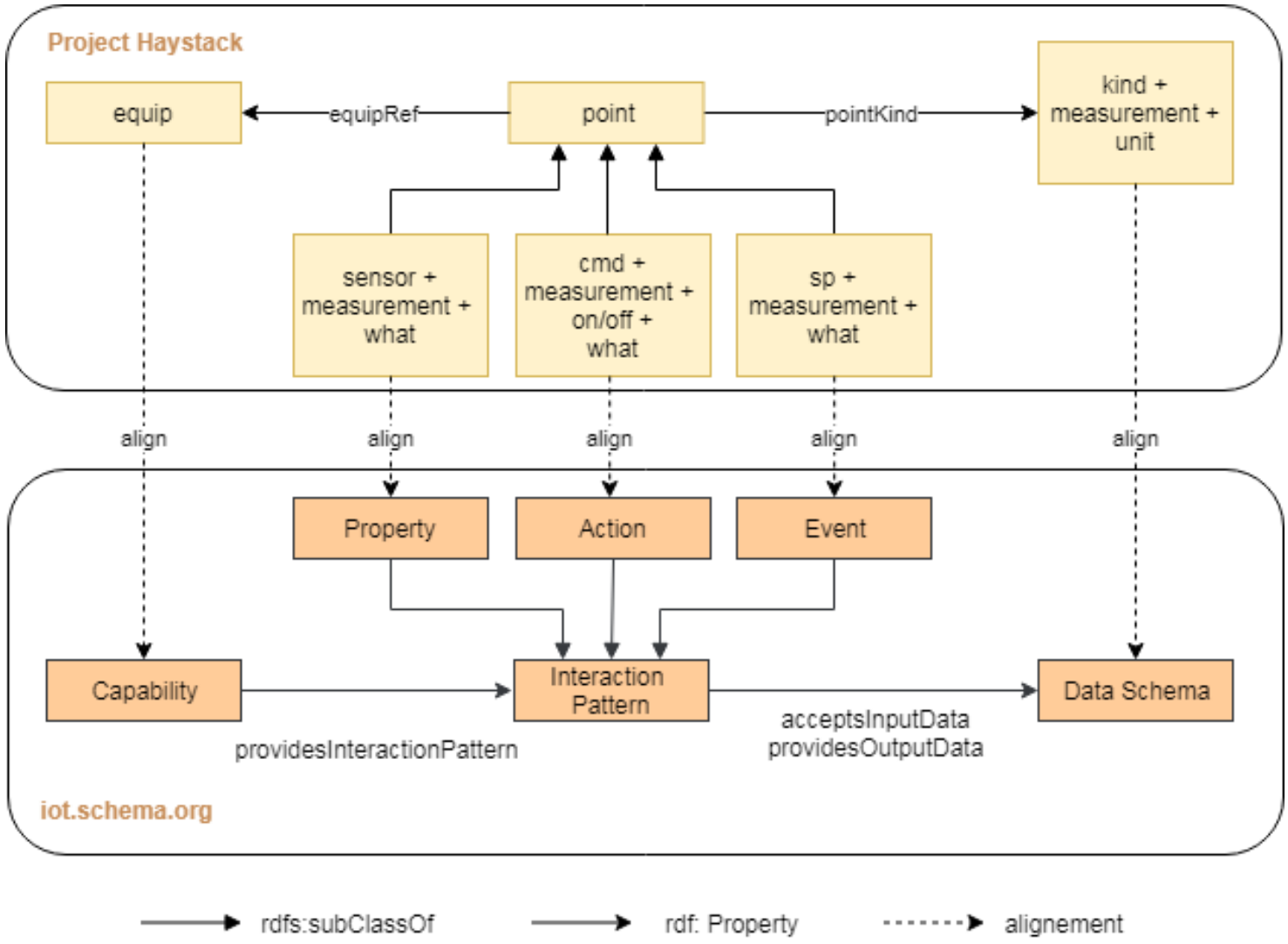
iot.schema.org Example Definition

```
{
  "@id": "iot:Temperature",
  "@type": "rdfs:Class",
  "rdfs:comment": "Temperature interaction property",
  "rdfs:label": "Temperature",
  "rdfs:subClassOf": { "@id": "iot:Property" },
  "iot:providesOutputData": {
    "@id": "iot:TemperatureData"
  },
  "iot:writable" : "schema:Boolean",
  "iot:observable" : "schema:Boolean"
},
{
  "@id": "iot:TemperatureData",
  "@type": "rdfs:Class",
  "rdfs:comment": " Temperature data",
  "rdfs:label": "TemperatureData",
  "rdfs:subClassOf": { "@id": "schema:PropertyValue" },
  "schema:propertyType": { "@id": "schema:Number" },
  "schema:unitCode": { "@id": "iot:TemperatureUnit" },
  "schema:minValue": "schema:Float",
  "schema:maxValue": "schema:Float"
}
```

Feature Of Interest Pattern



Integration of Haystack vocabulary in iot.schema.org



Feature of Interest Example

```
{
  @id": "iot:LiquidMixingSystem",
  "@type": "rdfs:Class",
  "rdfs:comment": "A LiquidMixingSystem is an Equipment.",
  "rdfs:label": "LiquidMixingSystem",
  "rdfs:subClassOf": {
    "@id": "iot:Equipment"
  }
},
{
  "@id": "iot:LiquidPipe",
  "@type": "rdfs:Class",
  "rdfs:comment": "A Liquid pipe is an Equipment.",
  "rdfs:label": "LiquidPipe",
  "rdfs:subClassOf": {
    "@id": "iot:Equipment"
  }
}
```

Thing Description Annotation


```
{  "@context": [{"iot": "http://iotschema.org/",
                 "festoPA":"http://example.com/FestoPA/"}  ],
   "@type": [ "Thing", "iot:Pump", "iot:Valve", "iot:FloatSwitch", "iot:UltrasonicSensing“ ],
   "iot:isAssociatedWith" : {"@id": "festoPA:FESTO-1", "@type": "iot:LiquidMixingSystem"},
   "name": "FestoLive",
   "id": "urn:dev:wot:siemens:festolive",
   "security": [{"scheme": "basic"}],
   "properties": {
   "PumpStatus": {
        "@type": "iot:OperationStatus",
       "isPropertyOf": {"@id": "festoPA:Pipe2", "@type": "iot:LiquidPipe"},
        "type": "object",
        "properties": {"PumpStatus": {"type": "boolean"}},
        "writable": false, "observable": false,
        "forms": [{  "href": "https://129.144.182.85/iot/api/devices/Festo/PumpStatus",
           "mediaType": "application/json“   }]   }
```

Tools and infrastructure

- Work in progess: definitions using RDF shapes
  - Flexible structure and value constraints
  - Augment the RDFS structural constraints
  - Alternative to PropertyValueSpecification for data types
  - Way forward for browsing definitions in HTML
- Constructor tool to populate WoT Thing Descriptions from iot.schema.org capability definitions
- Working on the roadmap to converge/merge with schema.org

iotschema.org Prototype Website

Capability

Canonical URL: http://iotschema.org/Capability

Capability

Capability Class

Property	Expected Type	Description
Properties from Capability		
`providesInteractionPattern`	InteractionPattern	A property that relates a capability with its interaction patterns.
`isAssociatedWith`	FeatureOfInterest	A relation between a Capability and the entity it belongs to.

Instances of Capability may appear as values for the following properties

Property	On Types	Description
`capability`	InteractionPattern	A property that relates an interaction pattern with its capability .

More specific Types

- MotionControl
- Pump
- AirConditioner
- TemperatureSensing
- LevelSwitch
- LightControl
- UltrasonicSensing
- HumiditySensing

W3C WoT CG Charter

- Subset of the iot.schema.org charter, for incubating contributions of definitions to iot.schema.org
- Develop the process for achieving community consensus to publish agreed definitions
- Contributor IPR regime is based on schema.org or W3C Community Group (either are OK)
- Creative Commons CC-BY License
- Ultimately expect to split off domain-specific community groups or use existing CG, for example automotive, based on this as a template

Events and Conferences

- W3C Web of Things face to face meeting in Korea
  - June 30-July 4th 2018
  - 2 day Plugfest with
  - Using iot.schema.org annotation including FoI
- WISHI at the IETF 102 hackathon
  - 2 days; July 14th and 15th
  - Using iot.schema.org annotation with WoT framework
  - Research questions around semantic annotation and hypermedia integration