iot.schema.org

Alignment with Brick Schema and Haystack Vocabulary,

Update on Feature of Interest

June 28, 2018

Haystack vocabulary in iot.schema.org

ALIGNMENT WITH BRICK SCHEMA

Project Haystack

- Background:
  - Aims to standardize semantic data models to unlock the value of data generated by building equipment.
  - It is an open source initiative to enable Internet of Things applications.
  - Applications include automation, control, energy, HVAC, lighting, and other environmental systems.
- Goal:
  - propose a concept to integration Haystack model with iot.schema.org
- Review of existing schemas, which provide an RDF/OWL model for Haystack, and a proposal for the integration

Berkeley-IBM-UVA Model for Haystack

- An RDF representation of Haystack tags and tagsets
- No schema available
- Example:

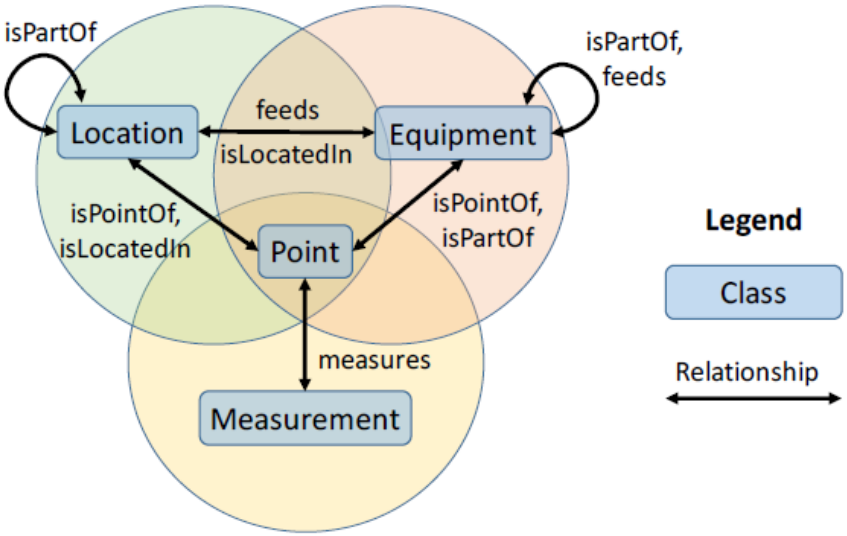
```
:ahu  rdfs:subClassOf     :HaystackMarker;
   rdfs:label    "AHU"@en;
   rdfs:seeAlso  <http://project-haystack.org/tag/ahu>;
   :usedWith     :equip;
   :usedWith     :rooftop.

:ahu_set  rdfs:subClassOf     :HaystackMarkerSet;
   rdfs:label  "AHU set"@en.

:ahu_discharge_air_temp_sensor  rdfs:subClassOf     :ahu_set;
   owl:equivalentClass [ rdf:type owl:Class ; owl:intersectionOf (
      [ rdf:type owl:Restriction; owl:onProperty :hasMarker; owl:someValuesFrom :ahu ]
      [ rdf:type owl:Restriction; owl:onProperty :hasMarker; owl:someValuesFrom :discharge ]
      [ rdf:type owl:Restriction; owl:onProperty :hasMarker; owl:someValuesFrom :air ]
      [ rdf:type owl:Restriction; owl:onProperty :hasMarker; owl:someValuesFrom :temp ]
      [ rdf:type owl:Restriction; owl:onProperty :hasMarker; owl:someValuesFrom :sensor ]
   ) ].
```

Source: https://github.com/arkaaloke/Berkeley-IBM-UVA

Brick Schema



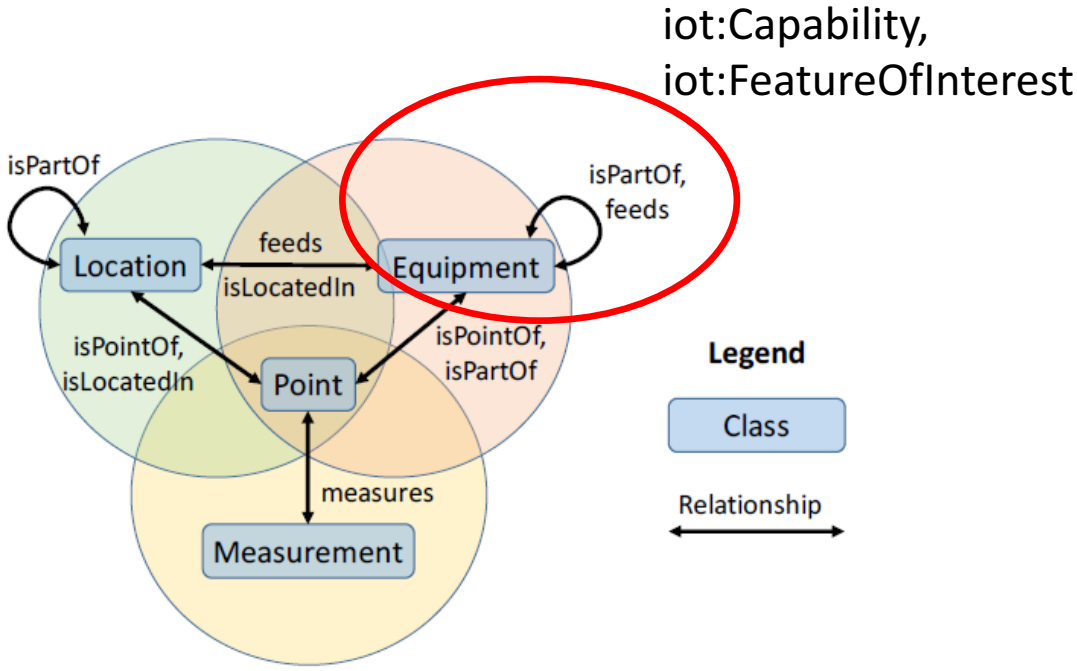
- RDF representation of Haystack tags and tagsets
- Brick has additional tags and tagsets
- Example:

```
brick:AHU_Discharge_Air_Temperature_Sensor

bf:usesTag
        :AHU,
        :Air,
        :Discharge,
        :Sensor,
        :Supply,
        :Temperature .
```

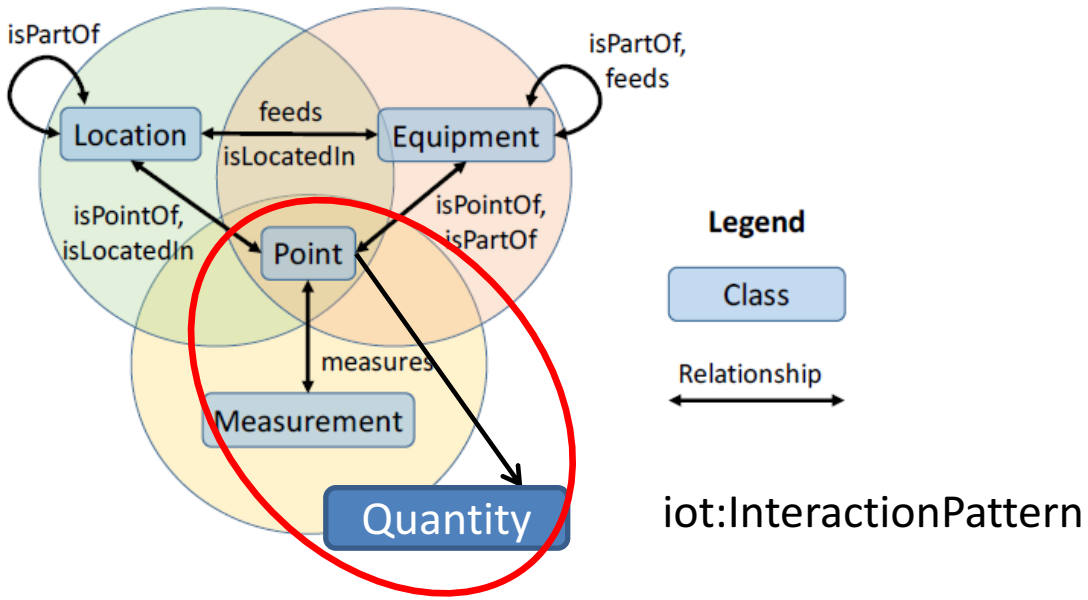
Source: http://brickschema.org/

Brick iot.schema.org Integration



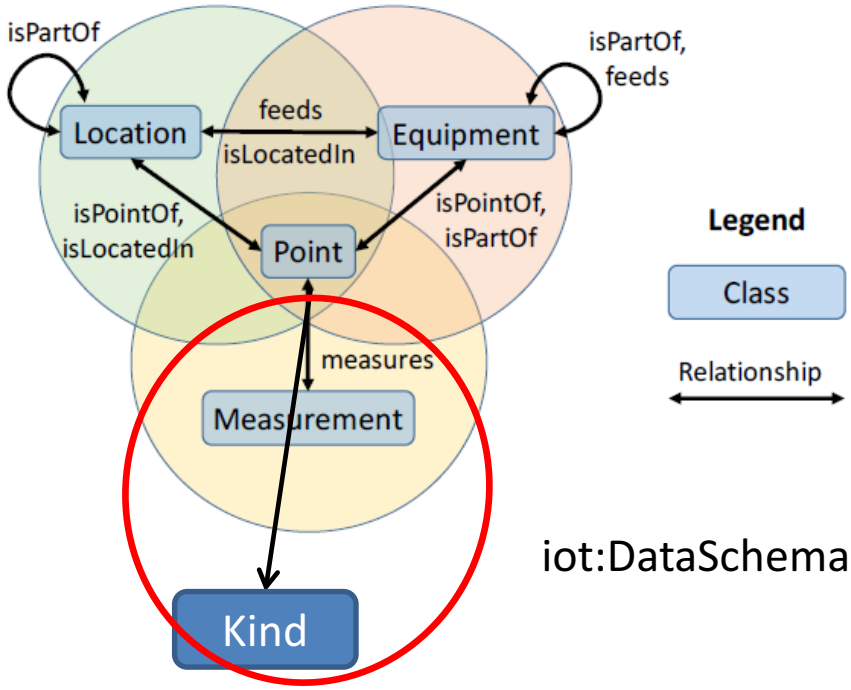
- Equipment aligns to iot:FeatureOfInterest and iot:Capability
- Example: Boiler equip →Boiler as a Capability and FeatureOfInterest

Brick iot.schema.org Integration



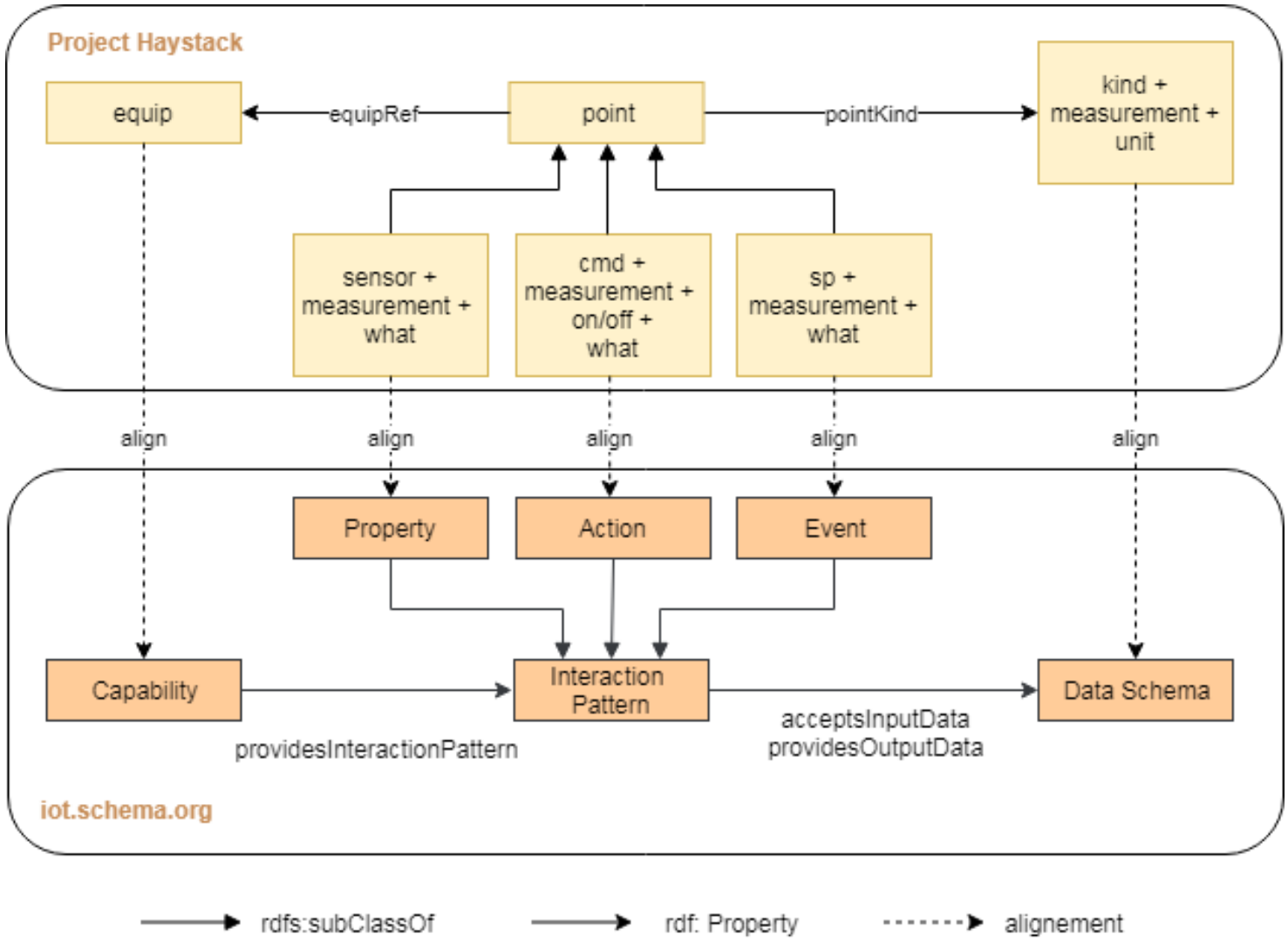
- Quantity is missing in Brick schema. A quantity is a measure of an observable phenomenon, that, when associated with something, becomes a property of that thing.
- Quantity is missing in Brick.
- Point, Measurement, and Quantity align to iot:InteractionPattern
- Example: chilled water delta temp sensor → InteractionPattern: ChilledWaterDeltaTemperature (Property)

Brick iot.schema.org Integration

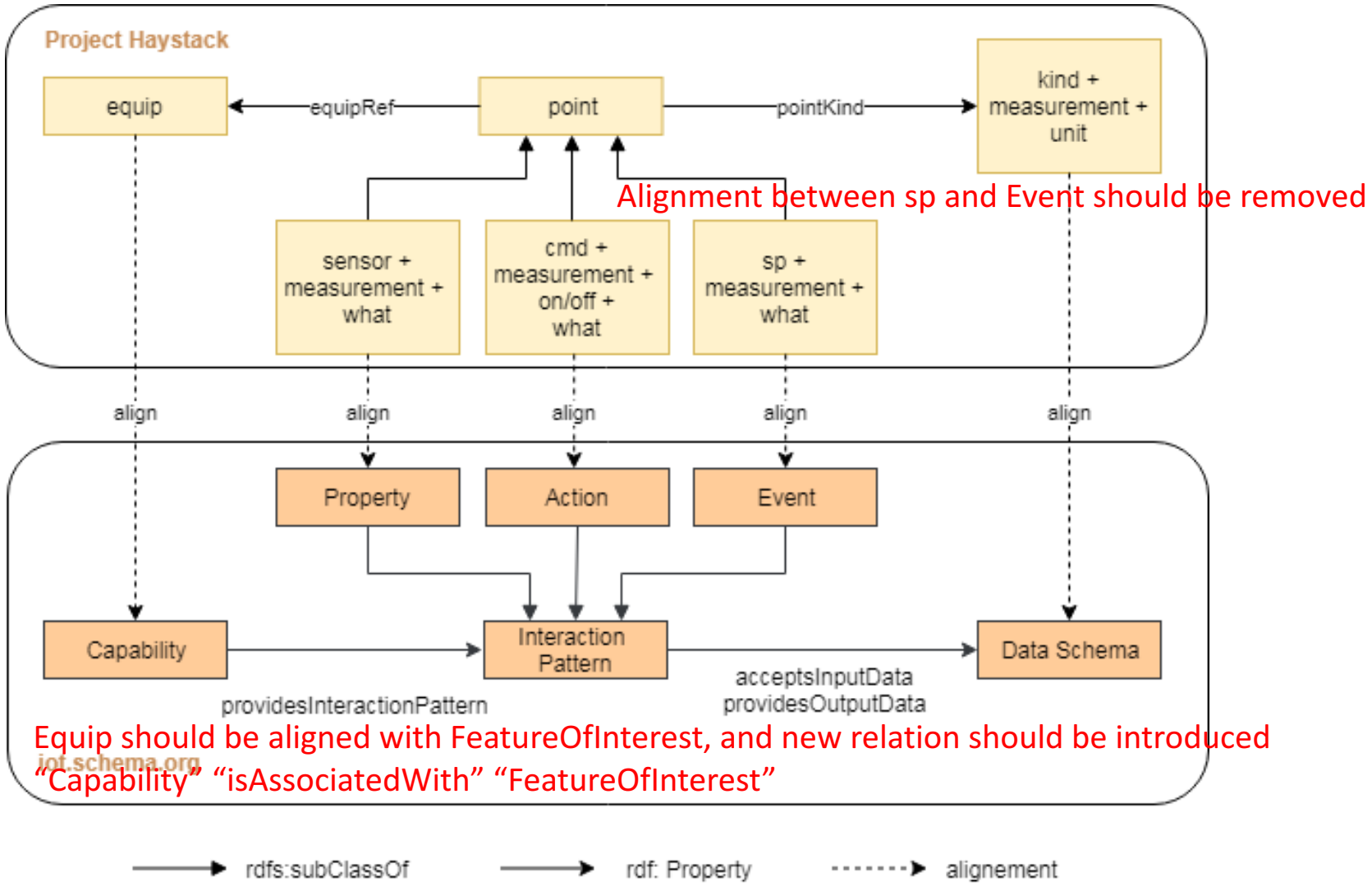


- Kind in Haystack defines a tag value type using a predefined string constant.
- Kind (DataSchema) is missing in Brick.
- For some points kind is missing in Haystack as well.
- Kind and Measurement align to iot:DataSchema

Update: Integration of Haystack vocabulary in iot.schema.org



Update: Integration of Haystack vocabulary in iot.schema.org



Example

iot:Capability: iot:Boiler

- subclasses: iot:HotWaterBoiler, iot:SteamBoiler, iot:OilBoiler etc

iot:InteractionPattern:

- iot:Action: iot:TurnOn, iot:TurnOff (run cmd)
- iot:Property: iot:RunStatus (run sensor)
- iot:Action: iot:CirculatePumpOn, iot:CirculatePumpOff (circ pump cmd)
- iot:Property: iot:CirculatePumpStatus (circ pump sensor)
- iot:Action: iot:CondensatePumpOn, iot:CondensatePumpOff (condensate pump cmd)
- iot:Property: iot:CondensatePumpStatus (condensate pump sensor)

Example

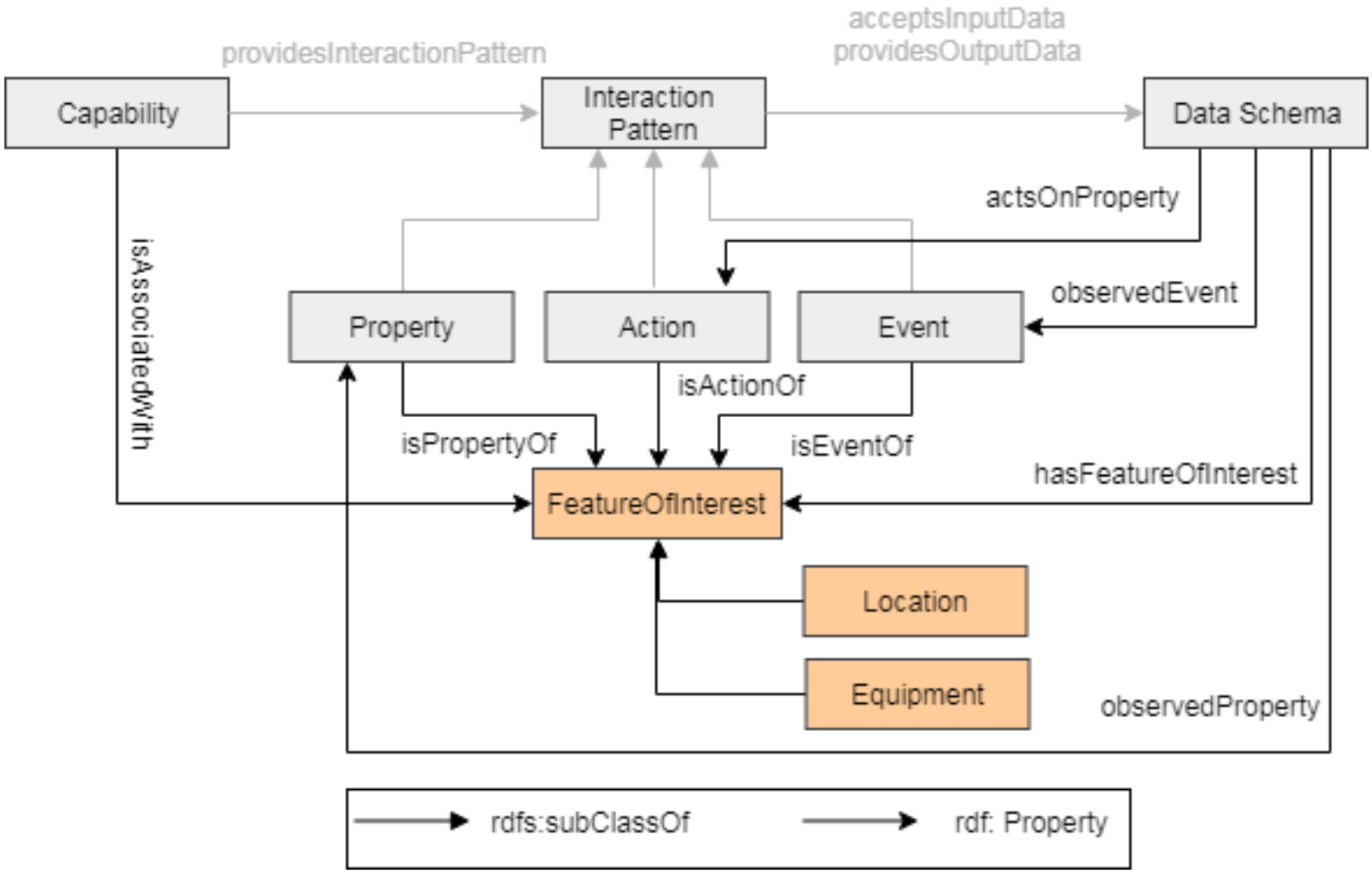
```
:Boiler a owl:Class ;
    rdfs:label "Boiler"@en ;
    rdfs:subClassOf bf:Tag ;
    skos:definition ""@en ;
    bf:usedBy brick:Boiler,
    brick:Boiler_On_Off_Status,
    brick:Boiler_Run_Time_Sensor,
    brick:Boiler_Start_Stop_Status .
brick:Boiler_Run_Time_Sensor bf:usesTag :Boiler,    :Run,    :Sensor,    :Time .
```

- Haystack defines more points for a Boiler than present in Brick, e.g. Brick does not define:
  - circ pump cmd, circ pump sensor, condensate pump cmd, condensate pump sensor

Update on the meta-model

FEATURE OF INTEREST

Feature Of Interest Pattern



Thing Description Example


```
{  "@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“   }]   }
```

Thank You!

Questions please…