# Poster Abstract: Schemas for IoT interoperability for Smart Cities

Abhay Sharma, Srikrishna Acharya, Vasanth Rajaraman, Rakshit Ramesh, Arun Babu, Bharadwaj Amrutur
RBCCPS, IISc, Bangalore

## ABSTRACT

One of the key aspects of smart city ecosystem is enabling easy collection and exchange of data to develop new applications. Providing good, open API's for smart city middleware along with standardising the data schemas will be vital for application and device ecosystem to evolve. In this paper we present the resource catalog component of the middleware along with a framework to develop data schemas for IoT devices. Using data schemas one can provide meta-data which enables effective use of the device data.

**ACM Reference Format:**
Abhay Sharma, Srikrishna Acharya, Vasanth Rajaraman, Rakshit Ramesh, Arun Babu, Bharadwaj Amrutur. 2017. Poster Abstract: Schemas for IoT interoperability for Smart Cities. In *Proceedings of 4th ACM Conference on Systems for Energy-Efficient Built Environments (BuildSys'17).* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3137133.3141466

## 1 INTRODUCTION

Many ongoing smart city solutions are designed to address specific issues/problems, e.g., improvement of traffic, surveillance, smart lighting etc., thus leading to passive, siloed solutions which do not interact with other solutions. Even within the same application domain, solutions from different vendors do not interoperate. While such an approach results in faster deployments for specific use cases, it leads to un-usability of data across different solutions and vendor stacks. Inspired by UIDAI-Aadhar[1] hourglass architecture, we advocated[9] orchestrating a city-wide, modular, inter-connected digital layer that promotes data sharing and inter-operability. The neck of the hourglass is the key middleware which, if chosen carefully and minimally, will not only break silos but also enable a whole host of new cross-domain smart city applications.

It has been well recognized that data/information connectivity is the key to smart-city intelligence and a key ingredient is the concept of linked data[2]. Simply speaking this implies that there should be some additional data about the data, also referred to as meta-data, which allows for a richer and more meaningful interpretation of data. For example, a temperature sensor may provide temperature readings but annotating it with information like location, manufacturer id, time-stamps, units etc. will make it much more useful. While easy to conceptualize, a key challenge is the lack of agreed upon standards and requirements to provide meta-information. In this work, we elaborate on the key architectural constructs of resource catalog, which is used to provide this additional meta-information. We also present a framework to develop data schemas which can be used for IoT devices to provide meta-data that will enable effective use of the device data.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
*BuildSys'17, November 8–9, 2017, Delft, The Netherlands*
© 2017 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-5544-5/17/11.
https://doi.org/10.1145/3137133.3141466

## 2 RESOURCE CATALOG

The resource catalog is one of the key components of the smart city middleware proposed in [9]. It contains the list of resources and their descriptions, including API endpoints, and other meta-data like access hints, ownership, providers, list of parameters and their descriptions etc. A useful analogy is the online shopping catalog, where a consumer can browse through available products, their features, reviews etc. and then decide to purchase a subscription or a copy of them. The resource catalog plays a similar role for the clients wishing to use the smart city resources and, as additional benefit, it can help nucleate the development of a new data/digital economy.

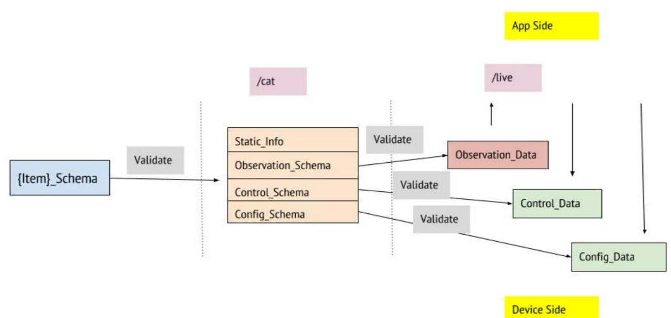


**Figure 1: Schema Validation Framework**

The current catalog implementation is accessible via REST APIs and stores the resource descriptions in JSON format which is both machine as well as human readable. Hypercat, a recent JSON based standard for catalog format[8], stores the meta-information as a bunch of RDF-like triples. Although our catalog uses JSON format, we do not use RDF-like triples and instead use the JSON schema framework[3]. Like Hypercat, the catalog supports search and discovery of various resources.

## 3 CATALOG AND DEVICE DATA SCHEMAS

JSON schemas are used to define the structure of each entry in the resource catalog. The schemas define specific fields to be included in the meta-data for a given class of devices. Apart from mandating the inclusion of certain fields, there exist provisions to include (optional) device/vendor specified fields, e.g., links to reference

```
{
  "type": "object",
  "properties": {
      "dataSamplingInstant": {
        "type": "number",
        "description": "Sampling Time in EPOCH format",
        "units": "seconds",
        "permissions": "read",
        "accessModifier": "public"
      },
      "caseTemperature": {
        "type": "number",
        "description": "Temperature of the device casing",
        "units": "degreeCelsius",
        "permissions": "read",
        "accessModifier": "public"
      },
      "powerConsumption": {
        "type": "number",
        "description": "Power consumption of the device",
        "units": "watts",
        "permissions": "read",
        "accessModifier": "public"
      },
      "luxOutput": {
        "type": "number",
        "description": "lux output of LED measured at LED",
        "units": "lux",
        "permissions": "read",
        "accessModifier": "public"
      },
      "ambientLux": {
        "type": "number",
        "description": "lux value of ambient",
        "units": "lux",
        "permissions": "read",
        "accessModifier": "public"
      }
  },
  "additionalProperties": false
}
```

**Figure 2: Example observation schema**


**Figure 3: Street Light Observation and Control Data**

ontologies, additional device information, usage hints etc., which enhances the usability of data by third party applications. The schemas are further used to validate the catalog entries at the time of on-boarding the device using the easily available json-schema validation tools.

Towards defining the structure of catalog items, we propose to organize the meta-data into following categories (see Figure 1): (a) static info: general information about the device, e.g., geo-location, type of device, ID, tags etc., (b) Observation schema: Meta-information for the observations made by the device, e.g., sensed parameters etc. (c) Control schema: Meta-information for the actuation parameters accepted by the device, e.g., control inputs etc. (d) Configuration schema: Meta-information for the configuration parameters of the device, e.g., sampling rates, sleep times etc.

A key idea is to store observation/control/configuration information within a catalog entry as a JSON-schema itself. Thus, the meta-data becomes an actionable specification that can be used to dynamically validate the data being sent from/to the device (see Figure 1).

We refer the reader to examples of two specific devices: streetlight ("ex_streetlight_item.json") and an electric meter[4]. The structure for streetlight is defined by the schema "generic_iotdevice _schema.json". Similarly, for electric meter the structure is specified by "electric_meter_schema.json". Here, in Figure 2, we give an example of an observation schema for streetlight. Further, Figure 3 shows the observation (up-arrow) and control (down arrow) data packets for the same device. An important point is that, the observation schema(Figure 2) , which is stored in catalog, can be directly used to validate the incoming observation packet (shown in Figure 3). The same applies to the control packets in the reverse direction.

From the examples, we also note that a field is provided to capture information about the data-serialization formats used by the device. In bandwidth constrained scenarios, e.g., LPWAN networks[5], JSON is not an efficient format. It may be required to use some efficient serialization formats, e.g., protocol-buffers[10], to send/receive information from the device. A key point is that, using above information, one can easily orchestrate data flows that hide these complexities. That is, a user may continue to interact with the device (through middleware) using JSON whereas the middleware/gateways communicate the devices using their preferred serialization formats.

## 4 CONCLUSIONS

In this work, we proposed a JSON-schema based framework for specifying meta-information about various data sources in a smart city deployment. Such a framework is critical to establish data connectivity and interoperability which are the most important factors for developing smart cross-domain applications. The proposed framework is flexible, extensible and also has readily available tools. Hence, it can serve as a starting point to develop standardised schemas for various classes of IoT devices using already existing data-models from ongoing standardisation efforts[6],[7].

## REFERENCES

[1] [n. d.]. https://www.slideshare.net/AmitRanjan/aadhaar-technology-architecture.
[2] [n. d.]. http://linkeddata.org/.
[3] [n. d.]. http://json-schema.org.
[4] [n. d.]. https://github.com/rbccps-iisc/smart_cities_schemas.
[5] [n. d.]. https://www.link-labs.com/blog/past-present-future-lpwan.
[6] [n. d.]. https://openconnectivity.org/.
[7] [n. d.]. https://www.fiware.org/.
[8] M. Blackstock and R. Lea. 2014. IoT interoperability: A hub-based approach. In *2014 International Conference on the Internet of Things (IOT)*. 79–84. https://doi.org/10.1109/IOT.2014.7030119
[9] Bharadwaj Amrutur et al. 2017. Poster Abstract: An Open Smart City IoT Test Bed *(IoTDI '17)*. ACM, New York, NY, USA, 323–324. https://doi.org/10.1145/3054977.3057303
[10] S. Popić, D. Pezer, B. Mrazovac, and N. Teslić. 2016. Performance evaluation of using Protocol Buffers in the Internet of Things communication. In *International Conference on Smart Systems and Technologies (SST)*. 261–265. https://doi.org/10.1109/SST.2016.7765670