

# Decentralized Linked Open Data in Constrained Wireless Sensor Networks

Bart Moons<sup>1</sup>, Flor Sanders, Thijs Paelman, Jeroen Hoebeke<sup>2</sup>

University of Ghent - imec, IDLab, Department of Applied Engineering

<sup>1</sup>bamoons.moons@ugent.be, <sup>2</sup>jeroen.hoebeke@ugent.be

Ghent, Belgium

**Abstract**—Data generated by sensors in Internet of Things ecosystems contains lots of valuable information, which is often not used to its full potential. This is mainly due to the fact that data is stored in proprietary storages and formats. Manufacturers of sensor devices often offer closed platforms to view and manage the data, which limits their reusability. Moreover, questions start to raise about true data ownership over data generated from monitoring our everyday lives. In order to overcome these issues several initiatives have emerged in the past to hand over data to the rightful owner. One of these initiatives is Solid, currently focusing on socially linked data. However, never before did one apply the Solid principles to Internet of Things data. Therefore, in this paper, a novel approach is presented where sensor data is handled from sensor to storage using open data formats and standards to ensure interoperability and reusability. It is shown that combining existing concepts can be helpful in designing decentralized Internet of Things data storages, on top of which data can be incorporated into the Linked Open Data cloud. This has been done by comparing the overhead of a regular approach, using Linked Open Data concepts on top of a sensor device, to an approach that was optimized for device management in constrained Internet of Things networks.

## I. INTRODUCTION

Many applications and innovations arose from the fact that an excessive amount of information is openly available due to the open vision Tim Berners-Lee had about the world wide web. Recent advancements within the area of the Internet of Things (IoT), will make even more information available. As governments, individuals and companies are rolling out sensor networks to manage their daily tasks more easily, even more data is added to the ever expanding datasphere that is predicted to grow from 33 Zettabytes (ZB) in 2018 to 175 ZB by 2025 [1]. The real time collected data from the wide spectrum of physical sources that defines the IoT, might be compared by Deep Learning algorithms to historically collected data and therefore holds even more information. However, different data sources should be easy to integrate in order to unlock new scientific insights and to empower innovations. A fairly old concept for doing so, which gained popularity in academia to ensure free access to academic data, is the concept of Open Data. In order to do so, data should be linked to other data in a Linked Open Data Platform (LDP) to provide context [2].

However, currently data is stored by application providers, which led to -after several scandals regarding privacy- innova-

tions in the field of data decentralization, with Solid (Socially Linked Data) as the leader. Solid enables data decentralization and reusability using open standards and protocols, following the 5-star Linked Open Data concept [3]. Nevertheless, as Solid's full form suggests, the project mainly focuses on social web applications, where users can easily control where they store and who they share their personal data with. A similar approach should be maintained for easy integration of services and development of the Internet of Things data cloud.

Currently, closed source products are delivered to the IoT consumer market such as smart thermostats, smart lightning and smart plugs where data can be accessed from a central platform, offered by manufacturers. Its use of proprietary based approaches results in interoperability issues between different end-points and limits data reusability. Moreover, manufacturers gain more insight in our everyday lives, without offering owners of data complete control over it. In order to diverge from vertical data silo's and to provide data openness, open standards and decentralized storage is required.

Therefore, in this paper a solution is presented to store data from sensor networks in a decentralized way by making use of well-known technologies and protocols. Our analysis shows that a naive approach of the Linked Open Data concepts would increase the energy consumption of wireless devices to a great extent and thus requires a more suitable ontology at the sensor side. We show that by choosing the right protocol for constrained networks, device management can be performed too, while decreasing the battery drain. This, however, requires a mapping from the wielded light-weight encoding schemes to any format supported by LDPs.

## II. RELATED WORK

Incorporating Internet of Things data to the Linked Open data is no novelty. The authors in [4] address the interoperability issues of the Internet of Everything, whereby proposing the idea of "Linking Everything". Their solution is based upon the 5-star Linked Open Data concept, replacing the Hyper Text Transfer Protocol (HTTP) by the Constrained Application Protocol (CoAP). Apart from their vision and enabling technologies, they mainly focus how real-time data can be queried from traditional data platforms such as Apache Jena, Virtuoso and Stardog. Similarly, in [5] a HTTP-CoAP proxy

is implemented to translate CoAP messages to HTTP requests. Additionally, new content-format types were introduced (i.e. `application/ld+json` and `text/turtle`) to support Linked Data Platform (LDP) features. Both articles assume that devices can be queried directly without constrained links, being exactly a fundamental property of Low Power Wide Area Networks (LPWANs) frequently used for connecting constrained IoT devices. Other work considers the Java Data Objects (JDO) standard as a data model for a resource framework incorporated in a generic, uniform data management platform [6]. The authors present a homogeneous service layer as an interface to heterogeneous data sources. For this purpose, they defined a novel API which can be used for data validation and resources handling, thereby centralizing and unifying the data model from a broad range of data sources. Furthermore, a transparent proxy is provided, which can deliver data to any SPARQL (SPARQL Protocol and RDF Query Language) endpoint. However, none of these solutions have incorporated data in a decentralized platform where the data producer can provide access control to the application of his choice.

### III. BACKGROUND TECHNOLOGIES

#### A. *Solid*

Solid provides a platform for decentralized data storage which allows users to easily switch between application providers without having to reorganize their data. Solid does this by offering a platform where each user can store their (semantically interoperable) data in an online storage space (called a pod). Data is exchanged using HTTP and stored using the Resource Description Framework or RDF, which enables self-describing, linkable, yet independent, data management mechanisms. RDF consists of graphs - formed out of self-descriptive subject-predicate-object triples - and datasets, used to organize collections of RDF graphs [7]. In order to structure the data to express a subject area and how their different properties are related, the use of an ontology is required. It is important for applications that their ontologies can be accessed by others so that data can be reused by groups of potential consumers. Ontology sharing is the first step towards semantic interoperability on a larger scale. A next step includes ontology mapping and matching, where resources from different ontologies can be matched with others. One of the few mapping tools which allows to map heterogeneous sources into RDF, is the RDF Mapping Language (RML), which provides a way to express custom mapping rules from heterogeneous data structures to the RDF data model.

Finally, different serialization syntaxes exist for storing and exchanging RDF such as Turtle (Terse RDF Triple Language), XML (eXtensible Markup Language) and JSON-LD (JavaScript Object Notation-Linked Data).

#### B. *Web of Things*

As the Web became the largest connected information system through the use of standardized, resource oriented protocols, the Internet of Things is aiming for a similar approach to become a true part of the Web ecosystem. Several initiatives

have emerged in the past years to make sensor devices available through Representational State Transfer (REST) operations [8] [9]. The need for a lightweight protocol became clear and the Constrained Application Protocol (CoAP) emerged as a lightweight version of the HTTP. This internet protocol shares the true vision of a Web of Things, where each sensor has a URI and can be treated like any other Web resource [10].

On top of the CoAP protocol, the Light-Weight Machine to Machine (LwM2M) specification from the Open Mobile Alliance (OMA) was developed in order to provide a generic API to perform device bootstrapping, registration and management. Besides these interfaces, an object model is defined to represent collections of mandatory and optional resources that offer semantics [11]. The provided object models are structured using pre-defined URI-templates, known to both client and server. An object can consist of several instances, which at their turn can consist of several resources. These resources can then be accessed using REST operations.

Apart from the CoAP, several other protocols for the IoT have emerged in the last couple of years. A popular approach for collecting sensor data is the publish-subscribe paradigm, where data is distributed asynchronously by means of interest. A publisher will categorize its messages using topics, to which subscribers can express interest. Published messages will end up at the subscriber, only if these are of interest. The most popular implementation of the publish/subscribe messaging pattern for Internet of Things applications is the Message Queue Telemetry Transport or MQTT. Another implementation; AMQP or Advanced Message Queuing Protocol, implements more advanced features and has therefore more overhead than MQTT.

#### C. *Wireless Technologies*

In this paper only wireless sensor network technologies are considered that are constrained in nature. Many of such technologies make use of the Industrial, Scientific and Medical (ISM) 915-MHz band (Region 2) or the license-free European Short-Range Device (SRD) 863870-MHz band. In this area, a multitude of technologies exist to cover different properties and use cases and can be divided roughly in two groups:

1) *Long Range Technologies*: Long range technologies in the unlicensed spectrum (ISM band) include LoRa (Long Range), Sigfox and Weightless among others. The Third Generation Partnership Project (3GPP) also came up with NB-IoT (Narrow-Band IoT) and long-term evolution machine type communications or LTE-M, for lower latency and higher bandwidth, in the licensed spectrum [12]. These technologies aim at sensor devices consuming very little energy, while covering a very long range up to 50 km.

2) *Short Range Technologies*: On the contrary, short range technologies aim at a higher data rate while covering a shorter range and still consuming as little power as possible. A non-exhaustive list includes DASH-7, Bluetooth Low Energy (BLE) and IEEE 802.15.4, which also forms the basis for protocols such as Zigbee and WirelessHART [13].

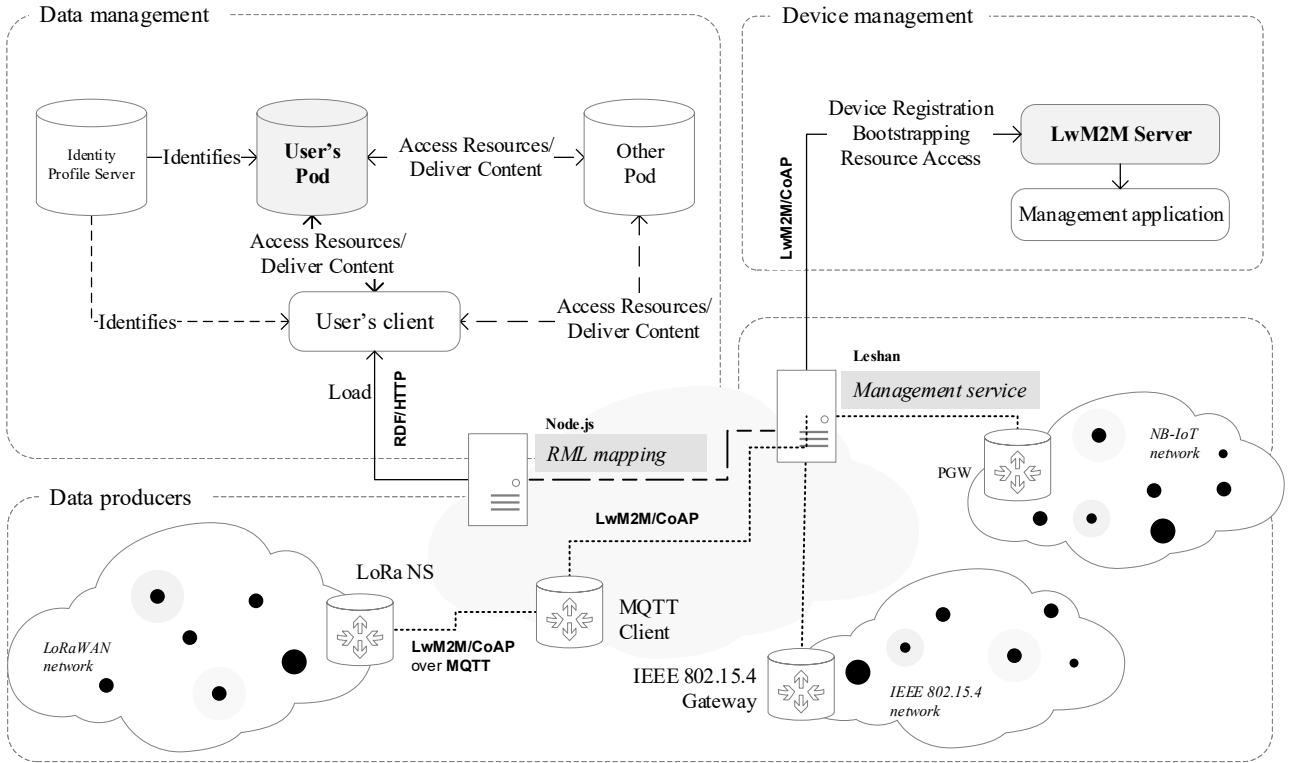


Fig. 1. Implemented system overview

TABLE I  
NETWORK TECHNOLOGY OVERVIEW

Technology		MTU (B)	Range (m)
LoRa	SF7 4/5 125kHz	242	2.575
	SF12 4/5 125kHz	51	5.350
Sigfox	Uplink	12	8.000
BLE	Native	244	150
IEEE 802.15.4	2-GFSK	127	860
NB-IoT	Uplink	125	2.000

An overview of the Maximum Transmission Unit (MTU) and the theoretical range for the given technologies are outlined in Table I.

As some of these technologies were used in the evaluation of this work, a brief overview of their presence in the network architecture is given in the next Section.

#### IV. ARCHITECTURE

##### A. System Overview

In order to connect the constrained Internet of Things to the Linked Open Data cloud, several interfaces are required. First of all, the produced data will be transmitted over constrained wireless links to a receiving gateway. The data producers,

depicted at the bottom of Figure 1, make use of LwM2M, a communication protocol that can fit every network and offering both device management and data transfer services. The generated IP packets are sent over MQTT from the LoRaWAN network server and IP for IEEE 802.15.4 and are distributed to a LwM2M management server, where after the registration of a device a data flow can be established. Finally, packets of interest (i.e. packets containing sensor measurements) are filtered and forwarded to the Solid pod. Other packets, such as bootstrap and registration requests are ignored. As Solid makes use of a Web Access Control (WAC) mechanism for decentralized cross-domain authorization, multiple authorized applications and parties can be allowed to read, write, append and control the data. This makes it possible not only to have multiple clients writing data to the same pod, but also for applications to gather and combine data from different pods in a decentralized manner, with the possibility of opening up the data to the Linked Open Data Cloud [3].

##### B. Implementation

As described in the previous Section device management and data collection from several networks, such as NB-IoT, LoRa and IEEE 802.15.4, can be performed by means of a LwM2M server. For this purpose, Leshan, an open-source OMA LwM2M server implementation is set up (illustrated in the upper right corner of Figure 1). In order to pass every

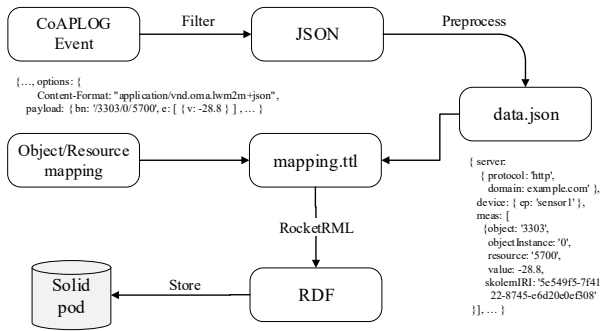


Fig. 2. Internal working of the translation mechanism

received message from Leshan to a supplementary data processor, Leshan exposes a RESTful communication API between the front- and back-end. The LwM2M/CoAP messages of interest are forwarded to a Node.js application, where an RML script translates them to an RDF graph, required to store them in the Solid data pod. The RDF graph consist of nodes uniquely identified by a URI for which relations are defined using predicates. Every node in the graph represents either a device, a sensor instance or a single measurement. Once the graph with the updated data is produced by the RML script, the measurements are sent to the Solid pod. Existing graphs will merge fluently since every node with an existing URI is considered the same and therefore only new elements are stored. An overview of the translation flow is given in Figure 2.

As Section III-A suggests, the description of RDF data requires an ontology, which acts as a lexicon for the data model. As a result of a study for the European Commission a basis of an ontology containing the OMA LwM2M data structure is publicly available [14]. However, in order to store historical sensor data, various adaptations were required to this ontology<sup>1</sup>. Firstly, following the OMA LwM2M registry, the mandatory objects and their resources were completed and predicates were added to represent the relations between them. Next, a device object was added to represent the OMA LwM2M client and its exposed objects. Finally, the `hasTimestamp` predicate was added to the `SensorValue` resource enabling the storage of historical data in a simple manner.

## V. EVALUATION

One of the main requirements for low power sensors is to operate on a single battery charge for multiple years and therefore consume as little energy as possible. Since wireless communication modules tend to have the highest energy consumption in IoT sensor devices, the communication overhead should be kept as small as possible. This Section will evaluate the aforementioned encodings and ontologies in order to select the most appropriate solution to initiate IoT sensor devices into the semantic web of things. Previous work around this

<sup>1</sup><https://iotsolidigent.inrupt.net/public/ontologies/omalwm2m.owl.ttl>

topic introduced additional content-format types in order to incorporate these sensors directly in the Web as Linked Data providers [5]. The authors released their implementation as open source<sup>2</sup>, which was used during the comparison of the different ontologies and encodings.

In order to decide which ontology suits constrained networks the best, the overhead of a CoAP GET request to the memory availability of the device and its corresponding response is evaluated for the Linked Data content-format types `application/ld+json`, `text/turtle` and the LwM2M `text/plain`, `application/vnd.oma.lwm2m+json` and `application/vnd.oma.lwm2m+tlv` data types.

### A. CoAP CON request

Due to the request-response nature of CoAP, every response is preceded by a request from a client. First, the linked data request is evaluated and can be seen in Listing 1.

```
CON, MID:38929, GET, TKN:61 62 63 64 65 66
67 68, /sensors/memory
```

Listing 1. CoAP JSON-LD GET request

On the contrary, the request in Listing 2 shows how the same resource can be queried using the predefined LwM2M URI-templates.

```
CON, MID:35534, GET, TKN:ee 10 01 6d 28 3b
79 6d, /3/0/21
```

Listing 2. CoAP LwM2M GET request

It can be seen from Figure 3 that LwM2M provides a higher efficiency in CoAP CON requests due to the object model known to both server and device. Even with UDP and IPv4 overhead, a LwM2M client will not surpass the fragmentation limit of LoRa using Spreading Factor (SF) 12, which would require multiple packets and consequently consume a lot more energy.

### B. CoAP response

In response to the GET request, the server will reply with the value of the queried resource. Listing 3 shows the payload of a response in the `application/ld+json` data format.

```
{ "@id" :
  "coap://192.168.2.16:5683/sensors/
  memory",
  "@type" :
  ["http://www.w3.org/ns/ldp#RDFSsource",
   "http://purl.oclc.org/NET/ssnx/ssn
   #SensorOutput"],
   "http://purl.oclc.org/NET/ssnx/ssn
   #hasValue" : [{
     "@type" :
     "http://www.w3.org/2001/XMLSchema
     #double",
     "@value" : "253952"
   }],
}
```

<sup>2</sup><https://github.com/sisinflab-swot/ldp-coap-framework>

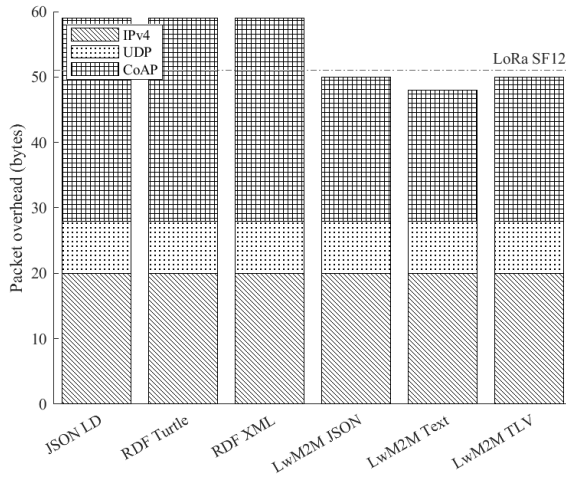


Fig. 3. GET request performed from client to server with different ontologies and encodings

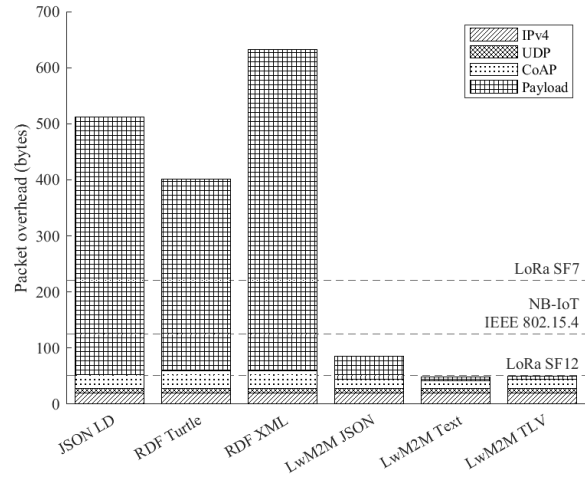


Fig. 4. Response to a GET request performed from server to client with different ontologies and encodings

```
"http://purl.org/dc/terms/created" : [{
  "@type" :
    "http://www.w3.org/2001/XMLSchema
    #dateTime",
  "@value" :
    "2020-06-16T09:59:14.141+02:00"
}]}
```

Listing 3. Payload of a JSON-LD response to a CoAP GET request

Listing 4 clearly shows that this is in sheer contrast with the payload size of a LwM2M request encoded as application/vnd.oma.lwm2m+json.

```
{"bn":"/3/0/21","e":[{"n":"","v":253952}]}
```

Listing 4. Payload of a LwM2M JSON response to a CoAP GET request

Figure 4 illustrates how the different encodings and ontologies relate to each other in response to a GET request. The fragmentation lines of different technologies are shown on the right and again indicate how LwM2M limits the number of packets required for a single response.

It can be seen from both figures and evaluations of the different ontologies and encodings that low power sensor devices can greatly benefit from the wielded approach where low power IoT ontologies are mapped using RML instead of using semantic web ontologies directly on top of IoT protocols. To further elaborate on this, the battery lifetime of a LoRa SF12 device that responds every hour with a new measurement to an initial OBSERVE request in different encoding schemes is shown in Figure 5. This clearly indicates how a light-weight encoding schema can decrease the energy consumption by a factor of almost 10.

### C. Performance Evaluation

As Solid works on a per file basis, sensor data is currently stored in a single file per sensor. Every measurement will issue a PATCH request to append the last measurement to the file. In order to lay out the feasibility of the applied approach, we

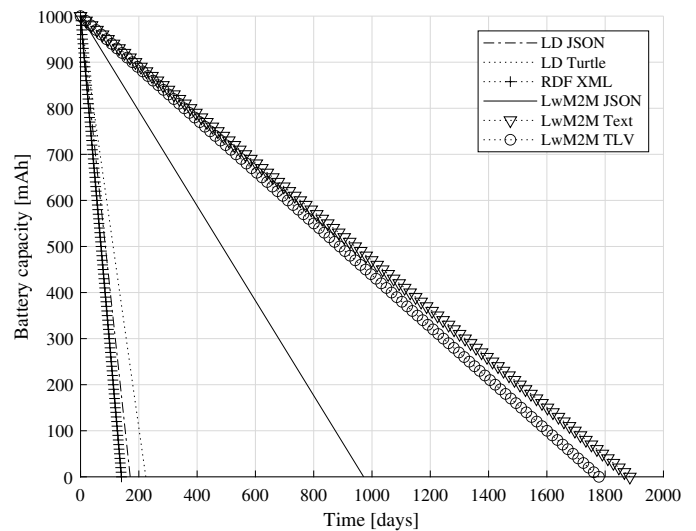


Fig. 5. Battery drain of a LoRa SF12 communicating with the server every hour, without self-discharge

evaluated this property of current Solid implementations. A log of delays between the PATCH requests sent by the RML mapper and the 200 OK response from the server is visualized in Figure 6 and Figure 7.

The Solid Community Server shows a linear relation between the file size of the total database and the loading time, with an increase of  $33.74 \frac{ms}{kB}$ . A file size of a mere 600 kB corresponds to a delay of almost twenty seconds. The Inrupt server shows a less consistent, however very similar trend, with a delay trend-line that increases with  $18.85 \frac{ms}{kB}$ . These tests, however, were not carried out on a self-hosted Solid server and therefore requires further analysis to pin down the true cause of these delays.

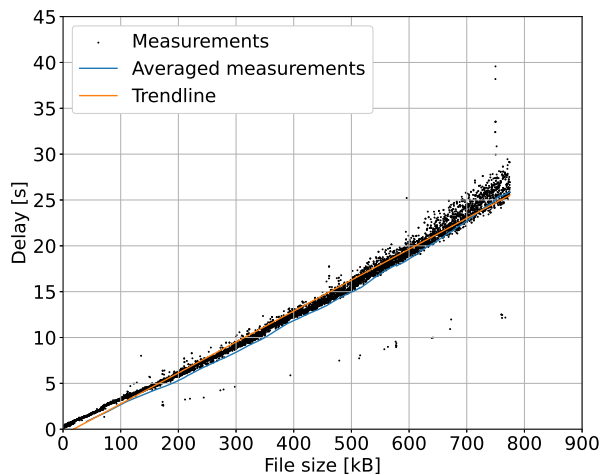


Fig. 6. Solid Community Server PATCH request delay

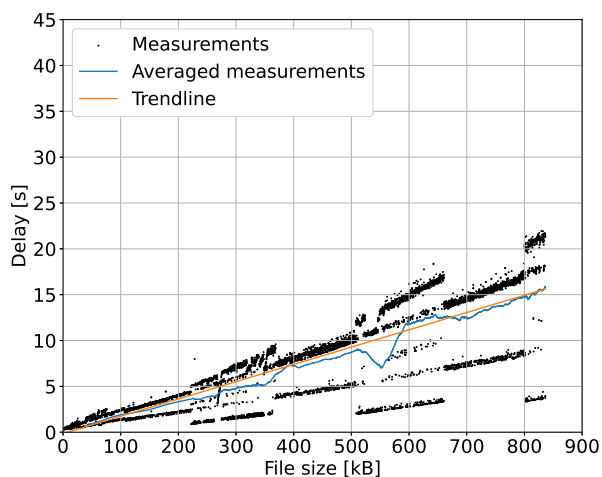


Fig. 7. Inrupt Server PATCH request delay

## VI. CONCLUSION

As current data storage lacks in true data ownership and data openness, novel principles are being developed, with Solid as the leader. However, as Solid focuses on Socially Linked Data, IoT was not considered before. Therefore, this paper showed that it is possible to use a standard based approach over constrained wireless links to enable the concepts of Linked Open Data in the Web of Things. We showed that using our implementation less packets, and consequently less energy, is required. By using RML as an ontology matching tool, we showed that it is possible to couple a set of constrained IoT devices using the LwM2M management protocol to an RDF-based Solid pod.

## VII. FUTURE WORK

Future work may include the elimination of the Node.js server in its entirety. In order to do so, the OMA LwM2M server implementation should be extended to enable external data processors to subscribe to event- or data streams by

the means of a standardized API. On the Solid side this would require the possibility of running data processors as a middleware service. Furthermore, as the use of an ontology corresponding specifically to the OMA LwM2M data structure made the mapping as straight-forward as possible, a more maintainable solution would consist of a mapping to an existing general-purpose ontology for sensor networks and smart applications such as the Smart Appliances REFERENCE (SAREF) ontology or the Semantic Sensor Network (SSN) ontology [15]. Finally, no optimizations for time series data are provided by Solid and as shown in Section V, growing file sizes increase delays for requests largely. Future work should analyze different aggregation models and which model suits different use-cases. Storing measurements of sensors reporting every 10 minutes can be outrageous, however only storing aggregated data can limit the possibilities of other applications.

## REFERENCES

- [1] D. Reinsel, J. Gantz, and J. Rydning, "The Digitization of the World from Edge to Core," p. 28, 2018.
- [2] "5-star Open Data," library Catalog: 5stardata.info. [Online]. Available: <http://5stardata.info/en/>
- [3] A. V. Sambra, E. Mansour, S. Hawke, M. Zereba, N. Greco, A. Ghanem, D. Zagidulin, A. Aboulnaga, and T. Berners-Lee, "Solid: A Platform for Decentralized Social Applications Based on Linked Data," p. 16.
- [4] D. Le-Phuoc and M. Hauswirth, "Linked Data for Internet of Everything," in *Integration, Interconnection, and Interoperability of IoT Systems*, R. Gravina, C. E. Palau, M. Manso, A. Liotta, and G. Fortino, Eds. Cham: Springer International Publishing, 2018, pp. 129–148, series Title: Internet of Things. [Online]. Available: [http://link.springer.com/10.1007/978-3-319-61300-0\\_7](http://link.springer.com/10.1007/978-3-319-61300-0_7)
- [5] G. Loseto, S. Ieva, F. Gramegna, M. Ruta, F. Scioscia, and E. D. Sciascio, "Linked Data (in resourceless) Platforms: a mapping for Constrained Application Protocol," in *The Semantic Web - ISWC 2016*. Springer International Publishing, 2016.
- [6] J. Pullmann and Y. Mohamad, "Linked Data Services for Internet of Things," in *Proceedings of the 2015 International Conference on Recent Advances in Computer Systems*. Hail, Saudi Arabia: Atlantis Press, 2016. [Online]. Available: <http://www.atlantis-press.com/php/paper-details.php?id=25847799>
- [7] "RDF 1.1 Concepts and Abstract Syntax." [Online]. Available: <https://www.w3.org/TR/rdf11-concepts/>
- [8] E. Wilde, "Putting Things to REST," in *UC Berkeley: School of Information*, 2007, p. 13.
- [9] T. Luckenbach, P. Gober, S. Arbanowski, A. Kotsopoulos, and K. Kim, "TinyREST a Protocol for Integrating Sensor Networks into the Internet," p. 6.
- [10] Z. Shelby, K. Hartke, and C. Bormann, "The Constrained Application Protocol (CoAP)," RFC Editor, Tech. Rep. RFC7252, Jun. 2014. [Online]. Available: <https://www.rfc-editor.org/info/rfc7252>
- [11] O. M. Alliance, "Lightweight Machine to Machine Technical Specification: Core v1.1," Aug. 2018. [Online]. Available: [http://www.openmobilealliance.org/release/LightweightM2M/V1\\_1-20180710-A/OA-TS-LightweightM2M\\_Core-V1\\_1-20180710-A.pdf](http://www.openmobilealliance.org/release/LightweightM2M/V1_1-20180710-A/OA-TS-LightweightM2M_Core-V1_1-20180710-A.pdf)
- [12] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low Power Wide Area Networks: An Overview," *arXiv:1606.07360 [cs]*, Jun. 2016, arXiv: 1606.07360. [Online]. Available: <http://arxiv.org/abs/1606.07360>
- [13] C. Buratti, A. Conti, D. Dardari, and R. Verdone, "An Overview on Wireless Sensor Networks Technology and Evolution," *Sensors*, vol. 9, no. 9, pp. 6869–6896, Aug. 2009. [Online]. Available: <http://www.mdpi.com/1424-8220/9/9/6869>
- [14] L. Daniele, F. den Hartog, and J. Roes, "Study on semantic assets for smart appliances interoperability : D-s4: Final report," 2015.
- [15] A. Haller, K. Janowicz, S. Cox, M. Lefrançois, K. Taylor, D. Phuoc, J. Lieberman, R. Garca Castro, R. Atkinson, and C. Stadler, "The modular ssn ontology: A joint w3c and ogc standard specifying the semantics of sensors, observations, sampling, and actuation?" *Semantic Web*, vol. 10, 08 2018.