A Self-managed Marketplace for Sharing IoT Sensors

Anas Dawod, Dimitrios Georgakopoulos, Prem Prakash Jayaraman
Department of Computer Science and Software Engineering
Swinburne University of Technology
Melbourne, Australia
{adawod, dgeorgakopoulos, pjayaraman}@swin.edu.au

Panos K. Chrysanthis
Department of Computer Science
University of Pittsburgh
Pittsburgh, PA, USA
panos@cs.pitt.edu

2023 IEEE 39th International Conference on Data Engineering Workshops (ICDEW) | 979-8-3503-2244-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICDEW58674.2023.00024

***Abstract*— Internet of Things (IoT) applications can significantly reduce the cost and timeframe of providing their benefits by taking advantage of the billions of existing IoT sensors and other IoT devices that have been deployed by others in IoT. This paper proposes a self-managed Sensor Sharing Marketplace (SenShaMart) that allows IoT applications to discover, integrate, and pay for the use of sensors that are managed by different parties. In addition to being a self-managed IoT marketplace, SenShaMart is fully autonomic and cannot be controlled by any specific party. IoT applications and IoT sensor providers interact via SenShaMart provided services for semantic description of IoT sensors and their data, semantic query processing, automatic integration of sensors and their data, and IoT sensor payment transactions that control the access to the flow of sensor data according to the sensor payment terms. Self-management and full autonomy are achieved via a specialized SenShaMart blockchain that keeps and manages all data needed by the SenShaMart services that encapsulate it. The paper presents the SenShaMart architecture and service organization, its specialized blockchain, as well as its current performance characteristics and future research.**

Keywords*— *Blockchain, IoT sensor discovery, IoT sensor integration, IoT sensor payment, Self-management

I. INTRODUCTION

Internet of Things (IoT) combines billions of sensors and other devices that can communicate via the internet (we refer to these as IoT sensors) and support the development of IoT applications that provide novel IoT services and/or products [1]. The IoT sensors (e.g., wearables, smart-phones, industrial machines, RFIDs, etc.) sense the physical world and send observation data (which we refer to as IoT data) to IoT applications that run in the cloud, on edge computers, and/or the IoT devices that host the sensors [2]. IoT sensors are currently owned by many different individuals or organizations who deploy and utilize them for their own purposes. Tens of billions of IoT sensor and other IoT devices are currently connected to the Internet and major industry players project that their number will reach anywhere between 25 to 125 billion in 2030 [3]. The vast number of IoT sensors provides an exceptional capability to observe the physical world and distil high-value information enabling solutions to major challenges that were hard to solve before due to lack of timely and accurate information. However, the potential of IoT is not currently fully realised, as IoT applications currently operate in silo, lacking the ability to use and share the costs of sensors that can be shared by other parties (which we refer to as sensor providers) [4]. Therefore, currently IoT application must incur the cost and effort needed to procure, deploy, and use their own sensors.

As an example, consider the negative impact of climate change in agriculture. This can be mitigated by 1) using IoT sensors that provide the information needed to determine how various plants perform under changing environmental conditions across Australia and the world [5], and 2) planting crops consisting of species of plants that tolerate best the challenging conditions (e.g., increased drought, annual solar radiation, soil deterioration, locust, etc.) at each region or farm. However, the procurement, deployment, and maintenance of IoT sensors that are needed to monitor micro-climate, soil humidity, solar radiation and crop performance are difficult to scale up and incredibly expensive. If a sensor sharing platform or marketplace exists, it will allow the sharing of existing IoT sensors that have been deployed by farmers and agribusinesses with IoT applications to use them for collecting the data needed for climate change mitigation. This will minimize the effort, cost, and timeframe for responding to the effects of climate change.

Sharing existing IoT sensors is currently severely hindered by lack of solutions in several areas including the following: 1) IoT sensor discovery that involves formulating and querying sensor descriptions that are supplied by their providers; 2) insufficient standards, and limited use of such standards in describing existing sensors–for example, although semantic technology is proposed by IoT community for providing IoT sensor and data description [6], leading standards for semantic description of sensors and their data, such as Semantic Sensor Network (SSN) ontology [7] and Sensor Observation Sample and Actuator (SOSA) ontology [8], do not include sensor identifications or support mobile sensors; 3) IoT applications must have an unfettered right to discover any sensor offered by any provider and do that with minimal (ideally zero) mediation cost and delay; 4) scalable, reliable and dependent management of rapidly expanding volume and variety of IoT sensors; and 5) support for cost-sharing between sensor providers and IoT applications (which we refer to as sensor clients) via a pay-as-you-go model. These can only be achieved by a self-managed [9] marketplace for sharing IoT sensors that is fully autonomic and trusted so it can guarantee that no party can ever prevent, restrict, manipulate, or monopolize access to any available IoT sensor. Existing attempts to create such a Marketplace (e.g., [10]) are embryonic, which fail to achieve these aims.

To address these considerable challenges, we propose *Self-managed Sensor Sharing Marketplace* (SenShaMart) that includes the following novel contributions:

- A novel self-managed, autonomic, and decentralized Marketplace architecture that prevents any external

2473-3490/23/$31.00 ©2023 IEEE
DOI 10.1109/ICDEW58674.2023.00024

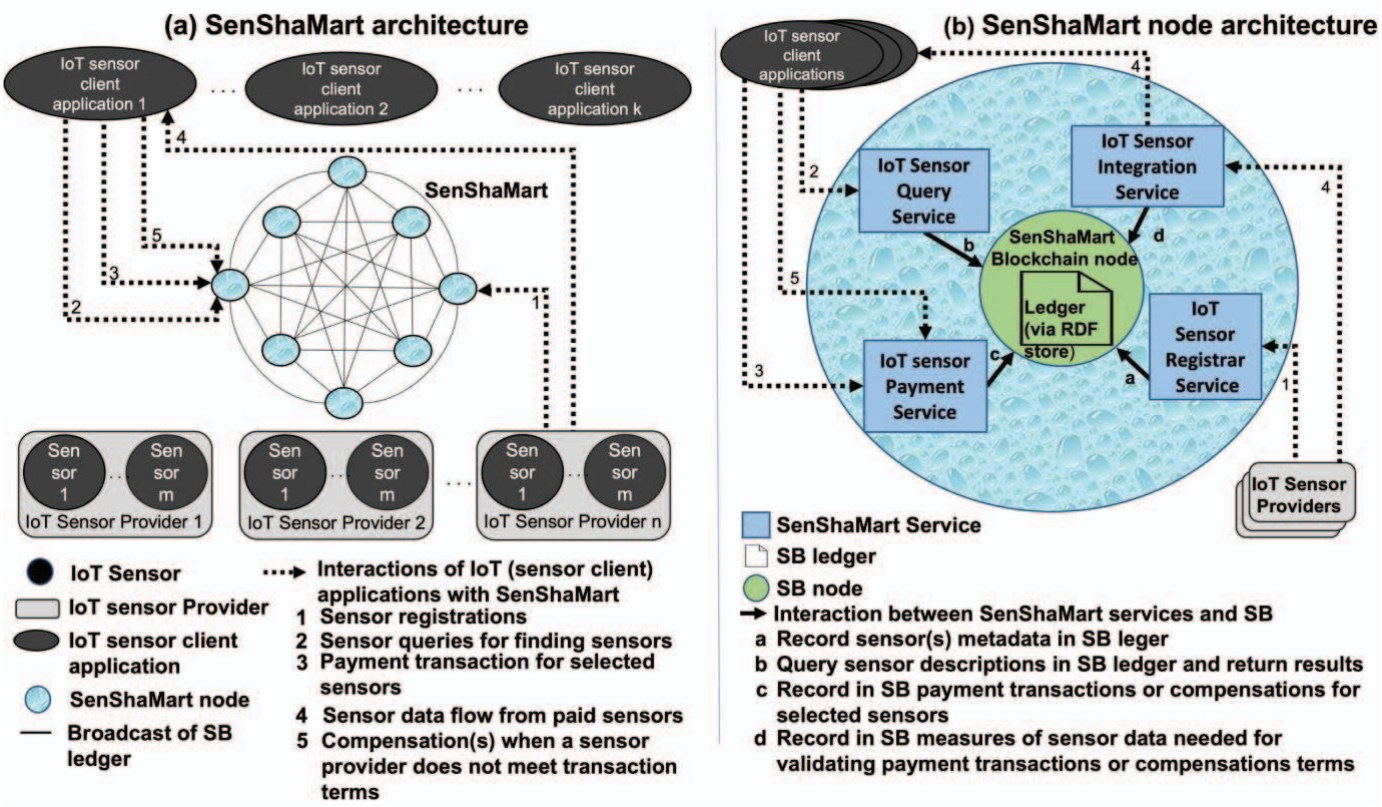

Fig. 1. SenShaMart high-level architecture (a) and node architecture (b)

entity from controlling the sharing (i.e., the discovery, integration, and payment) of IoT sensors.

- Enhanced semantic sensor descriptions using an extended SSN/SOSA ontology. To accommodate IoT sensors that have no extended SSN/SOSA-based description, we are developing sensor classifiers that generate extended SSN/SOSA-based sensor descriptions from their data streams.
- A specialized SenShaMart blockchain that maintains all sensor descriptions and other metadata that are necessary for IoT sensor discovery, integration, payment, and access control. SenShaMart does not store any sensor data, which is instead managed directly by the client IoT applications.
- Efficient semantic query processing of the IoT sensor metadata that is stored in the SenShaMart blockchain by incorporating a triple store in the blockchain nodes.
- Automatic integration of any IoT sensor by any client application via any existing IoT protocols, such as CoAP [11], MQTT [12], etc.

The remainder of this paper is organized as follows: Section II presents the architecture of SenShaMart. Section III describes the specialized SenShaMart blockchain. The sensor registration and query services are presented in Section IV, while Section V describes the sensor integration service. Section VI describes the payment transactions service. Section VII provides preliminary performance characteristics. Finally, Section VIII outlines future research directions and then concludes the paper.

II. Architecture of SenShaMart

As illustrated in Figure 1(a), SenShaMart is comprised of a collection of distributed SenShaMart nodes that interact via the *SenShaMart Blockchain* (SB) that stores all information required for registration, discovery, payment, and integration of IoT sensors.

The architecture of individual SenShaMart nodes is depicted in Figure 1(b). Each SenShaMart node is built around a corresponding SB node, and it is accessed only by the registration, query, payment and integration services of that SenShaMart node. The SB is a specialized blockchain because it is specifically devised to: 1) store (semantic) descriptions of all available IoT sensors and their data observations, 2) records the IoT sensor ID, the protocol used to communicate with IoT sensors (e.g., MQTT [12]), their network end point, and the payment terms for each IoT sensor, and 3) incorporates a triple-store enabling efficient processing of related semantic queries for sensor discovery. We use the term *IoT sensor metadata* to refer to all the above information for IoT sensors that is kept in the SB so we can distinguish it from the *data observations* produced by the IoT sensors, which are *not* stored in SB. The SB is discussed further in Section III.

As illustrated in Figure 1(b), each SenShaMart node includes IoT sensor registration, query, payment, and integration services which respectively allow 1) IoT sensor

providers to register their sensors, and 2) IoT (sensor client) applications to find, select, pay for, and integrate the IoT sensors they need. IoT sensor providers and IoT (sensor client) applications use SenShaMart services form different SenShaMart nodes, e.g., an IoT application can submit a sensor query in one node and pay for the sensors it selects via a different SenShaMart node. The SenShaMart-provided services are discussed further in Sections IV, V, and VI. The SB is presented next in Section III.

III. SENSHAMART BLOCKCHAIN

The SB is a self-managed registry of IoT sensor metadata that is needed to support autonomic IoT sensor discovery, query, payment, and integration services. Just like many other existing blockchains (e.g., the Bitcoin blockchain [13]), the SB allows SB nodes to generate new blocks, contribute to SB consensus, and verify newly generated blocks across the entire SB.

Unlike other existing blockchain-based solutions for IoT, SenShaMart uses SB to store only IoT sensor metadata that is required for IoT sensor description, registration, query, integration, payment, and access control. SenShaMart avoids blockchain-related bottlenecks by not storing any IoT sensor data in SB as this would have significantly reduce its scalability. Furthermore, the incorporation of a triple store in the nodes of SB further improves the efficiency of processing sensor metadata queries in SB.

Also, unlike other existing blockchains, SB provides the following novel features that support IoT sensor sharing via the SenShaMart services:

- A novel ledger (we refer to as the *SB ledger*) for storing IoT sensor metadata organized in the form of blocks, which we refer to as SB ledger blocks. The SB ledger uses an enhanced Semantic Sensor Network (SSN) [7]–based ontology for describing sensors and their data. This is discussed further in the following paragraphs.
- SB blocks that incorporate an RDF triple store that is used to record IoT sensor metadata as triples organized in SB ledger blocks. The SenShaMart's RDF triple store supports efficient processing of semantic queries involving IoT sensor metadata. This eliminates the need to extract the entire SB ledger and insert its data in an external triple store to process the queries that are submitted to the IoT sensor query service by IoT client applications, which considering the IoT-wide scope of SenShaMart would have been completely impractical.
- SB-based distributed data communications (via the SB ledger) between the SenShaMart nodes, which are the only clients of SB. When a new SB ledger block is created in a SenShaMart node, The SB node in this SenShaMart node broadcasts the newly generated SB leger block to all other SB nodes to make the SB ledger consistent across all SB nodes and corresponding SenShaMart nodes. Via this mechanism, the RDF triple store in each SB node is updated automatically with the triples metadata that contain the metadata of any newly registered IoT sensors stored in the new SB block. The broadcast mechanism used in SB is similar to that of other existing blockchains (e.g., Bitcoin [13]).
- The SB provides a unique self-managed, trusted, and not controlled by any party, semantic registry for IoT sensor metadata.

Blockchains that have been specifically developed for IoT (e.g., IOTA [15] and IoT Chain [16]) and other peer-to-peer systems (e.g., IPFS [14]) provide trusted storage for IoT sensor data and metadata. However, they are not specifically designed for managing IoT sensor descriptions and do not support semantic query processing for sensor metadata. For example, IOTA, IoT Chain, IPFS do not incorporate any ontology for semantic IoT sensor description or a blockchain-based triple store for efficient processing of semantic queries over IoT sensor metadata. Even assuming that an external IoT sensor description ontology is added to address the former limitation, the processing of semantic queries over the IoT sensor metadata will require extracting all IoT sensor metadata from these systems and inserting them in an external triple store that to enable processing of queries over the IoT sensor metadata they store. This solution is clearly very inefficient the used of the external triple store will make it untrustworthy.

In Section II we noted that SenShaMart (more specifically its SB ledger and its services) use and enhanced SSN-based ontology to semantic capture descriptions of IoT sensors and their data observations, which we refer to as the *SenShaMart ontology*. Beyond the ontology that is documented well in the SSN [7] and related SOSA [8] standards, the SenShaMart ontology incorporates the following extensions:

Sensor ownership: The SenShaMart ontology incorporates Public Keys (PKs) as identifiers for IoT sensor providers. This ensures anonymous and unique identifier for all IoT sensor providers. In addition, the SenShaMart ontology supports multi-level ownership of IoT sensors, e.g., a sensor can be owned by a department in organisation and managed by a few specific persons. Therefore, in the SenShaMart ontology the concept of sensor ownership includes multiple attributes (e.g., *provider organization, department*, and *person*), that permit multi-level ownership.

Sensor geospatial location: The SenShaMart ontology includes concepts necessary for capturing the geospatial information for IoT sensors. These include attributes for *latitude, longitude*, and *elevation*.

Sensor integration concepts: The SenShaMart ontology includes for IoT sensor integration. These include *Protocol, Unified Resource Identifier (URI), Topic, Token, Address,* and *Endpoint*. Protocol attribute captures information about the internet protocol used to integrate the IoT sensor (e.g., MQTT [12], which is used by the IoT sensor integration service to establish the communication via the provided protocol. The Endpoint captures an address or URL that allows IoT sensor integration service to communicate with the IoT sensor (this is used when the specified protocol is CoAP [11]). The URI

captures an address or URL that allows IoT sensor integration service to communicate with the IoT client applications via MQTT broker(s). The Address is used to capture an address or URL that allows IoT sensor integration service to communicate with the IoT sensor and it is used by any protocol. Topic is used by the IoT sensor integration service to signal and record the activation sensor data flow.

Sensors Cost: The SenShaMart ontology includes concepts necessary for capturing the cost of using IoT sensors. This concept currently includes attributes to describe the cost of using each IoT sensor, which are cost per a unit of time and data volume, and payment method. IoT sensor providers may specify time, volume, or both. The SenShaMart payment service calculates the required payment for using each IoT sensor and stops the sensor dataflow when the payment that has received from the IoT sensor client applications is less than that. The payment method specifies the unique ID(s) or one of more payment methods that can be used by the IoT client application to make payments. Each IoT sensor may accept multiple payment methods, such as PayPal, Credit card, and Bitcoin.

Sensor Unique Identification (ID): The SenShaMart ontology includes requires sensor IDs for all sensors as these are used to identify, use query results to select, pay, and integrate IoT sensors.

IV. IoT Sensor Registration and Query Services

The *IoT Sensor Registration* (SRS) service allows sensor providers to register their IoT sensors in SenShaMart to make them discoverable by IoT client applications via the *IoT Sensor Query Service* (SQS).

In SenShaMart, the Sensor providers are currently the parties responsible for submitting all sensor metadata we discussed in Section III via the SRS, which automatically records them in SB. However, SenShaMart ontology-based or even standard SSN-or SOSA-based descriptions of IoT sensors are currently rare due to the expertise and cost needed to develop them [18]. Therefore, we currently investigate developing machine learning-based sensor classifiers that 1) generate SSN-based metadata for IoT sensors and their data by analyzing the sensor data stream, and 2) continue relying on the sensor suppliers to provide non-SSN-related metadata, such as the sensor location and payment details. Early results on IoT data stream classification are presented in [19, 20]. In any case, classifiers and/or IoT sensor providers provide the IoT sensor metadata to SRS that stores them in SB in blocks consisting of triplets that are compliant with the SenShaMart ontology. In particular, the SRS service submits the IoT sensor metadata to SB via a blockchain transaction that is verified by the SB verification function and then stored in SB ledger's triple store. As soon as the metadata of an IoT sensor is stored in the SB, any sensor query submitted via the SQS of any SenShaMart node can "find" this IoT sensor.

When the metadata of a specific sensor is updated, the SRS updates the IoT sensor metadata by adding a new set sensor metadata to the SB but linking the previous and new versions of the sensor metadata in the SB via the same sensor ID. Note that in this case, both IoT sensor metadata versions will be visible in SB ledger as the information stored inside SB is immutable (as in all other blockchains). All metadata versions for a specific IoT sensor contain a timestamp that is used to determine which is the latest version.

To search for available IoT sensors, IoT (client) applications submit sensor queries to the SQS via any SenShaMart node. The SQS currently supports SPARQL queries that use the SenShaMart ontology and cover the entire spectrum of IoT sensor metadata we discussed in Section III. The sensor queries submitted to SQS are processed in the SB using its RDF triple store. For each sensor query submitted by a sensor client application, the SQS returns the sensor metadata (in the form of triplets) of all available IoT sensors that satisfy the submitted sensor query. The IoT applications are responsible for selecting which sensor to use form the sensor query results. They submit integration requests for the selected sensors to the Sensor Integration Service (SIS), which is discussed next in Section V.

V. IoT Sensor Integration Service

To present the *Sensor Integration Service* (SIS), we assume that all IoT sensors utilize the standard MQTT protocol [12]. This assumption does not reduce the generality of SenShaMart because: 1) MQTT is supported by virtually all existing IoT platforms, and 2) other widely supported communication protocols, such as CoAP [11], require similar SenShaMart support as MQTT.

To integrate the sensors they select, the IoT sensor client applications use the sensor query results they obtained from SQS to extract the IDs of the sensors they select. The sensor clients integrate the selected IoT sensor(s) by sending an integration request to the SIS service, which includes the ID of the requesting IoT sensor client application, the IDs of the selected IoT sensors, and the ID of sensor payment transaction, which is discussed further next, in Section VI, and includes the payment for the selected sensors that determines the duration of and/or amount of data each selected sensor will provide. The SIS service generates a broker for each Client application that performs the following autonomously: 1) integrates the selected IoT sensors, and 2) activates the flow of their sensor data observations to the sensor client application. The details of the protocol and mechanisms SenShaMart uses for (1) and (2) are presented in [21]. The *Sensor Payment Service* (SPS), which is discussed in Section VI, autonomously terminates the broker of each client application, and stops the sensor data flow to the IoT client application whenever the payment made by its sensor payment transaction runs out.

VI. IoT Sensor Payment Service

Existing blockchains do not support transactions unless they use smart contract technology like Ethereum [17] does [22]. Using a smart contract may require resources and high cost as it needs to run on the entire blockchain (i.e., all blockchain nodes at the same time).

The *Sensor Payment Service* (SPS) estimates the payments of IoT service client applications for the sensors they select, and it supports SB-based payment transactions that guarantee the payment transaction terms are met.

To compute IoT service client application payments, the SPS uses the IDs of the sensors selected by each sensor client to search the SB ledger and retrieve their metadata. The sensor metadata provides all the required information including the cost per unit of time of sensor dataflow and/or unit of sensor data volume. By using this information, the SPS service can calculate the total payment (*PaymentSi*) for a selected sensor *Si* by considering: 1) the cost of using *Si* per minute (*Cm*), multiplied by the number of minutes (*Nm*), and 2) the cost of IoT data per Kbyte (*Ckb*) multiplied by the number of delivered Kbytes (*Nkb*) for each selected sensor *Si*, as represented by:

$$PaymentSi = Cm * Nm + Ckb * Nkb$$

From that, SPS computes *TotalPaymentCj* (the total payment cost of for a sensor client application *Cj*) as the *Sum of (PaymentS1, …, PaymentSn)* when *{S1, …Sn}* is the set of sensors selected by *Cj*.

IoT client applications interact with SPS to get payment estimates and to set related terms for the sensors that have selected. IoT client applications use these estimates to submit payment transactions to SPS that ensures that the sensor payment terms are satisfied, or the payment is other cancelled or returned. SPS supported payment transactions, which include the sensor client ID (payer ID), the IDs of the selected sensors *S1…Sn*, the *PaymentSi* for each selected sensor Si and its terms (i.e., *Cm, Nm, Ckb, Nkb*), and the *TotalPaymentCj* and its payment method. Payment transactions are recorded in SB. As noted in Section IV, the SPS uses the transaction estimates to monitor the payment terms *Cm, Nm, Ckb, and* Nkb of each sensor *Si* selected by a sensor client *Cj* and terminates *Cj*'s sensor dataflow by stopping *Cj*'s broker when *Cj* payment runs out. These are accomplished as follows:

- Suppose that *t* is the time SIS activates dataflow from *S*i to *Cj*. SPS monitors if t $\leqslant t + Nm$ is true. If it is, SPS allows the IoT data flow from *Si* to *Cj*. Otherwise, it stops the IoT data flow to *Cj*.
- Suppose now that *d* is the number of Kbytes delivered from *Si* to *Cj*. At the time SIS activated dataflow from *S*i to *Cj*, *d* was set to 0 by SIS. SPS increments *d* with the size of data delivered from *Si* to *Cj* every time *Si* pushes new data to *Cj*. While $d \leqslant Ckb * Nkb$ SPS allows dataflow to continue. Otherwise, it stops the IoT data flow to *Cj*.

Next, we consider a situation where a sensor *Si* selected by a client application *Cj* if any IoT sensor fail to send data after SIS has successfully integrate *Si* or activated its dataflow to *Cj*. In the event of such a failure, the SPS calculates the required compensation from each sensor involved and generates a compensating transaction, which is linked to the original sensor payment transaction in SB via the same transaction ID. While this basic extended transaction model (e.g., Saga [23]) is not new, the blockchain-based transaction mechanism that supports it is novel.

VII. Performance Charecterstics

SenShaMart including the specialized blockchain and its provided services is implemented by using NodesJS V14.17.5 [24]. We have deployed SenShaMart on 20 nodes using Nectar research cloud, which is Australian's national research cloud that provides cloud computing services for Australian researchers. We have conducted a large-scale experiment to measure the *scalability* and the *performance* of SenShaMart in terms of register, query, and integrate IoT sensors as well as pay their providers.

In our large-scale experiments, we used up to 100,000 Sensor clients and 5,000,000 IoT sensors. We used 10 real IoT sensors and the rest are virtual sensors generate data similar to the real IoT sensor by using IoT-Data-Simulator tool [25]. In this experiment, we measured the following as shown in Figure 2:

1) the response time of the IoT sensors registration (i.e., storing the IoT sensor metadata in SenShaMart Blockchain) with respect to the increasing number of IoT sensors;

2) the response time of querying IoT sensors with respect to the increasing number of registered IoT sensors;

3) the response time of the integrating IoT sensors with respect to the increasing number of IoT sensors; and

4) the response time of paying IoT sensors with respect to the increasing number of IoT sensors.

The experiment results show a linear relationship between the increasing number of IoT sensors with the response time of IoT sensor registration, query, integration, and payment. Please note that the size of SB ledger reached 8,828 MB for storing the metadata of 5,000,000 IoT sensors, and sensor integration can scale up indefinitely. From the experiment results, we can claim that SenShaMart is a scalable self-managed marketplace for discovering, integrating, and paying IoT sensors. Although these evaluation results are sufficient for supporting many IoT applications, they also provide the baseline for research towards increasing the scalability of SenShaMart.

VIII. Conclusion

To fully realize SenShaMart, further research is required in the following areas:

Devising machine-learning classifiers for IoT sensors from analyzing their data streams and using them to generate SenShaMart ontology compliant description and other metadata. Earlier research in this area has provided encouraging results [19, 20].

Developing a simplified language for the specification of IoT sensor queries and a highly scalable query mechanism for processing SSN-based IoT sensor descriptions stored in the SenShaMart ledger. The simplified IoT sensor query language will significantly reduce the complexity (and related specification effort and expertise required) of sematic query languages, such as SPARQL, when they are used to specify IoT sensor queries involving many sparsely interconnected SOSA/SSN-based IoT sensor descriptions. Devising a highly efficient query processing mechanism for information dis-

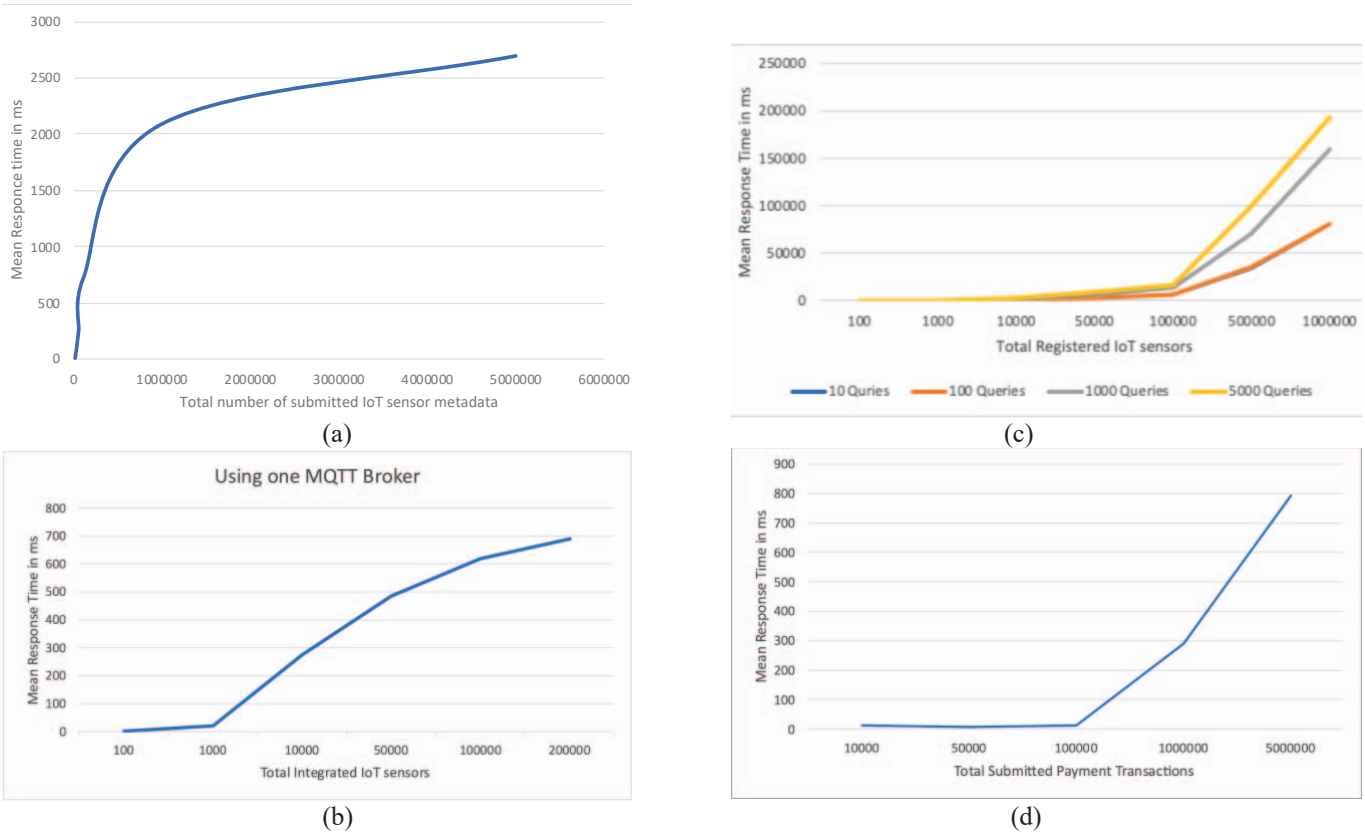

Fig. 2. Evaluation plots: The relationship between the mean response time and a) the number of submitted IoT sensor metadata, b) the number of integrated IoT sensors, c) the number of registered IoT sensors with varies number of queries, and d) the number of submitted payment transactions.

tributed across multiple nodes in the SB is critical for enabling sensor clients to quickly find the IoT sensors they need.

Devising a blockchain-based mechanism for ACID transactions as well as decentralized extended transaction models that support internet-scale sensor queries and payment transaction. Existing blockchains do not support ACID involving payment and use (i.e., activation and termination of sensor data streams) of multiple IoT sensors owned and managed by multiple providers. Existing blockchains completely lack transaction support for data stored in different blockchain blocks. This problem becomes more challenging as the number of IoT sensor clients, the number of IoT sensors, and/or the number of IoT sensor providers increase. Building on earlier work on chronological transaction scheduling [26] that includes highly decentralised transaction processing mechanisms may provide a pathway meeting both the blockchain and scale requirements of SenShaMart.

To conclude, this paper proposed a self-managed market place for sharing sensors in IoT (SenShaMart). To realize SenShaMart, this paper proposes a novel specialized blockchain and related services for IoT sensor discovery, integration, and payment transactions that are completely trusted (i.e., they are not controlled by any party) and can scale up to support millions of IoT sensors and sensor clients. SenShaMart benefits include: (1) makes IoT application development more efficient and cost-effective via enabling sharing and reusing of existing IoT sensors owned and maintained by different sensor providers, (2) eliminates the need for administration to share IoT sensors, promotes deployment of new IoT sensors supported by a revenue generation scheme for their providers, (3) reduces or eliminates the need to produce, deploy and maintain the IoT sensors each application needs, and (4) supports novel blockchain-based extended transactions to pay for IoT sensors and compensate.

IX. REFERRENCES

[1] D. Georgakopoulos and P. P. Jayaraman, "Internet of things: from internet scale sensing to smart services," *Computing,* vol. 98, no. 10, pp. 1041-1058, 2016.

[2] A. Dawod, D. Georgakopoulos, P. P. Jayaraman, and A. Nirmalathas, "Advancements towards Global IoT Device Discovery and Integration," in *2019 IEEE International Congress on Internet of Things (ICIOT)*, 2019: IEEE, pp. 147-155.

[3] M. Conoscenti, A. Vetro, and J. C. De Martin, "Blockchain for the Internet of Things: A systematic literature review," in *Computer Systems and Applications (AICCSA), 2016 IEEE/ACS 13th International Conference of*, 2016: IEEE, pp. 1-6.

[4] A. Dawod, D. Georgakopoulos, P. P. Jayaraman, and A. Nirmalathas, "An IoT-owned service for global IoT device discovery, integration and (Re)

use," in *2020 IEEE International Conference on Services Computing (SCC)*, 2020: IEEE, pp. 312-320.

[5] S. Chandler, "How The Internet Of Things Will Help Fight Climate Change," 5/11/2019. [Online]. Available: http://bit.ly/37VpGF3

[6] C. Perera, A. Zaslavsky, M. Compton, P. Christen, and D. Georgakopoulos, "Semantic-driven configuration of internet of things middleware," in *2013 Ninth International Conference on Semantics, Knowledge and Grids*, 2013: IEEE, pp. 66-73.

[7] M. Compton *et al.*, "The SSN ontology of the W3C semantic sensor network incubator group," *Web semantics: science, services and agents on the World Wide Web,* vol. 17, pp. 25-32, 2012.

[8] A. Haller *et al.*, "The modular SSN ontology: A joint W3C and OGC standard specifying the semantics of sensors, observations, sampling, and actuation," *Semantic Web,* no. Preprint, pp. 1-24, 2019.

[9] S. Chaudhuri and G. Weikum, "Self-Management Technology in Databases," in *Encyclopedia of Database Systems*, L. Liu and M. T. ÖZsu Eds. Boston, MA: Springer US, 2009, pp. 2550-2555.

[10] K. Mišura and M. Žagar, "Data marketplace for Internet of Things," in *2016 International Conference on Smart Systems and Technologies (SST)*, 2016: IEEE, pp. 255-260.

[11] Z. Shelby, K. Hartke, and C. Bormann, "The constrained application protocol (CoAP)," 2070-1721, 2014.

[12] U. Hunkeler, H. L. Truong, and A. Stanford-Clark, "MQTT-S—A publish/subscribe protocol for Wireless Sensor Networks," in *2008 3rd International Conference on Communication Systems Software and Middleware and Workshops (COMSWARE'08)*, 2008: IEEE, pp. 791-798.

[13] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[14] J. Benet, "IPFS-content addressed, versioned, P2P file system," *arXiv preprint arXiv:1407.3561,* 2014.

[15] S. Popov, "The tangle," *cit. on,* p. 131, 2016.

[16] O. Alphand *et al.*, "IoTChain: A blockchain security architecture for the Internet of Things," in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, 15-18 April 2018 2018, pp. 1-6, doi: 10.1109/WCNC.2018.8377385.

[17] V. Buterin, "A next-generation smart contract and decentralized application platform," *white paper,* 2014.

[18] D. Georgakopoulos, P. P. Jayaraman, and A. Dawod, "SenShaMart: A Trusted loT Marketplace for Sensor Sharing," in *2020 IEEE 6th International Conference on Collaboration and Internet Computing (CIC)*, 2020: IEEE, pp. 8-17.

[19] D. Madithiyagasthenna *et al.*, "A solution for annotating sensor data streams-An industrial use case in building management system," in *2020 21st IEEE International Conference on Mobile Data Management (MDM)*, 2020: IEEE, pp. 194-201.

[20] F. Montori, K. Liao, P. P. Jayaraman, L. Bononi, T. Sellis, and D. Georgakopoulos, "Classification and annotation of open internet of things datastreams," in *International Conference on Web Information Systems Engineering*, 2018: Springer, pp. 209-224.

[21] A. Dawod, D. Georgakopoulos, P. P. Jayaraman, A. Nirmalathas, and U. Parampalli, "IoT device integration and payment via an autonomic blockchain-based service for IoT device sharing," *Sensors,* vol. 22, no. 4, p. 1344, 2022.

[22] P. Ruan *et al.*, "Blockchains vs. distributed databases: Dichotomy and fusion," in *Proceedings of the 2021 International Conference on Management of Data*, 2021, pp. 1504-1517.

[23] P. K. Chrysanthis and K. Ramamritham, "ACTA: the SAGA continues," ed: Citeseer, 1992.

[24] *Node.js*. (2009). OpenJS Foundation. [Online]. Available: https://nodejs.org/en/

[25] *IoT-Data-Simulator*. (2018). Github. Accessed: 22/12/2022. [Online]. Available: https://github.com/IBA-Group-IT/IoT-data-simulator

[26] D. Georgakopoulos, M. Rusinkiewicz, and W. Litwin, "Chronological scheduling of transactions with temporal dependencies," *The VLDB Journal,* vol. 3, no. 1, pp. 1-28, 1994.