DATA MANAGEMENT ISSUES AND STUDY ON HETEROGENEOUS DATA STORAGE IN THE INTERNET OF THINGS

T.Kalai Selvi[1] and Dr.S.Sasirakha[2]

[1]Assistant Professor (SLG-I), Department of Computer Science and Engineering, Erode Sengunthar Engineering College, Erode
[2]Associate Professor, Department of Information Technology, Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam

ABSTRACT

The Internet of Things is a networking standard that connects various hardware, including digital, physical, and virtual things that may communicate with one another and carry out user-requested tasks. Traditional database management methods cannot be used in this entity because of the variety, large volume and heterogeneous data generated by them. The rapid growth of heterogeneous data can only be managed by distributed and parallel computer systems and databases. When it comes to handling vast amount of diverse data, most relational databases have a variety of drawbacks because they were designed for a certain format. One of the most difficulties in data management is investigating such heterogeneous data. Consequently, IoT data management system design has to be considered with some distinct principles. These various guiding concepts enable the suggestion of various IoT data management system strategies. The solution should provide a unified format for the conversion of various heterogeneous data which are generated by the sensors. The integration of generated data is made simple by some middleware or architecture-oriented solutions. Other methods also offer effective storage of the unified data generated. This paper surveys the challenges of IoT Data management and provides a survey about the storage of heterogeneous data and the type of data used.

KEYWORDS

Heterogeneous, Internet of Things, unified data, and data management

1. INTRODUCTION

The Internet of Items (IoT) is a live, expansive network infrastructure in which things are recognisable, independent, and self-configurable [2]. The subsystems can interact with one another by exchanging data that is produced through sensing and responding to control the physical world by starting operations [1]. The sources of data, the collecting of data, and the processing of data are just a few of the different characteristics of data in the IoT [2][3]. As shown in Figure. 1, these design primitives are arranged into three primary categories: data collection, data management system design, and processing. The detection and identification of Things and subsystems—which may be stationary or mobile—are the primary goals of data collecting elements. The IoT data repositories will receive the acquired data. The architecture of the data management system and the manner in which data will be archived are both covered by data management system design elements. The actual access to data repositories is dealt with by processing components.

DOI:10.5121/cseij.2022.12604

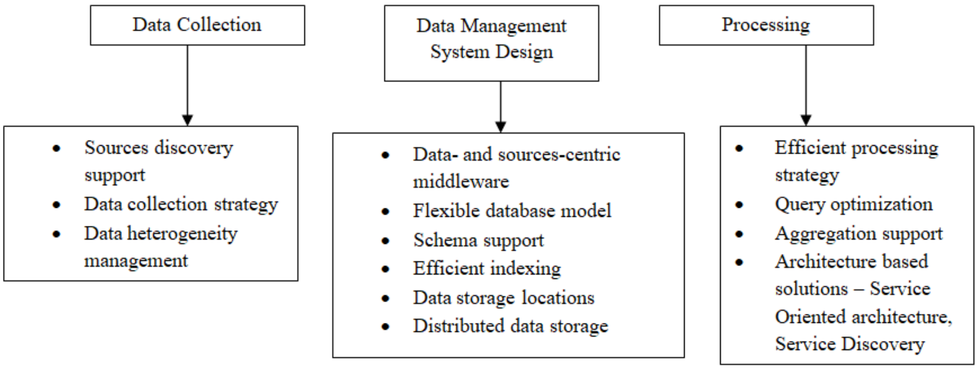

Figure 1. IoT Data Management Solution

2. DATA MANAGEMENT AND ITS BASIC CONCEPTS

One of the more valuable components of IoT is data. Their management is therefore an absolutely essential task and will be a major research issue. To enable the processing and assessment of data from IoT factors, particularly in RFID, Wi-Fi Sensor Networks (WSNs), and other related technologies, a number of works were completed. Numerous types of data may be recognised due to the diversity of the resources. Information in IoT environments originates from several types of devices totaling billions of devices. IoT data can be divided into a number of categories, as shown in Figure. 2. Structured, semistructured, and unstructured data are only a few of the formats in which IoT devices provide their data. Discrete sensor readings, device metadata, big image or video files, digital signals, and these data may also contain analogue signals, such as videos. Since IoT data is not homogeneous, there is no one-size-fits-all method for storing IoT data.

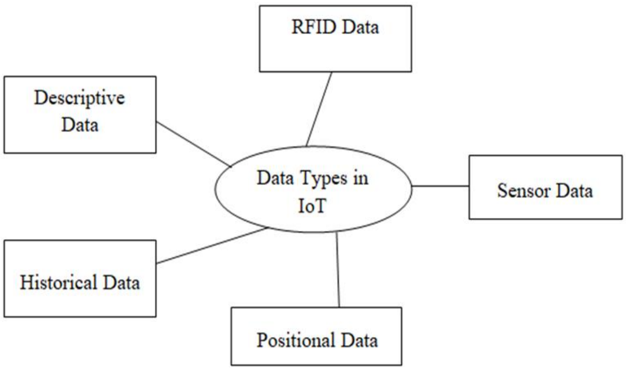

Figure 2. Different Classes of IoT Data

3. CHALLENGES IN HANDLING IOT DATA

- A suitable DBMS must be used to manage the enormous amount of heterogeneous data generated by sensors [4].
- To offer effective mechanisms for IoT data storage and query processing to meet IoT application requirements [5].
- IoT devices generate a lot of data, but there aren't enough tools to locate datasets that are right for the models you want [6].

- Demand for quick real-time data processing systems as well as secure systems to store and analyse sensitive data safely [7].
- Sensors produce enormous amounts of repetitive data [8].

4. LITERATURE SURVEY ON STORAGE OF IOT DATA

This section provides the literature survey on storage of IoT data.

A centralised method for storing massive amounts of diverse IoT data was presented by Li et al. [9] and is called a data storage management system. This offers a remedy that is based on NoSQL and which uses an architecture called as IOTMDB. To solve the problems of storing and managing the enormous amount of IoT data, NoSQL systems are effectively used. The data-generating apps are gathered and transmitted to the IOTMDB system. The Master node, Slave node, Data receiving node, and Standby node are the four key nodes in the system.

The Master node was the manager of the cluster whose function is to accept the requests from clients and check whether the maximum limit has been reached or not The data is obtained via a data reception node and the slave node is used to store the data. When the master node malfunctions, the standby node, which is a duplicate of the master node, continues to function. They had also proposed a strategy for storage of IoT data and also followed homogeneity in the way IoT data is arranged. Their solution displays data from a range of applications as a collection of SampleRecords. A SampleRecord, which is a collection of a set of sampleElement, is the basic unit of data storage in the IOTMDB.

The format of the sampleElement is defined as a key-value pair ie., is represented as follows SampleElement is equal to "key, value", where value is the actual value of the string and key is the name of the value, both of which are Strings.

SampleRecord contains both static and dynamic information about the item in two separate pieces. Static information is the data about an object that does not change, such as the object's ID or the field it belongs to. The examples of the dynamic information are time, location, speed, etc. These are the sample values which are changing values of the object. The SampleRecords are the collection that may contain data of different objects of the same application.
The following is a definition of SampleRecord:

$$\mathrm{SampleRecord} = ((\mathrm{objID}, \mathrm{field}, \mathrm{type} \ldots), (\mathrm{t}, \mathrm{loc}, (\mathrm{v_i})_{i=1}^{n}))$$

((objID:"b01", field:"agriculture", type:"shed"), (t:t1, loc: (40.1,20.5), (light:60, humidity:0.65, light:32.2) is an example of a SampleRecord. To address the needs of IoT data storage, architecture for managing IoT storage called IOTMDB based on NoSQL is proposed.

A data storage paradigm that incorporates both structured and unstructured data has been proposed by Jiang et al. [10]. This is suggested as a solution to the possible issues brought by the vast volume of IoT data. If the data from IoT devices is structured, their solution's database management module will be activated, and then the database management module will handle it and stores the data in the relational databases and NoSQL databases. In contrast, if data is an unstructured, then they will be handled by the next module called as the file repository module.
The file repository, database module, service module, and resource configuration module are just a few of the modules that make up the IoT-focused data storage framework.

File repository: Hadoop Distributed File System (HDFS) is used by file repositories to store unstructured files in a distributed setting. To enhance the file repository's capacity for handling small files, a file processor is employed.

Database module: The database module manages structured data by combining numerous databases and using both relational and NoSQL databases. For many databases, this module also offers object-entity mapping and a consistent API to bury implementation and interface differences.

Service module: Automatic RESTful service generation is embedded into the service module.
Resource configuration module: Static and dynamic data management are supported by the resource configuration module.

Only a few different kinds of database adapters are implemented in the present version of the data storage framework. In the future, more adaptors for additional NoSQL databases will be added.

For the purpose of finding IoT data in distributed in-networking, the authors of [11] developed a storing technique employing a machine learning algorithm. The user query is received by the Query Processing in their proposed framework, which then sends it to the Discovery Services (DSs), whose job it is to route the user's request and find a group of related Gateways (GWs) that may have references to connected resources that can provide a response (values) to the requested query.

The authors of [12] suggested using a specialised records management architecture that could be modified to meet various IoT requirements. By addressing node failure and unreliable exchange through the use of allocated data storage inspired by the RAID device as a data storage layer that sits between a device and the customised DBMS, this system improves health and trustworthiness.

The work of [13] offered to provide to a conventional vision of statistics collected in the form of motion filtered and continuously aggregated data toward a storage-focused vision. In this illustration, sensors are equipped with embedded structures and flash memories to store records locally at the nodes and help lower the cost of communication during query processing. A DBMS called StoneDB uses data from locally saved sensors to build a database that can support archived queries or even tasks like data mining. Only requests that are sent to the database and while receiving results are counted as transmission.

The authors of [14], Youssef Elbanoby., et.al., proposed a framework and they have saved the data in two ways. The RDBMS for the ERP, SCM, and CRM serves as a temporary storage location for data initially. The information is then sent to Bigdata in the public cloud. A programme called Oracle Bigdata SQL, which facilitates the quick integration of the RDBMS with the Bigdata using the Oracle Bigdata Connectors, was also used to integrate the RDBMS with Bigdata.

The authors of [15] in their work, they have proposed two RFID statistics warehouse. They employed a BigData Platform (Hadoop/Hive), a de-normalized schema with RDBMS (PostgreSQL), a normalised schema with RDBMS (one of which is the normalised schema), and performance testing with cloud computing.

By building a scalable sensor data collecting, analysis, and visualisation platform based on the IoT platform for smart farming applications, the work in [16] offered SmartFarmNet. They have displayed the platform's architectural layout, which primarily focuses on IoT devices. The cloud-based system employs NoSQL to obtain data and SPARQL (a language similar to SQL) to query the data.

The authors of [17] suggested a framework that stores the large number of data on cloud with less storage space. This framework is provided with several data compression techniques such as AES encryption methods with the increase in data security. Additionally, the framework demonstrates how data interacts with reporting and analytical tools via the cloud.

A framework with the MongoDB as NoSQL databases has been implemented by the authors [18]. The data storage framework implemented contains the Internet Gateway Device which connects the middleware to the Internet. Also the framework provides with the Web Service to process the data with the help of GET and POST function. The data storage framework utilizes the MongoDB and the GridFS. The data which is stored in the GridFS is divided into two sections. One is the chunk file and another section is the metadata. The MongoDB database stores sensor data in topic-by-topic order.

The authors of [19] proposed the storage mechanism for storing the large volume of sensor data with the various RDF (Resource Description Framework) storage mechanisms like data aware hybrid storage, triple store, table with vertical and horizontal partitions, a column storage, and a property table. Also they proposed the RDF data storage techniques, which process the RDF data faster that enables the interactive device network connectivity framework.

An effective architecture for efficient and secure data collecting from multiple sensors and methods for data transfer was proposed by the authors of [20]. Also, they proposed the framework with effective storage scheme in the cloud. This framework has been proposed for the Smart Ocean IoT subsurface. Real-time data and non-real-time data are the two categories into which the data collecting is divided in this method. The base stations and cloud servers both collect and store the data from the numerous underwater sensors.

The following Table.1 provides the comparison of the related works on the storage framework:

Table.1 Comparison Table

References	**Database used / Data stored in**	**Framework used / Algorithm used**
[9]	NoSQL	IOTMDB
[10]	NoSQL	IoT oriented data storage framework
[11][12]	RAID DBMS	Machine Learning algorithms
[13]	Local Storage	StoneDB
[14]	RDBMS, RDBMS with Bigdata	Oracle Bigdata SQL
[15]	PostgreSQL	Bigdata Platform (Hadoop / Hive)
[16]	NoSQL, SPARQL	SmartFarmNet
[17]	Stores data on Cloud	AES encryption techniques
[18]	MongoDB	GridFS with Web Service
[19]	Hybrid storage, triple store, column store, property table, vertically and horizontally partitioned table	Resource Description Framework
[20]	Cloud servers and the base stations	Underwater IoT of Smart Ocean

5. Objective of the Study

This qualitative, phenomenological study's objective was to comprehend the many difficulties associated with managing IoT data. This demonstrates that IoT data management is the process of shifting through all available data to extract the most important information. Large amounts of mixed information are sent by many different devices for many different uses. Controlling all of this IoT data needs the development of new architectures, policies and procedures for collecting the data, transforming and analyzing the data which provides better results to the users of the application. Also this study helps to understand, how the various frameworks are storing the raw data from various sensors and other IoT devices.

6. Results of the Study

This study contributes to the discovery, that traditional data management solutions cannot handle the vast volume of data produced by diverse IoT devices. The data available from the various connected sources are in different formats. The time can be saved by considering the unified data format for processing the data received from the connected objects. A storage management solution has to be identified with the needed algorithms to increase the speed of the analysis in order to get the proper results of the applications. Also found that the task can be automated which can save the time of processing. Some of the tools are identified which helps to collect, transform and process the data.

7. Conclusion

The issues of managing IoT data are discussed in this study, along with a survey on how heterogeneous data is stored. Various frameworks presented give the detailed description about the storage of the data from the IoT resources.

References

[1] Somayya Madakam, et. al.,(2015) "Internet of Things (IoT) : A Literature Review", *Journal of Computer and Communications,* 3, 164-173, http://dx.doi.org/10.4236/jcc.2015.35021

[2] M. Abu-Elkheir, M. Hayajneh, N. Ali, (2013) "Data management for the internet of things: design primitives and solution", *Sensors* 13, 15582–15612, https://doi.org/10.3390/s131115582

[3] Bassirou Diène, et.al.,(2020) "Data management techniques for Internet of Things", *Mechanical Systems and Signal Processing,* Volume 138, April 2020, 106564, Received 25 May 2019, Revised 6 November 2019, Accepted 11 December 2019, Available online 23 December 2019, https://doi.org/10.1016/j.ymssp.2019.106564

[4] MAHMOUD EYADA., et.al, (2020) "Performance Evaluation of IoT Data Management Using MongoDB Versus MySQL Databases in Different Cloud Environments", *IEEE Access,* Received May 26, 2020, accepted June 8, 2020, date of publication June 15, 2020, date of current version June 25, 2020.

[5] Ousmane Diallo., et.al.,(2019) "Data Management Mechanisms for Internet of Things: A position paper", *2019 International Conference on Computational Science and Computational Intelligence (CSCI),* DOI 10.1109/CSCI49370.2019.00228

[6] Keith Grueneberg, et.al., (2019) "IoT Data Management System for Rapid Development of Machine Learning Models", 2019 *IEEE International Conference on Cognitive Computing (ICCC),* DOI 10.1109 / ICCC.2019.00021

[7] Md Shihabul Islam, et.al.,(2019) "Secure Real-Time Heterogeneous IoT Data Management System", *2019 First IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA),* DOI 10.1109/TPS-ISA48467.2019.00037

[8] Narasimha Swamy S, et.al., (2018) "Repeated Data Management Framework for IoT: A Case Study on Weather Monitoring and Forecasting*", IEEE, 4th Int'l Conf. on Recent Advances in Information Technology , RAIT-2018*

[9] T. Li, Y. Liu, Y. Tian, S. Shen, W. Mao,(2012) "A Storage Solution for Massive IoT Data Based on NoSQL", in: *IEEE*, pp. 50–57, DOI 10.1109/GreenCom.2012.18

[10] Lihong Jiang, Li Da Xu, Hongming Cai, Zuhai Jiang, Fenglin Bu, Boyi Xu, (2014) "An IoT-Oriented Data Storage Framework in Cloud Computing Platform", *IEEE Trans. Ind. Inform. 10* 1443–1451. doi: 10.1109/TII.2014.2306384.

[11] Y. Fathy, P. Barnaghi, S. Enshaeifar, R. Tafazolli, 2016 "A distributed in-network indexing mechanism for the Internet of Things", *2016 IEEE 3rd World Forum Internet Things WF-IoT, Reston, VA, USA*, pp. 585–590.

[12] N. Siegmund, M. Rosenmüller, G. Moritz, G. Saake, D. Timmermann, (2009) "Towards robust data storage in wireless sensor networks", *IETE Tech. Rev. 26*, 335–340.

[13] Y. Diao, D. Ganesan, G. Mathur, P.J. Shenoy, (2007) "Rethinking Data Management for Storage-centric Sensor Networks", in: *CIDR, 2007*: pp. 22–31.

[14] Youssef Elbanoby, Mohamed Aborizka, Fahima Maghraby, (2019) "Real-Time Data Management For IoT In Cloud Environment", *2019 IEEE Global Conference on Internet of Things (GCIoT)*, DOI:10.1109/GCIoT47977.2019.9058394

[15] Yei-Sol Woo (2015). "RFID BigData Warehousing and Analytics in Cloud Computing Environment". *http://opus.ipfw.edu/masters_theses/44*

[16] Jayaraman, P., Yavari, A., Georgakopoulos, D., Morshed, A., & Zaslavsky, A. (2016). "IoT Platform for Smart Farming: Experiences and Lessons Learnt. Sensors", 16(11), 1884. doi:10.3390/s16111884

[17] Pallavi Srivastava, Navish Garg, (2015) "Secure and optimized data storage for IoT through cloud framework", *International Conference on Computing, Communication and Automation (ICCCA2015),* ISBN: 978-1-4799-8890-7/15-2015 *IEEE*

[18] Eko Sakti Pramukantoro, Widhi Yahya, Gabreil Arganata, Adhitya Bhawiyuga, Achmad Basuki,(2017) "Topic Based IoT Data Storage Framework for Heterogeneous Sensor Data", 978-1-5386-3546-9/ 17 – *2017 IEEE*

[19] Trupti Padiya, Minal Bhise, Prashant Rajkotiya, (2015) "Data Management for Internet of Things", *2015 IEEE Region 10 Symposium*, DOI 10.1109/TENSYMP.2015.26

[20] Chunqiang Hu, Yuwen Pu, Feihong Yang, Ruifeng Zhao, Arwa Alrawais, Tao Xiang, (2020) "Secure and Efficient Data Collection and Storage of IoT in Smart Ocean", DOI 10.1109/JIOT.2020.2988733, *IEEE Internet of Things Journal.*

AUTHORS

T.Kalai Selvi received M.E degree in Computer Science and Engineering from Mahendra Engineering College, Namakkal and presently she is working as Assistant Professor (Selection Grade-I) in Department of Computer science and Engineering at Erode Sengunthar Engineering College, Erode. She has published more than 12 papers in reputed Journals and more than 15 papers in various National and International conferences. Her Research interests include Cloud Computing, IOT. She is a Life member of ISTE and CSI.

S. Sasirekha has over 13 years of teaching experience and 1 year of industrial experience. She received her B.E (IT) degree from Periyar University, M.E. in Computer and Communication from SSN College of Engineering (Silver Medallist), Anna University, Chennai and Ph.D. from Anna University, Chennai in the area of SOA and Semantic based IoT middleware. She joined SSN College of Engineering in the year 2007 and is currently working as an Associate Professor in the Department of Information Technology. Her current research areas of interest include Web Based Application, Service Oriented Architecture, Internet of Things, Semantic Web Services, Ontological Engineering and Web of Things. To her credit she has published more than 35 papers in International and National Journals and Conferences. She is a recognized Anna University research supervisor to guide Ph.D. /M.S. candidates. She has guided about 37 UG projects, 7 PG projects and 17 student internal funded projects so far. She has organized / attended many externally / internally funded Conferences / Workshops / Seminars / Guest

Lectures at National and International levels. She is an active member of IEEE and ACM professional society. She is also an active reviewer for reputed publishers like IOS Press and IGI Global. She is also serving as the Technical Programme Committee member in various National and International conferences.