Available online at www.sciencedirect.com

ScienceDirect

Procedia Manufacturing 34 (2019) 893–902

Procedia MANUFACTURING

www.elsevier.com/locate/procedia

47th SME North American Manufacturing Research Conference, Penn State Behrend Erie, Pennsylvania, 2019

Provisioned Data Distribution for Intelligent Manufacturing via Fog Computing

Riddhiman Sherlekar [a], Binil Starly [a*], Paul H. Cohen [a*]

[a]*Edwards P. Fitts Department of Industrial and Systems Engineering*
North Carolina State University, Raleigh NC USA 27695

* Corresponding author. Tel.: +1 919-515-1815; fax: +1 919-515-5281. *E-mail address:* bstarly@ncsu.edu; pcohen@ncsu.edu

Abstract

The number of 'things' ranging from simple devices to complex machines on the factory floor connected at the enterprise level and to the broader internet is growing exponentially. This connection also leads to a tremendous amount of data generated leading to 'Data' now considered one of the core assets in the broader manufacturing industry. However, the availability of this asset is hardly made use of by Small and Medium scale manufacturing enterprises (SME) - the 'Mittelstand' of America. How can certain types of data be shared by SME companies, yet have the ability to retain ownership and control over their own data? How does SME leverage computing on these diverse forms of data for the benefit of its clients and itself? In this paper, we propose a decentralized data distribution architecture to democratize the potential availability of large amounts of data generated by the manufacturing industry using the Fog Computing paradigm. The architecture leverages an Industry scalable middleware extension of Cloud manufacturing that securely filters and transmits data from IoT enabled manufacturing machines on the shop floor to potential users over the cloud. This work also demonstrates a data-centric approach which allows peer-to-peer data sharing laterally within the fog layer to serve cloud users. We demonstrate the feasibility of the Fog middleware infrastructure through case studies that involves various types of manufacturing data.

Peer-review under responsibility of the Scientific Committee of NAMRI/SME.

Keywords: Fog Computing; MTConnect Industry 4.0; IIoT; Industrial Cyber-Physical Systems; Big Data; Digital Manufacturing, Data Distribution

1. Introduction

The prevalence of the vision towards Industry 4.0 is leading the transformation of traditional manufacturing environments and factory floors into cyber-physical systems which give more flexibility to the key stakeholders in the manufacturing industry. Forecasts expect that nearly 50 billion devices will be IoT enabled by 2020[1], and a major proportion of those devices will be on factory floors generating a tremendous amount of high variety and high-frequency data. Moreover, data-intensive applications which require lots of parallel computing will play a primary role in these IoT enabled environments [2]. This "Big Data" generated will cause storage problems and would result in high resource utilization in the current cloud manufacturing environment. Data transmission in this scenario will become problematic because of privacy and latency issues and hence would require an extension of existing Cloud manufacturing

Peer-review under responsibility of the Scientific Committee of NAMRI/SME.
10.1016/j.promfg.2019.06.158

services [3]. Also, if manufacturers want to develop data-driven strategies, perform advanced analytics on real-time data, the traditional cloud may limit flexibility. Also to note is that, once data is moved to the cloud, there is a notion that owners of such data lose control over how that data is shared and used. A solution to this concern is to distribute real-time data from manufacturers into decentralized computing nodes with storage and network capabilities which acts as an advanced data filter and transmits only the summary data to the Public Cloud. To transition towards Industry 4.0 revolution, the manufacturers must be able to use Big Data technologies judiciously [4]. A substantial research effort is being placed on how to efficiently distribute control and computing systems geographically [5,6,7] and securely transfer data from an IoT endpoint to the client or computing nodes [8,9,10].

Many small manufacturers have begun the journey to connect physical machines and devices on their shop-floor to integrated intra-enterprise and inter-enterprise systems. Communication protocols such as those enabled by OPC/UA and MTConnect have lowered the barriers to enable interoperability between machines and IT systems. This slow transformation now raises the issue of how small and medium scale manufacturers can leverage this data asset for direct benefit to the enterprise and more importantly how can such data assets be utilized for obtaining new business clients for small and medium scale manufacturers.

Platform and Infrastructure-as-a-service cloud platforms partially solve this problem by migrating data over from within enterprise to the entire cloud [11]. However, such ad hoc distribution of data across the major cloud service providers only partially addresses the problem of computing and is mostly done to conduct in-house data analytics (ex. prognosis of machine assets, production scheduling, etc.). Such forms of product and process data can also be valuable in search and discovery processes in matching job shop service providers with users who request them through an equitable and decentralized architecture where the control over data lies in hands of the small manufacturers instead of a centralized organization.

Consider the scenario of a design engineer at a startup company in search of the best job shop service providers who can work with the startup to conduct short prototype runs and in the future be able to conduct production quantity runs. A web search engine on recommended job shop service providers would yield superficial information on the service capabilities of the job shop. Personnel would have to conduct their own due diligence that involves physically visiting and confirming the capabilities of the service provider before any agreement is in place between the startup-company and manufacturing provider. What if there was a near real-time information resource available to engineers at the startup company to digitally verify technical capabilities offered by a service provider, prior historical performance, quality system certifications, materials worked with in the past and kinds of customers it has done business with. If such information is readily available, this would transform sourcing and selection of supply chain service providers. However, there are several challenges as to why this is not possible soon, even if all shop-floors at SMEs were fully connected. For one, how does SME share its private data with client requests for such access to data? How can SME continue to keep control over its data without data being compromised or sent to third-party services that may misuse them? In addition, client requests may also contain the need to aggregate information across multiple service providers which necessitates conducting a meta-analysis of individual data sources of each service provider. Also, if there are only a few cloud service providers aggregating such information, it increases the chance of data monopolization and consequent misuse. Manufacturers will be dependent on a centralized authority which may not always act according to their interest which further demotivates manufacturing service providers from further investing in the digital integration of its factories.

Solutions to the above problem transcend both engineering solutions and transformative business model processes. New technologies such as decentralization enabled by Blockchain and ubiquitous computing through fog computing can perhaps provide solutions to the problem. The concept of Fog computing is built upon the idea of stretching the cloud network and bringing it closer to the IoT endpoints. Fog computing introduces a distributed and decentralized framework which connects the edge and the Cloud, thereby reducing the latency and bandwidth bottlenecks in an IoT environment [12]. This paper introduces a generic fog computing-based architecture for manufacturers to share their data and leverage the distributed computing paradigm to bridge the gap between cloud users and individual service providers. Our architecture is built on top of the existing three-layer fog computing architectures [13, 14, 15]. The architecture intends to provide a framework that can address data-ownership, data-computing, data-sharing, and data-aggregation services with regards to the manufacturing data asset. Our contribution presents a solution to Data Distribution-as-a-Service (D-DaaS) enabled through the fog computing architecture. We present the architecture with fog nodes that act as an orchestrator in the distributed system which controls data transfer from the IIoT endpoints (example physical CNC machines) to the main High-Performance Cloud through which the user interacts to get access to the machine level data. The paper further discusses briefly the data distribution service, fog computing and the previous works in these domains.

The Section 2 in the paper reviews the literature while Section 3 details the proposed generic architecture proposed along with the technical details of the implementation, the architectural overview of each sublayer within the system. Further, Section 4 contains two case studies with their technical implementation. The Conclusions are drawn in Section 5 followed by future work to be done in Section 6.

2. Literature Survey

In this section, state-of-the-art research related to the domain of data distribution in manufacturing, fog computing in manufacturing and existing fog computing architectures are discussed.

2.1 Data Distribution in Manufacturing

Data generated by the manufacturers, shop floors, designers is increasing at an exponential rate. Moreover, the exchange of this unstructured data becomes more arduous when data from heterogeneous sources like sensors, CAD-based product models, Supervisory Control and Data Acquisition (SCADA) systems, smart machines are incorporated into the data pipeline. The concept of fog computing in data distribution for a smart manufacturing environment has not been studied. There have been few efforts in the past to implement Data Distribution Service (DDS) in cyber manufacturing environments for flexible manufacturing environment at an intra-enterprise level. A publish-subscribe model, independent of any platform was implemented by the authors of [16] to develop a modular fixturing system which used the sensor feedback to adjust to the clamping forces while manufacturing using the RTI DDS software. Ungurean and Gaitan [17] discussed the interoperability of Data Distribution Service with a SCADA system. The study has shown an architecture based on the OPC protocol which uses a DDS middleware to distribute the data generated by the SCADA system. Some research has also been done to evaluate the robustness of the Data Distribution System in comparison to conventional methods of distribution the manufacturing data over the World Wide Web. Authors in [18], have compared Data Distribution Service with traditional web sockets for communication between control systems. The research has successfully shown the importance of publish-subscribe models in a real-time data distribution framework which is a key aspect of the Industry 4.0 paradigm. In the case of data distribution in a cyber-physical environment, the efficiency of the system lies in user-oriented information logistics [19].

2.2 Fog Computing in Manufacturing

The term Fog computing was coined by Cisco who viewed it as an extension of cloud computing in order to bring intelligence closer to the edge [20]. The authors in [21] have discussed perspectives for fog computing in manufacturing with a focus on DAMA (Design Anywhere Manufacture Anywhere) concept which requires the availability of manufacturing data to key stakeholders, designers without delay and scalable integration of IT systems and manufacturing systems both vertically and horizontally [22]. In [23], the authors have proposed a framework to provide production ready machine learning models in factory floor operations. This research shows that fog computing is competent to address the Industry 4.0 design concerns of latency, lesser network dependency, and real-time analytics through a case study which used a Predictive Model Mark-up Language implementation to predict faulty heating operations. Wu et al. [24] demonstrated a framework for real-time machine prognosis using fog computing to predict tool wear. The research showed the significance of storing manufacturer's data at the edge and analysing it locally instead of transferring the large data to and from local server to a centralized server. The prototype developed by the authors used a machine learning algorithm which was used to detect discrepancies in milling operations of a CNC machine. Li et al. [25] went a step further to leverage the potential of fog computing by using Artificial Intelligence. The authors implemented a deep learning model to identify flawed products on an assembly line by adopting a convolutional neural network in a fog computing environment. This study shows how fog computing is a benefit for manufacturers who are interested in making data-driven decisions for their businesses.

2.3 Architecture of Fog Computing

The concept of fog computing is state-of-the-art with respect to current digitization in the manufacturing industry. Data transmission on a large scale requires a sophisticated architecture which can channel the data from the source to the destination with a high throughput rate to improve the end user experience. The architecture of Fog computing is highly distributed connectivity within nodes and loose device coupling [26]. The application of fog computing is mainly service oriented and as a result, the architecture totally depends on the service which is to be provided. Even if the service provided is similar, the individual components of the architecture may be adjusted to streamline data distribution. The Open Fog Consortium has set some guidelines for the architecture but the highly dynamic and variable nature of services which can be offered using fog computing makes it impossible to fix a specific architecture.

A sizeable amount of research has been done in the healthcare domain with fog computing. Nandyala et al. [27], Gu et al. [28], Cao et al. [29] have analyzed a large volume of healthcare data by computationally offloading it using fog computing and using different architectures. Research papers [2, 30] have also been published which explain the middleware and evaluate the architectural constraints within a fog computing environment. Aazam et al. [3] discuss the constraints within an IoT-Cloud integration and provide a succinct comparison of fog and cloud computing technologies.

3. The System Architecture

The architecture (Fig. 1) is made up of the Physical Layer consisting of IoT enabled manufacturing machines, the fog layer which consists of distributed computing nodes which act as an orchestrator and brings manufacturing intelligence closer to the Physical Layer and the high-performance cloud computing layer with large storage and significant computing capabilities. The architecture proposed in the paper is built on the guidelines suggested by the Open Fog

Reference architecture [31] an officially accepted IEEE Standard named as IEEE 1934™ [32]. The High-Level architecture of the Data Distribution Architecture is presented (Fig. 2).

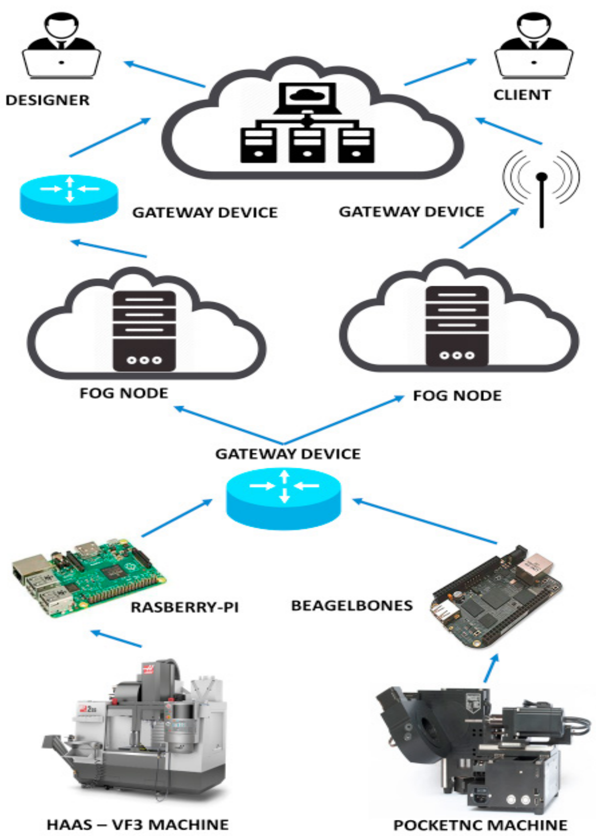


Fig. 1. The Network Architecture: The figure shows the flow of data from the data source (CNC machines) to the main cloud which serves the client via the Fog computing architecture.

The core fundamentals on which the architecture was built are as follows:

Security: All machines connecting to the fog node will be trusted and verified in our architecture. A combination of standardized MTConnect data model and OPC/UA for data transfer ensures secure data emission from the edge device. Further, the literature has proposed ways and means to secure the OPC/UA protocols [33]. Within the fog layer, the fog nodes communicate via the Publish-subscribe communication model which allows dynamic entry and exit into the system and hence necessitates secured software resources. For any fog node to join the network, it must register itself on a Loopback API [34] exposed at the Fog-Cloud Middleware node (Fig. 2) where it gets access to architecture via access tokens. The fog nodes can be owned physically by neutral third parties who have no conflict of interest with the manufacturers or the data consumers. These neutral third parties can be academic institutions, government agencies etc. Fog computing, being a highly virtualized platform, allows the architecture to incorporate even virtual fog nodes within the proposed architecture.

Scalability: The computing, storage and networking capabilities of the fog nodes can be scaled internally depending upon the number of machines connected and the throughput of data. The fog layers can be scaled vertically by adding more tiers within the hierarchy. Scalability can be achieved either by full virtualization or by container virtualization. Container virtualization is more cost effective and a light-weight solution compared to the Virtual migration when it comes to measuring the cost of scalability [35]. However, since the containers share the kernel of Operating System with other containers and have a root access, they are comparatively less secure and isolated than the Virtual migration. Moreover, when considering scalability, it should be kept in mind that the motivation for data movement is to present valuable data which can perform intricate analysis to help manufacturers [36]. Since an IIoT environment is a highly elastic and agile environment, the architecture supports the demand-driven expansion of individual components.

Openness: The main motivation of developing this architecture is the democratization of the manufacturing data by storing it in distributed databases instead of being managed by a central authority which makes the generated data and insights available to any interested party. The architecture is highly interoperable and can thoroughly exchange information at each level and within levels. The communication between the location-aware fog nodes is also open and transparent.

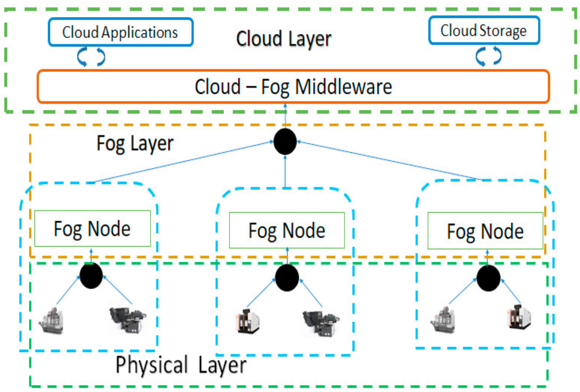


Fig. 2. The high – level three layered architecture

Autonomy: Every fog node is autonomous, and no central authority directs the fog node to act in a certain way. It is up to the discretion of the fog nodes to communicate with other fog nodes or with the cloud. Similarly, the manufacturers have full authority and control on whether to share data with a specific fog node or not. A manufacturer can opt out of the system by simply disabling the ingestion engine installed on their end and similarly a fog node can opt out.

Programmability: The architecture consists of highly adaptive deployments including support for programming at the software and hardware layers. In a virtual environment it very easy to set up n number of fog nodes using DevOps tools like Ansible, Kubernetes, and Jenkins. Fog nodes can also be created in Docker containers (a piece of software that packages the code and its dependencies to flexibly adjust to different operating systems) to maximize the resources by using several components. Fog nodes can also be created virtually in containers using the containerization technique which prunes the use of computing resources [37].

Agility: The architecture is agile if it supports key operational decisions at the fog node. In critical cases or operations that require immediate attention, for example when the machine is in an alarming state or the tool is about to wear out, the fog nodes can send an immediate response to the manufacturers.

Hierarchy: The architecture supports hierarchy both horizontally and vertically. As the number of machines connected increases or the throughput of data increases, the architecture can be deployed with clusters of fog nodes hierarchically by introducing tiers within the architecture.

RAS (Reliability, Availability and Serviceability): The reliability of the architecture lies in using reliable software and hardware computing resources. Moreover, the hardware and software resources used within the architecture are easily available and in case of downtime can be serviced in order to be functional again.

3.1 The Physical Layer

In the proposed architecture, the Physical Layer (Fig. 3) consists of manufacturing machines that live stream data from their respective geographic locations through sensor networks, communication adapters, and single board computers to a local network. The Physical Layer may consist of several heterogeneous data sources. Several communication protocols may be used within the Physical Layer, such as MTConnect and OPC Unified Architecture (UA), which integrate the existing specifications of the machines and implement the relevant industry standards. The data generated by these machines can then be routed to a database server. Within the Physical Layer, both old legacy machines and newer smart machines can be connected to facilitate data sharing from these machines.

3.2 The Fog Layer

The Fog Layer consists of location-aware computing nodes which can perform compute and storage operations. These nodes are the fog nodes which act as an intermediate entity in our distributed data sharing architecture. The fog nodes present in the Physical Layer and store it to perform short term analytics. These nodes are then further connected to the main Public cloud and have the networking capabilities to communicate with other fog nodes in the network.

The fog nodes can be physical systems having computation, storage and network capabilities which can be integrated into the D-DaaS architecture. The deployment of fog computing can also be driven by virtualization technology, which introduces a software abstraction between the computer hardware and the operating system and application running on the hardware.

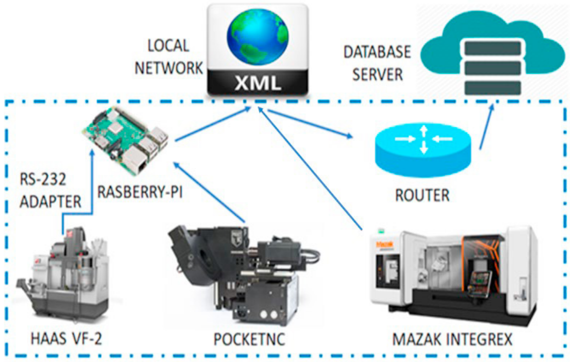


Fig. 3. The Physical Layer

Techniques such as Data Virtualization can be used within the Fog Layer to integrate data from various sources [38]. This abstraction layer acts as a controller of hardware resources and enables multi-tenancy within the fog nodes [39]. In many cases, fog may not be just middleware that provides quick response and other related services. Fog may have its own set of complete applications (like the cloud) where the cloud may act as a facilitator to access the applications running on the fog node [40] The Fog Layer consists of decentralized computing nodes with storage and network capabilities and are positioned logically between the IoT enabled manufacturing machines and the Public Cloud. The nodes in this layer are responsible to transmit the collected data via an IoT gateway device which can be a router, switch or via wireless communication to the Public Cloud. The data transmitted from the manufacturing machines in the Edge Layer to the fog nodes needs to be compressed before being transferred to the main Cloud.

The proximity of the fog layer to the manufacturing shop floor improves the efficiency of secure data transfer by resolving latency and bandwidth issues and simultaneously limiting the data transported to the High-Performance cloud for storage, processing, and long-term analytics.

3.3 The Cloud Layer

The cloud layer is the highest layer in our architecture which comprises of high-performance cloud computing resources that incorporate standards, procedures, and elements from normal cloud computing. HPC cloud computing will provide access to a massive computing infrastructure which will be able to cater to the needs of the continuously evolving manufacturing environment. It is also through this layer external users gain access to data contained at the lower level of the architecture.

The Cloud Layer receives all the filtered data from legacy CNC distributed fog nodes. Cloud layer has a middleware API which communicates with the fog nodes as per the request of the application users using the client-server REST architecture [41]. The cloud may also be used to perform long term analytics for the manufacturers.

3.4 Communication Protocols

Communication Protocols play a pivotal role in data-intensive manufacturing environments. Currently, no communication protocol has been specifically designed or accepted as an industry standard in a Fog computing environment [42]. Several communication protocols used within the fog computing architecture have been listed by authors of [43].

<u>HTTP</u>: The HTTP protocol, one of the most popular networking protocols is used in developing this architecture. In fog computing, the rest architecture can be integrated with HTTP protocol to handle requests and send quick responses to those requests via a client-server architecture. The prime advantage of using RESTful Web services with HTTP is that it allows CRUD (Create, Read, Update and Delete) operations on the data stored within the fog nodes. Moreover, using both in combination gives flexibility in using several data formats like JSON. Several frameworks like NodeJS offer flexibility in using REST with HTTP [42]. The transport protocol used in HTTP is TCP which can be useful in sending from one fog node to another fog node, but bottlenecks may occur. Moreover, the security mechanism used by HTTP is the TLS (Transport Layer Security).

<u>Publish-Subscribe</u>: The Publish-Subscribe is a messaging pattern where the publishers or the nodes which send a message, do not send the messages directly to the specific receivers of the message but instead send the messages to a message broker which then forwards the messages. Using the Publish-Subscribe pattern, the subscribers will receive only those messages that are of interest without the publisher's knowledge. A publish-Subscribe model ensures that the same state of information is maintained across all nodes within a decentralized environment. The Pub/Sub architecture ensures more security by the principle of minimal privileges.

4. Case Study

In this section, we present two case studies where we transmit the data generated from physical CNC machines and 3D CAD model data from a storage drive resource. The first case study involves data generated by a CNC machine and obtained via MTConnect protocol while the second one involves the data generated during CAD modelling activities.

4.1 Data Distribution of Machine Data

The first case study is for distributing the data generated by CNC machines, namely the HAAS VF-3, the PocketNC, and the MAZAK-Integrex within the Fitts Department of Industrial and Systems Engineering at NC State. The MAZAK-Integrex has an inbuilt adapter which streams data to a local server while the HAAS VF-3 and the PocketNC are the older legacy CNC machine. The HAAS VF-3 and the PocketNC machines were connected to a system-on-chip Raspberry Pi to interface with the machine control boards in order to transmit data via the MTConnect protocol through the university ethernet infrastructure [45]. The main motivation behind this case study is to demonstrate that manufacturers who may lack a highly sophisticated IT infrastructure can participate in Industry 4.0/Smart Manufacturing in order to share their data for a more collaborative manufacturing environment. This can increase business prospects for even small manufacturers.

A gateway device collects real-time data from the CNC machines in the Data Intensive Manufacturing Environment Laboratory and Centre for Additive Manufacturing and Logistics at NC State University through a network of sensors, MTConnect adapters, and I/O adapters. All the raw data is streamed to a local cloud database server through a Python ingestion framework and stored in a PostgreSQL Database Server. Further, in the data pipeline, the data is compressed, filtered and shared as pre-decided views of a SQL Database and transmitted to a fog node with computation, storage, and network capabilities via a Python-based oracle service that intermediates between the two servers.

The fog node can be a physical system or a highly virtualized environment. In our implementation, a MongoDB instance running on the fog node is an endpoint for the transmitted data via the oracle. The NoSQL database stores the data in the format (Fig. 4). The Fog Node follows a Client-Server architecture and can share data with the main Public cloud via an API or can act as an independent server to serve authenticated clients on the machine data sent to it. The fog node also maintains a registry of data sources and the metadata associated with it.


```
_id: ObjectId("5ba2b9a484b7814558b394e0")
M_id: "Hurco04"
type: "PartCount"
id: "Hurco04-path_4"
timestamp: "2018-09-05T02:20:05.256508"
name: "Part_Count"
sequence: "30030312581"
subtype: "ALL"
value: "UNAVAILABLE"
```


Fig. 4. The representation of data in a Fog Database

Laterally, the Fog Nodes are a part of the Data Distribution System wherein a Fog Node is both a Publisher and a Subscriber in a global data space. The fog nodes communicate with each other on topics whose data structures are pre-decided. The public Cloud is a data store for all the

Fig. 5. The first user interface for the end user

data collected from the Fog Nodes. Storing the data in local fog nodes is a viable solution to store and filter the real-time data [46]. The main cloud has applications deployed on the client-server architecture which serves users and shows the data collected from the IoT enabled machines via the Fog nodes.

The main User Interface (Fig. 5) of the application deployed on the cloud was built using a node JS framework and MongoDB database. The user has the freedom to view the data related to various machines. Once the user selects a specific type of machine, the user will be directed to available data streams which will further fetch the data from respective fog nodes via a Fog-Cloud Middleware API. The graphical user interface at the Fog node is built by the MongoDB database, Node JS for rendering web page content hosting the historical data related to machines. All functionality for access management, user authentication and checking the number of login times are built into Node JS Express server and a MongoDB database at the fog node. In case of requests, data was exchanged in JSON format.

4.2 Data Distribution of CAD Data

The second case study involves the distribution of unstructured file formats such as jpeg, JSON data existing on other private cloud platforms like Google Cloud. The motivation behind this case study is to show the ease with which the data required by designers can be made available over the internet through the fog computing architecture. The case study is used to show the ease of distribution of unstructured design data through our implementation of the Fog computing architecture. The data source for this case study is a Public cyber manufacturing infrastructure developed to support design and manufacturing research maintained at the Google Cloud platform and developed by researchers from the DIME Lab at North Carolina State University and the University of Southern California. The repository contains a plethora of CAD data which includes the metadata of the files in JSON format, STEP files, STL files and JPEG images (Fig. 6) available freely for sharing. Consider a case, where a designer is looking to design a specific part for his design and looks up for it on the web.

Our architecture facilitates read-only access to the image, metadata, data including make the metadata available via the user interface and an authenticated API access to download it with an agreement to the source. Using our architecture, the data from this CAD repository will be shared at will by the admin of the repository with the Fog databases using an authenticated Python program. The Python program will

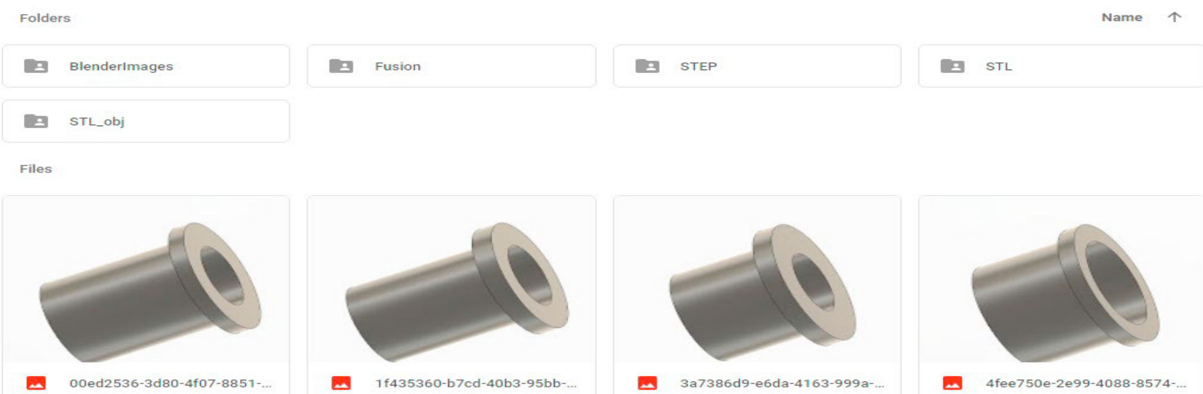
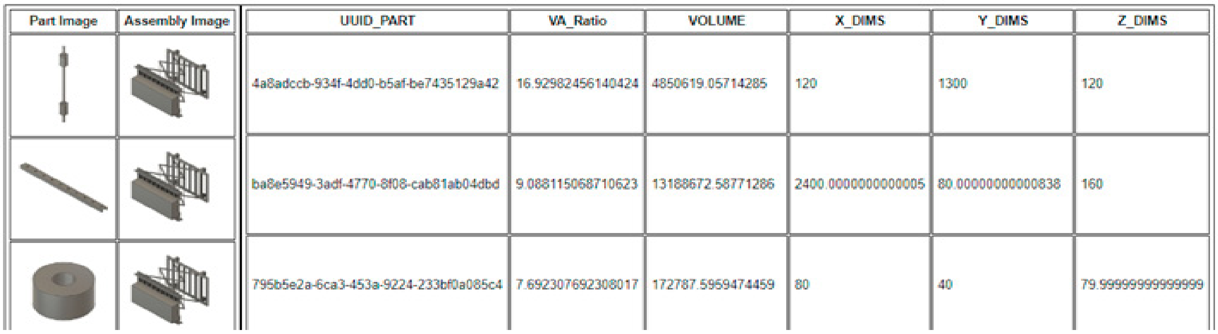


Fig. 6. The CAD data stored in a data source repository



Part Image	Assembly Image	UUID_PART	VA_Ratio	VOLUME	X_DIMS	Y_DIMS	Z_DIMS
		4a8adccb-934f-4dd0-b5af-be7435129a42	16.92982456140424	4850619.05714285	120	1300	120
		ba8e5949-3adf-4770-8f08-cab81ab04dbd	9.088115068710623	13188672.58771286	2400.0000000000005	80.00000000000838	160
		795b5e2a-6ca3-453a-9224-233bf0a085c4	7.692307692308017	172787.5959474459	80	40	79.99999999999999

Fig. 7. The CAD metadata and images presented in a read-only format to the end user.

fetch the data for read-only access from the source repository and this data will further be made available via the main cloud to the designer looking for a part via an API (Fig. 7).

Our Fog computing architecture provides the required data pipeline which facilitates SME's, academia, independent researchers to share their data from distributed computing nodes (Fig 8.) while keeping the access control of the data with them. An interested client or designer who sends a request via the Public Cloud does not have write-access to the data and his requests are served itself at the public cloud via the Fog-Cloud Middleware. The data sources (both CNC machines or CAD repositories in this case) have complete control of the data and it will be completely up to the repository admin to decide what data will be shared, whether to provide a download access or not and lastly whether to share data with a fog node or not.

5. Conclusion

In this paper, we present a decentralized data distribution architecture for democratizing the data generated in a smart manufacturing environment. We have implemented the fog computing architecture as per the IEEE standards.

In the case study, we have presented the ease of distributing both structured and unstructured data through the architecture and making the data requested by the end user of a service provider available in a read- only format to the client. The architecture allows the manufacturer to have control over their own data. The Kafka client embedded in the fog nodes allows communication with other fog nodes within the layer between the IoT enabled manufacturing machines and the main Cloud. This ensures that the fog nodes can inter-communicate to deliver query results.

Fog computing being a fledgling technology involves several different communication protocols for heterogeneous data transmission between fog to fog and fog to cloud which makes a coherent data distribution system possible. In addition to communication protocols, another compelling challenge lies in enhancing the user experience and hiding the backend working of the layers.

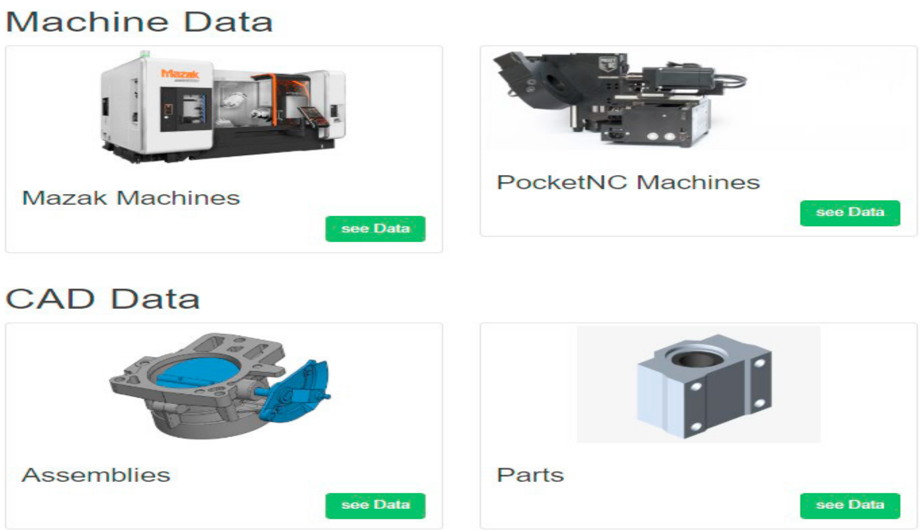


Fig. 8. The data streams provided to the end user corresponding to data from different fog nodes

6. Future Work

In the future, our efforts will be focused on working towards making the architecture more industry scalable using cloud orchestration. An important consideration is to test the decentralized data distribution architecture on networking performance and latency of response to complex query requests. A major challenge lies ahead in domain of the security of the architecture. Since the number of things connected with the distributed architecture will be huge, security will remain a main issue of concern. Further, the type of data to be distributed via the architecture will be heterogenous and with increase in data sources may require alternate network configurations for optimal performance, which might make node management tedious.

7. References

[1] Ray S., Jin Y. and Raychowdhury A.,(2016), "The Changing Computing Paradigm With Internet of Things: A Tutorial Introduction," in *IEEE Design & Test*, vol. 33, no. 2, pp. 76-96.doi: 10.1109/MDAT.2016.2526612

[2] Plebani, P., García-Pérez, D., Anderson, M., Bermbach, D., Cappiello, C., Kat, R.I., Pallas, F., Pernici, B., Tai, S., & Vitali, M. (2017). Information Logistics and Fog Computing: The DITAS* Approach. *CAiSE-Forum-DC*.

[3] Aazam M., Zeadally S.,Harras K. A.. (2018). Fog Computing Architecture, Evaluation, and Future Research Directions. IEEE Communications Magazine. 56. 46-52. 10.1109/MCOM.2018.1700707.

[4] Khan M., Wu X., Xu X. and Dou W. (2017), "Big data challenges and opportunities in the hype of Industry 4.0," 2017 IEEE International Conference on Communications (ICC), Paris, 2017, pp. 1-6. doi: 10.1109/ICC.2017.7996801

[5] Zhang P., Liu J. K.,. Yu F. R, Sookhak M.,. Au M. H and Luo X., (2018) "A Survey on Access Control in Fog Computing," in *IEEE Communications Magazine*, vol. 56, no. 2, pp. 144-149, Feb. 2018.doi: 10.1109/MCOM.2018.1700333

[6] Osanaiye O., Chen S., Yan Z., Lu R., Choo K.R. and Dlodlo M.,(2017) "From Cloud to Fog Computing: A Review and a Conceptual Live VM Migration Framework," in *IEEE Access*, vol. 5, pp. 8284-8300, 2017. doi: 10.1109/ACCESS.2017.2692960

[7] He S., Cheng B., Wang H., Xiao X., Cao Y. and Chen J.(2018), "Data security storage model for fog computing in large-scale IoT application," IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), Honolulu, HI, 2018, pp. 39-44.doi: 10.1109/INFCOMW.2018.8406927

[8] Osuwa A. A, Ekhoragbon E. B. and Fat L. T.(2017), "Application of artificial intelligence in Internet of Things," 2017 9th International Conference on Computational Intelligence and Communication Networks (CICN), Girne, 2017, pp. 169-173.doi: 10.1109/CICN.2017.8319379

[9] Calo S. B, Touna M., Verma D. C. and Cullen A., (2017) "Edge computing architecture for applying AI to IoT," 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, pp.3012-3016. doi: 10.1109/BigData.2017.8258272

[10] Zhang T. and Zuckerman D.N.(2018), "Fog computing and networking — What's next*?,"* 2018 IEEE International Conference on Consumer Electronics*(ICCE)*,LasVegas,NV,pp.1-1. doi: 10.1109/ICCE.2018.8326338

[11] Sabiri K. , Benabbou F. , (2015). Methods Migration from On-premise to Cloud. IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 17, Issue 2, Ver. IV ,PP 58-65

[12] Hassan S. F. , Fareed R., (2018).Video Streaming processing using fog computing. *2018* International Conference on Advanced Science and Engineering *(ICOASE)*, Duhok, 2018, pp. 140-144. doi: 10.1109/ICOASE.2018.8548869

[13] Bonomi F., Milito R., Natarajan P., Zhu J. (2014) Fog Computing: A Platform for Internet of Things and Analytics. In: Bessis N., Dobre C. (eds) Big Data and Internet of Things: A Roadmap for Smart Environments. Studies in Computational Intelligence, vol 546. Springer, Cham

[14] Pengfei H., Sahraoui D., Huansheng N., Tie Q.,(2017) Survey on fog computing: architecture, key technologies, applications and open issues, Journal of Network and Computer Applications, Volume 98,2017,Pages 27-42,ISSN 1084-8045,
https://doi.org/10.1016/j.jnca.2017.09.002.

[15] Bermbach D. et al. (2018) A Research Perspective on Fog Computing. In: Braubach L. et al. (eds) Service-Oriented Computing – ICSOC 2017 Workshops. ICSOC 2017. Lecture Notes in Computer Science, vol 10797. Springer, Cham

[16] Ryll M., Ratchev S.(2008). Application of Data Distribution Service for Flexible Manufacturing Automation, International Journal of Industrial and Manufacturing Engineering, Vol:2, No:5

[17] Gaitan N., Gheorghita V., Ungurean I.,(2016). An IoT Middleware Framework for Industrial Applications, International Journal of Advanced Computer Science and Applications. 7. 10.14569/IJACSA.2016.070905.

[18] Yang J., Sandström K., Nolte T. and Behnam M.,(2012). "Data Distribution Service for industrial automation," Proceedings of 2012 IEEE 17th International Conference on Emerging Technologies & Factory Automation, Krakow, 2012, pp. 1-8. doi: 10.1109/ETFA.2012.6489544

[19] Michelberger B., Andris R., Girit H., Mutschler B., (2013).A literature survey on information logistics. In: Proceedings of the 16th International conference of business information systems, BIS 2013, Poznan, Poland, June 19–21, Springer International Publishing, Heidelberg.

[20] Bonomi F., Milito R., Zhu J., Addepalli. S.,(2012). Fog computing and its role in the internet of things. In Proceedings of the first edition of the MCC workshop on Mobile cloud computing (MCC '12). ACM, New York, NY, USA, 13-16. DOI=http://dx.doi.org/10.1145/2342509.2342513

[21] Pizoń, J., & Lipski, J. (2016). Perspectives for fog computing in manufacturing. *Applied Computer Science, 12*(3), 37-46.

[22] Yu C., Xu X., X, Lu Y., (2015). "Computer-Integrated Manufacturing, Cyber Physical Systems, and Cloud Manufacturing—Concepts and Relationships," Manuf. Lett., 6, pp. 5–9

[23] O'Donovan P., Bruton K., Gallagher C., O'Sullivan T. J, (2018). A fog computing industrial cyber-physical system for embedded low-latency machine learning Industry 4.0 applications. Manufacturing Letters.15.10.1016/j.mfglet.2018.01.005.

[24] Dazhong W., Shaopeng L., Zhang Li., Terpenny J, Gao R.X, Kurfess T., Guzzo, J.A.(2017)A fog computing-based framework for process monitoring and prognosis in cyber-manufacturing, Journal of Manufacturing Systems, Volume 43, Part 1,2017,Pages 25-34,ISSN 0278-6125,https://doi.org/10.1016/j.jmsy.2017.02.011.

[25] Li L., Ota K., and Dong M.,(2018). "Deep Learning for Smart Industry: Efficient Manufacture Inspection System with Fog Computing," in *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4665-4673.

[26] Nath, S.B., Gupta, H., Chakraborty, S., & Ghosh, S.K. (2018). A Survey of Fog Computing and Communication: Current Researches and Future Directions. *CoRR, abs/1804.04365*.

[27] Nandyala C.S. and Kim H K.,(2016). "From cloud to fog and IoT-based real-time u-healthcare monitoring for smart homes and hospitals," International Journal of Smart Home, vol. 10, no. 2, pp. 187–196.

[28] Gu L., Zeng D., Guo S., Barnawi A. and Xiang Y. (2017), "Cost Efficient Resource Management in Fog Computing Supported Medical Cyber-Physical System," in *IEEE Transactions on Emerging Topics in Computing*, vol. 5, no. 1, pp. 108-119, Jan.-March 2017. doi: 10.1109/TETC.2015.2508382

[29] Cao Y., Hou P., Brown D., Wang J., and Chen S., (2015). "Distributed analytics and edge intelligence: Pervasive health monitoring at the era of fog computing," in Proceedings of the 2015 ACM Workshop on Mobile Big Data. ACM, pp. 43–48.

[30] Clemente J., Valero M., Mohammadpour J., Li X. and Song W., "Fog computing middleware for distributed cooperative data analytics," *2017 IEEE Fog World Congress (FWC)*, Santa Clara, CA, 2017, pp. 1-6. doi: 10.1109/FWC.2017.8368520

[31] OpenFog Consortium Architecture Working Group OpenFog Architecture Overview White Paper, 2017 https://www.openfogconsortium.org/wpcontent/uploads/OpenFog_Reference_Architecture_2_09_17-FINAL.pdf

[32] IEEE 1934-2018 -IEEE Standard for Adoption of OpenFog Reference Architecture for Fog Computing,2018. https://standards.ieee.org/news/2018/ieee1934-standard-fog-computing.html

[33]Wei, M., Zhang, S., Wang, P., & Kim, K. (2018). An authentication and key agreement mechanism for OPC Unified Architecture in industrial Internet of Things. International Journal of Distributed Sensor Networks. https://doi.org/10.1177/1550147718754793

[34] Loopback API - https://apidocs.strongloop.com/

[35] Zhang, Q., Liu, L., Pu, C., Dou, Q., Wu, L., & Zhou, W. (2018). A Comparative Study of Containers and Virtual Machines in Big Data Environment. 2018 IEEE 11th International Conference on Cloud Computing (CLOUD), 178-185.

[36] Cappiello C., Pernici B., Plebani P., Vitali M..,(2017).Utility-driven data management or data-intensive applications in fog environments. In: Advances in Conceptual Modelling, ER Workshops, pp 216–226. 10.1007/978-3-319-70625-2_20.

[37] Vaughan-nichols S.J, "New Approach to Virtualization Is a Lightweight," in *Computer*, vol. 39, no. 11, pp. 12-14, Nov. 2006. doi: 10.1109/MC.2006.393

[38] Munir A., Kansakar P.and Khan S.U, "IFCIoT: Integrated Fog Cloud IoT: A novel architectural paradigm for the future Internet of Things.," in *IEEE Consumer Electronics Magazine*, vol. 6, no. 3, pp. 74-82, July 2017. doi: 10.1109/MCE.2017.268498

[39] Cao Y. et al.,2015. "FAST: A Fog Computing Assisted Distributed Analytics System to Monitor Fall for Stroke Mitigation," Proc. 10th IEEE International Conf. Networking, Architecture, and Storage (NAS 15), 2015, pp. 2–11

[40] N. Marz N, J.,2014. Warren.Principles and Best Practices of Scalable Realtime Data Systems, Manning Publications Co., Shelter Island.

[41] Fielding R. T., Software, D. , Taylor R. N.. (2002). Principled design of the modern Web architecture. Proceedings - International Conference on Software Engineering. 2. 10.1109/ICSE.2000.870431.

[42] Dizdarevic, J., Carpio, F., Jukan, A., & Masip-Bruin, X. (2018). A Survey of Communication Protocols for Internet of Things and Related Challenges of Fog and Cloud Computing Integration. *ACM Comput. Surv., 51*, 116:1-116:29.

[43] Sarkar S., Chatterjee S., Misra S., (2015). Assessment of the Suitability of Fog Computing in the Context of Internet of Things. IEEE Transactions on Cloud Computing. PP. 1-1. 10.1109/TCC.2015.2485206.

[44] Silva, L.D., Vasconcelos, R., Alves, L.R., André, R., & Endler, M. (2013). A DDS-based middleware for scalable tracking, communication and collaboration of mobile nodes. *Journal of Internet Services and Applications, 4*, 1-15.

[45] Singh S. K. ,(2017).A Cloud-Based Middleware Architecture for connecting manufacturing machines to enable digital thread in Product Lifecycle Management, MS Thesis, North Carolina State University.

[46] Pérez J. L, Gutierrez-Torre A., Berral J. L., Carrera D.(2018),A resilient and distributed near real-time traffic forecasting application for Fog computing environments, Future Generation Computer Systems, Volume 87,2018,Pages 198-212,ISSN 0167-739X, https://doi.org/10.1016/j.future.2018.05.013.