An Extensible Edge Computing Architecture: Definition, Requirements and Enablers

Volkan Gezer and Jumyung Um and Martin Ruskowski
Innovative Factory Systems (IFS)
German Research Center for Artificial Intelligence (DFKI)
Kaiserslautern, Germany
Emails: {name.surname}@dfki.de

***Abstract*—Cloud computing is highly being used for several years for various purposes. From daily tasks, such as reading e-mails, watching videos to the factory automation and device control, it changed where the data is being processed and how it is accessed. However, increasing number of connected devices brings problems, such as low Quality of Service (QoS) due to infrastructure resources and high latency because of the bandwidth limitations. The current tendency to solve the problems that the Cloud computing has is performing the computations as close as possible to the device. This paradigm is called Edge Computing. There are several proposed architectures for the Edge Computing, but there is no an accepted standard by the community or the industry. Besides, there is not a common agreement on how the Edge Computing architecture physically looks like. In this paper, we describe the Edge Computing, explain how its architecture looks like, its requirements, and enablers. We also define the major features that one Edge Server should support.**

Keywords–Edge computing; requirements; enablers; Fog computing.

I. INTRODUCTION

With the increased tendency towards Internet of Things (IoT), number of connected devices to the Internet are increasing day by day. In 1992, the connected devices count was around one million which went up to 500 million in 2003 with increased usage of notebooks. Later, IoT became even more popular and made three billions of devices connected. In 2012, with the inclusion of wearable devices this number went high as 8.7 billion. In 2013, this number was 11.2 billion thanks to connected home appliances and in 2014, 14.4 billion with smart grids. The numbers increased in the upcoming years due to involvement of small personal objects, such as toothbrushes, traffic lights, and table watches. Finally, even door levers are expected to be part of smart objects in 2020 [1].

Connected devices are expected to be around 50 billion by 2020 [1][2]. This number is high as the Cyber-Physical Systems (CPS) and more intelligent components being used even for simple tasks. Using different standards, a single infrastructure to keep the system reliable is becoming even more complex, causing difficult and costly maintenance. Relying on a single information technology (IT) infrastructure can also increase the downtime of communication which disrupts the service leading to non-productive time. The bandwidth for communication is also becoming a problem to transmit that amount of data.

Cloud Computing [3] is an emerging technology which allows machines/people to access the data ubiquitously. It enables on-demand sharing of available computing and storage resource among its users which could be either human or machine, or even both. Today, it is even possible for a simple device to share its status or get information over Internet with millions of users. In Cloud Computing, the communication between the device and the infrastructure which provides the service is direct, without involvement of other tiers. However, increased usage of Cloud increases latency and the load on the server and on the network. Having billions of devices and processing the data produced by each of them is a troublesome task for centralized systems [4]. Figure 1 shows some examples for Cloud Computing, such as E-Mail services, Cloud Storage systems, Video hosting web sites, etc.

A layer is a logical organisation of set of services, devices, or software with the same/similar specific functionality, mainly defined for abstraction of tasks. A tier is, however, a physical deployment of layers for scalability, security and to balance performance [5].

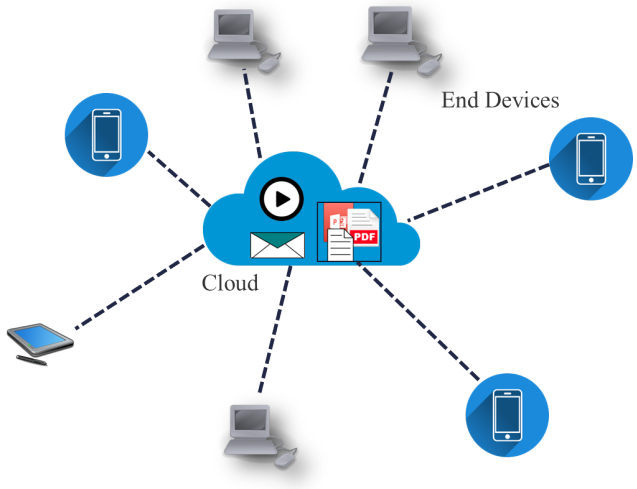

Figure 1. Some of daily usage examples of Cloud Computing, such as e-mails, music/video streaming, and data storage.

Edge Computing is a recent paradigm, which moves computing application and services from centralized units into the logical extremes or at the closest locations to the source and provides data processing power there. It adds an additional tier between the Cloud and the end-devices as depicted in Figure 2. Increase in Edge nodes within a location will reduce the number of devices connected to a single Cloud and eliminate the problems of the Cloud Computing. Examples to Edge Computing can be listed as Smart Cities, Machine to Machine communication, Security Systems, Augmented Reality, Wearable Health Care Systems, Connected Cars, and Intelligent Transportation. For example, a plane produces gigabytes of

 ISBN: 978-1-61208-598-2

data per second [6], which cannot be handled by a single base infrastructure due to bandwidth limitations. Another example is a Formula One car which produces approximately 1.2 GB/s data [7] that requires gathering, analysis, and acting in-time to stay competitive in the race [8]. Edge Computing is believed to solve these issues by aggregating and pre-processing the data in Edge, before transmitting to the Cloud or even deciding the next steps on the Edge.

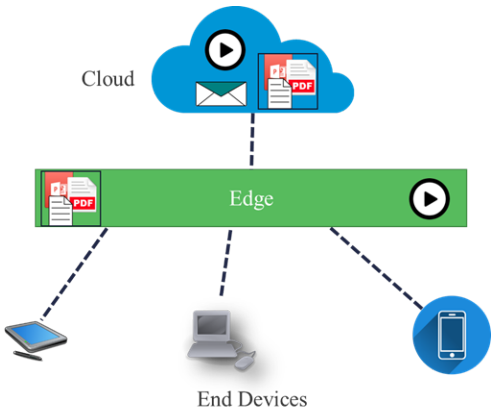

Figure 2. A simplified version of communication using Edge Computing.

Both Edge Computing and Cloud Computing are part of Internet of Things (IoT) and allow accessibility of the data ubiquitously. To build an architecture, the issues on the current Cloud or IoT systems must be identified, requirements must be specified, enabling technologies must be listed, and then a concept must be given. Later, the concept can be implemented in an architecture, validated, and evaluated.

This paper presents an ongoing work on Edge Computing with its clear description. It also explains its requirements and enablers to solve the introduced issues because of high usage of Cloud and IoT.

The paper is structured as follows. In Section II, a short overview on related work in Edge Computing domain is given. In Section III, the concept of Edge Computing is explained. Later, in Section IV, its requirements, and in Section V, enablers are explained. In Section VI, the major functionalities of the proposed architecture is explained. The paper is concluded in Section VII with the future work.

II. Related Work

Although usage of the term “Edge Computing” is recent, there are already several proposed architectures available, each considering different aspects to meet the requirements of the Edge Computing. Below, some of the existing proposed architectures will be discussed.

The architecture proposed by IBM considers the requirements for autonomy and self-sufficiency of production sites. The architecture is three-layered to balance the workload between the Edge, the Plant, and the Enterprise. The challenges of the architecture are listed as productivity gains for high throughput, failure prevention for reliable system and high product quality, and flexibility while hiding the complexity and allowing reconfiguration without a lot of effort [9].

Another reference architecture is proposed by OpenFog Consortium [10]. This architecture names the core principles as pillars. Pillars group requirements within their scope. These pillars are Security, Scalability, Openness, Autonomy, Agility, and Programmability. OpenFog Reference Architecture is proposed by covering industrial use cases.

Another recent initiative to build a common platform for Industrial IoT Edge Computing is EdgeX Foundry [11]. It was launched by Linux Foundation and initial contribution made by Dell. However, similar to OpenFog Consortium, it is also open for new memberships. EdgeX Foundry is a vendor-neutral open source software platform that interacts at the Edge of the network. It defines its requirements in architectural tenets as follows: platform agnostic in terms of hardware and operating system, flexible in terms of replacability, augmentability, or scalability up and down, capable in storing or forwarding data, intelligent to deal with latency, bandwidth, and storage issues, secure, and easily manageable. A similar framework called Liota is being developed by VMware and it also aims at easy to use, install, and modify. Secondarily, it targets for a general, modular and enterprise-level quality. This framework is also open source and governed by VMware [12].

The aim in this research is not simply to build another architecture, but to analyse the existing architectures and consider industrial requirements to make up a generic reference architecture which is vendor-independent and extensible. The architecture is also able to execute real-time tasks. To the best of our knowledge, this is not considered in any of the aforementioned reference architectures.

III. Concept

One of the main goals of Edge Computing is to reduce latency and to keep the Quality of Service (QoS) as high as possible. As seen in Figure 1, in Cloud Computing, the Cloud infrastructure communicates with the end-devices directly. Edge Computing intends to solve the issues of Cloud Computing or IoT by adding an additional tier between the IoT devices and back-end infrastructure for computing and communication purposes. As depicted in Figure 3, this tier also has intermediate components for the first gathering, analysis, computation of the data. These intermediate components are called *Edge Servers*. Several architecture types for IoT-enabled applications are proposed [13]. In this paper, a three-tier architecture is used.

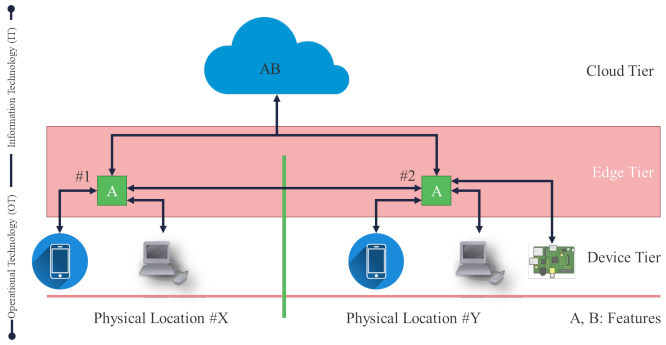

Figure 3. Edge Computing is an additional tier between Cloud and the Devices. The Edge Servers can be in the same or different physical locations.

As seen in Figure 3, the proposed architecture for Edge

Computing consists of *Cloud Tier*, *Edge Tier*, and *Device Tier*. In the Device Tier, there are end-user devices. The green blocks in the Edge Tier are Edge Servers. These servers gather, aggregate, analyse, and process the data before offloading them to the Cloud Tier. The end-devices can be in the same location, or in different physical locations as depicted in the figure. When an end-device needs to communicate with the Cloud, first, the request is sent to the Edge Server which is at the closest location. Then, if the Edge Server is capable of completing the task by itself, it automatically handles the data and responds to the end-device with the result. If not, the data is offloaded to another server in the same tier provided that it exists. Otherwise, the data is offloaded to the Cloud. The decision process is made by considering available resources in other available servers in the same network, physical distance, and time requirements.

Assume that the Cloud provides functionalities A and B. When one of the devices in *Physical Location X* intends to do task *B*, first the data is passed to the server #1. Since this server is not capable of performing this task, it passes the data either to the Cloud. As Cloud is capable of performing task B, the data is processed here and sent back to the originating end-device. The challenge here is to decide on functionalities in the Edge Tier by keeping the costs at minimum and the QoS at maximum. However, deciding on the count and available resources of Edge Servers are also big challenges and big trade-offs. There are several aspects to consider before passing the data to the Cloud. For example, if a device located at Y needs task A to be done, and if the Edge Server #2 is busy with servicing other two connected devices, another trade-off will be existent. In this case, the server #2 needs to offload the task either onto server #1 or the Cloud. However, depending on the urgency of the task, the server #2 needs to calculate a function to decide on the best recipient of the data. According to this, the function should consider the priority of the task, resource utilization of the servers, computing cost for the task, and the physical distance or distance cost of the servers that is going to be used.

IV. REQUIREMENTS

Edge Computing is a paradigm which uses Cloud Computing technologies and gives more responsibilities to the Edge tier. These responsibilities are namely, computing offload, data caching/storage, data processing, service distribution, IoT management, security, and privacy protection [4].

Without limiting the Cloud Computing features, Edge Computing needs to have the following requirements, some of which are also defined for Cloud Computing [14][15]:

1) Interoperability: Servers in Edge Computing can connect with various devices and other servers. In Cloud Computing, IoT allows countless number of devices to communicate with humans or each other. This creates a big market for manufacturers of these devices. For this reason, there is the issue of interoperability with connected device using different communication protocols. Advanced Message Queuing Protocol (AMQP), Message Queue Telemetry Transport (MQTT), and TCP/IP are widely used and should supported by Edge Computing. Using a widely-used and widely-known standard will remove the technology and language barriers, increasing interoperability among the devices.

2) Scalability: Similar to Cloud services, Edge Computing will also need to be adapted for the size of its users and sensors. First deployment enables small number of users and devices while few Edge Servers should handle higher number. Additional deployment of Edge Servers is costly and small number of Edge Servers is desirable in terms of economical aspects. For this reason, high scalability is also mandatory.

3) Extensibility: Computing technology is developing rapidly. After 2-3 years of deployment, clock speeds, memory size and program size increase, too. Easy deployment of new services and new devices with small effort is required for essential goal of Edge Computing. New functions and devices should be integrated without (re)configuration of the Edge network. Therefore, the system should allow extensibility with hardware and software components.

4) Abstraction: For the seamless control and communication, the abstraction of each Edge Node and group of nodes is required. Moreover, abstraction helps the topology of an Edge network to be flexible and reconfigurable. Fundamentally, an Edge node is located between device tier and Cloud tier. In other words, an Edge tier is a border between Information Technology (IT) and Operational Technology (OT). This tier can consist of one or more Edge nodes and groups. In this case, one Edge node of the group can share tasks or nodes in the group can be prioritized. Utilization of Application Programming Interfaces (APIs) in abstraction is useful to provide backward compatibility for the new functionalities or big changes in the architecture.

5) Time sensitiveness: Below OT, the operations may be near-real-time or real-time. Edge Computing is expected to solve time issues which Cloud computing cannot guarantee. Unlike Cloud Computing, physically close distance is one strength of reliable and fast communication without worrying about traffic problem. Video streaming service is one of expected applications of Edge Computing. It is required for real-timeness of the service provision. In addition, time-sensitiveness adds big benefits to providers of reactive services, such as location-based advertisements and user-status based guide systems.

6) Security & Privacy: Using Cloud Computing services has a trade-off for enterprises like manufacturing and high-tech companies because there is a concern about the leakage of high knowledge and business activities outside their own organization. Edge Computing is a way to secure data contents, which is different from firewall which only controls external access into the network. It is also important to isolate the data by preventing access from even non-authorized users.

7) Reliability: Edge Servers provide real-time or non-real-time control for the devices. Real-time tasks may be vital which involve human safety. Therefore, it is vital to have a reliable system which reacts when it is needed and how it is needed. The physical reliability requirements for Edge servers providing services is similar to Cloud Computing. Harsh environments, such as factories and construction yards, require water-proof ceiling, fanless computers and dust-proof system. In power plant, magnetic shield is equipped by sensor gateways.

8) Intelligence: Multi-sensor generates tremendous amount of data and uploads into Cloud, directly. It causes network congestion and heavy load on the Cloud server. Edge Computing

 ISBN: 978-1-61208-598-2

supports first and second filtering of these data by converting into higher level of data contents. Data filtering is implemented by rule-based engines or machine learning algorithms. In the case of multi-camera system like security systems, Edge Computing supports image processing, computer vision and enables object detection before transferring the data into the Cloud. Another example is predicting the failure or abnormalities in a production line by analysing the sensor data and taking the precautions for prevention or informing the user. These kinds of intelligent functions are necessary for Edge Computing.

9) Power: Unexpected shutdown or blackout is the cause of breakdown of Edge Server. Uninterruptible power supply (UPS) is required to give an ample amount of time to protect the electronic units and data storage in case of an unexpected shutdown due to power outage.

V. ENABLERS

Edge Computing uses wide range of technologies and brings them together. Within this domain, Edge Computing utilizes many technologies, such as wireless sensor networks (WSN), mobile data acquisition, mobile signature analysis, Fog/Grid Computing, distributed data operations, remote Cloud services, etc. Additionally, it combines the following protocols and terms:

1) 5G communication: It is the fifth generation wireless system which aims at higher capacity, lower power consumption, and lower latency compared to the previous generations. Due to increased amount of data between the data, 5G is expected to solve traffic issues which arose with the increased number of connected devices.

2) PLC protocols: Object Linking and Embedding for Process Control Unified Architecture (OPC-UA) is a protocol developed for industrial automation. Due to its openness and robustness, it is widely used by industries in the area of oil and gas, pharmaceutical, robotics, and manufacturing.

3) Message queue broker: MQTT and TCP/IP are popular message protocols of smart sensors and IoT devices. Supporting these message brokers, Edge Computing increases the device count that it connects. For the problem of MQTT security, AMQP is useful in the communication with Cloud Computing server.

4) Event processor: After messages of IoT arrive in the Edge server, event processor analyses those messages and creates semantic events using pre-defined rules. EsperNet, Apache Spark, and Flink are some examples for this enabler.

5) Virtualisation: Cloud services are deployed as virtual machines on a Cloud server or clusters. Using virtual machines allow running multiple instances of operating systems (OS) on the same server.

6) Hypervisor: As well as virtual machine, performance evaluation and data handling are required and realized by hypervisor to control virtual machines in the host computer.

7) OpenStack: Managing multiple resources could be challenging. OpenStack is a Cloud operating system that helps control of pools of computing and storage resources at ease through a control panel and monitoring tools.

8) AI platform: Rule-based engine and Machine learning platform supports data analysis in local level. As stated in Section IV, this is quite important to reach one of the goals of Edge Computing which is to gather, analyse, and perform the first filtering of the data.

9) Hyperledger: Blockchain technology is currently used for highly sensitive areas, such as digital currencies like BitCoin. It is also considered as useful for the data protection in Cloud Computing. By using this technology, secure data can be shared with external persons and servers with high security.

10) Docker: Virtual machines work with installation of operating systems. Unlike virtual machines, Docker is a Container as a Service (CaaS) which can use a single shared operating system and run software in isolated environment. It only requires the libraries of the software which makes it a lightweight system without worrying about where the software is deployed.

VI. ARCHITECTURE DESIGN

Edge Computing adds an additional tier between the Cloud and IoT devices for computing and communication. The data produced by the devices themselves are not directly sent to the Cloud or back-end infrastructure, but initial computing is performed on this tier. Considering the number of connected devices and the data they produced, this tier is used to aggregate, analyse, and process the data before sending it into the upper layer, the infrastructure.

Figure 4 depicts the proposed core functionalities for an Edge Server.

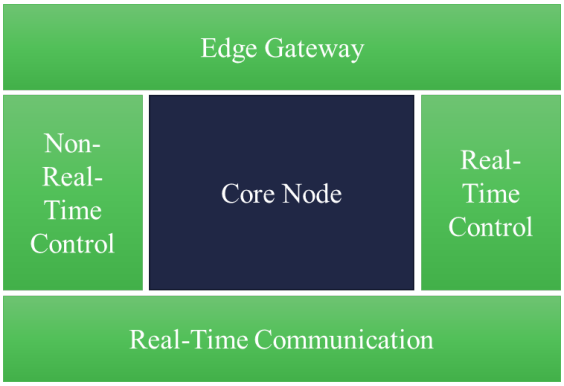

Figure 4. View of the proposed extensible Edge server architecture with its major functionalities, where green blocks extend the functionalities for the blue core node.

The proposed Edge Server architecture is to be designed modular and should provide functionalities for real-time and non-real-time control, as well as real-time communication. Core node runs on an operating system and tracks resources and makes decisions on where to execute a task. In the proposed architecture, addition of a new hardware or software modules enable new functionalities and improve the usability of the server. For example, in the case that machine learning algorithms are desired to be executed on the server, connecting a dedicated artificial intelligence (AI) module with dedicated Graphics Processing Unit (GPU) should require none to minimal configuration to be active.

 ISBN: 978-1-61208-598-2

As mentioned in Section IV, scalability is quite important to accomplish the tasks. In the scope of scalability, one server is expected to be aware of its neighbouring servers along with their functionalities. Using the previous example, in case an AI module is connected to one server, other servers are informed with this functionality and they can utilize this server more often for AI-related tasks. The decision, of course, depends on the conditions required by the task, such as deadline.

VII. Conclusion and Future Work

Edge Computing is a recent term which moves the services from the Cloud to the device as close as possible. It is a borderline between the Cloud and the device tier. Although the Cloud Computing has brought many advantages in the previous years, increased number in the connected devices raised some issues, such as latency and low QoS problems. Edge Computing is believed to solve these issues by analysing the issues and considering the requirements of real world use cases.

This paper showed an ongoing work on how *Edge Computing* physically looks like together with its requirements and enablers. It also explained the basics on how the communication between the end-devices and Edge servers are expected to be.

There are already several existing proposed architectures in the domain of Edge Computing, such as EdgeX Foundry, Liota, and OpenFog Reference Architecture. Although they are also extensible and they allow inter-connectivity, they do not talk about the real-timeliness of the architectures. This work will be focusing on real-time computing and communication for the given tasks. Of course, it will also be available for non-real-time tasks. The work is being developed by considering the real-world use cases of the industrial partners. The validation will be performed with these use cases and the comparison with the legacy systems will be made.

In the future, internal software and hardware components for the Edge Server will be decided. Later, they will be simulated as an initial work for the architecture design. Next, the software components will be individually implemented in the simulation environment. By analysing the simulator results, a hardware benchmarking will be performed and a hardware will be chosen to be used as the Edge Server solution. The final task will be to realize the components by deploying them on the chosen hardware.

Acknowledgment

This research was funded in part by the H2020 program of European Union, project number (project FAR-EDGE). The responsibility for this publication lies with the authors. The project details can be found under project website at: `http://www.far-edge.eu`

References

[1] NCTA, “The Growth of The Internet of Things,” Infographic, May 2014, [retrieved: Sep 2017]. [Online]. Available: https://www.ncta.com/platform/industry-news/infographic-the-growth-of-the-internet-of-things/

[2] D. Evans, “The Internet of Things - Cisco,” Cisco, White Paper, April 2011, [retrieved: Sep 2017]. [Online]. Available: https://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf

[3] M. Peter and G. Timothy, “The nist definition of cloud computing,” in National Institute of Standards and Technology Technical report, September 2011.

[4] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, “Edge Computing: Vision and Challenges,” IEEE Internet of Things Journal, vol. 3, no. 5, October 2016, pp. 637–646.

[5] R. Lhotka, “Should all apps be n-tier?” Blog, 2005, [retrieved: Sep 2017]. [Online]. Available: http://www.lhotka.net/weblog/ShouldAllAppsBeNtier.aspx

[6] S. Higginbotham, “Sensor Networks Top Social Networks for Big Data,” Article, 2010, [retrieved: Sep 2017]. [Online]. Available: https://gigaom.com/2010/09/13/sensor-networks-top-social-networks-for-big-data-2/

[7] T. Valich, “Big Data In Planes: New P&W Gtf Engine Telemetry To Generate 10GB/s,” Article, 2015, [retrieved: Sep 2017]. [Online]. Available: https://vrworld.com/2015/05/08/big-data-in-planes-new-pw-gtf-engine-telemetry-to-generate-10gbs/

[8] F. Bi, “How Formula One Teams Are Using Big Data To Get The Inside Edge,” Article, 2017, [retrieved: Sep 2017]. [Online]. Available: https://www.forbes.com/sites/frankbi/2014/11/13/how-formula-one-teams-are-using-big-data-to-get-the-inside-edge/

[9] I. C. A. Center, “IBM: Internet of Things,” Cloud Garage Method, 2017, [retrieved: Sep 2017]. [Online]. Available: https://www.ibm.com/devops/method/content/architecture/iotArchitecture/industrie_40

[10] “OpenFog Consortium Reference Architecture,” Website, 2017, [retrieved: Sep 2017]. [Online]. Available: https://www.openfogconsortium.org/ra/

[11] “EdgeX Foundry Architectural Tenets,” EdgeX Foundry Wiki, 2017, [retrieved: Sep 2017]. [Online]. Available: https://wiki.edgexfoundry.org/display/FA/Introduction+to+EdgeX+Foundry

[12] “VMware Introduces Liota,” Website, 2017, [retrieved: Sep 2017]. [Online]. Available: https://www.vmware.com/radius/vmware-introduces-liota-iot-developers-dream/

[13] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of things: A survey on enabling technologies, protocols, and applications,” IEEE Communications Surveys Tutorials, vol. 17, no. 4, 2015, pp. 2347–2376.

[14] G. Orsini, D. Bade, and W. Lamersdorf, “Context-Aware Computation Offloading for Mobile Cloud Computing: Requirements Analysis, Survey and Design Guideline,” Procedia Computer Science, vol. 56(1), December 2015, pp. 10–17.

[15] J. Shamsi, M. A. Khojaye, and M. A. Qasmi, “Data-intensive cloud computing: Requirements, expectations, challenges, and solutions,” Journal of Grid Computing, vol. 11, no. 2, Jun 2013, pp. 281–310.

 ISBN: 978-1-61208-598-2