Class of Service in Fog Computing

Judy C. Guevara, Luiz F. Bittencourt and Nelson L. S. da Fonseca
Institute of Computing - State University of Campinas
Campinas, Brazil
jguevara@lrc.ic.unicamp.br, bit@ic.unicamp.br, nfonseca@ic.unicamp.br

***Abstract*—Although Fog computing specifies a scalable architecture for computation, communication and storage, there is still a demand for better Quality of Service (QoS), especially for agile mobile services. Both industry and academia have been working on novel and efficient mechanisms for QoS provisioning in Fog computing. This paper presents a classification of services according to their QoS requirements as well as Class of Service for fog applications. This will facilitate the decision-making process for fog scheduler, and specifically to identify the timescale and location of resources, helping to make scalable the deployment of new applications. Moreover, this paper introduces a mapping between the proposed classes of service and the processing layers of the Fog computing reference architecture. The paper also discusses use cases in which the proposed classification of services would be helpful.**

Keywords— Fog Computing, Resource Management, Cloud Computing, Internet of Things (IoT), Edge Computing, Mobile Cloud, Classes of Service, Quality of Service (QoS).

I. INTRODUCTION

Although Cloud Computing has become a mature technology offering elastic infrastructure in a pay-as-you-go business model, the fundamental limitation of this technology is the connectivity between the Cloud and the end devices [1]. Such connectivity is provided by the Internet, which is usually associated with high transfer delays not suitable to latency-sensitive applications. Fog Computing has emerged as a new computing paradigm to address some of the limitations of Cloud computing, bringing the Cloud to the edge. Fogs extend computing, storage, and network services provided by Clouds, making possible processing near to the end-user. Fog computing differs from previous initiatives that aimed at bringing the Cloud close to the end devices such as edge computing by allowing new layers to the end-to-end principle.

Fog has been a promising technology for the "Cloud to the Thing" as a recognition that Cloud computing is not viable to the Internet of Thing due to several issues such as the huge volume of data transfer through the Internet. However, fogs will support a wide spectrum of applications such as those involving Content Delivery Networks (CDNs), connected cars, e-health, smart grids, and smart home to name a few. Fog computing will allow not only time-sensitive applications but also will bring efficiency, rapid innovation, affordable scalability, and infrastructure to cognition to current computing and communication systems. A variety of new and heterogeneous applications and services will benefit from the deployment of fogs [2]. Besides low latency, these applications will need interactivity, location awareness, personalization, mobility, control and ubiquity to access content and services, and will demand a large spectrum of Quality of Service (QoS) requirements.

Fog and Cloud will be integrated, composing a distributed computational environment in which nodes have different processing and storage capacities, and can be connected through multiple switches and links at different layers [3], [4]. Fogs will be composed by densely distributed nodes, which can vary from proxies, mini-clouds, smart edge devices, routers, cellular base stations and access points [5].

These devices will have diverse computing and storage capabilities as well as network connectivity. Tasks, services, and code need to be scheduled on these resources as well as on resources in the Cloud to satisfy the needs of end-devices. In this context, scheduling becomes more challenging, since the scheduler must consider multiple variables to guide the decisions on where tasks and virtualized software (e.g. virtual machines or containers) should run, considering the state of resource availability and the cost of resources [6].

To map the Quality of Service requirements of applications onto fog/Cloud resources, Class of Service need to be defined to support fog providers and administrators to cope with such demands as well as to efficiently manage the available resources. The employment of Class of Services (CoS) is common in communications network technologies that support QoS, and the employment of CoS will empower administrators of both public and private fogs to build efficient systems.

In line with that, this paper defines a set of Class of Service for fog computing. These classes were defined after analyzing the requirements of potential applications that will run on fogs. This is a first step in the definition of essential components of Quality of Service frameworks and will also facilitate the proposal of business models to these emerging computing systems.

The present paper is organized as follows. Section II summarizes the related work. Section III proposes a set of class of service to fog computing. Section IV presents a mapping between the proposed classes of service and the layers of the reference architecture introduced by the OpenFog Consortium

978-1-5386-2098-4/17/$31.00 ©2017 IEEE

as well as some use cases. Finally, Section V concludes the paper.

II. Related work

In this section, we review literature that proposes QoS strategies in Fog and Cloud computing based on services differentiation.

Aazam et al. [7] formulate a resource management framework for Fog computing, based on the service type. Thus, a mapping between virtual resource values and the physical resource pool is made by the Cloud service provider (CSP) according to the type of service being provided. Some types of services used as example are Video on Demand (VoD) and health monitoring.

Souza et al. [8] propose an integer linear programming (ILP) model in the Fog-Cloud context. The objective is the optimization of latency in planning scenarios based on the capacity (number of slots) required by each service to be allocated. To this end, services are categorized into two groups: mice, representing the high amount of services requiring few slots, and elephants, representing the low amount of services requiring a higher number of slots.

In [9], Aazam et al. present a model that covers resource prediction in Fog, customer type based resource estimation and reservation, advance reservation, and pricing for new and existing IoT customers, on the basis of their characteristics, and the type of customer defined by the number of times that a specific service is requested by the same client.

These works evidence the need to establish a relationship between the type of service to be processed, the QoS requirements of the applications, and the mechanisms implemented in the fog to schedule and allocate resources to the application tasks. However, the classification of fog applications and Class of service is somehow limited in these papers. In this paper, we present a comprehensive proposal for the establishment of a class of service in fog computing.

III. Class of Service

A. QoS requirements

Fog computing will enable new applications especially those with strict latency constrains. The massive amount of data generated by IoT, Smart Cities, and Smart Homes will be possible to be processed in real time close to the end user, thus avoiding the expected overwhelming bandwidth demand as a result of the deployment of sensors everywhere. Moreover, mobile applications such as those in vehicular networks not possible before will become be feasible. The diversity of applications will be a main characteristic of processing in fog computing, which implies in highly heterogeneous demands of resources.

Fog administrators will face not only the challenge of providing the required quality of service for heterogeneous applications but also the need for dealing with the heterogeneity of the capacity of fog nodes and network links. The complexity of a fog node can range from a smart device to a "mini-cloud" or cloudlet. Such degree of heterogeneity makes even harder the decision on which resource a task should be allocated. Moreover, the flow of demand requests can fluctuate dynamically due to the users' mobility, the time scale of the application as well as the rate of data generation. Such fluctuation leads to the need of the adoption of dynamic resource allocation mechanisms.

In communications networks, network flows with similar resource demands and traffic profiles are gathered in groups that clearly identify these two. These groups, called Class of Service, are used not only to define the offering of network services, but also to identify the needs of flows for the traffic control mechanisms such as schedulers and buffer managers. The tagging of a packet by traffic classifier determines the treatment that the network flows will receive in the network core. Frameworks for Quality of Service provisioning such as Interserv and Diffserv define Class of Services as well as traffic control mechanisms to support diverse quality of service requirements of network flows.

Fog nodes will be interconnected by a fog network composing a heterogeneous distributed system, which can federate resource with other fogs and be integrated with other Clouds. The complexity of the fog infrastructure and the diversity of fog applications calls for the definition of Class of Service so that the offering of services as well as the resource demands by fog components can be clearly identified.

Next, the quality of service requirements of fog applications as well as other requirements will be presented by identifying the applications which will run on fogs, especially those enabled by the deployment of fogs.

Bandwidth – Users should be able to require the minimum amount of bandwidth for an application. This can be translated in having Guaranteed Bit Rate (GBR) requirement and a Non-GBR (NGBR) requirement for best effort applications.

Delay sensitivity – Delay bounds need to be assured for real-time applications such as face recognition in crowds.

Packet loss – Loss sensitive applications such as financial data may demand lossless transfer services.

Reliability – Some applications need to have failed fog components reestablished quickly so that tasks can be performed within latency bounds.

Availability – Consists in a measure of how often the resources of the fog are accessible to end-users during the application execution. High availability is needed by applications and services that must be running all the time, such as mission-critical applications, while those applications or

services which execution can be postponed within a given period do not require that resources be available all the time.

Security – Applications will transfer personal and critical information. Information security mechanisms or mission-critical applications need to be deployed. Moreover, network security is essential since the fog network will be the highway for all the data generated by end-users.

Data location – Data can be stored locally at the end device, near a Fog node or in a remote repository in the Cloud. Requirements of data location for an application depends on factors such as the response time constraints, the computational capacity of each layer of a fog, available capacity on network links connecting the fog end-users, the fog nodes, and the Cloud.

Mobility support – Connectivity should be provided continuously even for highly mobile end-users. Connectivity is also critical for collaborative tasks either in different fog devices or fogs; for instance, in applications such as rendering the end devices furnish processing capacity, and continuous connectivity is a must to execute the processing needed.

Scalability – The number of users in a fog can fluctuate due to the mobility of users as well as the activation of applications or sensors. Streams of data in big data processing may need to be processed within a time frame. The demand on fogs can fluctuate and resource elasticity needs to be provided to cope with such demands to make viable such applications.

B. Classes of services

After identifying the requirements of most probable applications running on fog, they were grouped in a minimum number of classes with a distinct set of requirements. Keeping the number of meaningful classes as small as possible is desirable since the complexity of allocations mechanisms is directly proportional to the number of classes. Five classes are proposed for a fog system considering the requirements described previously. They are Mission-critical, Realtime-interactive, Streaming, CPU-bound, and Best-effort. Next, the proposed classes will be described.

Mission-critical - Comprises applications with a low event to action time bound, regulatory compliance, military grade security and privacy, and applications in which a component failure causes a significant increase in the safety risk for the people and/or environment involved. Examples include traffic generated by armed forces combat systems, drone operations, some healthcare systems, hospitals, and ATM banking systems [10]. If the network supports priority services, this class of service should be assigned a high priority level. Moreover, applications related to context analysis of data streams for the identification of specified objects or hazardous events should also be mapped onto this class.

Realtime-interactive. Includes real-time, interactive and conversational applications. The real-time applications have delay constraints. Some examples are industrial control systems, some IoT and smart cities applications, online gaming, virtual and augmented reality. In the interactive applications, the user can be an end device or an individual. Examples of these services are the interactive television, object hyperlinking (RFID, NFC, QR-Code), and data transaction services such as e-commerce. Conversational applications include real-time conversation performed between peers or groups of humans. Conversational applications are delay-sensitive but loss-tolerant. Thus, the QoS requirements are given by the human perception. Examples of this last category are Voice over Internet Protocol (VoIP) and videoconference.

Streaming - This class covers applications with long file transfers such as streaming of stored content and streaming of live content. In applications involving the streaming of stored content, the content is prerecorded and placed on servers. Users request access to this content on demand, as well as interact with the content. Among the main companies that provide streaming video are YouTube, Netflix, and Hulu. Applications based on streaming of live content allow a user to both broadcast and watch live audio or video events transmitted over the Internet.

CPU-Bound – This class should be used by applications involving complex processing models such as those in decision making, which may demand hours, days or even months of processing. Examples include applications with video or images processing such as those involving distributed cameras or rendering.

Best-effort – This class is dedicated to traditional best effort applications over the Internet such as: www, e-mail, chat, FTP and p2p file sharing.

Table 1 presents the relationship between the applications supported by Fog computing and requirements of each Class of Service. The first column shows the recommended priority level of each class for potential adoption in scheduling systems.

IV. Class of Service and The Fog Reference Architecture

The OpenFog reference architecture (OpenFog RA), defined by the OpenFog consortium [10], provides an architectural model based on several tiers of nodes. Tiers differ by the amount and type of work processed on them, the number of sensors, the capacity of the nodes, the latency between nodes, reliability and availability of nodes. Nodes at the edge are involved in sensor data acquisition/collection, data normalization, and command/control of sensors and actuators. In the next tier (or layer), nodes typically execute data filtering, compression, and transformation. They may also provide analytics capability required for critical real-time or near real-

TABLE 1 CLASS OF SERVICE AND THEIR REQUIREMENTS

Allocation Priority	Class of Service	Service Quality Requirements																Applications
		Bandwidth	*Reliability*	*Security*		*Availability*		*Data location*			*Mobility*		*Scalability*			*Delay sensitivity*	*Loss sensitivity*	
				High	*Low*	*High*	*Low*	*Local*	*Vicinity*	*Remote*	*High*	*Low*	*None*	*High*	*Low*			
1	Mission-critical	GBR	✓	✓		✓		✓	✓		✓	✓	✓	✓		Yes	Yes	Healthcare, criminal justice, financial, biological traits, residence and geographic, military, emergency.
2	Realtime-interactive	GBR	✓	✓		✓		✓	✓		✓	✓	✓	✓		Yes	No	Online gaming, IoT deployments, industrial control, virtual and augmented reality, interactive television, object hyperlinking, voice messaging, VoIP, videoconference, telemetry, Telnet.
3	Streaming	GBR	✓		✓		✓	✓	✓	✓	✓	✓	✓	✓		Yes	No	Internet radio, Video (one-way), High quality Streaming audio.
4	CPU-Bound	GBR	✓	✓		✓			✓	✓		✓	✓		✓	Yes	Yes	Face recognition, animation rendering, speech processing, distributed camera networks.
5	Best-effort	NGBR			✓		✓			✓	✓	✓	✓	✓		No	Yes	Network signaling, all non-critical traffic such as TCP-based data: www, e-mail, chat, FTP, p2p file sharing, progressive video and other miscellaneous traffic

time processing. Nodes closer to Cloud or in the Cloud aggregate data and transform data into knowledge. As one moves farther away from the edge, the overall system intelligence and capacity increase.

Fig. 1 presents a distributed multi-layer architecture, based on the OpenFog RA, which is composed of four layers: The Cloud, at the top, a layer of end devices at the bottom and two intermediate fog layers. Fig. 1 also illustrates on which layer tasks from each of the proposed Class of Service could be processed. Application requests come from the bottom layer, where smart end-devices perform data pre-processing and compression, while each one of the other three layers process certain types of traffic according to the available computational capacity and QoS requirements of the applications. Not all layers are involved in the processing of all tasks.

Mission-critical applications could be processed at layers below the Cloud layer to avoid potential security threats typical of multi-tenant Clouds [11]. Realtime-interactive and Streaming applications such as online sensing and stored streaming are delay-sensitive. Therefore, these applications must be processed as close as possible to the end user, preferably in nodes located at the first fog layer. The fog layer 2 can carry out more intense processing without affecting the Quality of Service on end-user such as tasks related to business, deep data analysis, and even non-viral content for streaming applications. In both cases, the fog layer 2 can execute these tasks without affecting the Quality of Service. CPU-bound applications require a lot of processing resources. For this reason, all layers of the reference architecture can be involved in the processing of tasks in this class. Best-effort applications such as e-mails can be processed in the Cloud since there is no delay constrains for this class.

Table 2 details a mapping of the proposed classes of service and the layered architecture in Fig.1, which is based on the Fog reference architecture proposed by the OpenFog Consortium [10].

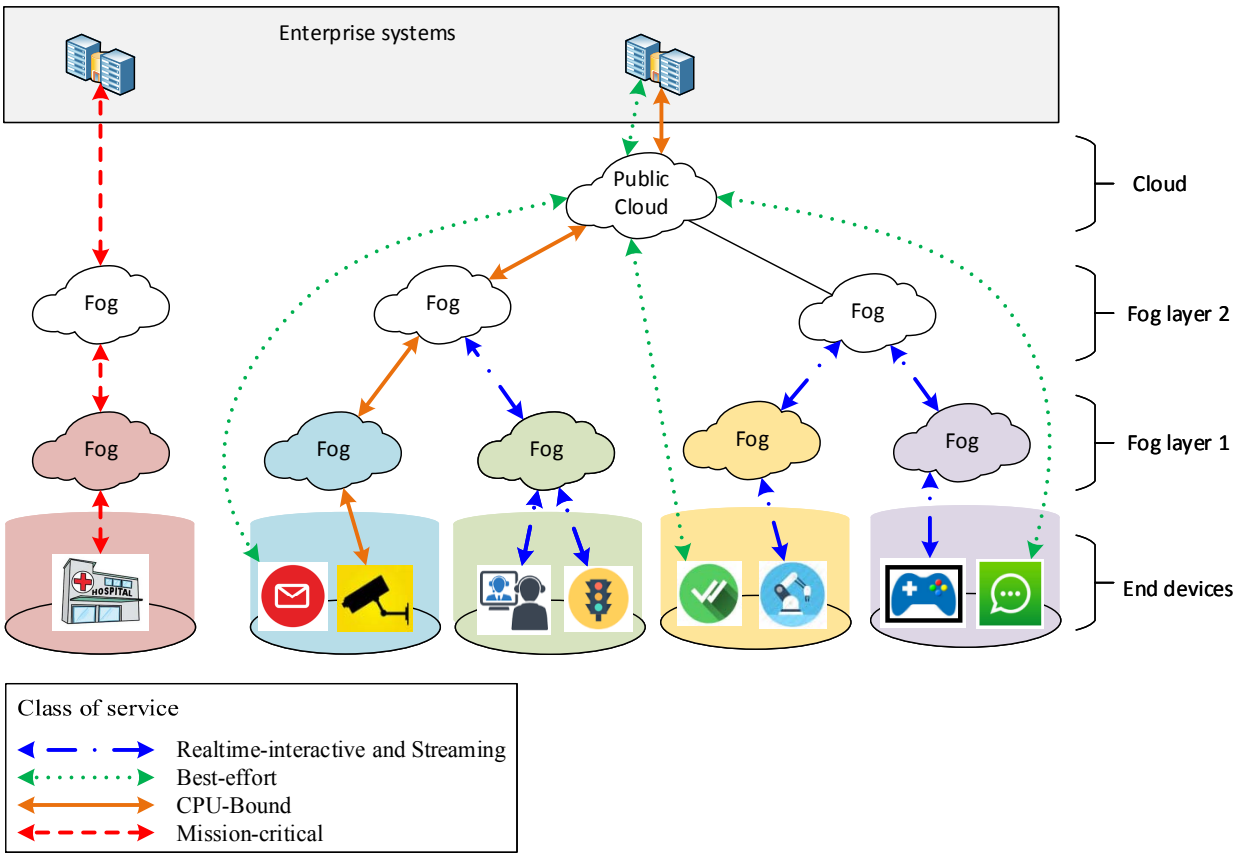

Fig. 1 Fog computing architecture implementing Class of Services

TABLE 2. MAPPING BETWEEN CLASSES OF SERVICE AND PROCESSING LAYERS OF FOG REFERENCE ARCHITECTURE

Class of service	*Functionalities of the Cloud layer*	*Functionalities of the Fog layer 2*	*Functionalities of the Fog layer 1*
Mission-critical	NA	• Provide security and privacy • Resource management • Costs management • Run complex jobs	• Support mobility • Realize data pre-processing
Realtime-interactive	NA	• Realize In-depth data analysis • Data caching • Computation offloading • Resource management • Costs management	• Collect data from sensors • Realtime data processing. • Maintain the on-board geographic information • Provide the real-time navigation
Streaming	NA	• Contents caching • Resource management • Cost management	• Process request from users. • Select and cache strategic content • Provide the most desirable services to mobile users according with their location.
CPU-bound	• Store the inputs used for processing. • Save the results of processing. • Massive parallel data processing. • Big data management. • Big data mining. • Machine learning.	• Data processing. • Workload allocation • Resource management • Cost management	• Perform pre-processing • Receive requests from users • Monitor resource utilization • Manage results of data query returned from nodes
Best-effort	• Process information • Store information • Resource management • Cost management	NA	NA

V. Conclusions and Future Research Directions

This paper proposed class of services for fog computing. The introduction of class of services in Fogs and Fog integrated to the Cloud is a first step towards the definition of Quality of Service frameworks for Fogs. These classes can be used to prioritize network traffic and processing demands by schedulers and resource allocation mechanisms. Prioritization will support the processing of delay sensitive applications, moving non real-time applications farther to the edge. By promoting load balancing among the layers of a fog, it is most likely that the fog will be able to support a higher number of requests, contributing to the scalability of the Fog. Moreover, the definition of Class of Service can facilitate the assignment of functionalities to different fog layers for the processing of typical demands of applications in each Class of Service. In addition, business models will largely benefit from the definition of Class of Services to fog computing.

The management and operation of Fogs present numerous challenges which call for innovative solutions. The interface and functionality assignment to fog layers should allow efficient management and dynamic allocation of resources [12]. Reliability schemes are necessary to assure that Fogs will continuously provide low latency, connectivity, and processing even when failures occur. Moreover, self-adaptation and cognition in the management of fogs need to be understood towards the deployment of autonomic fogs. The Class of Services introduced in this paper can facilitate addressing all the challenges that lay ahead, since they help in coping with the requirements of the applications under limited availability of resources.

Acknowledgment

This work was supported in part by the Brazilian Research Agency CNPq and the Academy of Sciences for the Developing World (TWAS), under process 190172/2014-2 of the CNPq-TWAS program.

The research leading to these results received funding from the European Commission H2020 program under grant agreement no. 688941 (FUTEBOL), as well from the Brazilian Ministry of Science, Technology, Innovation, and Communication (MCTIC) through RNP and CTIC.

References

[1] V. B. Souza, X. Masip-Bruin, E. Marin-Tordera, W. Ramirez, and S. Sanchez, “Towards Distributed Service Allocation in Fog-to-Cloud (F2C) Scenarios,” in 2016 IEEE Global Communications Conference (GLOBECOM), 2016, pp. 1–6.

[2] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog Computing and Its Role in the Internet of Things,” in Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing, New York, NY, USA, 2012, pp. 13–16.

[3] S. Li, M. A. Maddah-Ali, and A. S. Avestimehr, “Coding for Distributed Fog Computing,” IEEE Commun. Mag., vol. 55, no. 4, pp. 34–40, Apr. 2017.

[4] I. Petri, J. Diaz-Montes, O. F. Rana, Y. Rezgui, M. Parashar, and L.F. Bittencourt. Coordinating Data Analysis & Management in Multi-Layered Clouds. In: International Conference on Cloud, Networking for IoT systems (Cn4IoT), 2015.

[5] T. H. Luan, L. Gao, Z. Li, Y. Xiang, G. Wei, and L. Sun, “Fog Computing: Focusing on Mobile Users at the Edge,” ArXiv150201815 Cs, Feb. 2015.

[6] N. L. S. da Fonseca and R. Boutaba, Cloud Services, Networking, and Management. John Wiley & Sons, 2015.

[7] M. Aazam, M. St-Hilaire, C. H. Lung, and I. Lambadaris, “PRE-Fog: IoT trace based probabilistic resource estimation at Fog,” in 2016 13th IEEE Annual Consumer Communications Networking Conference (CCNC), 2016, pp. 12–17.

[8] V. B. C. Souza, W. Ramírez, X. Masip-Bruin, E. Marín-Tordera, G. Ren, and G. Tashakor, “Handling service allocation in combined Fog-cloud scenarios,” in 2016 IEEE International Conference on Communications (ICC), 2016, pp. 1–5.

[9] M. Aazam and E. N. Huh, “Fog Computing Micro Datacenter Based Dynamic Resource Estimation and Pricing Model for IoT,” in 2015 IEEE 29th International Conference on Advanced Information Networking and Applications, 2015, pp. 687–694.

[10] “OpenFog Reference Architecture: OpenFog Consortium”. Available: https://www.openfogconsortium.org/ra/ [Accessed: 24/05/2017].

[11] M. Chiang, S. Ha, C. L. I, F. Risso, and T. Zhang, “Clarifying Fog Computing and Networking: 10 Questions and Answers,” IEEE Commun. Mag., vol. 55, no. 4, pp. 18–20, Apr. 2017.

[12] M. Chiang and T. Zhang, “Fog and IoT: An Overview of Research Opportunities,” IEEE Internet Things J., vol. 3, no. 6, pp. 854–864, Dec. 2016.