CSSE7014 Distributed Computing
Assignment 2
Semester 1, 2017

Paul Kogel (44644743), Ramdas Ramani (44743767)

May 6, 2017

Contents

1   Introduction

Fog computing is a new, exciting computing paradigm [1].

The introduction is clear with several definitions of the computing paradigm under study for comparison. The structure of the report is presented.

2   Architectures and Models

Compare and contrast different architectures and models with examples to back the arguments.

RAM

3   Common Issues

In their "survey on fog computing" [2], Yi et al. discuss issues related to fog computing, and present possible solutions from current research. This section uses their discussion as foundation.

3.1   Networking

3.2   Quality of Service

Based on Yi et. al's assessment of quality of service-related issues for the fog, we infer the following basic metrics to be used to describe the QoS:

- Service availability

- Resource usage

- Delay

- Reliability

"Service availability" describes the availability of specific services to a node. According to [2], this is greatly influenced by the network topology. This makes intuitively sense: if a node is, for example, connected to a router over an unreliable link, it will probably not be able to access network services at all time.

Naturally, resources as bandwidth and storage are limited in the fog. [2] suggest that placing data effectively can help addressing this issue. In addition, they discuss several techniques for "computation offloading", i.e., the movement of tasks from resource-limited devices like mobile phones to more powerful nodes that are located in the fog, or the cloud. This offloading, however, introduces several new issues. Mainly, these are centred around the difficulty of deciding which parts of the application should be offloaded, while accounting for the constant changes in the network due to node mobility.

Since many applications for fog computing are set to process data in real-time, maintaining a low delay is critical. The aforementioned techniques of data placement, and choice of network topology can be used here.

Finally, to improve reliability, [2] suggest that replication might be used. Traditional techniques like checkpointing, or rescheduling, in contrast, are argued to introduce much delay, and are therefore unfit for real-time applications.

3.3   Interfacing and Programming Model

As stated in section XX, the fog is dynamic in regards to network topology, and resource availability. In addition, fog nodes might by based on different platforms and system architectures. Resulting from this, developing applications in a fog computing context is expected to be challenging [2]. To counter this issue, Bonomi et al. [3] propose a "fog abstraction layer" that hides the underlying heterogeneity, and

provides developers with a "uniform and programmable interface". Yi et al. make a similar suggestion by calling for a "unified interfacing and programming model" [2].

Though the necessity of these abstraction layers is obvious, we expect that the initially mentioned characteristics of the fog will make an actual implementation highly challenging. Moreover, if devices from different vendors are set to work together, an industry-wide standard has to be developed.

4 Applications

Various examples (across different disciplines) provided with clear arguments why they are relevant.

5 References

[1] 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*, pp. 13–16, ACM, 2012.

[2] S. Yi, C. Li, and Q. Li, "A survey of fog computing: concepts, applications and issues," in *Proceedings of the 2015 Workshop on Mobile Big Data*, pp. 37–42, ACM, 2015.

[3] F. Bonomi, R. Milito, P. Natarajan, and J. Zhu, "Fog computing: A platform for internet of things and analytics," in *Big Data and Internet of Things: A Roadmap for Smart Environments*, pp. 169–186, Springer, 2014.