



بسم الله الرحمن الرحيم

∞∞∞∞

تم رفع هذه الرسالة بواسطة / حسام الدين محمد مغربي

بقسم التوثيق الإلكتروني بمركز الشبكات وتكنولوجيا المعلومات دون أدنى

مسئولية عن محتوى هذه الرسالة.

ملاحظات : لا يوجد





Ain Shams University  
Faculty of Computer and Information Sciences  
Computer Science Department

# **A Layer-Based Runtime Framework for Enabling Cloud Computing into Fog Stratum**

**A Thesis**  
**Submitted in Partial Fulfillment of the Requirements of the**  
**Degree of Doctor of Philosophy in Computer Science**

**By**

**Ahmed Hosny Mohammed Ibrahim**

Assistant Lecturer  
Computer Science Department,  
Faculty of Computer & Information,  
Assiut University

## **Supervisors**

**Prof. Zaki Taha Fayed**

Professor of Computer Science  
Computer Science Department,  
Faculty of Computer & Information Sciences  
Ain Shams University

**Prof. Hossam Eldin Mostafa Faheem**

Professor of Computer Systems  
Computer Systems Department,  
Faculty of Computer & Information Sciences  
Ain Shams University

Cairo, Egypt  
2022

بسم الله الرحمن الرحيم

## شكر و عرفان

كان فضلُ الله عليَّ عظيماً في كل أمور حياتي صغيرها وكبيرها، وفي رحلة تعلّمي وتحضير لي هذه الرسالة والتي منَّ الله عليَّ فيها بكثيرٍ من الخير، فالحمد لله وحده حمداً كثيراً طيباً مباركاً فيه.

إلى أستاذي القدير الذي شملني بتوجيهاته ونصائحه ومتابعته المستمرة، إلى الأستاذ الدكتور/ زكي طه فايد، جزيل الشكر والعرفان لسيادتكم وجزاكم الله خيراً.

إلى أستاذي الفاضل الذي تعلمت واتعلم منه معاني العلم والبحث والعمل الجاد، إلى الأستاذ الدكتور/ حسام الدين مصطفى فهيم، أدين لك بالفضل بعد الله عز وجل الانتهاء من هذا البحث وأدين لك بالكثير، أسأل الله العظيم أن يجعل كل ذلك العلم والعمل في ميزان حسناتك.

إلى الذي حثني على العلم والعمل، إلى قدوتي ومعلمي وأستاذي الأول، إلى أبي العزيز الأستاذ الدكتور/ حسني محمد إبراهيم، تعجز كل كلمات الشكر والثناء عن الوفاء ببعض حقك، فأسأل الله العظيم أن يُجزيك عني خير الجزاء.

إلى أُمي الحبيبة، كم أزرنتي بكلماتها المُشجعة، وادخلت السرور على قلبي بابتسامتها وتفاؤلها المستمر، هي من علمتني معنى العطاء وعلمتني معاني الحياة، جزاك الله عني خير الجزاء.

إلى مؤنستي ورفيقة دربي ومُنيرة طريقي، إلى زوجتي الغالية، أهدي إليك هذا العمل الذي طالما رجونا من الله عز وجل الانتهاء منه، أهديه إليك فقد كنت لي نِعَم الداعمة.

إلى بناتي مريم وملك، قرّة عيني، سعدت بطفولتكم طوال فترة هذا العمل، أسأل الله العظيم أن يُنبّتكم نباتاً حسناً ويجعلكم من الصالحات.

إلى أبي الثاني و والد زوجتي الأستاذ الدكتور/ هاني عبدالمعطي علي، وإلى أُمي الثانية و والدة زوجتي الأستاذة الدكتور/ إيمان محمد حسين خضر، كنتم لي خيرَ ناصح و مُشجع، أدام الله فضلكم وبارك فيكما ورفع قدركما وجزاكم الله عني خيراً.

# Abstract

Evolution of computing paradigms has been one of the main reasons of IT revolution over the past few decades. Cloud computing has been playing a major role in this revolution, offering different kinds of demands as services, such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). Cloud computing helps in solving many issues of datacenters as resource utilization, resource sharing and scalability. Yet, the cloud computing still has some challenges tackling latency-sensitive applications such as Fire detection, Firefighting, Healthcare applications, robot control and content delivery. This is caused mainly because of the connectivity between the cloud and the end devices. Another reason can be the intra-communication delay occurring between different cloud instances to provide the final service or answer.

Fog Computing is a new computing paradigm that has been introduced to offer solutions for the cloud computation challenges. Introducing a new conceptual layer between Cloud computing and Edge devices called Fog opened new horizons for computation offloading close to the consumers. Offloading computations near the Edge layer plays an important role in reducing the latency of the services. There is a lack of Fog application development tools and frameworks that are easily adopted by software engineers in the software community. In this thesis, we provide tools and frameworks that utilizes the Fog stratum in an easily abstracted manner for software engineers.

Two main contributions are proposed in our thesis to tackle the lack of fog stratum utilization. A Fog based CDN framework is proposed to facilitate using fog nodes as a CDN nodes. Also, a new annotation and meta language is proposed along with its compiler to facilitate generating a service that can be deployed in fog architecture. These tools and framework provide better quality of service (QoS) for cloud services in terms of latency and reliability.

# Table of Contents

<i>شكر و عرفان</i>	<i>II</i>
<i>Abstract</i>	<i>II</i>
<i>List of Figures</i>	<i>III</i>
<i>List of Tables</i>	<i>IV</i>
<i>List of Definitions</i>	<i>V</i>
<i>List of Abbreviations</i>	<i>VI</i>
<b>Chapter 1 Foundation</b>	<b>9</b>
<b>1.1 Overview</b>	<b>9</b>
<b>1.2 Computation Paradigms (Cloud, IoT, Edge, Fog)</b>	<b>10</b>
1.2.1 Cloud Computing	11
1.2.2 Internet of Things	14
1.2.3 Edge Computing	18
1.2.4 Fog Computing	19
<b>1.3 Fog Architecture</b>	<b>21</b>
<b>1.4 Comparison Between Cloud, Fog, Edge Computing Paradigms</b>	<b>24</b>
1.4.1 Fog Computing vs. Cloud Computing	24
1.4.2 Fog Computing vs. Edge Computing	24
<b>1.5 Problem Statement</b>	<b>25</b>
<b>1.6 Motivation</b>	<b>25</b>
<b>1.7 Objectives</b>	<b>26</b>
<b>1.8 Thesis Organization</b>	<b>27</b>
<b>Chapter 2 Fog Computing Review: Frameworks, Software Tools, and Content Delivery Network</b>	<b>28</b>
<b>2.1 Introduction</b>	<b>28</b>
<b>2.2 Fog Computing Software Tools and Programming Models</b>	<b>29</b>
2.2.1 Programming Models	30
2.2.2 Software Tools	30
<b>2.3 Fog Computing Framework</b>	<b>31</b>
2.3.1 General Purpose Fog Computing Frameworks	31
2.3.2 Specific Purpose Fog Computing Framework	32
<b>2.4 Fog Computing Content Delivery Network (CDN)</b>	<b>33</b>
<b>2.5 Discussion</b>	<b>35</b>
<b>2.6 Summary</b>	<b>36</b>
<b>Chapter 3 Fog-Based CDN Framework for Minimizing Latency of Web Services using Fog-Based HTTP Browser</b>	<b>37</b>
<b>3.1 Overview</b>	<b>37</b>
<b>3.2 Fog-Based CDN Framework Components</b>	<b>37</b>
3.2.1 Fog Registry (FR)	37
3.2.2 Fog Service Locator (FSL)	38

3.2.3	Event Channel	39
3.2.4	Fog Worker (FW)	39
3.2.5	Fog Browser	39
<b>3.3</b>	<b>Fog-Based CDN Framework Workflow</b>	<b>39</b>
3.3.1	Offloading and Serving as a CDN Node using Fog Worker (FW)	40
3.3.2	The Fog Registry (FR) Workflow	41
3.3.3	The Fog Browser Workflow and the Fog Service Locator (FSL) Role	43
<b>3.4</b>	<b>Summary</b>	<b>44</b>
<b>Chapter 4</b>	<b>FAML: Fog Descriptor Language for Fog Service Development and Deployments</b>	<b>45</b>
<b>4.1</b>	<b>Overview</b>	<b>45</b>
<b>4.2</b>	<b>Fog Annotation and Meta language (FAML) Overview</b>	<b>45</b>
<b>4.3</b>	<b>FAML Compilation Process</b>	<b>48</b>
<b>4.4</b>	<b>Experimental Use Cases</b>	<b>49</b>
4.4.1	Image Enhancement Service	49
4.4.2	Weather Forecast Service	50
4.4.3	Bulk Update Service	51
4.4.4	The Experimental Use Cases Meta File	52
<b>4.5</b>	<b>FAML Comparison</b>	<b>53</b>
<b>4.6</b>	<b>Summary</b>	<b>55</b>
<b>Chapter 5</b>	<b>Experimental Results</b>	<b>56</b>
<b>5.1</b>	<b>Overview</b>	<b>56</b>
<b>5.2</b>	<b>Fog-Based CDN Framework Implementation and Experimental Results</b>	<b>56</b>
5.2.1	Prototype Implementation	56
<b>5.3</b>	<b>FAML Experiment Setup</b>	<b>62</b>
5.3.1	One Fog Node Environment with Raspberry Pi	62
5.3.2	One Fog Node Environment with Proxy Server	63
5.3.3	Containerized Environment with Multiple Fog Nodes	63
<b>5.4</b>	<b>Experiment Result Discussion</b>	<b>63</b>
<b>5.5</b>	<b>Thesis Summary</b>	<b>64</b>
<b>Chapter 6</b>	<b>Conclusion and Future Work</b>	<b>66</b>
<b>6.1</b>	<b>Thesis Conclusion</b>	<b>66</b>
<b>6.2</b>	<b>Future Work</b>	<b>66</b>
<b>Appendix A</b>		<b>68</b>
	<b>Data Models</b>	<b>68</b>
	<b>Broadcast Process</b>	<b>69</b>
	<b>Fog Node Registration Implementation</b>	<b>70</b>
	<b>Offloading Process</b>	<b>72</b>
<b>Appendix B</b>		<b>76</b>
<b>List of Publications</b>		<b>77</b>
<b>References</b>		<b>78</b>





# List of Figures

Figure 1. Cloud computing evolution timeline [6].	11
Figure 2. The Cloud network architecture conceptualization [10].	12
Figure 3. The Conceptual Cloud Computing Reference Architecture	13
Figure 4. Reference architecture for IoT proposed in [18].	16
Figure 5. The Abstract Three Layer Fog Architecture Model	22
Figure 6. OpenFog N-tier architecture of Fog Computing [23].	23
Figure 7. Expanded Fog Network Architecture [10].	34
Figure 8. Fog-based CDN architecture and components.	38
Figure 9. High-level architecture of the fog-based CDN.	40
Figure 10. Functions of the Fog Worker in a sequence diagram.	41
Figure 11. The web service registration steps.	42
Figure 12. The FSL processes flowchart.	44
Figure 13. Image Enhancement Pseudo Code with FAML Annotations.	48
Figure 14. FAML Meta for Image Enhancement Service (fog.yml).	48
Figure 15. High Level Flow of the FAML Compiler.	49
Figure 16. Image Enhancement Service in Golang.	50
Figure 17. Annotated Image Enhancement Service in Golang.	50
Figure 18. The Generated Service Directory Structure.	50
Figure 19. Sample Payload for the Generated Service.	50
Figure 20. Weather Forecast Fog Service Pseudocode with FAML Annotations.	51
Figure 21. Weather Forecast Service in Golang.	51
Figure 22. Weather Forecast Service Payload.	51
Figure 23. Bulk Update Fog Service Pseudocode with FAML Annotations.	52
Figure 24. Annotated Bulk Update Service in Golang.	52
Figure 25. FAML Meta for Image Enhancement, Weather Forecast, and Bulk Update Services (fog.yml).	53
Figure 26. Fog-based CDN latency vs. cloud-based CDN latency.	60
Figure 27. Fog-based CDN latency (different network distances) vs. cloud-based CDN latency.	61
Figure 28. Fog-based CDN latency (multiple fog nodes) vs. cloud-based CDN latency.	62
Figure 29. Fog Node and Hop Counts Experiment Live Setup	76
Figure 30. Retrieving Service Resources from Different Fog Nodes Experiment Live Setup.	77

# List of Tables

Table 1. FAML Annotations Summary.....	47
Table 2. Summary of Research Directions Towards Fog Application Development Facilitation.....	55
Table 3. A summary of the FW different commands.....	57
Table 4. List of the websites studied. ....	58

# List of Definitions

<i><b>Terminology</b></i>	<i><b>Definition</b></i>
Cloud Computing	Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction
Edge Computing	The Edge computing is the network layer encompassing the end devices and their users, to provide, for example, local computing capability on a sensor, metering or some other devices that are network-accessible. This peripheral layer is also often referred to as IoT network.
Edge Devices	Edge devices are located as the immediate first hop from the end devices. These devices are capable of computing, storing, and/or network routing.
End Devices	End devices are the devices that requests a service or upstream data to the cloud data centers.
Fog Computing	A horizontal, system-level architecture that distributes computing, storage, control, and networking functions closer to the users along a cloud-to-thing continuum.
Fog Node	The fog node is the core component of the fog computing architecture. Fog nodes are either physical components (e.g., gateways, switches, routers, servers, etc.) or virtual components (e.g., virtualized switches, virtual machines, cloudlets, etc.) that are tightly coupled with the smart end- devices or access networks and provide computing resources to these devices. A fog node is aware of its geographical distribution and logical location within the context of its cluster.
Internet of Things	The term "Internet of Things" (IoT) denotes a trend where many embedded devices employ communication services offered by Internet protocols. Many of

	these devices, often called "smart objects", are not directly operated by humans, but exist as components in buildings or vehicles, or are spread out in the environment
--	--

## List of Abbreviations

<i>Abbreviation</i>	<i>Definition</i>
CC	Cloudlet Computing
CDN	Content Delivery Network.
CI/CD	Continuous Integration / Continuous Delivery (Deployment)
CoAP	Constrained Application Protocol
DBaaS	Database as a Service
FA	Fog Architecture
FAML	Fog Annotation and Meta Language
FaaS	Function as a Service
FC	Fog Computing
FN	Fog Node.
FPGA	Field Programmable Gate Array
GPU	Graphic Processor Unit
HTTP	Hyper-text Transfer Protocol
IaaS	Infrastructure as a Service
IAB	The Internet Architecture Board
IoT	Internet of Things.
ITU	The International and Telecommunication Union
KPI	Key Performance Indicator

MEC	Multi-access Edge Computing
MQTT	Message Queue Telemetry Transport
NIST	National Institute of Standards and Technology
Offloadable	A version that can be deployed in another network layer closer to the consumer
PaaS	Platform as a Service
QoS	Quality of Service
RAN	Radio Access Network
RFID	Radio Frequency Identification
SaaS	Software as a Service
TPU	Tensor Processing Unit

# Chapter 1

## Foundation

### 1.1 Overview

Smart machines, IoT sensors, real-time applications and internet enhanced devices are growing rapidly, generating zettabytes of data. This data is pushed to the network affecting the network capacity, efficiency and performance [1]–[3]. Furthermore, this increase of data transferring via the network comes with an increase of numbers of the drop-off packets, causing service unavailability issues. While some of these online services can neglect those drop-off requests or accept degraded availability, other services are delay-sensitive and cannot accept any degradation in the quality of service (QoS). Health care systems and real time surveillance services are good examples of how critical the QoS should be nowadays [3], [4]. On the other side, connected devices are generating data at an exponential rate, and cloud-centric services are experiencing major challenges due to the tsunami of computations and data streaming. Some of these challenges are bandwidth, latency, uninterrupted requests, continuous data streaming by the end devices, and computation dependability [5]. Current cloud-centric service architecture is not helping such critical systems to maintain their high availability and high QoS [6]. The solution for such a problem is to move the computation and the service closer to the consumers, instead of serving from the cloud data centers. Hence, Fog computing architecture is proposed as one of the most solid approaches towards solving service cloud centralization problems.

With education becoming online, working from home rising as a new standard, and sharing knowledge becoming mandatory, many cloud services have been affected due to network high loads. Given that the centralization nature of the cloud computation, cloud services are facing challenges due to the boost of data streams and computation requests in the networks. One of the main challenges is to achieve the quality of service (QoS) requirements stated in the service level agreement (SLA). In the coming years, cloud computing will be no longer capable of handling such increasing volumes of data and requests. The need of a new computation paradigm that

can work with the cloud computing to tackle these challenges becomes a must. The new computing paradigm should try to decentralize the computations but keeping the advantages inherited from the cloud computing such as: scale as you go, powerful computation power, and unlimited storage capacity. Fog computing is a new computation paradigm that tries to move the cloud capabilities nearby the consumers. Yet, a lot of challenges in this computation paradigm need to be solved so that it gets adopted by the industry. In this thesis, we focus on two main key challenges in this paradigm that -with our contributions- can push towards more usage of such paradigm. Thesis area of contributions are: moving the resources nearby consumer effectively in autonomous manner and provide tools for service engineers to easily build services in this new paradigm.

In this chapter, we provide an introduction and foundation of Fog computing. We discuss the different computation paradigms in section 1.2. Conceptual model of Fog architecture is presented in section 1.3. In section 1.4, different definitions related to Fog architecture are listed. Comparison between Cloud, Fog, and Edge computation is explained in the same section 1.4. The problem statement is stated in section 1.5. Thesis motivation is presented in 1.6, and objectives are stated in section 1.7. Thesis organization is presented in section 1.8.

## 1.2 Computation Paradigms (Cloud, IoT, Edge, Fog)

Computation paradigms have its major shifts and transformation from the early 1960s. In this section, Cloud, Fog, and Edge computing will be discussed along with the Internet of Things. The aim of this section is to provide an overview for these three computing models and provide a clear difference between them as well. It will also discuss each of these computing model architecture and characteristics. Furthermore, it states some open challenges in each model.