
UNIVERSITÀ
degli STUDI
di CATANIA

Università degli Studi di Catania
Dipartimento di Matematica e Informatica (DMI)
Corso di Dottorato in Matematica e Informatica

---

# Smart Computing for a New Society: How Cloud and IoT Technologies Can Improve Education, Security and Support Decision-Making Processes

*Candidate*
Ing. Riccardo Di Pietro

*Supervisor*
Prof. Salvatore Distefano

*PhD Coordinator*
Prof. Giovanni Russo

---

ciclo XXII, 2016-2017

*"What are you going to do to promote technological innovation and excellence for the benefit of humanity?"*

– *Riccardo Di Pietro*

*IEEE 94384791*

# Acknowledgments

*"If I have seen further, it is by standing on the shoulders of giants."*
Bernard of Chartres compared us to the dwarves perched on the giants' shoulders. He stressed that we see farther and farther than our predecessors, not because we have a sharp vision, nor greater height, but because we are raised and carried hight on their gigantic stature and knowledge.
Thank you, `Engineering and Computer Science ECS-UNIME`, for the height you offered me.

*"A friend is someone who understands your past, believes in your future, and accepts you just the way you are. A real friend is one who walks in when the rest of the world walks out."*
Thank you `Marta e Olga`.

# Sommario

Negli ultimi decenni l'umanità ha assistito ad uno sconvolgimento senza precedenti del suo modo di organizzare e vivere la vita rispetto a come aveva imparato a fare nell'ultimo secolo. Da quando l'umanità ha imparato a registrare e trasmettere i fatti storici, non è certamente la prima volta che si assiste a questi balzi evolutivi. Basti pensare all'impatto che hanno avuto sullo stile di vita del singolo uomo e sull'organizzazione sociale delle comunità alcune scoperte tecnologiche quali stampa, motore a vapore, motore a combustione interna, elettricità, mass media.

Oggi l'umanità naviga a vista sull'onda lunga di un processo di innovazione tecnologica molto complesso ed eterogeneo che potremmo azzardare a riassumere col temine *"digitalizzazione"*. Non a caso di questo processo si parla come di una vera e propria *"rivoluzione"* o *"trasformazione"*.

La `rivoluzione digitale` consiste in un progresso tecnologico sistemico che sta trasformando profondamente le interazioni umane. È un cambiamento nel paradigma sociale, economico e culturale della società. In questo contesto, ci si riferisce al concetto di *``Società"* nella sua accezione più ampia. La società vista come un'entità formata da insiemi di individui uniti da rapporti di varia natura e in cui si instaurano forme di cooperazione, collaborazione, divisione dei compiti, che assicurano la sopravvivenza e la riproduzione dell'insieme stesso dei suoi stessi membri.

Ogni nuova tecnologia introdotta in questo processo di trasformazione ha messo in atto una sorta di *"selezione naturale"* che ha premiato coloro che sono stati in grado di sfruttare il suo potenziale e penalizzato quelli che non sono stati in grado di cogliere le nuove opportunità. Quando la tecnologia introdotta è *"distruttiva"* essa provoca cambiamenti radicali che influenzano tutti gli aspetti della vita. Tali innovazioni non restano confinate unicamente alla sfera scientifica, oppure specificatamente al campo di appartenenza, ma influiscono la società per intero. Spesso, molti cam-

biamenti derivanti dall'adozione di una nuova tecnologia, sono difficili da prevedere inizialmente. Questo principio è valido soprattutto per le innovazioni introdotte dal digitale il cui spettro di applicazione è senza dubbio più completo di quello di qualsiasi altra tecnologia del passato in quanto riesce a includere quasi tutte le aree del mondo reale.

Oggi, più che in altre epoche, la società umana dovrebbe cogliere *opportunità e sfida* di progettare, sviluppare, promuovere e sostenere tecnologie realmente al servizio dei cittadini e dello sviluppo umano e sociale, e non viceversa. La `sfida finale della tecnologia` non dovrebbe risiedere unicamente nell'arrivare a chissà quale dirompente novità tecnologica, quanto piuttosto nel favorire la sostenibilità e l'adozione nuove tecnologie, limitando il più possibile i possibili impatti negativi sull'organizzazione sociale delle comunità, e quindi di conseguenza nella vita delle singole persone.

La `ricerca scientifica` dovrebbe andare unicamente nella direzione di mettere al riparo le persone dai rischi provenienti dall'adozione di tecnologie troppo invasive della loro `privacy`, per la loro `salute fisica, biologica e psicologica`. Ancora, la ricerca scientifica dovrebbe favorire la creazione di sistemi, applicazioni e nuovi servizi in grado di proporre modelli di `sostenibilità ambientale, economica ed energetica`, al fine di soddisfare le esigenze di una società che si ritrova ad affrontare delle sfide sempre più complesse e sfidanti.

Nell'attuale scenario di cambiamento, tecnologie come il `Cloud`, il `Mobile`, l'`Intelligenza Artificiale`, i `sistemi Edge`, i `Big Data` e l'`Internet delle Cose`, rappresentano delle opportunità robuste ed affidabili che la società deve necessariamente cogliere in modo costruttivo per competere con le nuove sfide globali.

Gli individui e le comunità dovrebbero orientarsi verso l'adozione sistematica di strumenti che consentano `processi decisionali più snelli e flessibili`, che consentano specificamente di `trarre beneficio dalle più svariate fonti di dati disponibili`.

Inoltre, la società dovrebbe concentrarsi interamente sulla `valorizzazione del capitale umano`, per quanto riguarda i talenti e l'eccellenza.

In questa direzione, l'introduzione dei sistemi di `istruzione digitale` ha aiutato gli studenti ad apprendere in modo più efficiente ed efficace, anche attraverso nuove modalità, ad esempio l'apprendimento da remoto. Tuttavia, l'istruzione digitale è accusata di causare numerosi effetti dannosi. Tra questi, viene spesso menzionato il suo ruolo significativo nell'impoverimento

delle capacità cognitive di alcune fasce d'età. È proprio durante l'infanzia e l'adolescenza che questo deficit viene maggiormente percepito.

Malgrado ciò, l'adozione di sistemi `apprendimento da remoto` è stato l'unico modo per garantire il diritto allo studio ad una generazione di studenti vittime di una società umana colta impreparata dal recente scoppio del virus SARS-COV-2 e della relativa `pandemia di COVID-19`.

In conclusione, la rivoluzione tecnologica ha generato incredibili opportunità negli ultimi anni, ma senza un impatto sociale positivo, rischia di passare alla storia come una rivoluzione mancata.

La vera sfida è orientare l'innovazione verso la soluzione dei grandi problemi che affliggono le società contemporanee.

*Riccardo di Pietro*
Aprile 2020

# Abstract

In recent decades, humanity has witnessed an unprecedented upheaval in its way of organizing and living life compared to the means employed to do it in the last century. Since humankind learned to record and transmit historical facts, this is certainly not the first time that we have experienced these evolutionary leaps. Consider by way of illustration the impact that some technological discoveries have had on the lifestyle of the individual man and on the social organization of communities, such as printing, steam engines, internal combustion engine, electricity, and mass media.

Today humanity is sailing on sight on the long wave of a very complex and heterogeneous process of technological innovation that we could venture to summarize with the term *"digitization"*. There is no coincidence in calling this process a real *"revolution"* or *"transformation"*.

The `digital revolution` consists in systemic technological progress that is profoundly transforming human interactions. It is a change in the social, economic, and cultural paradigm of society. In this context, the term *``Society"* refers to the concept of society in its broadest sense, that is, as an entity made up of groups of individuals united by relationships of various kind. Forms of cooperation, collaboration, division of tasks are established, in a way that ensures the survival and reproduction of the whole of the members.

Each new technology introduced in this transformation process brought about some type of *"natural selection"* that rewarded those who were able to exploit its potential and penalized those unable to seize the new opportunities. When the technology introduced is *"destructive"*, it causes radical changes that affect all aspects of life. These innovations do not remain confined only to the scientific sphere, or specifically to the field they belong to, but they influence the whole society. Often, many changes resulting from the adoption of new technologies are difficult to predict initially. This principle

is especially valid for the digitally-introduced innovations whose application spectrum is undoubtedly more complete than that of any other technology of the past as it manages to include almost all areas of the real world.

Today more than in other eras, human society should take the *opportunity and challenge* of designing, developing, promoting, and supporting technologies that are indeed at the service of citizens and of human and social development, and not vice versa. The `final challenge of technology` should not consist only in aiming at reaching the next disruptive technological novelty, but rather in promoting sustainability and the adoption of new technologies, by limiting as much as possible the possible negative impacts on the social organization of communities, and therefore consequently in the life of individuals.

`Scientific research` should only go in the direction of protecting people from the risks arising from the adoption of technologies that are too invasive of their `privacy`, for their `physical, biological, and psychological health`. Still, scientific research should encourage the creation of systems, applications, and new services capable of proposing models of `environmental, economic, and energy sustainability` to meet the needs of a society that finds itself facing increasingly complex and challenging challenges.

In the current scenario of change, technologies such as `Cloud`, `mobile`, `AI`, `edge systems`, and `IoT` represent robust and reliable opportunities that society must necessarily take constructively to compete with new global challenges.

Furthermore, individuals and communities should move towards the systematic adoption of tools that enable more streamlined and `flexible decision-making processes`, which specifically allow profiting from the most varied `sources of data available`.

Moreover, society should focus entirely on the `enhancement of human capital`, regarding talents and excellence.

In this direction, the introduction of `digital education systems` helps students learn more efficiently and effectively, including through new ways, for example, remote learning. However, digital education is blamed for causing numerous harmful effects. Among the aforementioned effects, its significant role in the impoverishment of some age groups' cognitive abilities is often mentioned. It is during childhood and adolescence that this deficit is mainly perceived.

Despite this, the adoption of `remote learning` systems was the only

way to guarantee the right to study for a generation of students who are victims of a human society caught unprepared by the recent outbreak of the SARS-COV-2 virus and of the related `COVID-19 pandemic`.

In conclusion, the technological revolution has generated incredible opportunities in the last few years but without a positive social impact, it risks going down in history as a missed revolution.

The real challenge is to direct innovation towards solving the big problems contemporary societies face.

*Riccardo di Pietro*
April, 2020

# Contents

# List of Figures

# List of Tables

# Introduction

Some technologies such as Cloud, Mobile, Artificial Intelligence, Edge systems, Big Data, and IoT are changing our lives like never before opening up new scenarios for solving complex social issues such as climate change, health inequalities, the scarcity of energy resources, urban development and industrial growth. It is, therefore, from the appropriate implementation of technological innovation and the digital revolution that the opportunity arises to build a fair, prosperous, safe, and sustainable world for human beings understood as a social being over an economic one.
Smart computing is an essential multi-disciplinary area where advanced computational methods and technologies are combined with engineering approaches to create systems, applications, and new services that meet the needs of society. In such a context, innovation requires both conceiving new applications and services, as well as improving the security, efficiency, reliability, and sustainability of the existing ones.
During my Ph.D. program, I carried out a high-profile scientific training on building software systems and applications that meet the needs of users and society through the main exploitation of advanced methods of computer science and engineering.

The training events I have attended during my learning path influenced my research activities. Below is a list of the training activities that I consider the most significant for the way of growth during these years of study activity.

- *AdHoc-Now 2017 - 15th International Conference on AdHocNetworks and Wireless.* University of Messina, Messina, (Italy). September 20th-22th, 2017.

- *Multimedia security* - Ph.D. Course. ICT International Doctoral School. University of Trento, Italy. May 29th - Jun 01st, 2017.

- *ICVSS 2017 - International Computer Vision Summer School 2017 - "From Representation to Action and Interaction.* Hotel Village Baia Samuele, Ragusa, Sicily, (Italy). Jul 09th-15th, 2017.
- *I training school - MULTIFORESEE Cost Action CA 16101 - "CSI requirements for evidence handling within the application of imaging solutions".* West Yorkshire Police Training And Development Centre Carr Gate Complex Bradford Road, Wakefield, (United Kingdom). August 21st-25th, 2017.
- *IEEE S3C-EU 2017 - IEEE European Summer School on Smart Cities - "Improving the citizens' quality of life".* Polo Scientifico e Tecnologico "Fabio Ferrari", Trento (Italy). September 04th-08th, 2017.
- *ICIAP 2017 - International Conference on Image Analysis and Processing.* Monastery of San Nicolò l'Arena, Catania, Italy. 11st-15th September, 2017.
- *SSC 2018 - 4th International Workshop on Sensors and Smart Cities.* Giardini Naxos, (Messina), Italy. June 18th, 2018.
- *SMARTCOMP 2018 - 4th IEEE International Conference on Smart Computing.* Giardini Naxos, (Messina), Italy. June 18th-20th, 2018.
- *Lipari School on Computational Complex and Social Systems - "From swarm intelligence to digital democracy: new tools for a complex society".* Hotel Giardino Sul Mare - Lipari Island (Italy). July 19th-25th, 2018.
- *1st International Staff Training Week at Dicle University - "Erasmus+ Staff Mobility Programme".* Dicle University, Diyarbakir, Turkey. April 15th-19th, 2019.
- *SSC 2019 - 5th International Workshop on Sensors and Smart Cities.* Washington, DC, United States. June 12th, 2019.
- *SMARTCOMP 2019 - 5th IEEE International Conference on Smart Computing.* Washington, DC, United States. June 12th-14th, 2019.
- *Lipari School on Computational Complex and Social Systems - "Data Science".* Hotel Giardino Sul Mare - Lipari Island (Italy). July 19th-25th, 2019.

- *6th International Staff Training Week of the University of La Rioja - "Erasmus+ Staff Mobility Programme"*. University of La Rioja, Edificio de Rectorado, Logroño, Spain. September 16th-20th, 2019.

- *2019 IEEE Technical Meeting on Reliable, Safe, Secure, and Time-Deterministic Intelligent Systems*. Grand Hotel Majestic Già Baglioni, Bologna, Italy. December 6th, 2019.

- *1st IEEE Computer Society Global Chapter Summit*. School of Engineering of the University of Bologna, Bologna, Italy. December 7th, 2019.

- *"Interdisciplinary Collaborations: linking Creative Cultures and Business" at Staffordshire University - "Erasmus+ Staff Mobility Programme"*. Staffordshire University, Stoke-on-Trent, United Kingdom. January 20th-24th, 2020.

These activities taught me how to combine theoretical and practical aspects with improving my understanding of how to design and build computing systems. Moreover, I learned how to use computing technology to design and develop applications and services and make human life better.

## Structure of this Dissertation

This dissertation provides an overview of some of my recent research developments in the field of smart computing and its applications.
All the research activities described in this manuscript have been carried out with a single broader vision, aimed at leveraging science, technology, and engineering to benefit human welfare.
To facilitate the readers' understanding of my study path, I have organized my thesis into three mains Sections with a total number of ten chapters, excluding the Introduction and the Conclusion.
Each part contains chapters related to each other, which refer to a macro-area of issues that I have studied in-depth and which I dealt with trying to give new solutions, even using new approaches and methodologies.
All these chapters contain my contributions in terms of new systems applications and services. These contributions share the same *fil rouge*, which is to foster technological innovation and excellence for the benefit of humanity.

Each chapter outlines a single research activity introducing the specific problem addressed, the purpose of the study, the motivations the technological choices of the new resolution strategies proposed and presenting the background of existing solutions already present in the literature as a comparison method.

A brief introduction of the parts is as follows.

Section I discusses `confidentiality` and `integrity` issues concerning `data` stored on Cloud storage services, which are part of `Multi-Cloud` Environments. Chapters from 1 to 4 focus on individual aspects that outline the `innovative approach` identified as possible solutions to the challenges mentioned above. Chapter 1 presents a *new cloud storage service* that embodies the proposed approach. Chapter 2 proposes a *new mobile application* to extend and enable this approach also to the mobile world represented by smart devices; Chapter 3 intends to *evaluate the goodness of this approach* in a quantitative way comparing it with other commercial solutions under specific conditions. Chapter 4 shows a *new Web Framework* compliant with the proposed innovative approach.

Section II focuses on the need to respond to the lack of tools that allows more `flexible decision-making processes` allowing profiting from the most varied sources of data available. The focus is on the *decision support simulation tools* of *Cloud Service Providers (CSPs)*. CPSs need both new simulation tools and also new strategies to evaluate the conceived Cloud algorithms and policies before their actual development and deployment to avoid unnecessary consumption of resources. Chapters from 5 to 7 intend to provide tools and strategies to give the answer to the problem mentioned above. Chapter 5 discusses the design and development of a *simulation-based tool for Cloud Service Providers (CSPs) Brokerage ecosystems*. Chapters 6 and 7 deal with the creation of a *new Multi-Criteria Decision Making (MCDM) strategy* and a *new algorithm* that, in the right order, address the strategic perspectives to capture business value from the IoT-Cloud union or the O&G industries and select best green choice for processing's resource allocation for data centers in a Community Cloud ecosystem context.

Section III focuses on the need for `enhancement of human capital` by

creating new digital education systems able to specifically improve the students' learning process. In this regard, Chapters from 8 to 10 show how it is possible to build added value in learning activities through the design and implementation of new software solutions which integrate *Machine Learning* and *Gamification* concepts along with *Cloud and IoT Technologies*.
Chapter 8 presents a mobile app designed and implemented to help *digital natives* in the development of the correct study method by interacting with a digital virtual study assistant able to assess and suggest ways that help them improve their learning performance, while building a personalized path based on individual interactions. Chapter 9 talks about a novel *Intelligent Tutoring System - ITS* able to detect and process data obtained from a user's teaching activities during the learning of a theoretical concept taken from a written text and, therefore, to return assessments on acquired skills and predictive analysis. Chapter 10 discusses *IOT-OPEN.EU*, an educational project within the *Erasmus+ Key Action 2 framework*, which represents an *excellent and timely solution* that may be adopted worldwide by schools and universities that were forced by their governments to quickly shutdown due to the recent outbreak of the SARS-COV-2 virus and the related *COVID-19 pandemic*.

Details on the individual chapters are as follow.

- Chapter 1 presents and discusses a **new secure storage Cloud approach** oriented to guarantee *confidentiality* and *integrity* issues concerning information and data which are spread and stored in Multi-Cloud environments distributed all over the world. The ***Secure Storage in Multi-Cloud Environment - SSME Cloud Service*** arises as a novel solution to combine symmetric and asymmetric cryptography by offering user-friendly and dynamic management of both fragmentation schema and the pool of Cloud storage services available during the Cloud storage operations. It also responds to the needs of protection against insider attacks.
  The Chapter addresses aspects such as security reasons, technological choices, previous articles that motivated this scientific research, and an overview of current knowledge in that area.
  This study ends with an in-depth ***performance analysis*** section, which includes several experiments by considering the experimental implementation of the SSME Cloud Service as a real Cloud storage ser-

vice hosted at the *Cloud Data Center of the University of Messina* [98]. The analysis focuses on evaluating the behavior of the security service considering the system's *"overall response time"* related to the *Upload* and *Download Modes*, as well as sub-processing phases. The analysis considers the variation of the *"response time"* of the system varying the size of the files to be processed and the size of the fragmentation schema to be applied.

- Chapter 2 presents and discusses a **new *mobile* secure storage approach**, which is primarily oriented to guarantee *confidentiality* and *integrity* issues concerning data stored on mobile smart devices that are part of a multi-Cloud environment. The ***"ARIANNA" mobile application*** consists of an Android mobile application that represents the software *enabler*, which allows extending the experimental multi-Cloud system, already discussed in Chapter 1, towards the mobile world represented by the smart devices.
  The Chapter includes sections that provide security reasons, technological choices, previous articles that motivated this scientific research, and an overview of current knowledge in that area.
  This study ends with the ***performance analysis*** section, which reports several performance assessments considering the implementation of the ARIANNA app in a real multi-Cloud environment scenario which includes the SSME Cloud Service hosted at the *Cloud Data Center of the University of Messina* [98].
  The analysis focuses on the evaluation of the behavior of the ARIANNA app considering the *"overall response time"* of the system both as regards the *Upload* and *Download Modes* as well as sub-processing phases. Besides, the analysis considers time variations that occur when the ARIANNA app is used in different mobile networks, such as *"ADSL"* and *"4G"*.

- Chapter 3 presents and discusses a ***performance study*** that compares some commercial Cloud storage services such as *Google Drive* [66], *Dropbox* [51] and *OpenStack Swift* [106], with the multi-Cloud approach enabled by the "ARIANNA" app, already introduced and motivated in Chapters 1 and 2. The Cloud storage services used as a reference are well-known and highly appreciated commercial solutions that offer a certain level of free service without providing benefits in

terms of confidentiality and data integrity to their users.
This research aims to evaluate ***"how much"*** the *overall time overhead* introduced by the ARIANNA approach costs.
The Chapter reports a ***quantitative performance analysis*** section that compares the ARIANNA app with the other Cloud storage services in a real scenario. The analysis focuses on measuring the *"overall response time"* of the system. The evaluation considers the time of upload and download of a single file to and from a Cloud system (both for the individual Clouds and multi-Cloud) that users perceive.
The study ends with a numeric answer on the *"how much"* question. That means to evaluate if it is useful using the multi-Cloud solution enabled by the ARIANNA application rather than other single storage services. In order to quantify the differential introduced by the ARIANNA app, the study presents two indicators of *Percentage Difference* both for upload and download phases.

- Chapter 4 presents and discusses a Web client application, which is part of a ***Web Framework*** compliant with the SSME-middleware policies and protocol described in Chapter 1. The Web app consists of a branch of the end-user application already discussed in Chapter 2.
  The Chapter discusses the implementation of the proposed Framework of Web client secure storage and its key features. Furthermore, it includes sections that provide security reasons, technological choices, previous articles that motivated this scientific research, and an overview of current knowledge in that area.
  The study ends with a ***performance analysis*** section, which reports some performance assessments considering the implementation of this Web app in a real multi-Cloud environment scenario.
  The analysis focuses on evaluating the behavior of the security service in terms of system's *"overall response time"* related to the *Upload* and *Download Modes*, as well as sub-processing phases. In addition, the analysis considers the time variations that occur when the Web application is used in different networks, such as the public Internet or a private Virtual Private Network service.

- Chapter 5 mainly presents and discusses the study which led to the design and the development of a **new simulation-based tool for Cloud Brokerage ecosystems**. This new tool consists of an evolu-

tion of the **J2CBROKER** *Simulation Tool* [60], redesigned according to the Cloud *Software as a Service* (SaaS) model and integrated in the OpenStack environment.
The goal of this research is to provide *decision support simulation tools* to *Cloud Service Providers* (CSPs) who need timely, repeatable, and controllable methodologies that evaluate the conceived Cloud algorithms and policies before their actual development and deployment.
The Chapter also includes sections that provide motivations, software architecture design choices, previous articles that motivated this scientific research, and an overview of current knowledge in this area.
This study ends with a section dedicated to a ***case study*** in which J2CBROKER is deployed as a Service in a real CSP Brokerage scenario.
The *case study* introduces a new **sustainability-Cost Model** geared towards finding the "best choice" for *resource allocation* in a given scenario. The proposed model combines some *sustainability metrics*, i.e. metrics that can define *"how green a datacenter is"*, with *availability* and *monetary cost* criteria in an innovative way by using a *multi-criteria approach*.

- Chapter 6 presents, discusses and evaluates a ***new Multi Criteria Decision Making (MCDM) brokerage strategy*** allowing cooperative small-medium size IoT-Cloud Service Providers to satisfy the request for ***IoT-Cloud services***, while establishing a good compromise between service level and business for the ***O&G industries***.
The novelty element introduced by this research is the proposed ***multi-criteria approach*** that fits the strategic perspective to capture business value from the IoT-Cloud union.
The proposed strategy consists of an evaluation function that considers the following criteria: *Operational Availability, Storage Capacity Service Price, Data Analytics Service Price, Cybersecurity Level, Support Level*.
The Chapter also includes sections that provide motivations, previous articles that motivated this scientific research, an overview of both the IoT-Cloud brokerage scenario and the proposed MCDM strategy.
This study ends with a section dedicated to a ***case study*** carried out using the **J2CBROKER** (Chapter 5) in a real CSP Brokerage sce-

nario .
The analysis focuses on the evaluation of the scenarios that take into account different weight distributions both for the five criteria identified and various typologies of requests for IoT-Cloud services.

- Chapter 7 introduces a study that addresses medium and small size Cloud Service Providers towards solutions allowing them to compete with large Cloud providers in a more **sustainable service marketplace**.
  The novelty element introduced by this research is the ***low carbon strategy*** designed to make the best choice in resource allocation, based on sustainability, availability, and costs.
  The proposed ***energy-aware Brokering Algorithm (eBA)*** allows pushing down carbon dioxide emissions through the Community Cloud ecosystem, by running instances at the most convenient sites.
  The Chapter also includes sections that provide motivations, previous articles that motivated this scientific research, a description of the sustainability metrics proposed in the algorithm and a report of the energy-aware resource allocation approach.
  This study ends with a section dedicated to a ***case study*** section, carried out using the **J2CBROKER** (Chapter 5) in a real CSP Brokerage scenario .
  The analysis demonstrates the goodness of the proposed *energy-aware Brokering Algorithm (eBA).*

- Chapter 8 presents and discusses the software architecture, technological, and philosophical reasons behind the development of the Android mobile application ***"Virtual Study Partner - VSP"***. VSP represents the first output of a more extensive research activity still in progress that aims to provide study support to young people at school-age. The study seeks to solve ***digital natives' school productivity problems***.
  The study explores how, in spite of having all the knowledge they need within a *"click"*, digital natives pay the price of an unfavorable socio-economic situation, which has negative repercussions on their school productivity.
  VSP proposes a novel technological solution to help learners in the development of the correct study method by interacting with a digital

virtual study assistant able to assess and suggest methods to improve their learning performance by building a personalized path based on individual interactions.
The Chapter includes sections that describe the VSP features', technological choices, previous articles that motivated this scientific research, and an overview of current knowledge in that area.
This study ends with a section that delves into the description of the ***metrics*** of the ***Google Cloud Natural Language Service*** [38]. The extraction and processing of these metrics constitute a crucial part of the VSP's application, both for ***Basic Knowledge Creation*** and ***Training***. These are *Sentiment Analysis, Entity Analysis, Syntactic Analysis, and Content Classification.*

- Chapter 9 presents and discusses technological and philosophical reasons which lead to the design and the development of a new ***Intelligent Tutoring System - ITS***. The intent of this new ***cognitive tool***, named ***"Virtual Study Buddy"***, is to create a *Learning Platform* able to detect and process data obtained from a user's teaching activities during the learning of a theoretical concept captured from a digital text and, therefore, to return assessments on acquired skills and predictive analysis based on a systematic comparison between the digital text and the text of the speech.
  The novelty element introduced by this research is the design of a tool that uses the best technological innovations (Machine Learning, Mobile development, Cloud Computing services) to promote its use systematically and sustainably by teachers, and more generally, from the education system. These features allow the ITS to overcome technical maintenance problems, also overcoming well-known longevity problems.
  The Chapter also includes sections that provide motivations, choices, theoretical and technological considerations about the concept of Gamification in education and about the real worth of using ITS in current school systems.
  The Chapter documents how all research throughout the last 30 years suggests that ITSs is more effective in student learning over the traditional education community. Furthermore, it tries to answer the two long-standing questions: *"Why did ITSs never come out of university*

*research laboratories?"*, and *"Why didn't they prosper?"*.

- Chapter 10 presents and discusses the ***IOT-OPEN.EU remote laboratory infrastructure and IoT courses***, which were designed and implemented as part of the IOT-OPEN.EU Erasmus+ project.
  IOT-OPEN.EU is an educational project within the ***Erasmus+ Key Action 2 framework***, oriented towards a Strategic Partnership between Higher Education (HE) and commercial bodies.
  The valuable experience gained by this project represents an ***excellent and timely solution*** that may be adopted by schools and universities around the world which were forced by their governments to the lockdown due to the recent outbreak of the SARS-COV-2 virus and related ***COVID-19 pandemic***. Lots of universities and schools these days have switched from physical classrooms to virtual or online classes. This approach is working well for theoretical subjects and courses, but it is not straight forward in the case of laboratory subjects and courses that require access to hardware resources.
  The Chapter discusses IoT related technologies and platforms that the IoT training sector can lever. The presented solution has been introduced successfully into the participating universities' curriculum on the Internet of Things.
  Pilots performed in the Silesian University of Technology, covering classical, online courses and use of VREL labs and IOT-OPEN.EU project-created content in the years 2017-2020, present and prove usability and reasonable approach to distance learning with this kind of tools, as well as indicating the growing popularity of the mixed learning model, where students use both on-site and online materials.
  The Chapter also includes sections that provide motivations and technical details about the IOT-OPEN.EU VREL lab implementations, current advances in distant learning, and remote laboratory models.

An ending Chapter, which is not numbered, wraps up this work with the conclusions and proposes further work related to the presented subjects.

This dissertation interpolates material from several published papers by the author:

- Papers [47], [43], [44] and [116], respectively for Chapters 1, 2, 3 and 4;

- Papers [46], [61], [59] and [63], respectively for Chapters 5, 6 and 7
- Papers [41] and [45], respectively for Chapters 8, and 9;
- Paper [125] for Chapter 10.

# Part I

# Part: How Enhancing Confidentiality and Integrity on Data in Multi-Cloud

# Chapter 1

# Secure Storage as a Service in Multi-Cloud Environment

## 1.1 Brief Introduction to the Problem

The growing number of requests to store and share files in Cloud environments results in an ever-growing number of user-friendly Cloud services to meet these demands.
Cloud storage allows you to store data in multiple remote sites, generally owned by "top" companies and to run the solutions of their respective owner, for example, Google Drive, Dropbox, Amazon Simple Cloud Storage Service (S3). In addition to the proprietary solutions mentioned above, there are other open-source solutions to provide Cloud storage services. For example, users can use "Swift" the OpenStack Object Storage service to store large amounts of data efficiently and economically.
The development of the Cloud storage market depends, in particular, on the ability to build economies of scale. As part of the *Digital Single Market Strategy*, *European Union* establishes a free flow of data in Europe, facilitating data portability and the passage of Cloud service providers. The *"SMART 2013/0043 - Uptake of Cloud in Europe"* [31] study indicates that Cloud developments could lead to the growth of the European Cloud market from €9.5bn in 2013 to €44.8bn by 2020 (i.e., almost five times the market size in 2013). In 2014, around 19 % of EU businesses used Cloud computing primarily to host their email systems and electronic file storage. Besides, further estimates from this study highlight that four out of ten companies

(39%) using the Cloud reported that the risk of a security breach was the main limiting factor in the use of Cloud computing services. Figure 1.1 [64] shows the use of Cloud computing services in business for 2014 and 2016 purposes (% of companies using the Cloud). The Results confirm the most significant growth in the percentage quota of companies that use Cloud to store files: from 53% to 62%.

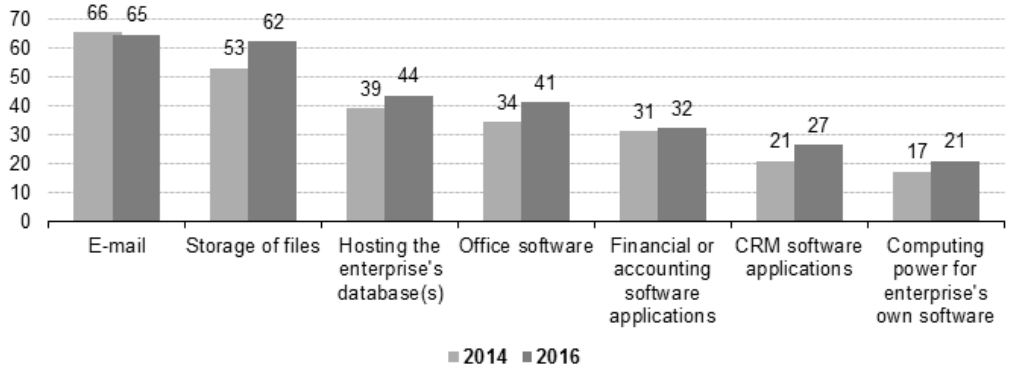


Figure 1.1: Use of Cloud computing services in enterprises, by purpose, 2014 and 2016 (% of enterprises using the Cloud) [47].

In such a scenario, users who wish to move data to the Cloud should consider avoiding:

- costs of building and maintaining a private storage infrastructure;
- costs to avoid unauthorized access by third parties;

without reducing legal certainties. In fact, because of the easy file storage and sharing services widely accessible by different types of customers and contexts (e.g., companies, academics, and many others), more and more personal and confidential data may be exposed to privacy and security vulnerabilities.

This Chapter presents the **Secure Storage in Multi-Cloud Environment - SSME** architecture, which addresses the confidentiality and integrity issues concerning data store and dissemination in worldwide distributed Cloud environments. The architecture implements a novel solution which mainly uses encryption at client-side, and a worldwide distributed middleware for data splitting, distribution, and retrieval.

The rest of this Chapter is organized as follows. Section 1.2 discusses the motivation of the study. Section 1.3 gives a brief overview of existing

literature about the use of multi-Cloud storage using cryptographic data splitting. Section 1.4 describes the SSME architecture. Section 1.5 details the SSME implementation. Section 1.6 intends to evaluate the results of the performance analysis.

## 1.2 The Goal of the Study

Cloud Computing Services generally provide resource management's functional, which guarantees security in accessing services and in data communicating. However, they ofter lack data protection from malicious system-wide direct access. It is mandatory to ensure data protection mechanisms to deny the intelligibility of data to unauthorized users, even when they are (local) system administrators.

The study aims to provide a software mechanism capable of storing data in a multi-Cloud environment in a secure way, providing an answer to the confidentiality and integrity issues of information and data disseminated and stored in remote distributed machines all around the world. SSME intends to introduce the above-mentioned data protection mechanism by combining both symmetric (AES256) and asymmetric (RSA) encryption. Besides, users can use dynamic management of both the fragmentation schema and the pool of Cloud storage services to use to create their multi-Cloud environment whenever they want to store a file on the Cloud. In this proposal, all configurations on the client-side are invisible on the server-side. Each execution of the SSME application isn't tracked anywhere on the system, nor the physical disk of servers. Using the *Trusted Control Service* (TCS, see the Section 1.4.1), users are able to *integrate* Cloud services they trust for the significant Cloud computing features of *authentication* and *backup-storage*. This opportunity significantly increases their confidence in using the SSME application.

Many works in the literature concern Cloud and multi-Cloud storage systems using cryptographic data splitting. The next Section presents the background and how the existing approaches and solutions differ from the SSME.

## 1.3 Background of the Study

In [26], the authors present an architecture for a cryptographic storage service. It consists of four components: a server which processes and encrypts (AES256) data before it is sent to Cloud, a private Cloud that holds the meta-data information, and two Clouds that respectively archive one half of each user's file. The authors assume that the remote server is trusted without specifying any information on "how" this trustiness is implemented. The meta-data information (e.g., passwords, secret keys of each file, encrypted access paths) are securely stored in the private Cloud. If compared with our dynamic approach, which allows specifying the size of the split fragments, here data splitting is statically fixed on half of each user's file.

In [19], the authors propose a new method for securing the user's data using the multi-Clouds in an untrusted mobile Cloud environment. This method splits data into segments that are successively encrypted, compressed and distributed via multi-Clouds while keeping one "segment" on the mobile device memory. Keeping one segment in the user's device will prevent any attempt to recover the distributed data, thus to avoid the grabbing of all the segments together with the key by possible unauthorized users. In contrast to our approach, their solution requires a Mobile Cloud Computing (MCC) architecture and keeps a segment in each device.

In [129], the authors present an architecture that makes the data splitting of the uploaded file size by three. Their architecture consists of a System Database and a Middleware for data slicing and merging and for data encryption and decryption. Also, in this work, data splitting is statically fixed on a third of each user's file. Another difference is they use the System Database to store the information necessary for the Middleware operations without specifying its content (i.e., the type of information).

In [111], the authors propose a reliable storage system, *TrustyDrive*, that takes care of both the document anonymity and the user anonymity. The system architecture consists of three layers: the "end-users" that use a storage system as a service that splits and encodes the files; the "dispatcher" that provides an entry point for both the end-users and the Cloud providers; "Cloud providers" that offer different storage space in terms of costs and performances. At the client-side (the end-user), documents are split into blocks, and for each block, meta-data are associated (user meta-data). The user breaks its meta-data into chunks to be sent to the dispatcher who,

in turn, computes missing information to store meta-data blocks on Cloud providers. However, the authors do not describe the secure communication between end user-dispatcher and dispatcher-Cloud provider. Moreover, differently from their architecture, our mechanism of splitting and dissemination of the fragments is configurable by the user, which can specify the size of the fragments and the pool of Cloud providers.

The TwinCloud client-side encryption solution presented in [30] focuses on the *secure sharing* on Clouds without explicit key management. To this end, they highlight the Public Key Infrastructure (PKI)-based solution problems, i.e., costs due to get a certificate from a Certification Authority (CA) and PKI-infrastructure maintenance. Differently from the TwinCloud solution, our architecture uses both symmetric and asymmetric cryptography.

In all the previous contributions, the authors do not give specific information on how they manage the keys, neither they describe where the keys are stored and how and who can access them. Moreover, in the contributions mentioned above, none of the virtual access points (e.g., gateway, dispatcher, server) to the various multi-Cloud environments which represent the central nodes of the processing offer a scheme quickly and dynamically configurable by users. Neither adopts and describes the use of a secure communication protocol for its communications nor works only in volatile memory. It means that the processed data are not protected against insider attacks [76].

## 1.4 Proposed SSME Architecture

Fig. 1.2 shows a general scheme of the SSME architecture. It consists of a JAVA client-server application that uses a stateless RESTful approach for its communication, where a client cooperates with the middleware where most of the computation is done. The *SSME middleware* is, in turn, made up of two main components: a "Trusted Cloud Service" (TCS) implementing all the fundamental interfacing functions with SSME clients and a "Server" where all the middleware file manipulations are provided.

An SSME compliant application can work in two different modes: *Upload Mode* and *Download Mode*.

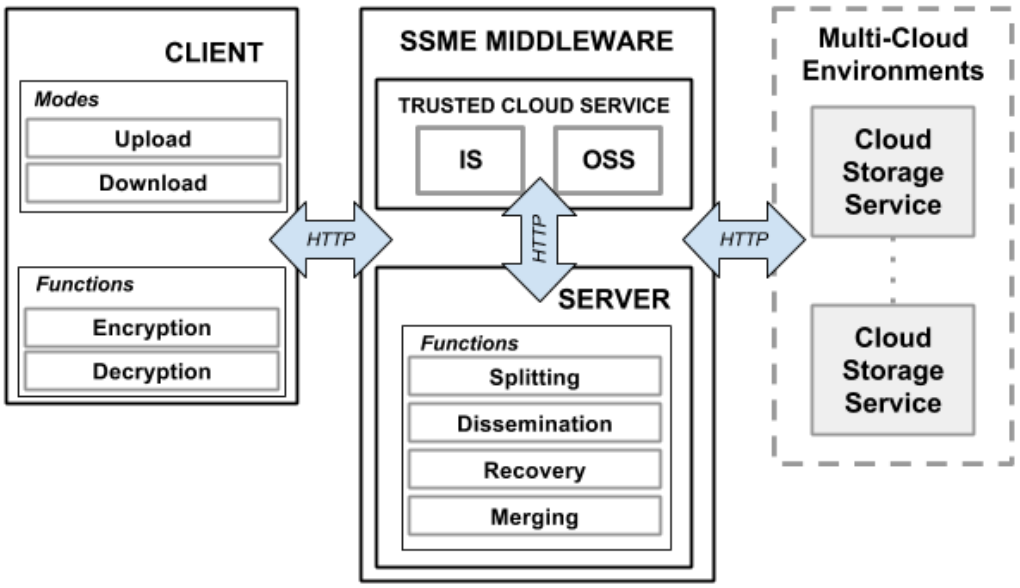


Figure 1.2: SSME architecture general scheme [47].

### 1.4.1 SSME Middleware

#### Trusted Cloud Service

The **"Trusted Cloud Service" (TCS)** is the architectural component that provides some significant Cloud computing functionalities to the SSME client-server application. It is composed in turn into a two different sub-services: the **"Identity Service" (IS)** and the **"Object Storage Service" (OSS)**. As the name implies, the IS identifies the own trusted Identity Service the user may want to adopt by running the SSME client-server application. The IS, by exploiting its own token service mechanism, provides authentication and authorization on Cloud for the SSME service requests. In the same way, the OSS identifies the own trusted Object Storage Service that the user may want to adopt by using the SSME Application. The OSS provides temporary backup needed during the SSME internal operations. The OSS can also be used to store the encrypted JSON file that results as the output of the SSME application. This output file represents the only *guiding light* to retrieve that particular file from the Cloud. SSME adopts the approach of externalizing the primary functions of Identity and Object Storage because each SSME user could have his/her own trusted Cloud services, for example, personal or provided by his company, and may want to use these

in the SSME scenario. In our testbed scenario, we adopt an OpenStack [104] compliant TCS.

#### SSME Server

All processing of the server is carried out without leaving any trace on local storage (hard disk) because it works only in volatile memory. In brief, the functions performed by the server are the following:

- uploading and downloading files to and from the TCS;
- fragmentation, recovery, and merging of fragments to and from the Cloud services which are part of the multi-Cloud environment used at the moment;
- decryption of the information contained in the Headers of the HTTP requests received from the client;
- creation and encryption of the *json-encrypted-file* file which represents the output of the *Upload Mode*. This file represents the only way to recover the file from the Cloud.

### 1.4.2 SSME Client

In order to work, the client needs the presence of a mandatory JSON configuration file called *json-conf-file* file. For smooth functioning of our service, the *json-conf-file* file must be filled in the proper way. As the name implies, the *json-conf-file* file contains information about the configuration of the client, in particular:

- its internal settings (including the symmetric encryption key used);
- the communication with the Server (including the public key of the Server);
- the services that make up the TCS;
- the Cloud storage service providers which are part of the multi-Cloud environment the user wants to use.

In brief, the functions performed by the client, when working in either *Upload Mode* or *Download Mode*, are the following:

- encryption of the file you want to send to the multi-Cloud environment;
- decryption of the file you want to receive from the multi-Cloud environment;
- encryption of the information contained in the Headers of the HTTP requests sent from both the server and the TCS;
- decryption of the information contained in the Headers of the HTTP requests received from both the server and the TCS.

### Upload Mode

During the *Upload Mode*, the client reads the information inside the *json-conf-file* file in order to instruct itself on how to contact the service. According to the reported information, the client starts the processing. The client sends to the server some HTTP requests according to the rule described in Section 1.4.3. Once the server received all the needed information, it elaborates them and returns the client an AES256 encrypted JSON file, called *json-encrypted-file* file. The *json-encrypted-file* file contains all the relevant information in order to retrieve and rebuild all the fragments scattered among the Cloud storage services. Referring to classical mythology, this file serves as a modern *Ariadne's thread*. In the absence of the *json-encrypted-file* file, or if it is damaged, it is not possible to recover and rebuild the fragments stored in the Cloud. The *Upload Mode* will return a *json-encrypted-file* file for each file uploaded in the Cloud. The template structure of the *json-encrypted-file* file is shown in Fig. 1.3.

```
{ "File": { "FileName": "", "DirName": "", "SliceSize": "" },
  "Fragments": [ { "FragmentName": "", "FragmentNumber": "", "FragmentMD5": "", "ServiceType": "dropbox",
                   "DropboxToken": "" },
                 { "FragmentName": "", "FragmentNumber": "", "FragmentMD5": "", "ServiceType": "openstack",
                   "OpenStackUser": "", "OpenStackPassword": "", "OpenStackTenant": "", "OpenStackUrl": "" },
                 { "FragmentName": "", "FragmentNumber": "", "FragmentMD5": "", "ServiceType": "gdrive",
                   "GDriveJsonFile": "", "GDriveUrlFile": "", "GDriveIdFile": "" }
               ]
}
```

Figure 1.3: The *json-encrypted-file* file template structure [47].

The structure is composed of two main parts: the field "File" and "Fragments". The field "File" contains the name of the original file (i.e. FileName), the name of the container/directory where the fragments are stored

inside the Cloud storage services (i.e. DirName) and the dimension of the single fragment (i.e. SliceSize). The field "Fragments" contains all the information needed to retrieve all the fragments. This field is structured as a JSON vector. In the actual implementation, the elements of this vector can assume three different typology: *dropbox*, *openstack*, *gdrive*. Each of these represents the typology of the Cloud storage services used by our SSME testbed.

#### Download Mode

During the *Download Mode*, the client reads the information inside the *json-encrypted-file* file to follow the *Ariadne's thread*. The *json-encrypted-file* file is related to the original file that the client wants to retrieve from the Cloud. The client decrypts *json-encrypted-file* file by using the symmetric key specified in the *json-conf-file* file and extracts the information contained to process it. The client sends to the server all the information contained in the *json-encrypted-file* file according to the rule described in Section 1.4.3. This information is needed to the server in order to retrieve and rebuild all the fragments of the file the user wants to download from its own multi-Cloud environment. These fragments were previously scattered among the Cloud storage services during the *Upload Mode*.
After the *Download Mode*, the client will obtain its original file (still encrypted). At this time, firstly the client deletes the *json-encrypted-file* file related to the file just receive, then in order to remove all the fragments stored in the multi-Cloud environment, the client sends some `delete-fragment` requests to the server. In the end, the client will decrypt the original file.

### 1.4.3 Communication between the Client and the Server

All the HTTP communications between the client and the server are authenticated using the token service provided by the chosen IS. The testbed uses the token service provided by the OpenStack Identity Service v2.0 (Keystone). Each HTTP request sent by the client embeds some fields that contain the information necessary to carry out a given task at the server-side. All these fields are symmetrically encrypted by using the AES256 algorithm. The key to decrypt these fields is also embedded in each request inside a field called "key". The client encrypts this field by using the RSA asymmetric algorithm with a key of 2048 bytes, by using the server's public key. The

server, through its own private key, can decrypt the "key" field and using it to decrypt the content of all the other fields stored in the HTTP request body. It is worth noting that the public key of the server is known and is freely downloadable from the web.

## 1.5 Implementation Details

In this Section, deeper details on how the client and server interact in the proposed architecture are presented. Using the communication protocol here described it is possible to design a client compliant with a public service based on SSME.

### 1.5.1 Client-server Communication Protocol in "Upload Mode"

The communication phase between the client and the server is done in seven different steps: (see Fig. 1.4)

1. First of all, in order to verify if the server is alive, the client sends to it an HTTP request (**1.A**). If the test is successfully done (**1.B**), then the client can move to the next step (**2.A**).

2. In order to be authenticated by the "Trusted Control Service" (TCS), the client sends an authentication request to the "Identity Service" (IS) (**2.A**). If the IS returns the token (**2.B**), which means that it is successfully authenticated, then the client can move to the next step (**3.A**).

3. In order to test the validity of the Cloud storage services described in the *json-conf-file* file, the client checks each of them. The client sends an HTTP request to each of the Cloud storage services included in the list (**3.A**). If a Cloud storage service returns a positive response (**3.B**), the client sends another Cloud storage service test request (**3.A**), repeating the checking until the end of the storage service list.

4. In order to put the file to store in our multi-Cloud environment on the OSS:

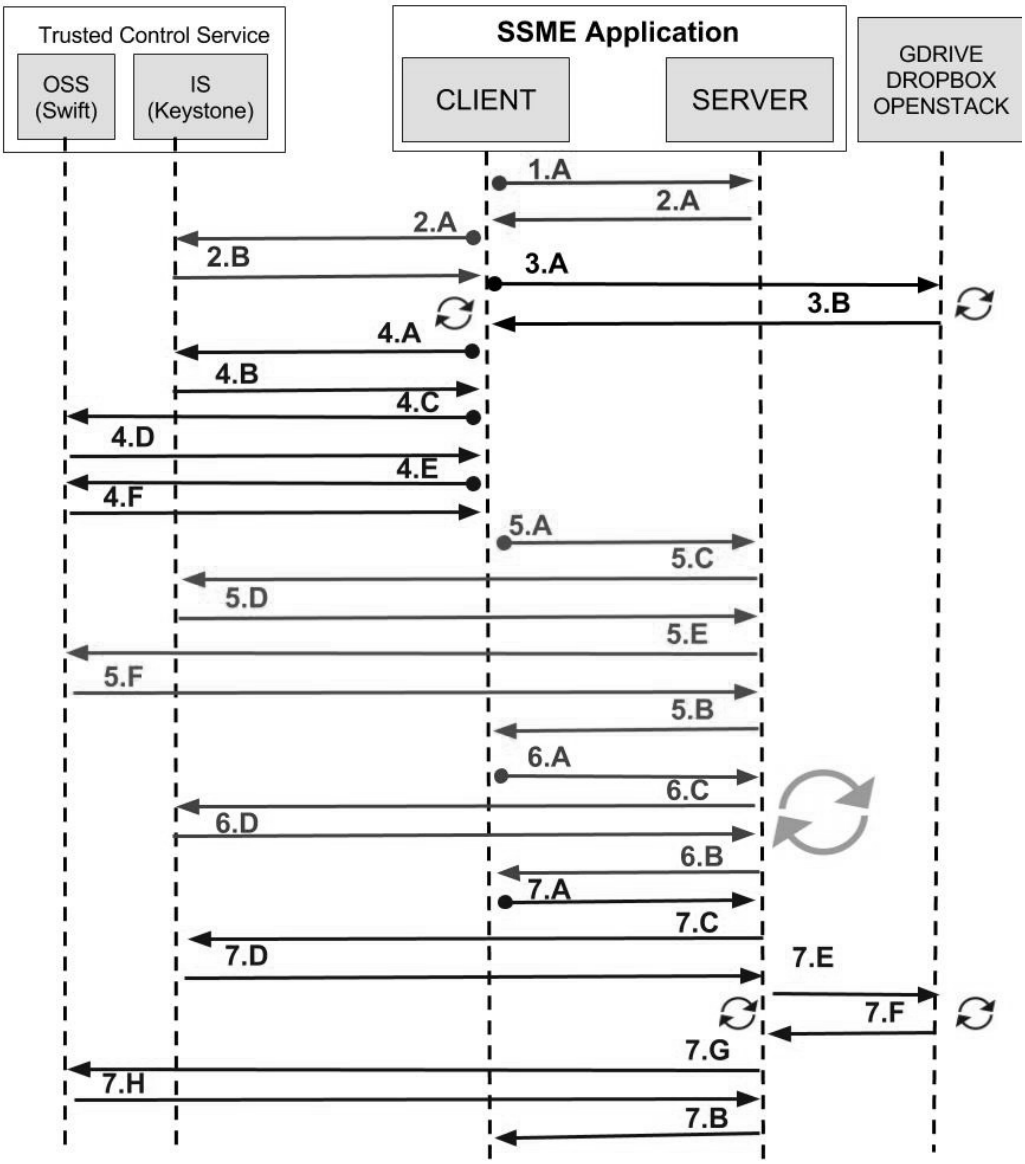


Figure 1.4: Client-server Communication Protocol in "Upload Mode" [47].

(a) firstly, the client sends an authentication request to the IS (**4.A**). If the IS returns the token (**4.B**), then the client can move to the next step (**4.C**);

(b) then, the client sends an HTTP request to the OSS (**4.C**). If the container creation is successfully done (**4.D**), then the client can move to the next step (**4.E**);

(c) finally, the client sends an HTTP request to the OSS **(4.E)**. If the the create-object request was successfully done (**4.F**), and the client can move to the next step (**5.A**).

5. To start the server-side processing phase, the client sends an HTTP request to the server (**5.A**). The server checks the received request by interrogating the TCS (**5.C**), and if the request is successfully authenticated (**5.D**), then the server starts the download of the file the user want to put on the multi-Cloud environment (**5.E**) from the OSS. When the server completes the download with success (**5.F**), it returns a response (**5.B**), and the client moves to the next step (**6.A**).

6. In order to transfer all the information about the Cloud storage services to the server, the client uses an HTTP request for each Cloud storage service (**6.A**). Each request contains the information about a particular Cloud storage service the client wants to transfer at that moment. The server checks the received request by interrogating the TCS (**6.C**) and, if the request is successfully authenticated (**6.D**), the server returns a positive response (**6.B)**), and the client sends another Cloud storage service request, repeating the process until the end of the Cloud storage service list.

7. To complete the server-side processing phase, the client sends a `split` HTTP request to the server (**7.A**). The server checks the received request by interrogating the TCS (**7.C**) and, if the request is successfully authenticated **(7.D)**, the server starts the splitting phase of the file just downloaded from the OSS. After the splitting, the server sends each fragment to a different Cloud storage service by following a random mechanism (**7.E**). If the upload-request for a fragment is successfully done (**7.F**), the server sends another upload-request, and so on, until the end. When the server has sent all the fragments to the Cloud storage services, a `delete-fragment` HTTP request is sent to the OSS (**7.G**) and, if successful (**7.H**) the server moves to the next step (**7.B**). At the end, the server returns to the client the *json-encrypted-file* file (**7.B**).

### 1.5.2 Client-server Communication Protocol in "Download Mode"

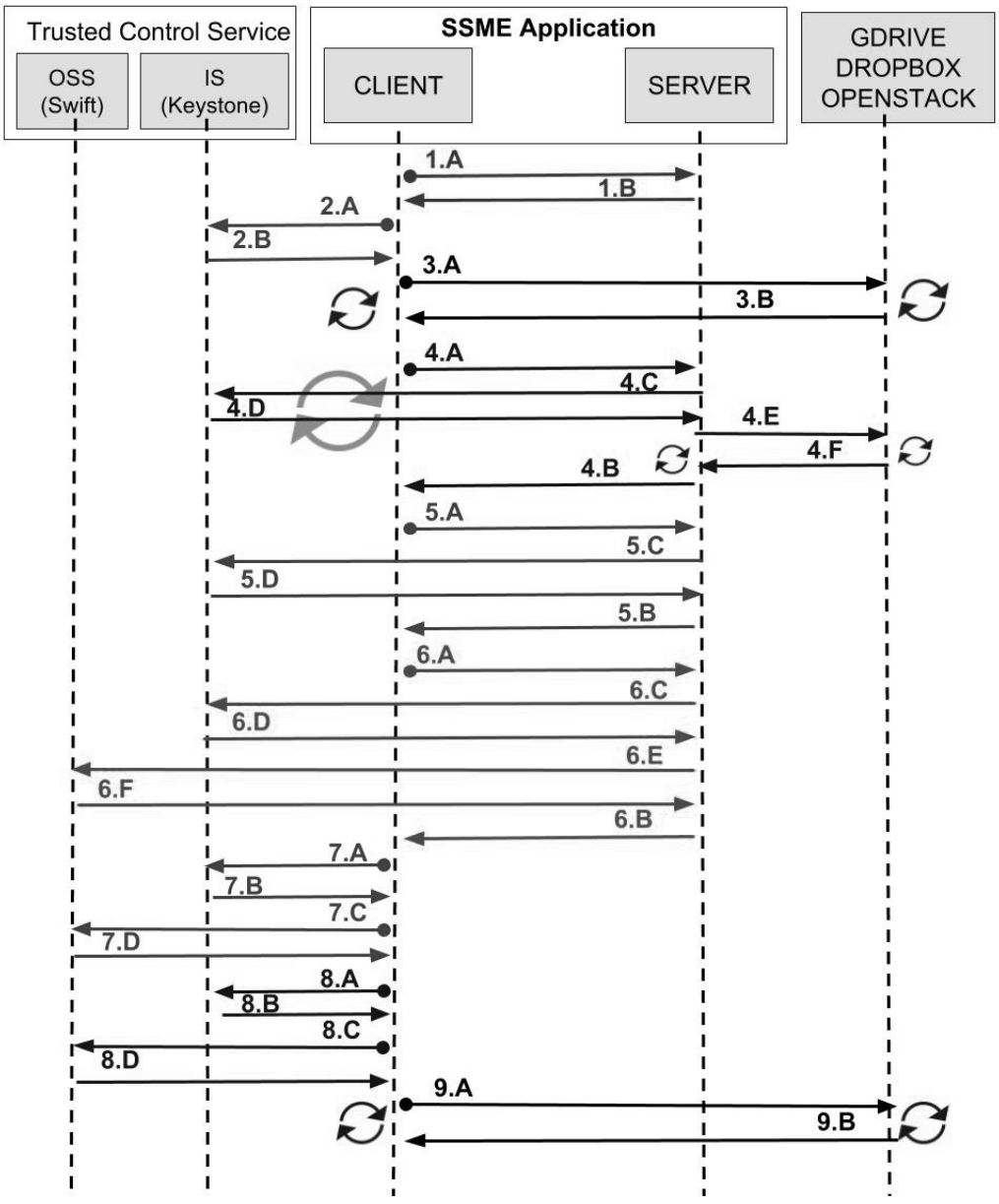


Figure 1.5: Client-server Communication Protocol in "Download Mode" [47].

The communication phase between the client and the server is done in nine different steps (see Fig. 1.5):

1. First of all, in order to verify if the server is alive, the client sends to it an HTTP request **(1.A)**. If the test is successfully done (**1.B**), then the client can move to the next step (**2.A**) otherwise it completes.

2. In order to be authenticated by the "Trusted Control Service" (TCS), the client sends an authentication request to the "Identity Service" (IS). The client sends to IS an HTTP request with a JSON file containing its credentials (**2.A**). If the IS returns the token (**2.B**) the client is successfully authenticated and it can move to the next step (**3.A**).

3. In order to test if the Cloud storage services described in the *json-encrypted-file* file are active and available, the client checks each of them. It sends an HTTP request to each of the Cloud storage services included in the list. All these requests (**3.A**) contain information about a particular Cloud storage service that the client wants to check. If the Cloud storage service returns a positive response (**3.B**), which means that the Cloud `storage-service-test` request was successfully done, the client sends another Cloud
`storage-service-test` request, and so on, until the end of the list.

4. To start the server-side processing phase, the client transfers to the server all the information about the fragments described in the *json-encrypted-file* file. The client uses one HTTP request for each fragment. All these requests contain encrypted parameters inside the headers (**4.A**). The server checks the received request by interrogating the TCS (**4.C**). If the request is successfully authenticated (**4.D**), the server uses the information contained in the request to download a fragment from the related Cloud storage service (**4.E**). If the Cloud storage service returns a positive response and a file (i.e. the fragment) (**4.F**), meaning that the Cloud storage service `download` request was successfully done, the server confirms the successful completion to the client (**4.B**). Then the client sends another `slice` request, and so on, until the end of the fragments and the client can move to the next step **(5.A)**.

5. To continue the server-side processing phase, the client sends an HTTP request to the server (**5.A**). The server checks the received request by interrogating the TCS (**5.C)**). If the request is successfully authenticated (**5.D**), the server starts the merge operation of the fragments

previously gathered server side. When the server successfully completes the merge operation, it returns a response (**5.B**).

6. To complete the recover of the file from the Cloud, the client sends an HTTP request to the server (**6.A**). The server checks the received request by interrogating the TCS (**6.C**). If the request is successfully authenticated (**6.D**), the server uploads the file just merged on the OSS (**(6.E)**). When the server completes the upload with success (**6.F**), it returns a response (**6.B**), and the client can move to the next step (**7.A**).

7. In order to download the file from the "Object Storage Service" (OSS):

   (a) firstly, the client sends an authentication request to the IS (**7.A**). If the IS returns the token **(7.B)**, the client can move to the next step (**7.C**);

   (b) then, the client sends an HTTP request to the OSS (**7.C**). If the the `download-object` request is successfully done (**7.D**), the client can move to the next step (**8.A**).

8. Once the client receives the file from the OSS:

   (a) firstly, the client sends an authentication request to the IS (**8.A**). If the IS returns the token (**8.B**), the client is successfully authenticated;

   (b) then, the client sends an HTTP request to the OSS (**8.C**). If the the `delete-object` request is successfully done (**8.D**), the server moves to the next step (**9.A**).

9. In the end, the client takes care to delete all the fragments scattered on the Cloud storage service constituting the multi-Cloud environment. The client sends an HTTP request to delete each of the fragments which are present in the *json-encrypted-file* file. All these requests (**9.A**) contain the information about a particular fragment the client wants to delete at that moment. If the Cloud storage service returns a positive response (**9.B**), the client sends another `delete-fragment` request, and so on, until the end. Then the client can finish.

## 1.6 Performance Analysis

To evaluate the system, numerous experiments were conducted by considering the implementation of the SSME application as a Service in a real scenario. The system was developed using the JAVA programming language for both client and server sides. The SSME application server was launched on a virtual machine equipped with Ubuntu Server 14.04 and hosted on an IBM BladeCenter LS21. Instead, the client machine used in our experiments is equipped with the following hardware configuration: a CPU Intel(R) Core(TM) i7-4700MQ 2.4GHz Dual-Core, 16GB of central memory, Linux Ubuntu server 14.04.5 LTS 64 bit operating system and a SATA HD with 1TB of disk storage. The middleware interacts with eight different Cloud storage services among three different Cloud storage providers: Google Drive (one instance), Dropbox (four instances) and OpenStack Swift (three instances). In the experiments, were considered different file sizes in each performed test: specifically, were used files from $10MB$ to $2GB$. Therefore, we split each of them in fragments with two different sizes: $10MB$ and $100MB$ (this latter with large files at least $500MB$).

In *Upload Mode*, performance analysis consists in evaluating: the system response time (*Overall*), the encryption time (*Encryption*) and the time due to splitting and fragment dissemination (*Split*). In *Download Mode*, performance analysis consists in to evaluate: the system response time (*Overall*), the time to receive and merge all the fragments to recompose the original file (*Merge*), and the decryption time of the file just recomposed (*Decryption*).

Each experiment was repeated 30 times and analyzed the collected data considering 95% as a confidence level.

Fig. 1.6 and 1.7 show a graphical representation of the monitored times, $500MB$ to $1GB$ as file size, in *Upload Mode*.

Fig. 1.10 and 1.11 show a graphical representation of the monitored times, $500MB$ to $1GB$ as file size, in *Download Mode*.

The results obtained with smaller file sizes were not depicted just to make more evident the graphs without introducing any further information. As can be noted the system response time linearly increases with the file size both in *Upload Mode* and *Download Mode* and independently on the fragment size used for splitting the files. Both in *Upload Mode* and *Download Mode*, the impact of encryption and decryption, respectively, turn out to be negligible with respect to the overall response time. When in *Download Mode* also the

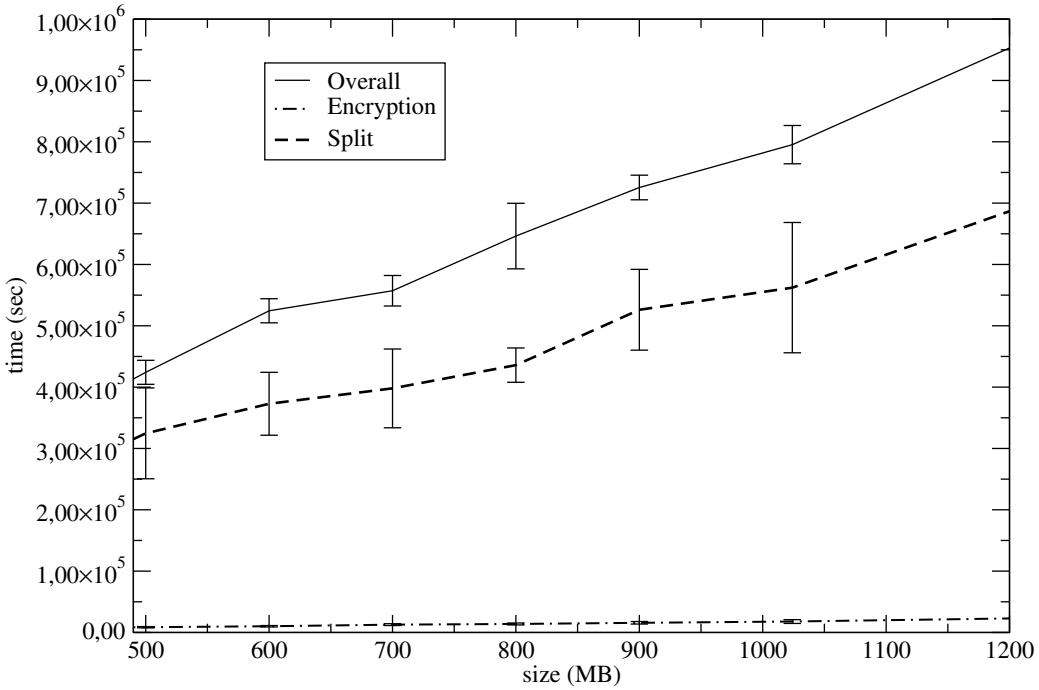


Figure 1.6: Measured times in *Upload Mode* with Fragment Size of $10MB$ [47].

merge phase does not affect the performance, instead the splitting phase has an impact on the overall time in *Upload Mode*. In particular, in *Upload Mode* (Fig. 1.6 and 1.7)

- encryption assumes a value between 1.5% and 5% of the overall time depending on the file size,
- Splitting and Fragments Dissemination is between 49% and 74% of the overall time depending on the file size.

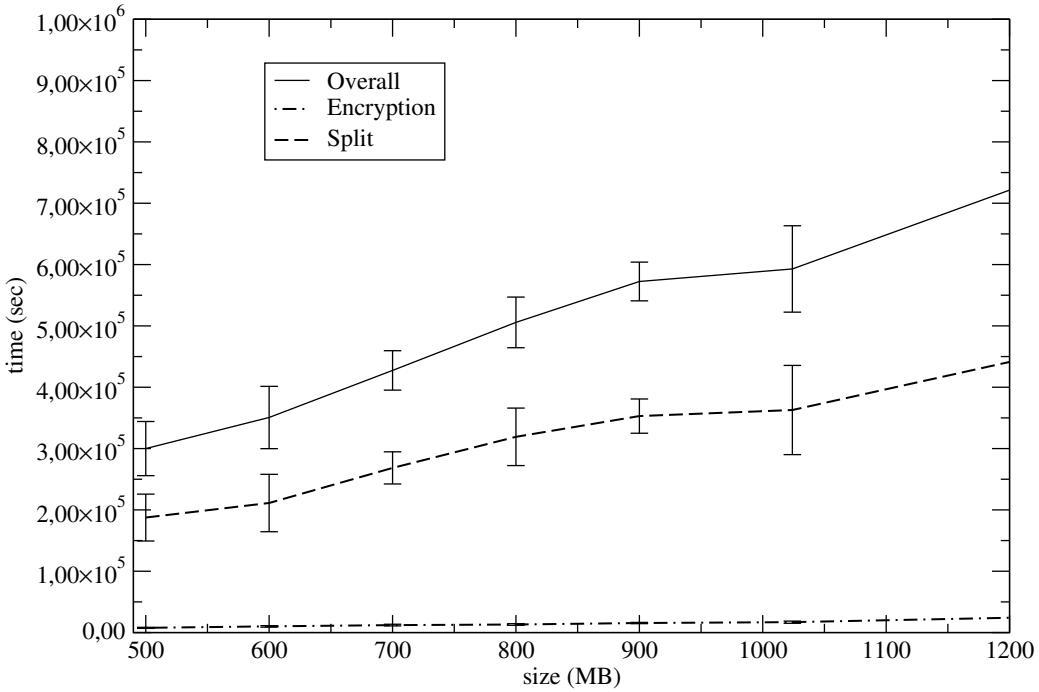


Figure 1.7: Measured times in *Upload Mode* with Fragment Size of $100MB$ [47].

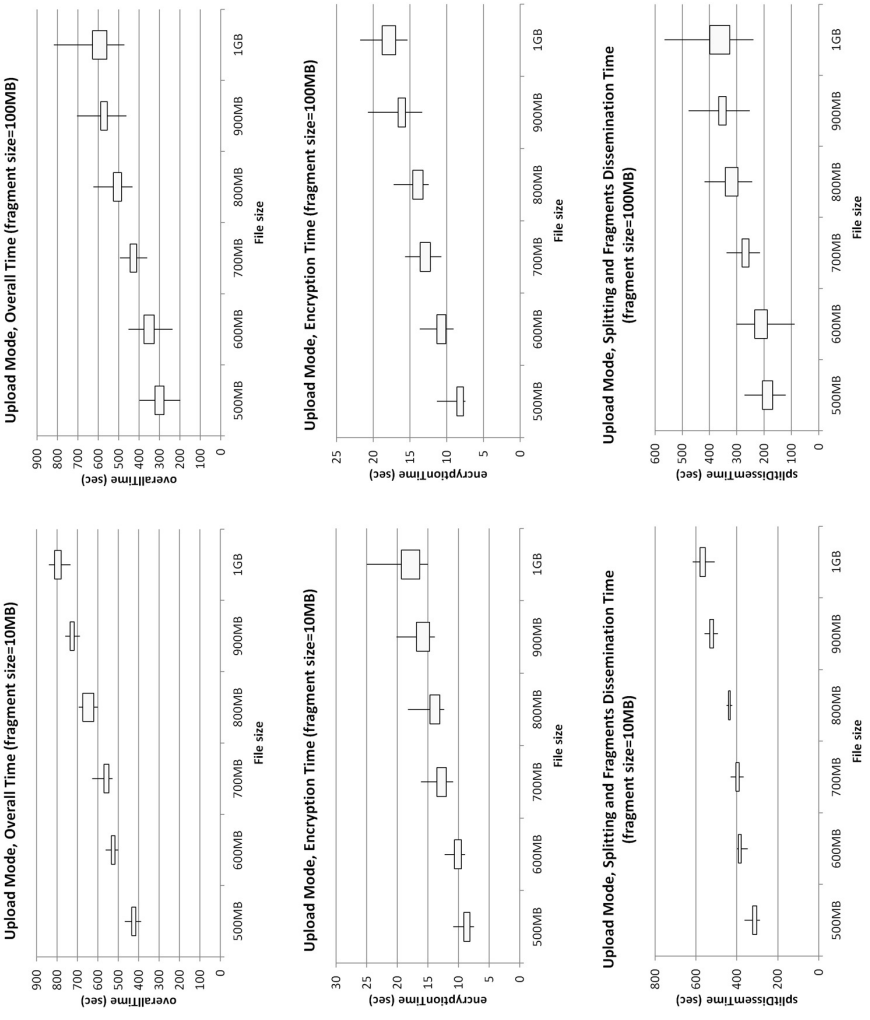


Figure 1.8: Graphical representation of monitored times in the Upload Mode [47].

In *Download Mode* (Fig. 1.9),

- Receive and Merging affects the Overall Time between 2.7% and 11% depending on the original file size.
- Decryption affects the Overall Time between 3.8% and 8% depending on the original file size.

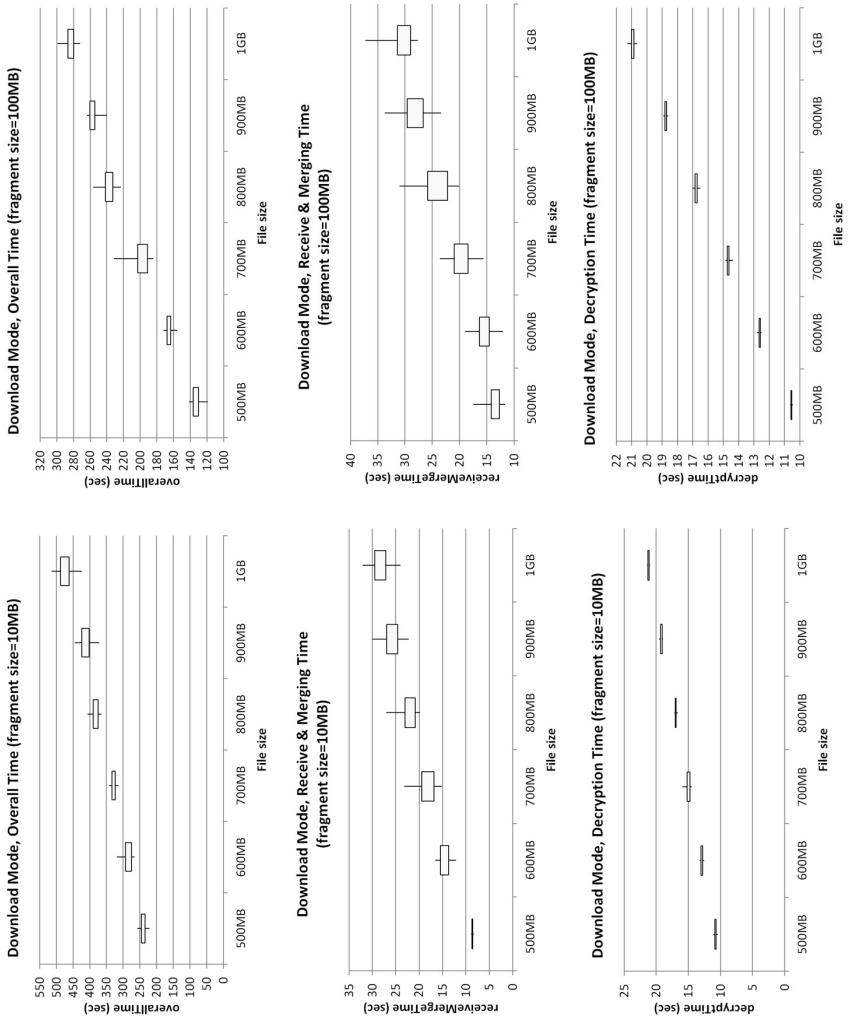


Figure 1.9: Graphical representation of monitored times in the Download Mode [47].

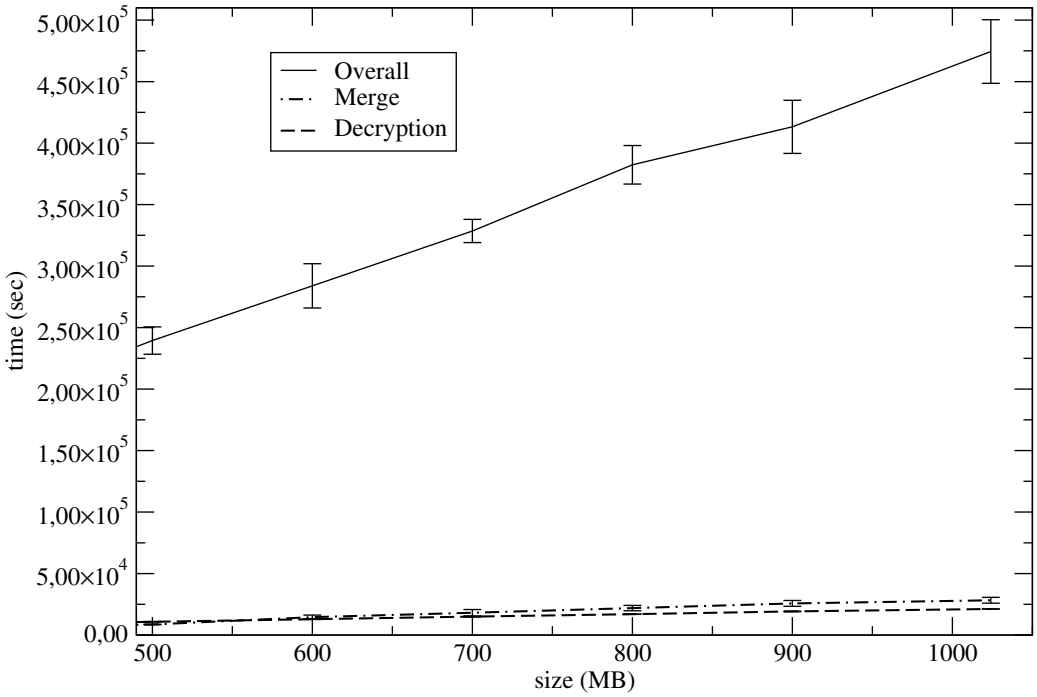


Figure 1.10: Measured times in *Download Mode* with Fragment Size of $10MB$ [47].

In both cases, the results show that Overall Time improves in performance when greater fragment sizes are used. For example, in *Upload Mode*, this trend is confirmed for $800MB$ file size: the average value is 646 *sec* for $10MB$ fragment sizes and 505 *sec* for $100MB$ fragment sizes. *Download Mode* had a similar trend using the same file: the average response time is 382 *sec* when $10MB$ fragment size is used and 237 *sec* with $100MB$.

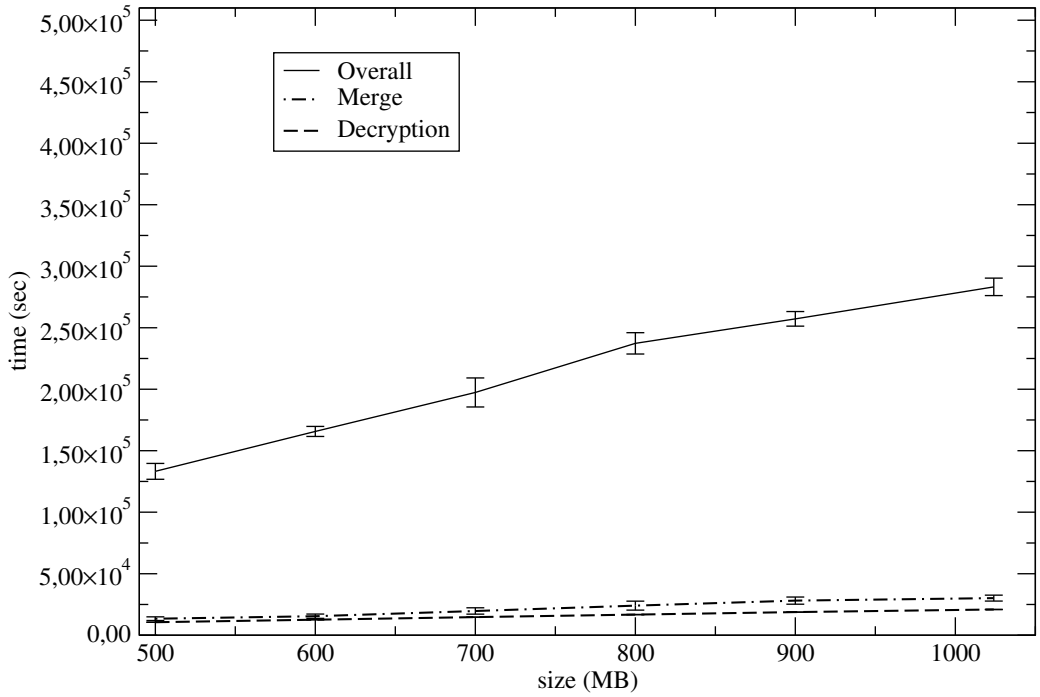


Figure 1.11: Measured times in *Download Mode* with Fragment Size of $100MB$ [47].

# Chapter 2

# An Approach to Enhancing Confidentiality and Integrity on Mobile Multi-Cloud Systems: The "ARIANNA" Experience

## 2.1 Brief Introduction to the Problem

Cloud computing has emerged as a new paradigm over the past decade, but today *"Cloud is the new normal"* [20]. Recent technological trends show that a Cloud today may not be enough. *Hybrid solutions* and *multi-Cloud* approaches are gaining high popularity in corporate Cloud computing strategies. The reasons for this success are easy to understand. If an enterprise wants to compete in today's digital economy, it will not be able to do it without having a more flexible, agile, and scalable IT infrastructure without vendor lock-in. These features make the multi-Cloud suitable for use in a wide range of sectors, applications, and industries. Therefore, multi-Cloud is becoming the new Cloud.

The spread of mobile devices is changing our daily lives. Mobile devices allow users to access their data when and where they want. Compared to the past, currently, the market offers users a considerable amount of low-cost, high-performance devices. However, as the quality (e.g., audio and video quality of multimedia contents) and frequency of our activities (e.g., social networks, remote work, etc.) are increasing, users cannot fully benefit

from this high-performance. In such a context, due to the inherent characteristics of mobile devices (in particular constrains in terms of battery size, processing, and storage capacity), the limitations persist, although in different proportions. From these premises, it seems clear how natural the process of integration between mobile devices and Cloud computing is.

From the union between Cloud technologies and mobile devices, the new concept of *"Mobile Cloud Computing (MCC)"* has emerged. The MCC offers core functionalities from Cloud technologies (pooling resources, elasticity, on-demand, pay-per-use, flexibility, scalability, and ubiquitous access) by adding portability and more convenient use of mobile devices in highly dynamic contexts. MCC allows users to collect and integrate data from various sources quickly, regardless of where it resides, inheriting existing security issues in terms of data *confidentiality* and *integrity*.

This Chapter presents **"ARIANNA**, an Android app that can be used as software *enabler* to extend the *"SSME Cloud Service"* [47] towards the mobile world.

*SSME Cloud Service* is an experimental multi-Cloud system implemented as a service at the *Cloud Data Center - University of Messina* [98].

ARIANNA app implements an approach primarily oriented to guarantee *confidentiality* and *integrity* issues concerning data stored on mobile smart devices that are part of a multi-Cloud environment.

The rest of this Chapter is organized as follows. Section 2.2 discusses the motivation of the study. Section 2.3 gives a brief overview of existing approaches and applications about mobile multi-Cloud systems. Section 2.4 describes the ARIANNA scenario. Section 2.6 intends to evaluate the results of the performance analysis.

## 2.2 The Goal of the Study

Nowadays, the Cloud is no longer the *"new normal"*, but it is more than a standard solution for the Enterprise world. From Banking & Financial services to Industrial & Manufacturing sectors, every big industry in the world has started migrating to the Cloud. Now, those industries are consolidating strategies to build and manage their services and products in the Cloud, but no one talks about security concerns and issues anymore.

According to IBM [85], insiders were responsible for 60% of attacks against U.S. companies surveyed in 2015, compared with 55% in 2014. The

23.5% were inadvertent actors. An insider threat may be an attacker who consciously or unconsciously extrapolates data, sabotages company IT systems, or manipulates its data and systems. In a 2016 study, IBM and the Ponemon Institute calculated that on a global average, the cost of a data breach totals USD 4 million. Such a loss can cause irreparable damage to an organization's brand and shakes the trust of its customers. Unfortunately, all the new Cloud approaches (e.g., multi-Cloud, Hybrid Cloud, and mobile multi-Cloud) suffer in some way of the lack of data protection mechanism against direct malicious access at the system level. Moreover, the market needs solutions that guarantee data intelligibility to unauthorized users, even when they are insiders (e.g., system administrators).

The study extends the experimental multi-Cloud secure storage approach towards the mobile world. To do this, ARIANNA app integrates the *SSME Cloud Service* [47].

Preliminary results of experimentations that use the ARIANNA app are already available in the literature [42].

*SSME Cloud Service* consists of an experimental Cloud computing system that aims to enhance the integrity and confidentiality of the data stored on multi-Cloud environments by combining, in a smart way, the concepts of data fragmentation, symmetric (AES256) and asymmetric (RSA) encryption. Moreover, *SSME Cloud Service* offers user-friendly and dynamic management of both fragmentation schema and the pool of Cloud storage services available to be used during the Cloud storage operations. In this way, contrary to what happens in other solutions described in the literature, users can decide how to customize in detail their own multi-Cloud environments time after time. The name ARIANNA is related to the Greek mythology, in particular to the myth of the labyrinth presented in *"The Ariadne's Thread"*. ARIANNA is the Italian name for *Ariadne*, the daughter of Minos, King of Crete [81]. Ariadne was the woman who gave Theseus the ball of red thread, which allowed him to find his way back out once he penetrated the labyrinth [80]. Following the idea of the myth, if we consider a multi-Cloud environment as a sort of labyrinth made up of physical and virtual server machines scattered all over the world, it is evident that the first need is to create a mechanism that allows easy extraction of elements from this labyrinth. For this reason, we created this mobile app that allows users to easily manage their files (in our scenario fragments of data) on dynamic multi-Cloud environments without worrying about their real location and how to recover

them. The idea behind it is straightforward. Both the name and the principles draw inspiration from the red thread of the myth. As will be discussed later, the *Upload phase* of ARIANNA gives back a JSON encrypted file for every single file the user moves from the mobile device to his multi-Cloud environment. Each JSON encrypted file can be stored on the mobile device itself (in fact, the app also acts as a collector), or more generally, wherever the user feels appropriate in terms of the trust. The critical aspect of being emphasized is that each of these JSON encrypted files represents the only *guiding light* to retrieve that particular file from the *labyrinth* represented by the multi-Cloud (precisely as the red thread of the myth).

Many works in literature deal with mobile Cloud storage, multi-Cloud storage, and mobile multi-Cloud storage systems. The next Section presents the background and how the already existing approaches and solutions differ from the ARIANNA app.

## 2.3 Background of the Study

In [19], the authors present an architecture for a cryptographic storage service. It consists of four components: a server that processes and encrypts (AES256) data before sending them to Cloud, a private Cloud that holds the meta-data information, and two Clouds that archive half of each file, respectively. The authors assume that the remote server is trusted without specifying any information on "how" this trustiness is implemented. The meta-data information (e.g., passwords, secret keys of each file, encrypted access paths) are securely stored in the private Cloud. If compared with our dynamic approach that allows specifying the size of the split fragments, here data splitting is statically fixed on half of each user's file.

In [117], the authors propose and implement a mobile Cloud storage system that authenticates users using a gesture-based password, and stores private information securely in the Cloud using the Amazon Simple Secure Storage Service (Amazon S3). Their solution makes use of the AES and SHA-1 algorithms to protect data, while also making use of SSL to transfer data between the mobile device and the Amazon S3 buckets. Their solution ensures a high level of security for the user's personal and private information. Still, with respect to our approach, we use a fragmentation mechanism for the files we want to store in the Cloud, and we allow the users to use a pool of Cloud storage services that users can customize from time to time.

The TwinCloud client-side encryption solution presented in [29] focuses on the secure sharing on Clouds without explicit key management. To this end, they highlight the Public Key Infrastructure (PKI)-based solution problems, i.e., costs due to get a certificate from a Certification Authority (CA) and PKI-infrastructure maintenance. Differently from the TwinCloud solution, our architecture uses both symmetric and asymmetric cryptography.

In [16], the authors present a mobile Cloud middleware, which aims to provide mobile clients with a flexible storage area and data integrity service. They demonstrate that the middleware can serve a large number of simultaneous users. In their architecture, they introduce a "Security/Storage Module," but in the explanation, they talk about a *trusted third party* without specifying any information on "how" this trustiness is implemented. Moreover, if referred to our dynamic approach, they use a static method both for the management of the fragmentation schema and the pool of Cloud storage services.

In [137], the authors propose a collaborative storage algorithm called MECCAS that can adaptively allocate resources at individual nodes. They state it can satisfy the mobile edge cloud technology features (i.e., variable characteristic information, weak computation, and dynamic nodes). They describe how MECCAS can minimize some aspects (i.e., delay of tasks execution, power usage effectiveness, risk of nodes withdrawal), meanwhile ensuring reliability and full integration of local heterogeneous information. However, this approach does not take into account confidentiality and integrity issues concerning information and data.

In [140], the authors propose and implement an authentication mechanism based on homomorphic encryption to secure access for mobile users to the remote multi-Cloud servers. Their approach guarantee data confidentiality and integrity. If compared to our approach, we use both symmetric and asymmetric cryptography, and we offer a configurable and flexible schema both for the management of the fragmentation schema and the pool of Cloud storage services.

All the previous contributions describes architectures, approaches or frameworks that include encryption phases do not give specific information on how the keys are managed, at least it is not described where the key is stored and how and who can access them. All the multi-Cloud environment descriptors are static and not dynamically configurable by the user. Furthermore, all processing nodes (e.g., servers) which represent the virtual access point to

the various multi-Cloud environment not work only in volatile memory, that means without saving data on the disk during the processing. The processed data at this point could not be protected against insider attacks [76].

## 2.4 ARIANNA Scenario

Figure 2.1 shows a high-level outline of ARIANNA, mainly identifying the app, the SSME middleware, and the multi-Cloud environment is communicating via HTTP. For more details, readers can refer to [47].

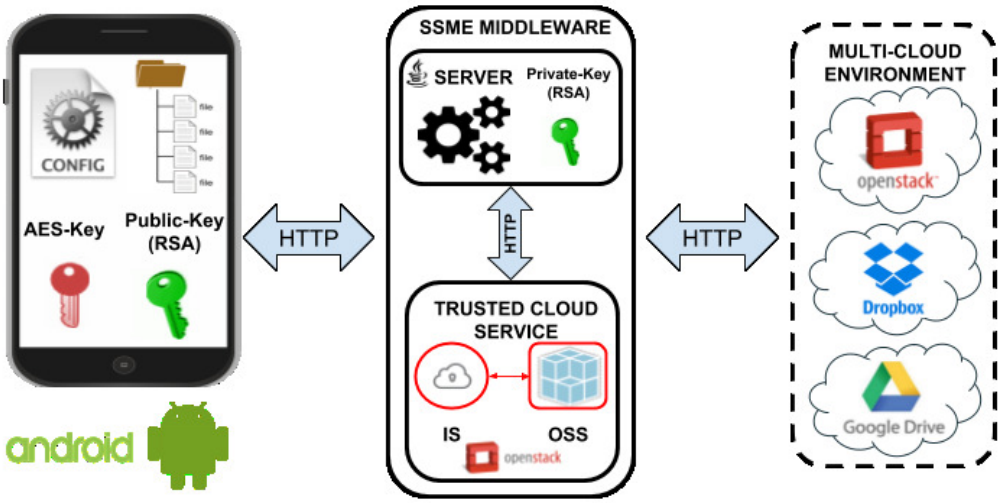


Figure 2.1: High-level outline of ARIANNA [43].

### 2.4.1 Multi-Cloud environment

A multi-Cloud environment is a heterogeneous set of Cloud storage services that are dynamically specified by the user at the application level and are automatically managed without any control by the SSME server. Nowadays, as well as, of course, a lot of paid Cloud storage services, it is possible to find hundreds of Cloud storage services that have free accounts. These free accounts usually come with some limitations, such as the amount of storage they provide or the size limit on files users can upload. Assuming the third party Cloud storage services are "available" and "reliable" from all points of view, deciding what the best Cloud storage service solution

available is is not easy. This because, in this case, the *quality* of service is not strictly related to its *quantity*. For the experimental purposes, until this time, this study approaches some commercial solutions of Cloud storage services that offer some level of free service. Specifically, GoogleDrive [66] and Dropbox [51] (even if it offers only 2GB of storage space available) are used in this scenario. In addition, the choice also fell on the OpenStack Cloud storage solution (Swift [106]). This choice because besides being open source, OpenStack represents the "de facto" standard solution for deploying private/public Cloud services.

### 2.4.2 Secure Storage Cloud Services: Security Threats and Requirements

Security threats in Cloud storage services can be categorized in different ways.

- *Threats from external attackers*, that come from outside the organization and which can be further subdivided into many other kinds (i.e., Human, Physical, Legal, Software, Network).

- *Threats from insiders* that can take many forms and can be categorized as either malicious or accidental. These threats are usually coming from all the people who have had a past or present relationship with the organization [96].

According to [124], threats present four significant security requirements:

- *data confidentiality* refers to the property with which data is not made available to unauthorized users. In this way, only authorized users can access the data.

- *data integrity* refers to the certainty that the data stored on Cloud cannot be modified, damaged, or deleted by anyone, voluntarily or involuntarily.

- *data access controllability* refers to the opportunity offered to the user to be able to perform a selective restriction of access to his data on Cloud.

- *privacy protection* refers to the property that users' access behaviors and habits are not be traced back by any other actor in the Cloud. Moreover, it refers to the opportunity to hide the user identity while he uses the Cloud service.

### 2.4.3 SSME Cloud system

The SSME Cloud system consists of a middleware composed of two main components: the **SSME server**, where all the processing takes place, and a **"Trusted Control Service" (TCS)**, that consists of the architectural component containing trusted Cloud services integrated into the middleware itself.

**SSME Server** The SSME server consists of a JAVA RESTful web server built using the RESTful architectural style. All the requests received from ARIANNA are managed transparently by the SSME server. The SSME server works in two modes: *Upload phase* and *Download phase*. During the *Upload phase*, the server retrieves from the OSS the encrypted file sent by the user. After the fragmentation step, the server takes care of randomly allocating each fragment to a specific Cloud storage service, which is part of the pool specified by the user. At the end of this dissemination step, the server deletes the original encrypted file sent by the user, which was temporarily stored on the OSS. The output of the *Upload phase* is the JSON encrypted file called *json-encrypted-file* file. The *json-encrypted-file* file represents the only *guiding light* to retrieve the related file from the labyrinth of the multi-Cloud. It's important to highlight that all the server-side operations are tracked nowhere on the middleware system, neither on the physical disk of the server host.

**Trusted Cloud Service** The TCS consists of two selectable Cloud services:

- the **"Identity Service" (IS)** that integrates a trusted token service mechanism which provides authentication on Cloud that is mandatory to process all the HTTP request received by the SSME server;

- the **"Object Storage Service" (OSS)** that integrates a trusted object storage service providing temporary backup-storage that is manda-

tory to run some communication steps during the *Upload phase* and the *Download phase*.

In this scenario is used an OpenStack compliant TCS. It means that the service used as IS and OSS are *"Keystone"* (the OpenStack Identity Service [105]) and *"Swift"* (the OpenStack Object Store Service [106]), respectively.

**How ARIANNA enhances *Confidentiality* and *Integrity*** The communication scheme adopted in the SSME Cloud system is detailed "step by step" in [47]. It meets the requirements of authenticity, confidentiality and integrity. In particular, *authentication* is performed using the token service identified by the IS. Moreover, each HTTP request sent by ARIANNA to the SSME server embeds some HTTP Headers, which contain data and directives related to the server-side tasks the server needs to set up. The information inside the HTTP Headers are encrypted by using the AES256 algorithm. The key to decrypt HTTP Headers is also embedded in each HTTP request itself, inside a field called *"key"*. However, this last field is encrypted by using the RSA asymmetric algorithm by using the SSME server's public key (this key has a length of 2048 bytes). Doing so, only the SSME server will be able to decrypt the *"key"* HTTP Header and therefore decrypt the other HTTP Headers AES256-encrypted. The AES256 encrypted key that is sent to the server in every HTTP request must not be confused with that is used by the user to encrypt on the device his file before to put it on the Cloud. These are different keys. This latter is maintained in the app configuration file. *Confidentiality* and *Integrity* derive from a smart use both symmetric and asymmetric techniques of encryptions that are applied for all HTTP requests exchange. This makes the scheme proposed by ARIANNA immune to all types of *"man in the middle"* attacks. Taking into account the requirements introduced in subsection 2.4.2, *data confidentiality* and *data integrity* are perfectly enhanced because of using cryptography unauthorized users will not be able to access the files, including internal staff at the CSPs. The *data access controllability* is not enhanced because the system doesn't allow the users to perform policies of selective restriction of access to the data. While the *privacy protection* is not enhanced because if on the one hand, the system trace nowhere in the middleware any kind of information about the processing, on the other hand, the system must identify the user that requires to use the service. Fig. 2.2 summarizes the security requirements enhanced by the ARIANNA app.

| Threats from Insiders | ARIANNA |
|---|---|
| data confidentiality | ✓ |
| data integrity | ✓ |
| data access controllability | ✗ |
| privacy preservability | ✗ |

Figure 2.2: Security requirements enhanced by ARIANNA app [43].

## 2.5 ARIANNA Application

This Section describes how the user-level application works, as well as introducing the features of the mobile app.

### 2.5.1 Configurations Management: "Configuration Phase"

The configuration phase essentially allows users to start ARIANNA for the first time to set up the preferences (i.e., the parameters chosen for the multi-Cloud environment) by a graphical user interface. The result is that the ARIANNA app automatically generates a JSON configuration file at the end of this phase. Once approved the configuration parameters, the system is available to be used based on the setup.

### 2.5.2 File Management: "Upload Phase"

To upload a file from the mobile device to the multi-Cloud environment previously set, the user has to tap on the "upload" button (Fig. 2.3). The application loads a screen wherein the user has to select the file to upload from the filesystem of the used mobile device. Therefore, the application presents a screen containing a non-mandatory option used to modify the dimension of the splitting unit (Fig. 2.3). Once the "upload phase" has been started, the communication with the server begins. The file is considered as "successfully uploaded" to the multi-Cloud environment when ARIANNA receives back the JSON encrypted file (*json-encrypted-file* file) from the SSME server. This file serves as a modern Ariadne's thread. It represents the only way to recover and rebuild the fragments stored in the labyrinth represented by the Cloud. The template structure of the *json-encrypted-file* file is described

in [47]. When the upload confirmation is received, the local copy of the file is deleted from the device, and the user will be brought back to the main screen. The main screen allows users to select another file to be uploaded in the same multi-Cloud environment. The user could decide to maintain the same configurations, even in terms of fragment size and encryption keys (AES256, RSA2048). It is important to emphasize that the user can decide time by time to change these configurations. As already said, this flexibility provides added value, which is not present in other proposals in the literature. Moreover, by adopting a dynamic combination of configurations for storing their files, specifically in terms of encryption keys, fragment size, and different Cloud storage services, the user surely will increase the protection level of his data in terms of *confidentiality*. All the single *json-encrypted-file* files generated by the different upload phases are stored in an internal directory managed by the application. After the first uploading, it is now visible a new button on the main screen of the ARIANNA (Fig. 2.4). By tapping on this new button, it is now possible to browse the list of these files, but for trivial security reasons, ARIANNA does not allow to open or modify them. These files are AES256-encrypted, and their content represents the only way to pull out the original data from the related multi-Cloud environment. The user will take care to handle these files correctly, for example, creating backups and moving them to his own trusted storage Cloud service.

### 2.5.3 File Management: "Download Phase"

To download a file from a multi-Cloud environment to the mobile device, the user has to tap on the "download" button (Fig. 2.3). ARIANNA shows a screen wherein the user has to select the file he wants to move from the multi-Cloud environment to the filesystem of the mobile device. The user can choose among the collection made available by the application itself. This collection consists of the list (Fig. 2.4) of the *json-encrypted-file* files stored on the mobile device by ARIANNA. After the choice, ARIANNA decrypts the *json-encrypted-file* file selected and starts a test phase related to the Cloud storage services needed to gather all the fragments specified inside the *json-encrypted-file* file. After this test phase, ARIANNA starts the "download phase," and the communication with the SSME server begins. The file is considered as "successfully downloaded" from the multi-Cloud environment when ARIANNA downloads the original encrypted file from the

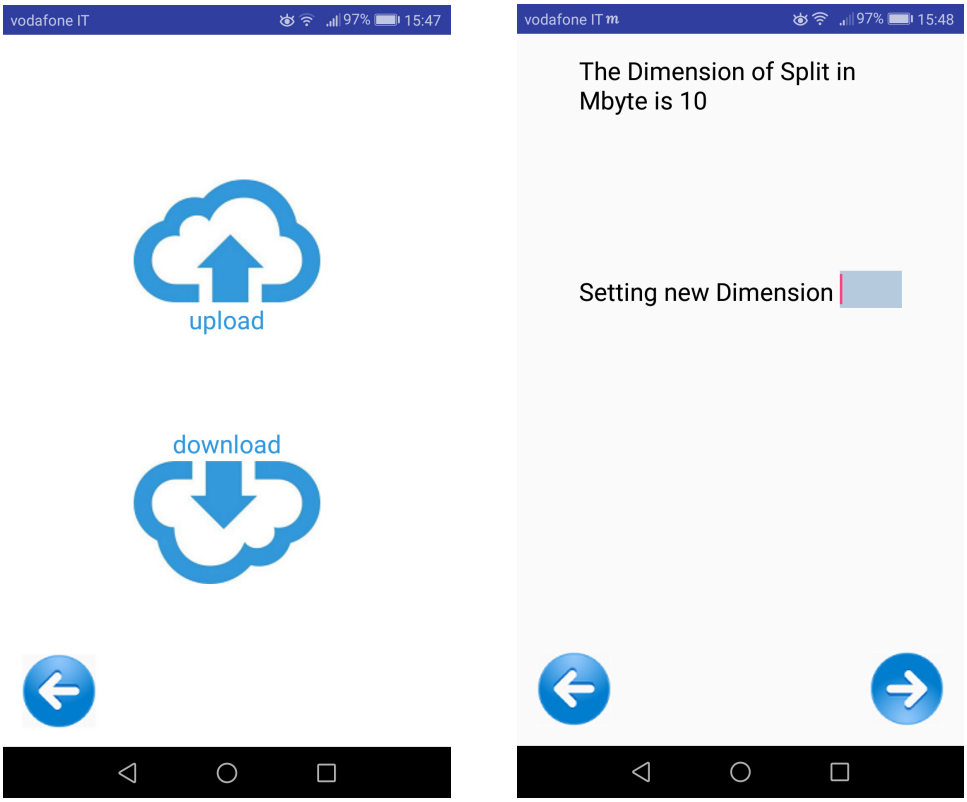


Figure 2.3: ARIANNA: selection tab [43].

OSS that was just merged by the SSME server and then moved there. Once the file is successfully downloaded from the multi-Cloud environment that is specified inside the *json-encrypted-file* file, the SSME server deletes the fragments disseminated on the Cloud storage services that were part of that particular multi-Cloud environment. As a consequence, no longer having any usefulness, even the *json-encrypted-file* related to the just downloaded file is deleted. In the end, the original file is then decrypted, and the user will be brought back to the main screen. The original file will now be available in the "Download" directory created by the ARIANNA application.

## 2.6 Performance Analysis

To evaluate the approach enabled by the ARIANNA mobile application, we conducted several experiments measuring the performance of ARIANNA integrated with the *SSME Cloud Service* [47] in terms of *response time*. In

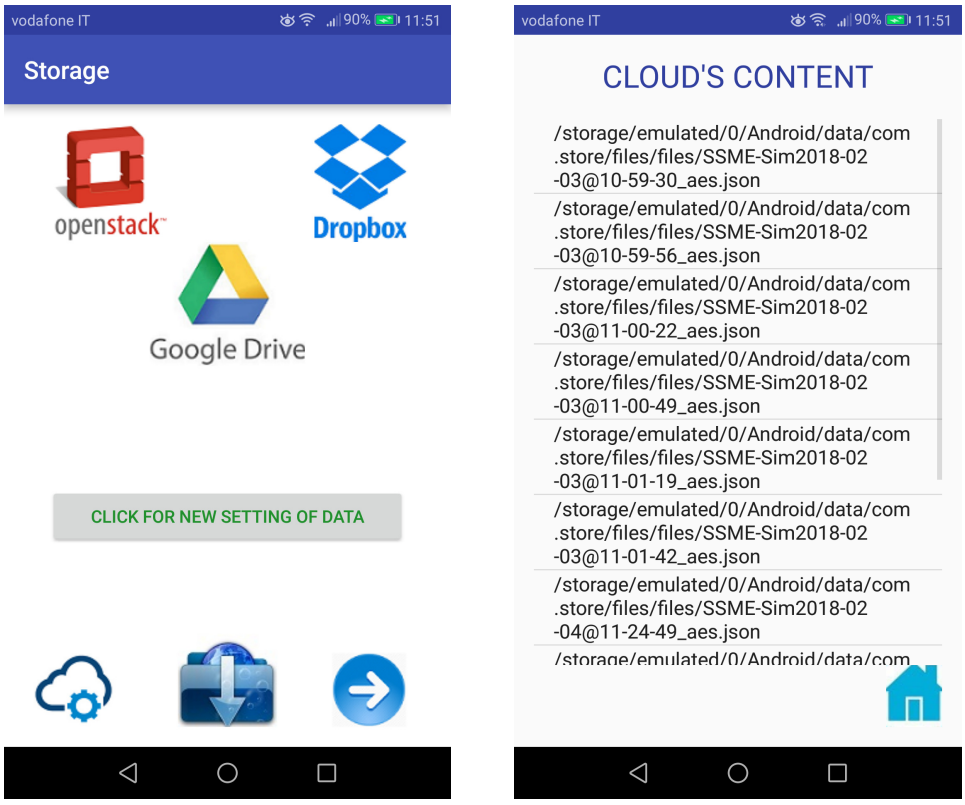


Figure 2.4: ARIANNA: main screen after the "Upload Phase" [43].

such a context, the response time can be defined as the time the presented solution takes to react to a given request (i.e., to execute the request task by the user successfully). The *SSME Cloud Service* was deployed on a virtual machine equipped with Ubuntu Server 14.04 and hosted on an IBM BladeCenter LS21.

As device deputed to run the ARIANNA app, the choice fell on a *LG Nexus 5* Android smartphone [72]. The smartphone was equipped with a CPU Qualcomm Snapdragon 800 2.30 GHz (4 core), a chipset Qualcomm MSM8974 Snapdragon 800, 2 GB of central memory, Android OS 4.4 KitKat and 16 GB of disk space. The pool of Cloud storage services we used for the dynamic composition of our multi-Cloud environments was composed of eight different Cloud storage services among three various providers:

- Google Drive (one service);
- Dropbox (four services);
- OpenStack Swift (three services).

To investigate how the performances of ARIANNA are related to the speed of the mobile network connection in use and the file size, we chose to use different combinations of these factors. Specifically, we used a set of files whose size ranged from 10 $MB$ to 100 $MB$, and we split each of them in fragments with a static size of 10 $MB$. Moreover, since the upload and download speeds may greatly vary depending on network speed, we chose to analyze the performance of ARIANNA by using two different network connections, "*ADSL*" and "*4G*", respectively.

We analyzed the ARIANNA behavior into both the (*Upload phase* and *Download phase*), and we evidenced some of the most relevant sub-phases. In *Upload phase*, we measured the overall system response time (*Overall*), the encryption time (*Encryption*) and the time due to splitting and fragment dissemination (*Splitting and Fragments Dissemination*). In *Download phase*, we evaluated:

- the overall system response time (*Overall*);
- the time to receive and merge all the fragments to recompose the original file (*Receive and Merging*);
- the decryption time of the file just recomposed (*Decryption*).

We repeated each experiment 30 times and analyzed the collected data considering 95% as confidence level.

### 2.6.1 Performance of ARIANNA

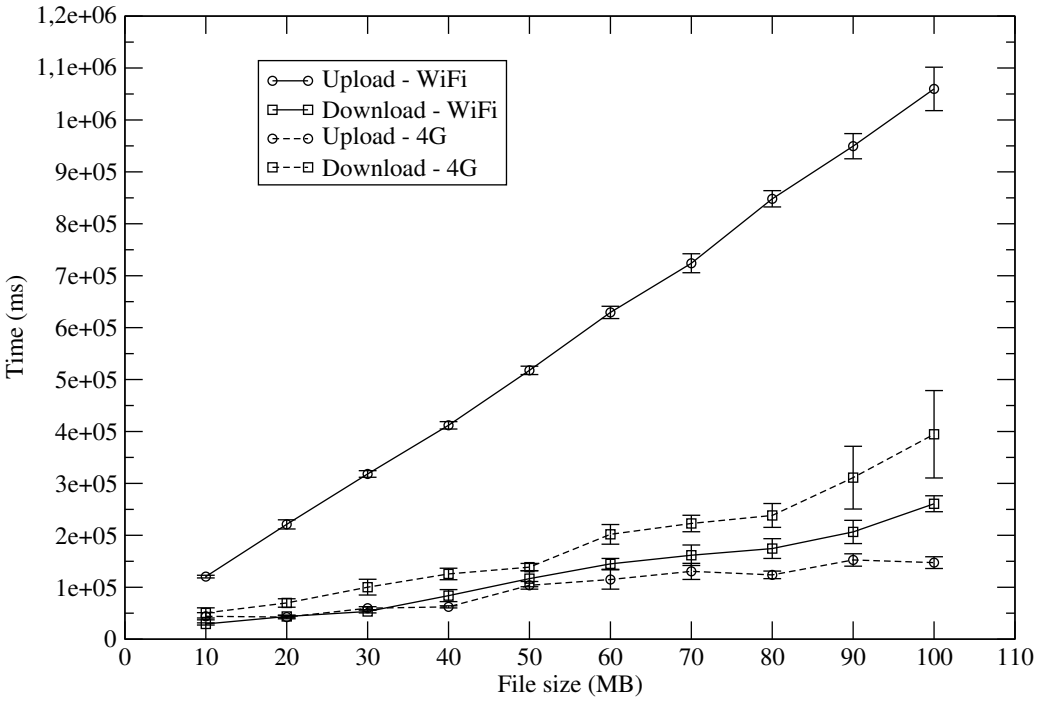


Figure 2.5: ARIANNA response times [43].

Fig. 3.1 shows the ARIANNA response time related to the *Upload phase* and *Download phase*, from 10 $MB$ to 100 $MB$, under the two different network mobile environments.

#### "*ADSL*" case

Experiments in this test-bed were characterized by a WiFi access point where the mobile device was connected; this latter also operates as a bridge to the Internet by using an ADSL connection. Thus the available bandwidth was 384 *Kbps* in upload mode and 7 *Mbps* in download mode. As can be noted in Fig. 3.1, the system response time linearly increases with the file size both in *Upload phase* and *Download phase*. Measurements show that in *Upload phase* and *Download phase*, the impact of the encryption and decryption,

respectively, turn out to be negligible with respect to the overall response time. When in *Download phase*, the merge sub-phase does not affect the performance, instead the splitting sub-phase has an impact on the overall time in *Upload phase*. In particular, in *Upload phase*, Encryption assumes a value between 5.15% and 1.38% of the Overall time depending on the file size, Splitting and Fragments Dissemination is between 8.30% and 2.31% of the Overall time, depending on the file size. In *Download phase*, Receive and Merging affects the Overall time between 2.71% and 0.56% depending on the file size, Decryption affects the Overall time between 21.14% and 6.64% depending on the original file size.

**"*4G*" case**

The second set of tests have been done by providing the mobile device with a *4G* connection given by the service provider; its available bandwidth was 50 *Mbps* in upload mode and 150 *Mbps* in download mode. Also in this case (Fig. 3.1), the system response time linearly increases with the file size both in *Upload phase* and *Download phase*. Measurements show that in *Upload phase* and *Download phase*, the impact of the encryption and decryption, respectively, turn out to be negligible concerning the overall response time. When in *Download phase*, the merge sub-phase does not affect performance, instead the splitting sub-phase has an impact on the overall time in *Upload phase*. In particular, in *Upload phase*, Encryption assumes a value between 15.17% and 8.28% of the Overall time depending on the file size, Splitting and Fragments Dissemination is between 16.31% and 3.22% of the Overall time, depending on the file size. In *Download phase*, Receive and Merging affects the Overall time 1.74% and 0.56% depending on the file size, Decryption affects the Overall time between 11.35% and 4.80% depending on the original file size.

# Chapter 3

# How much enhancing Confidentiality and Integrity on data can affect Mobile Multi-Cloud: The "ARIANNA" Experience

## 3.1 Brief Introduction to the Problem

ARIANNA is an Android app compliant and integrated into the *SSME Cloud Service* [47], a novel experimental framework concerning secure storage service in a multi-Cloud environment deployed and maintained at the *Cloud Data Center - University of Messina* [98]. *SSME Cloud Service* [47] proposed a secure storage approach primarily oriented to guarantee *confidentiality* and *integrity* issues concerning data stored in a multi-Cloud environment.

*SSME Cloud Service* [47] implements a multi-Cloud approach, also called *"ARIANNA approach"*, which consists of an architectural schema characterized by some distinctive features that can be summarized as follows:

- encryption at client-side;
- a worldwide distributed middleware for data splitting, dissemination, and retrieval;

- a data protection mechanism which combines both symmetric (AES256) and asymmetric encryption (RSA);
- client-side configuration invisible on the server-side;
- tracking of nothing on the system, nor on the physical disk of servers;
- dynamic management of both the fragmentation schema of the files and the pool of single Cloud storage services for creating a dynamic multi-Cloud environment whenever the user wants to store its file.

As introduced in [42], and deeply explained later in [43], ARIANNA app represents the software *enabler* which allows to extend the *SSME Cloud Service* [47] to the mobile world. Moreover, [43] demonstrated how the *"ARIANNA approach"* addresses the *confidentiality*, the *integrity* and partially the *privacy protection* issues described in [124] and concerning data stored in the mobile devices that are distributed in worldwide Cloud environments [96].

## 3.2 The Goal of the Study

The aim of this study is not to present the ARIANNA approach, not the improved security requirements enabled by the ARIANNA app, nor any architectural consideration about the Cloud infrastructure behind [47], nor the background of existing approaches and how they differ from ours. In this regard, readers can find interesting reading Chapters 1 and 2.

This study aims to provide an overview of the costs of using the ARIANNA approach. Chapter 2 discussed the *benefits* of this approach in terms of the security requirements enhanced on data. This study focuses to the *overhead* added to the *response time* of the system. In the context of computer technology, the *response time* consists of the elapsed time between an inquiry on a system and the response to that inquiry. The *response time* values to consider are the times of upload and download of a file to and from a Cloud system (single Cloud or multi-Cloud) that is perceived by the user. To measure the worth of the *overhead* introduced by using the ARIANNA approach, this study presents a quantitative performance analysis comparing some commercial Cloud storage services with the ARIANNA app. The study measures *"how much"* is useful using the multi-Cloud solution enabled by the ARIANNA app rather than other single storage services. Of

course, the choice of Cloud storage services to compare with ARIANNA was not random. The study considered only Cloud storage services which were used for the composition of the dynamic multi-Cloud environments of the testbed used [2] ( [43]). These services are *Google Drive* [66], *Dropbox* [51] and *OpenStack Swift* [106]. These services consist of well-known and highly appreciated commercial solutions that offer a certain level of free service without providing benefits in terms of confidentiality and data integrity to their users.

## 3.3 Performance Analysis

This Section evaluates *"how much"* the overhead introduced by the ARIANNA approach costs. The experiments refer to the use of the ARIANNA mobile app in real and dynamic multi-Cloud scenarios.

### 3.3.1 Testbed Description and definition of the tests

The *SSME Cloud Service* [47] used in the simulation was deployed on a VM equipped with Ubuntu Server 14.04 and hosted on an IBM BladeCenter LS21. The ARIANNA app used in the simulation was installed on a *LG Nexus 5* Android smartphone [72] characterized by the following technical specifications:

- CPU Qualcomm Snapdragon 800 2.30 GHz (4 core);
- chipset Qualcomm MSM8974 Snapdragon 800;
- 2 GB of central memory;
- Android OS 4.4 KitKat and 16 GB of disk storage.

The pool of Cloud storage services used to compose the multi-Cloud environments consists of eight services divided as follow:

- 1 Google Drive instance;
- 4 Dropbox instances;
- 3 OpenStack Swift instances.

Tests consider the varying both the characteristics of the network connection and the size of the files. ARIANNA app was tested in two different mobile network environments, *ADSL*, and *4G*, respectively.

In the *ADSL case*, we connected the smartphone to a WiFi access point, which operated as a bridge to the Internet by using an ADSL connection. Thus the available bandwidth was 384 *Kbps* in upload mode and 7 *Mbps* in download mode.

In the *4G case*, we connected the smartphone to an Internet Service Provider. Thus the available bandwidth was 50 *Mbps* in upload mode and 150 *Mbps* in download mode.

Both in the *ADSL case* and the *4G case*, we used the ARIANNA app with a set of files whose size ranged from 10 $MB$ to 100 $MB$, and we used a static splitting schema with the size of 10 $MB$. We repeated each experiment 30 times and analyzed the collected data considering 95% as confidence level.

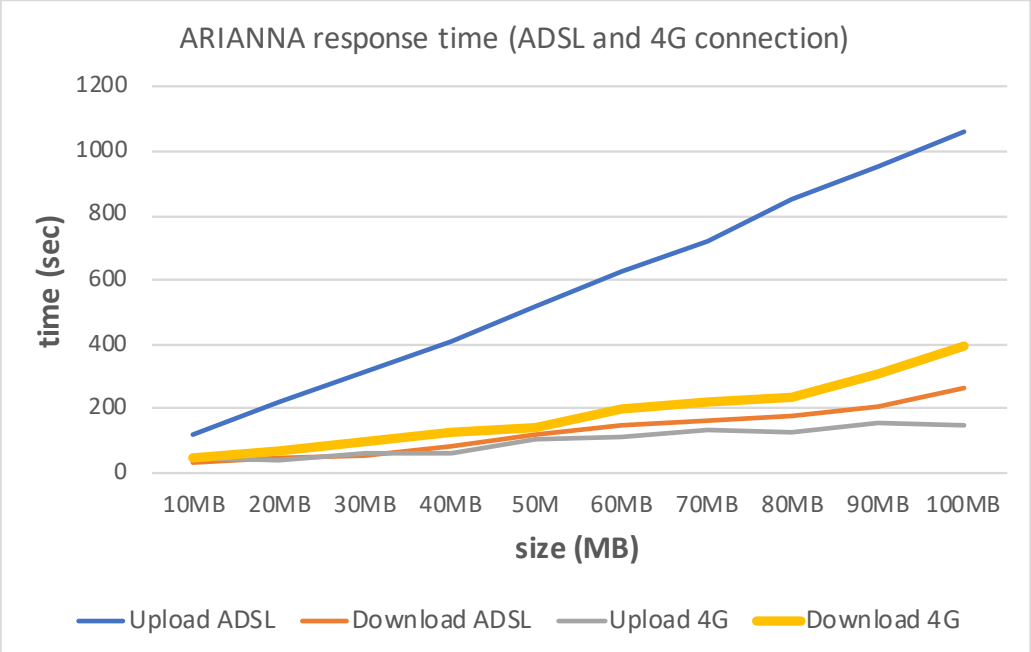


Figure 3.1: ARIANNA response times [44].

### 3.3.2 Performance of ARIANNA

Table 3.1 shows the the system response time of ARIANNA app in seconds. These measures refer to the *"Upload phase"* and *"Download phase"* [43], from 10 $MB$ to 100 $MB$, in both network environments considered. As explain in *Section 3.2*, these phrases refer to the time of upload and download of a file to and from a multi-Cloud system.

Fig. 3.1 depicts the values of Table 3.1. As can be seen in Fig. 3.1, the system response time of the ARIANNA app linearly increases with the file size both in *Upload phase* and *Download phase*, in both network environments considered.

Table 3.1: ARIANNA *Upload phase* and *Download phase* Response time: times in seconds [44].

| File size | Up-ADSL | Dow-ADSL | Up-4G | Dow-4G |
|---|---|---|---|---|
| 10 $MB$ | 120, 56 | 29, 32 | 42, 14 | 50, 57 |
| 20 $MB$ | 221, 11 | 43, 80 | 44, 49 | 69, 71 |
| 30 $MB$ | 318, 14 | 53, 32 | 59, 62 | 100, 17 |
| 40 $MB$ | 411, 88 | 83, 89 | 62, 25 | 125, 61 |
| 50 $MB$ | 517, 75 | 116, 51 | 103, 78 | 138, 75 |
| 60 $MB$ | 629, 25 | 144, 98 | 114, 88 | 201, 88 |
| 70 $MB$ | 723, 96 | 161, 59 | 130, 56 | 222, 83 |
| 80 $MB$ | 848, 11 | 174, 54 | 123, 81 | 238, 27 |
| 90 $MB$ | 949, 52 | 206, 55 | 152, 52 | 311, 08 |
| 100 $MB$ | 1059, 78 | 260, 82 | 147, 40 | 394, 64 |

### 3.3.3 Performance of Cloud storage services

As reported in Paragraph 3.3.2, the performances of the ARIANNA app varies according to the characteristics of the used network connection. However, to refine the performance analysis of the ARIANNA approach, we also evaluated the overhead introduced concerning the raw Cloud storage services that are part of the pool used in our testbed. We refer to those ones we used to make our dynamic multi-Cloud environments. Tests have been done by measuring these response times, both uploading and downloading files with different sizes from 10 $MB$ to 100 $MB$. We repeated every single

test 30 times, and we analyzed the collected data considering 95% as confidence level. Moreover, we repeated the tests at different times of the day and night. We realized that the measurements were influenced by several factors, mainly by the time slot, but also by climatic conditions, such as in the case of *4G* connectivity. The results can be seen in Tables 3.2, 3.3, 3.4, and 3.5. In these tables we reported the average response time of uploading and downloading a file by considering Dropbox [51], GoogleDrive [66] and OpenStack Swift [106] storage services.

We used the OpenStack Swift service deployed at the *Cloud Data Center* at the University of Messina [98]. Table 3.2 and 3.3 refer to the measures obtained by using the *ADSL* connection, whereas Table 3.4 and 3.5 by using the *4G* connection. As shown in Table 3.2, in the 10 $MB-30$ $MB$ range, all the Cloud storage services behave similarly in terms of upload response time. When the file size increase to 40 $MB-70$ $MB$, the response times of Dropbox and OpenStack are almost equivalent, whereas Google Drive behaves better. In the 80 $MB-100$ $MB$ range, the response time of Dropbox has the lowest response times.

Table 3.2: *"ADSL"* connection: *upload* times in seconds [44].

| File size | OpenStack | GDrive | Dropbox |
|---|---|---|---|
| 10 $MB$ | 93.69 | 93.20 | 99.40 |
| 20 $MB$ | 193.97 | 184.66 | 195.22 |
| 30 $MB$ | 280.72 | 278.24 | 292.20 |
| 40 $MB$ | 381.42 | 364.68 | 374.48 |
| 50 $MB$ | 465.38 | 453.58 | 463.32 |
| 60 $MB$ | 557.40 | 544.54 | 557.05 |
| 70 $MB$ | 641.10 | 727.33 | 639.80 |
| 80 $MB$ | 764.45 | 740.03 | 728.00 |
| 90 $MB$ | 848.80 | 828.86 | 819.18 |
| 100 $MB$ | 934.62 | 943.55 | 919.89 |

As shown in Table 3.3, the download response times of Dropbox service are almost always lower than that of Google Drive and OpenStack Swift. Google service reacts a little bit slowly in the 80 $MB-100$ $MB$ range, whereas OpenStack Swift shows the highest response time values. Consequently, OpenStack Swift is the *"storage worst choice"* in downloading in our multi-Cloud configuration when a *ADSL* connection is used.

Table 3.3: *"ADSL"* connection: *download* times in seconds [44].

| File size | OpenStack | GDrive | Dropbox |
|---|---|---|---|
| 10 $MB$ | 8.49 | 5.60 | 6.50 |
| 20 $MB$ | 17.14 | 10.34 | 12.33 |
| 30 $MB$ | 25.02 | 16.00 | 17.10 |
| 40 $MB$ | 35.97 | 21.64 | 21.10 |
| 50 $MB$ | 47.00 | 26.45 | 26.50 |
| 60 $MB$ | 58.90 | 30.88 | 31.34 |
| 70 $MB$ | 66.70 | 36.81 | 37.21 |
| 80 $MB$ | 78.20 | 41.20 | 43.70 |
| 90 $MB$ | 87.30 | 59.20 | 48.80 |
| 100 $MB$ | 97.40 | 71.42 | 54.84 |

As shown in Table 3.4, in the 10 $MB - 80$ $MB$ range, OpenStack Swift behaves better than the others in uploading files, whereas Dropbox has the worst performances. The upload response time of OpenStack Swift assumes almost always the lower values of upload response time. In comparison, the response time values of Dropbox are worse (higher) if compared with the upload response times obtained by using Google Drive. In 90 $MB - 100$ $MB$ range, the upload response time of OpenStack Swift increases significantly (it is the worst service), while the upload response time values of Google Drive decrease becoming the best.

Table 3.4: *"4G"* connection: *upload* times in seconds [44].

| File size | OpenStack | GDrive | Dropbox |
|---|---|---|---|
| 10 $MB$ | 7.3 | 8.15 | 7.24 |
| 20 $MB$ | 16.85 | 12.94 | 12.81 |
| 30 $MB$ | 17.85 | 18.58 | 22.57 |
| 40 $MB$ | 22.53 | 24.55 | 24.66 |
| 50 $MB$ | 27.31 | 30.21 | 37.1 |
| 60 $MB$ | 31.73 | 36.65 | 39.83 |
| 70 $MB$ | 35.8 | 39.48 | 41.14 |
| 80 $MB$ | 41.96 | 43.82 | 49.15 |
| 90 $MB$ | 61.64 | 50.68 | 55.55 |
| 100 $MB$ | 73.32 | 62.34 | 66.81 |

Table 3.5 shows storage service behaviors in downloading a file. In this case, Dropbox and Google Drive behave similarly, whereas OpenStack Swift is the worst with the highest download response times. In essence, it represents the *"storage worst choice"* in the download stage for our multi-Cloud configuration when a *4G* connection is used.

Table 3.5: *"4G"* connection: *download* times in seconds [44].

| File size | OpenStack | GDrive | Dropbox |
|---|---|---|---|
| 10 *MB* | 9.35 | 3.48 | 3.43 |
| 20 *MB* | 25.28 | 5.67 | 5.97 |
| 30 *MB* | 42.48 | 9.14 | 8.49 |
| 40 *MB* | 57.06 | 10.68 | 10.72 |
| 50 *MB* | 74.09 | 13.42 | 12.41 |
| 60 *MB* | 85.85 | 16.57 | 14.7 |
| 70 *MB* | 86.46 | 18.64 | 17.5 |
| 80 *MB* | 111.56 | 21.02 | 20.64 |
| 90 *MB* | 119.03 | 22.31 | 23.24 |
| 100 *MB* | 135.71 | 24.94 | 24.00 |

### 3.3.4 ARIANNA vs Cloud storage services

Measurements show that performances of Cloud storage services greatly vary in *upload* and *download* operations on different services; none of them always performs better than the others. Thus it is not possible to set priorities to improve performances of ARIANNA. Smarter, and possibly adaptive, policies could be designed, but this is out of the scope of this work. Anyway, these response times are out of our control and may vary depending on the availability, policies and features provided by Cloud storage service providers. Thus, here, we can just evaluate how augmented confidentiality in ARIANNA degrades performances in terms of response times. This type of comparison is based exclusively on *quantitative* reasoning. The use of ARIANNA involves benefits in terms of confidentiality and data integrity not provided by the other storage services, and we want to investigate how much these features cost.

Starting from these premises, *"how much"* does the use of ARIANNA cost in terms of time, rather than the use of other Cloud storage services? We

answer by comparing ARIANNA response times with the values measured in Paragraph 3.3.3.

In this regard, Fig. 3.2, 3.3, 3.4 and 3.5 show the comparison between the overall time of the ARIANNA *Upload* and *Download phase* with the *upload* and *download* time values measured from the raw Cloud storage services. Fig. 3.2 and 3.3 refer to *ADSL* connection, while Fig. 3.4 and 3.5 refer to the *4G* connection.

Tables 3.6 and 3.7 show the performance comparison per file of 100 $MB$ considering the *ADSL* and *4G* connection, respectively. We introduced and evaluated two metrics we called *U-Diff %* and *D-Diff %*. These new metrics represent the *Percentage Difference* of ARIANNA response time with respect to the raw Cloud Storage Services in *Upload* and *Download* phase, respectively.

We decided to calculate the *Percentage Difference* values because this operator quantify the change from one time value (e.g the upload time of Dropbox, $V_C = 919.89\ s$) to another (e.g the *Upload phase* of ARIANNA, $V_A = 1059.78\ s$), expressing the change as an increase or decrease. A positive change is expressed as an increase amount of the percentage value (in our example *U-Diff %* $= 15.21\ s$; *D-Diff %* $= 375.57\ s$), while a negative change is expressed as a decrease amount of the absolute value of the percentage value.

In particular:

- *U-Diff %* is defined as $\{[(V_A - V_C)/|V_C|] * 100\}\%$, where $V_A$ is the measured time to upload the file with ARIANNA and $V_C$ is the time to directly upload the file into the public storage;

- *D-Diff %* is defined as $\{[(V_A - V_C)/|V_C|] * 100\}\%$, where $V_A$ is the measured time to download the file with ARIANNA and $V_C$ is the time to directly download the file into the public storage.

**_"How much"_ does it cost to enhance Confidentiality and Integrity on data in term of time?**

As shown in Table 3.6, the use of ARIANNA to secure on Cloud (multi-Cloud) a 100 $MB$ file by using a *ADSL* connection, getting an increase in privacy and protection against insiders, costs an adding time that, in the best case is 12.32% higher in upload (GDrive case) and 167.79% in download

(OpenStack Swift case), and 15.21% in upload (Dropbox case) and 375.57% in download (Dropbox case) in the worst cases.

As shown in Table 3.7, the use of ARIANNA to secure on Cloud (multi-Cloud) a 100 $MB$ file by using a $4G$ connection, getting an increase in privacy and protection against insiders, costs an adding time that, in the best case is 101.04% higher in upload (OpenStack Swift case) and 190.79% in download (OpenStack Swift case), and 136.45% in upload (GDrive case) and 1544.34% in download (Dropbox case) in the worst cases.

It is worth noting that different performances for upload and download operations in $ADSL$ connection scenarios are because the access point used by the mobile device was connected to the Internet through an ADSL line, which is an asymmetric communication channel. This latter set of experiments shows how much the bandwidth, along with the response times of Cloud providers, affects the app behavior.

Table 3.6: Performance comparison per file of 100 $MB$ considering a $ADSL$ connection [44].

| Cloud Service | Upload | Download | U-Diff % | D-Diff % |
|---|---|---|---|---|
| Dropbox | 919.89 | 54.84 | 15.21 | 375.57 |
| OpenStack | 934.62 | 97.40 | 13.39 | 167.79 |
| GDrive | 943.55 | 71.42 | 12.32 | 265.21 |
| **ARIANNA** | **1059.78** | **260, 82** | - | - |

Table 3.7: Performance comparison per file of 100 $MB$ considering a $4G$ connection [44].

| Cloud Service | Upload | Download | U-Diff % | D-Diff % |
|---|---|---|---|---|
| Dropbox | 66.81 | 24.00 | 120.63 | 1544.34 |
| OpenStack | 73.32 | 135.72 | 101.04 | 190.79 |
| GDrive | 62.34 | 24.94 | 136.45 | 1482.37 |
| **ARIANNA** | **147.40** | **394.64** | - | - |

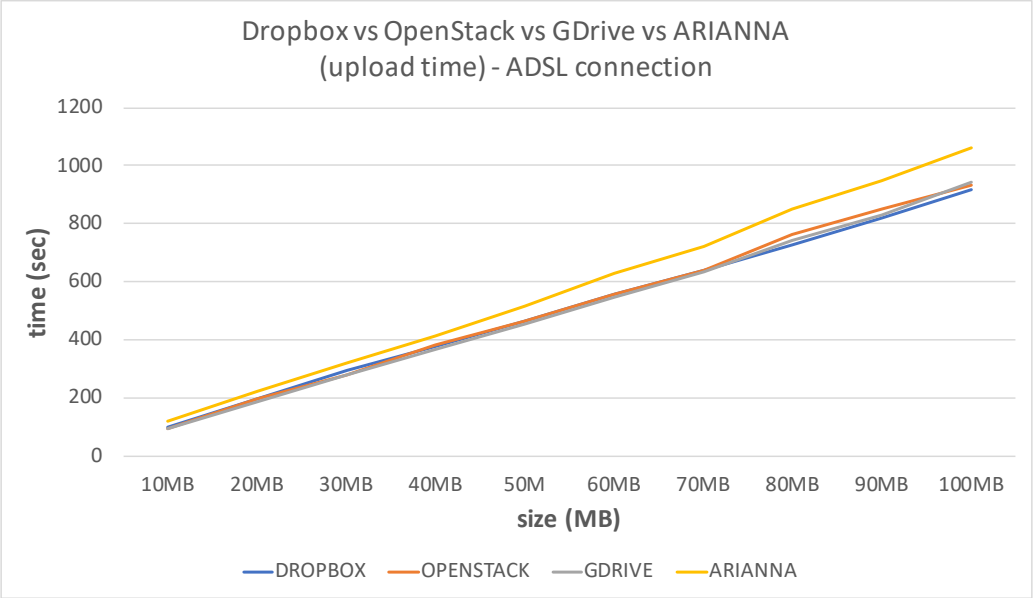


Figure 3.2: Performance comparison: ARIANNA vs Cloud storage services (upload - ADSL) [44].

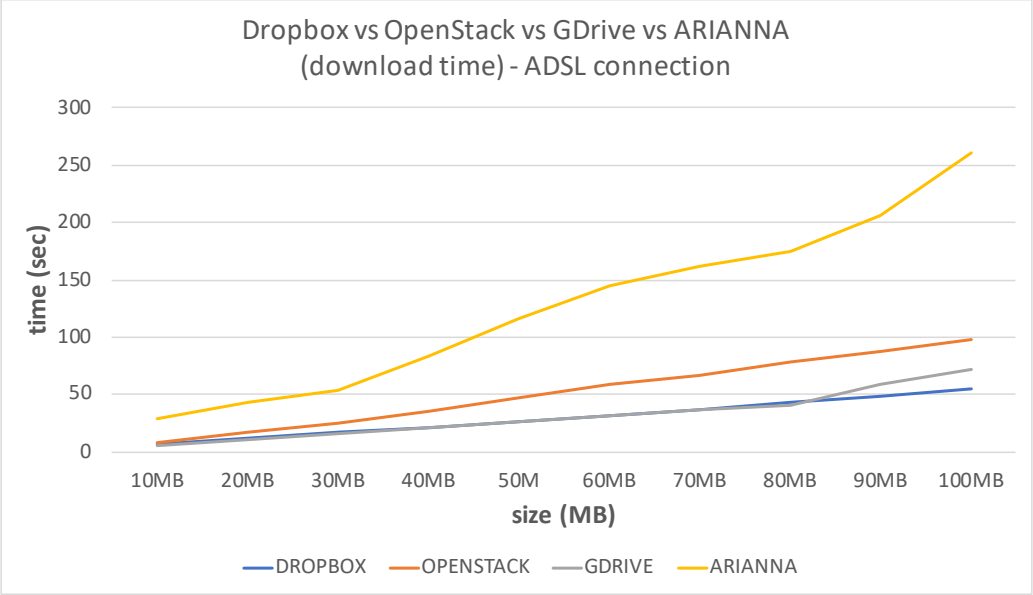


Figure 3.3: Performance comparison: ARIANNA vs Cloud storage services (download - ADSL) [44].

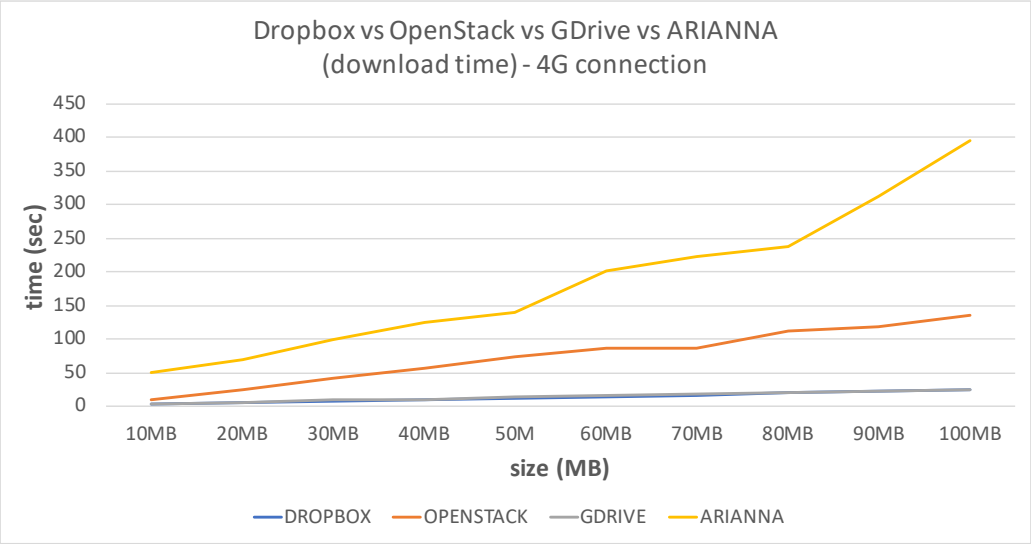


Figure 3.4: Performance comparison: ARIANNA vs Cloud storage services (upload - 4G) [44].

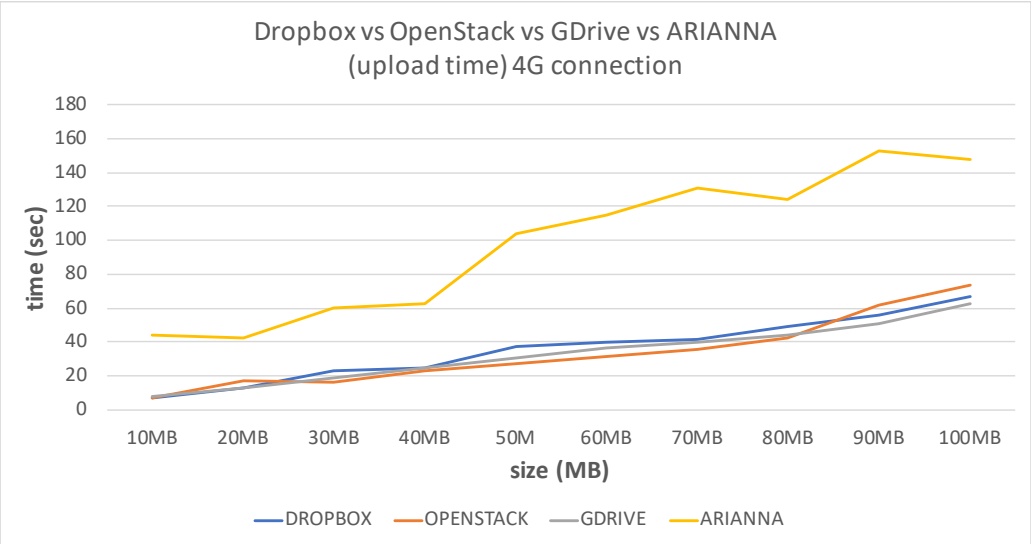


Figure 3.5: Performance comparison: ARIANNA vs Cloud storage services (download - 4G) [44].

# Chapter 4

# A Web Client Secure Storage Approach in Multi-Cloud Environment

## 4.1 Proposed Framework: Web Client Secure Storage

Fig. 4.1 shows an overview of the Web client application architecture, composing the front-end Web services, the *SSME-middleware* and the multi-Cloud service provider interfacing via HTTP protocol.

### 4.1.1 Web Client Service Application

The Web application provides user-friendly interface access at the multi-Cloud environment; the interface developed in PHP, HTML and Javascript, provides a GUI for the user to perform upload and download functions through its main menu as in Fig. 4.2. The Apache HTTP Server instance is hosted on a private server within the Cloud infrastructure. However, the Web server instance is separated from the *SSME-middleware* and acts solely for Web hosting service purposes. In this Web application, a user account is provided as any typical online form would have. It is worthy of mentioning that this first level authentication only provides the necessary access to the user page and does not provide any direct access to the files.

Fig. 4.3 shows the page flow operation based on the interaction of several

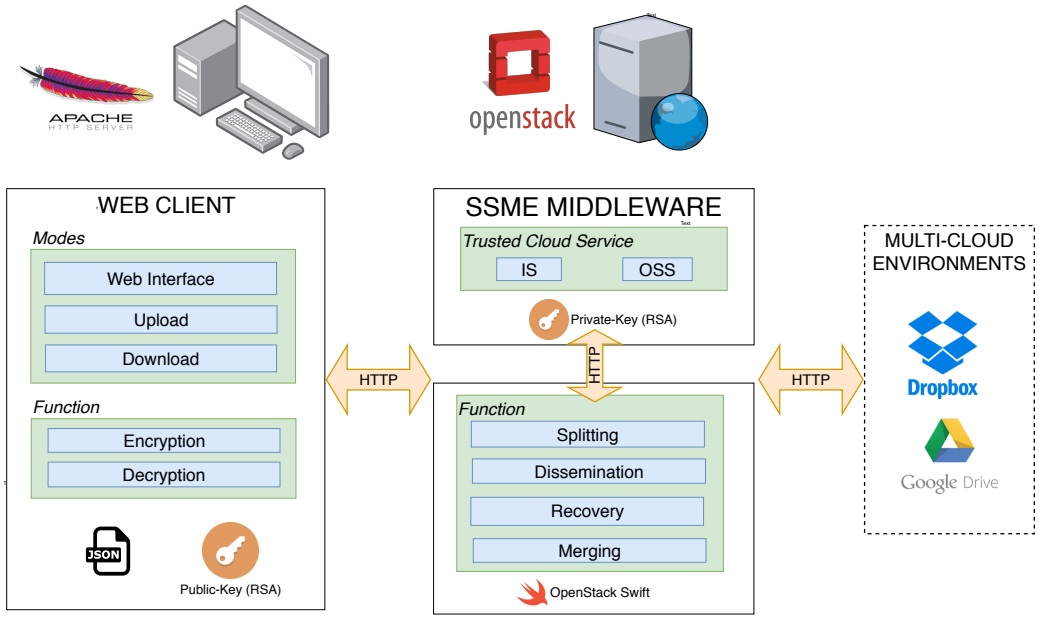


Figure 4.1: Overview of Web Client Secure Storage [116].

PHP files related to each other in accessing the primary function of the upload and download process. More specifically, the implemented components are described below:

- *index.php*: It is the main file of the project, it manages all section of the application and user log;
- *login.php*: It handles user authentication for using the Web application;
- *imlogin.php*: It supports files to login.php and represents the front end page;
- *upload.php*: It collects information on file to upload on the multi-Cloud environment and their configuration;
- *jarrun.php*: It creates the command line with the parameters which executes the java instance;
- *SSME-ClientPut-Web.jar*: A Java executable application for initiating the HTTP protocol.


Figure 4.2: Main interface for Upload and Download Selection [116].

### Configurations Management

The configuration mainly allows the user to have an initial setup of related authentication to the *SSME-middleware* server, which includes Identity Service, encryption key files, and multi-Cloud API tokens. The application will generate an SSME compatible JSON format file containing all parameters set by user access roles, as in Fig. 4.4. The JSON file is used in both the upload and download phase of the application.

Each JSON configuration format contains all the needed information in the five different fields described below:

- *WebService*: It contains the *SSME-middleware* server information and listening port access.
- *TrustedControlBlock*: It contains the information for authentication to the Trusted Cloud Service (TCS) and Identity Service (IS).
- *Criptograpy*: It has all the information related to encryption, including the path to the public key.
- *OpenStack*: It indicates the available OpenStack user account (each node has all parameters for authentication).

- *DropBox*: It indicates the multi-Cloud storage services available on DropBox (each node has an API token).

### File Upload

For uploading a file from the Web application to the multi-Cloud environment, the user will select the upload function from the main page, as in Fig. 4.2. Next, the user will choose the desired file for uploading from a standard filesystem browser as in the desktop environment for the Web upload page, as in Fig. 4.5. At this point, all configuration set earlier from the JSON structure is assigned for managing the upload process.

Once the upload is initiated through the interface, the *jarrun.php* will start to execute the Java backend instance *SSME-ClientPut-Web.jar* for HTTP request protocol in communicating with the SSME server. The HTTP response from the server will return a *JSON encrypted* file obtained by the SSME Server. Later this unique *JSON encrypted* file will be used as an object for rebuilding the fragments of the stored file.

Next, once the confirmation of the upload process is received, the fragmentation of the file in the *SSME-middleware* will be automatically deleted and distributed across the multi-Cloud environment. The only visible file in the multi-Cloud environment is the fragmented encrypted file, which is meaningless for the cloud service provider. Furthermore, dynamic configuration, specifically with encryption keys, fragment size and different Cloud storage services, ensures the confidentiality of user files.

After the first uploading, the list of the uploaded file will be available on the download page of the Web application, as in Fig. 4.6. These links of files are not the actual file link sources; however, the URL link represents a trigger protocol to download the original file from the related multi-Cloud environment. These will be further explained in the next Section on file download.

### File Download

To download a file from the multi-Cloud environment to the Web application, the user has to choose the download option from the main menu, as in Fig. 4.2. The list of the available file is represented as a trigger protocol of reconstructing and decrypting all the fragments from the multi-Cloud

environment. This list of links consists of JSON encrypted file stored at the Web server-side (Fig. 4.6).

Upon the selection of the *download* link, the Web application will decrypt the JSON encrypted file and starts the HTTP protocol backend process by executing the *SSME-ClientGet-Web.jar*. This process will collect all associated encrypted, fragmented files from the multi-Cloud environment to the Object Storage Service (OSS). Once the file is successfully downloaded from the multi-Cloud environment, the fragmented file is merged in the SSME Server and downloaded to the Web client application. Once this process is over, the SSME Server will perform a clean up of the associated fragments in the multi-Cloud environment and also in the OSS.

Therefore, the fragments that reside in the multi-Cloud environment are no longer traceable by any parties; this includes cloud storage provider and the *SSME-middleware* server. The original file is now available in the desktop directory in the user desktop.

### 4.1.2 SSME Cloud system

The *SSME-middleware* cloud system is composed of two main integrated components: the SSME Server and a Trusted Control Service (TCS). It is similar to what is described in Chapter 1 ( [47]) and to the approach used for mobile application in Chapter 2 ( [43]). However, we further explain it below in the context of our Web application implementation.

#### SSME Server

The SSME Server uses the RESTful architectural style; the resources are transferred and accessed by using Uniform Interface operation POST and GET of HTTP Methods. Currently, the Web application is running on the Apache HTTP server, which easily enables us to use the RESTful approach. In brief, as mentioned in the previous topic, the SSME server deploys two functions: Upload and Download functions.

During the Upload phase, the server retrieves from the OSS the encrypted file sent by the user. After the fragmentation step, the server allocates each fragment to a specific Cloud storage service from the pool specified by the user. By the end of the completed upload to multi-Cloud, the server deletes the original encrypted file, which was stored on the OSS.

Once the process above completed, the server will return a *JSON encrypted* file as the token source in the Web application list of available download file. Here, we ensure there is no trace of the original file to be found in the *SSME-middleware* and the Web application Server.

### Trusted Cloud Service (TCS)

In our implementation, we use OpenStack standard components: we used Keystone OpenStack Identity Service [107] as IS, and the Swift OpenStack Object Store Service [108] as OSS. In its usual activities, the TCS exploits the two Cloud services components as in the following:

- *Identity Service (IS)* provides a trusted token service for authentication on Cloud before allowing all process of HTTP request received by the SSME server;
- *Object Storage Service (OSS)* provides a trusted object storage service as temporary storage of the fragmented file before uploading or downloading in the multi-Cloud environment.

### Confidentiality and Integrity

In ensuring data confidentiality and integrity, the Web application adopts the full protocol scheme adopted in the SMME Cloud System. In specific, the authentication is validated by the IS component. The HTTP request from the Web application to the *SSME-middleware* server contains parameters and information which the *SSME-middleware* uses for managing the process. Hence, this request information is encrypted using AES256 by obtaining the key in the HTTP Header. Confidentiality and integrity derive from smart use of both symmetric and asymmetric techniques of encryption applied for all HTTP requests exchange.

In this proposed framework, the authentication layer ensures a high resistance concerning the man in the middle attacks using cryptography because other parties are not able to access the files. These include insiders form multi-Cloud service providers who have had a past or present relationship with the company. Besides, external threats would have fewer possibilities to obtain the original information, and data integrity could be maintained because no changes can be made to the fragmented encrypted files.

### 4.1.3 Multi-Cloud environment

As the multi-Cloud environment, we mean a different set of Cloud storage services that are dynamically specified by the user at the application level and are automatically managed without any control by the SSME server. Nowadays, as well as, of course, a lot of paid Cloud storage services, it is possible to find hundreds of Cloud storage services that have free accounts. These free accounts usually come with some limitations, such as the amount of storage they provide or the size limit on files users can upload.

Assuming that the third party Cloud storage services are "available" and "reliable" from all points of view, deciding which is the best Cloud storage service solution available is not easy. In this case, the quality of service is not strictly related to its quantity. For our experimental purposes, until this time, we chose to approach some commercial solutions of Cloud storage services that offer some level of free service. Specifically, in this scenario, we used GoogleDrive [74] and Dropbox [52], which offer an API token authentication access to the file directory of the cloud storage.

## 4.2 Performance Evaluation

For performance evaluation purposes on our approach of the Web application with the SSME cloud Service [47], we conducted experiments measuring the response times. In this context, we evaluated the time between the start and completion of each user request during download and upload operations. These measures show how much effective our approach is in the real-time usage of the proposed architecture.

Currently, we are using the same specification of SSME Cloud Service, as described in [43, 47]. The physical server was deployed on a virtual machine running on Ubuntu Server 14.04 and hosted on an IBM BladeCenter LS21. The client machine used in our experiments is equipped with the following hardware configuration: an AMD A10-7300 Radeon R6, 10 Compute Core 4C+6G, 16GB DDR3 of central memory, Windows 10 64-bit operating system and an SSD with 500GB of disk storage.

The distributed Cloud storage services we used in the experiment comprise four instances of DropBox, which each account was created separately with its own assigned API token. Here, we emphasize the effectiveness of our method for multi-account users. In this experiment, we set two main pa-

rameters for measuring the performance; the first is the network connection access base on both the public Internet connection and private VPN service. The upload and download speed varies depending on the network bandwidth allocation. The connection speed during the evaluation for the internet was 30 $Mbps$ in download mode and 10 $Mbps$ in upload mode. Meanwhile, the connection speed during the evaluation for the VPN was 10 $Mbps$ in download mode and 5 $Mbps$ in upload mode. Next, we chose a sample file range from 10 $MB$ to 100 $MB$ of 10 $MB$ incrementally varied.

We observed the Web application behavior in both processes of Upload and Download mode and provided each overall result with its relevant subphases. In the Upload phase, we measured the overall system response time; this includes encryption time and time due to splitting and fragment dissemination to the multi-Cloud environment. In the Download phase, we evaluated the overall system response time, which consists of the time to receive and merge all the fragments to rebuild the original file, and the decryption time of the single merger file.

Fig. 4.7 shows a graphical representation of the monitored response times, from 10 $MB$ to 100 $MB$ as file size, in Upload Mode and the Download Mode for the "internet" and "VPN" respectively. We also included measures of a "direct" upload and download to the Cloud storage without using any of our methods. However, this measure is used as a basis to observe the overhead (extra time) when using the Web application.

From the response time results, we unexpectedly obtained a linear increase relative to the sample file size in both the Upload and the Download mode. However, due to different bandwidth allocation, the session with the internet superseded all results for the VPN.

Both in Upload Mode and Download Mode, the impact of encryption and decryption, respectively, turn out to be insignificant with respect to the overall response time. When in Download Mode also the merge phase does not affect the performance, instead the splitting phase has an impact on the overall time in Upload Mode. The conclusions we can derive from the experiments for Upload Mode when Internet and VPN are used are:

- encryption value ranges from 1.67% to 2.85%, and from 1.45% to 2.33%, respectively, depending on the file size;
- Splitting and Fragments Dissemination is from 22.30% to 35.8%, and from 36.88% to 44.67%, respectively, depending on the file size.

Similarly, in Download Mode, when Internet and VPN are used, we obtain the following results:

- Receive and Merging affects the Overall Time from 3.04% to 6.83%, and 3.21% to 7.05%, respectively, depending on the file size;
- Decryption affects the Overall Time from 2.45% to 3.75%, and 2.53% to 3.84%, respectively, depending on the file size.

Next, "Direct" here refers to a normal download and upload usage of the cloud storage in Dropbox as what commonly used by users. In Fig. 4.8, we compared the overhead time taken for Internet and VPN in download phase compared to "direct" methods The average overhead time percentage for the Internet and VPN is 81.41% and 86.55%, respectively.

In Fig. 4.9, we compared the overhead time taken for the Internet and VPN in the upload phase compared to "Direct" methods. The average percentage is 74.41% and 78.77%, respectively.

It is evident from the time overhead results, our method has taken more time as usual "Direct" upload and download, as more processes added in between. However, taking into consideration the security measure introduced in our method, we believe that this brings a much more significant factor in data privacy. Hence, we do hope to improve the time performance in the future.

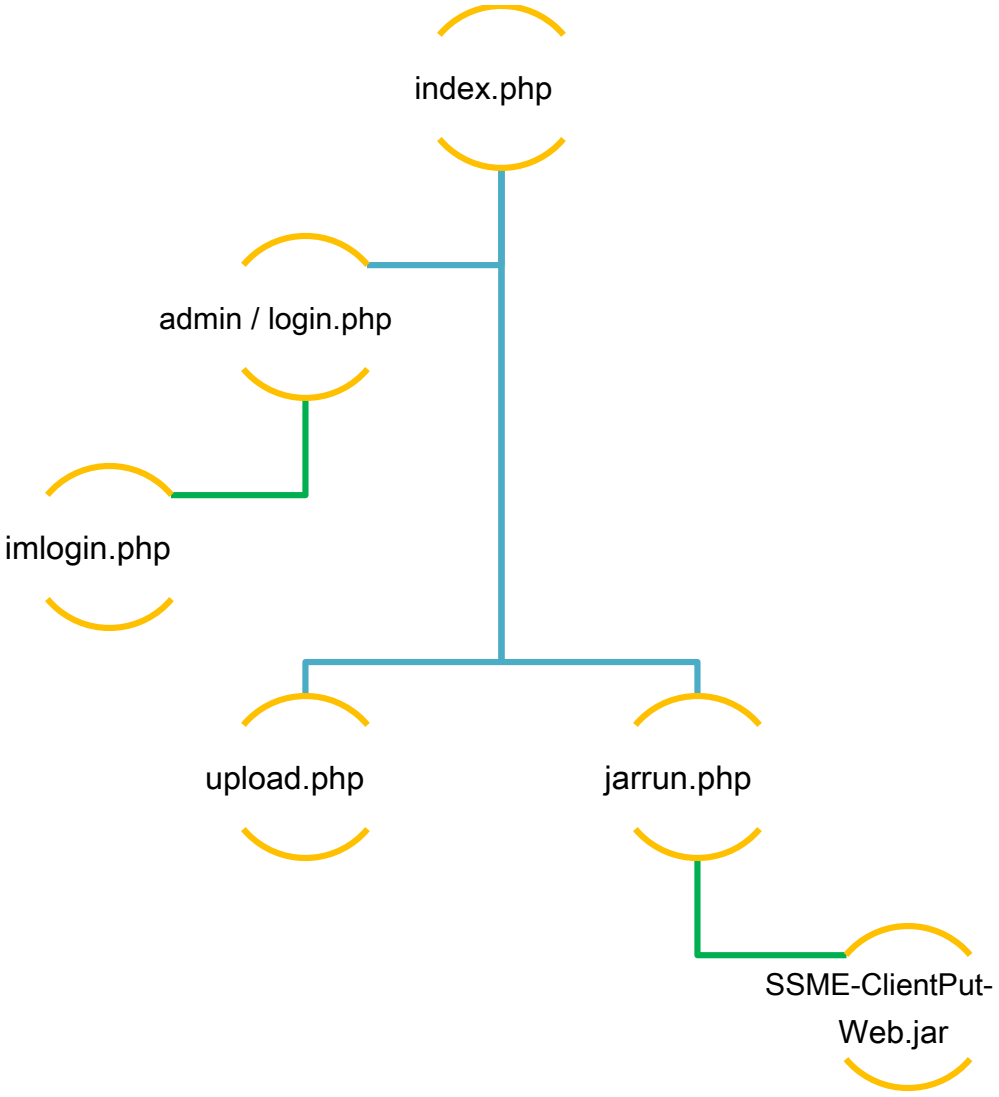


Figure 4.3: Schema of interaction of PHP Web application files [116].

```
{"WebService": {
"wsHostname": "xxx.xxx.xxx.xxx",
"wsPort": "xxxx"
},
"TrustedControlBlock": {
"Type": "openstack",
"IdentityServiceIp": "xxx.xxx.xxx.xxx",
"IdentityServicePort": "xxxx",
"IdentityServiceUserName": "e1d16f8ce3e84cbd",
"IdentityServiceUserPassword": "2027bd1893a8",
"IdentityServiceIdProject": "89f318d9c0374a74ad"
},
"Criptograpy": {
"AES256PasswordFileEncryption": "qwertyuiopas",
"AES256PasswordHttpEncryption": "jklzxcvbnmqg",
"RSAPublicKey": "folder\public_key.txt"
},
"OpenStack": [{
"Name":"useraccount1",
"OpenStackUser": "3a92df8bad004c0ca673907d0",
"OpenStackPassword": "3jCXPKAws2nwaKMwW",
"OpenStackTenant": "89f318d9c0374a74ad7c59b",
"OpenStackUrl": "http://xxx.xxx.xxx.xxx:xxxx"}],
"Dropbox": [
{"DropboxToken":"h_9c_OwKvHAAAAAAAAAACf3"},
{"DropboxToken":"IZjemoCKw4AAAAAAAAAACeR"}]
}
```

Figure 4.4: JSON configuration for SSME-Middleware [116].

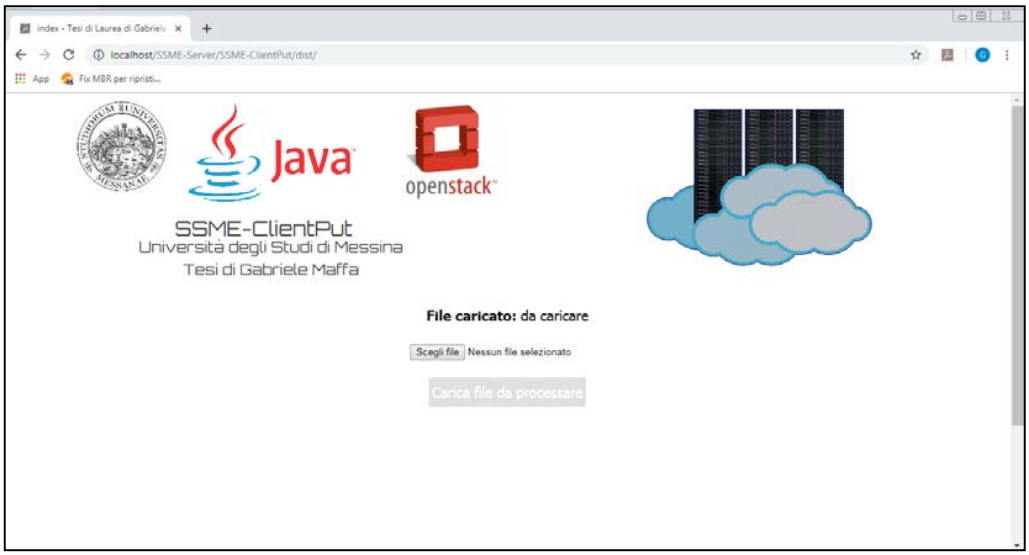


Figure 4.5: Upload File [116] .

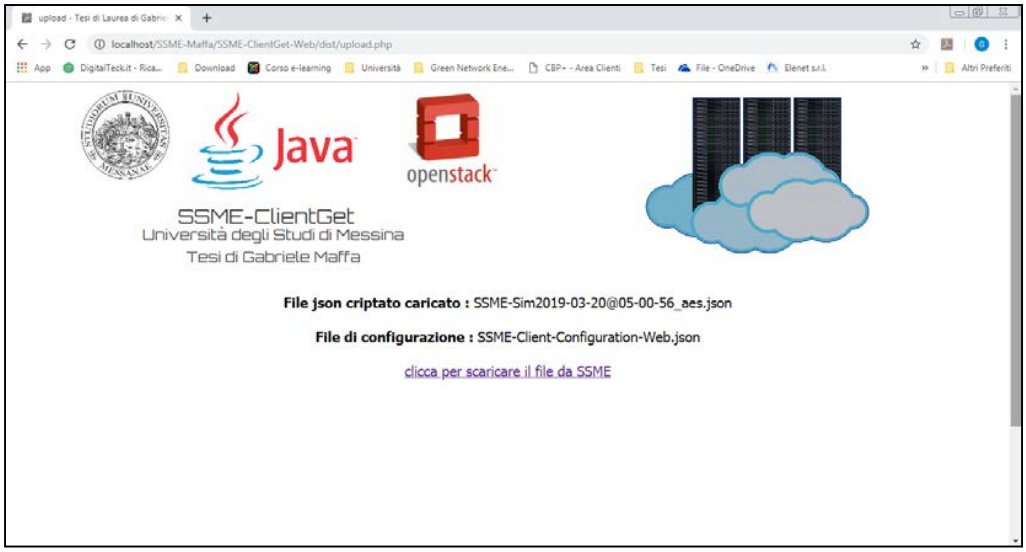


Figure 4.6: Download File [116] .

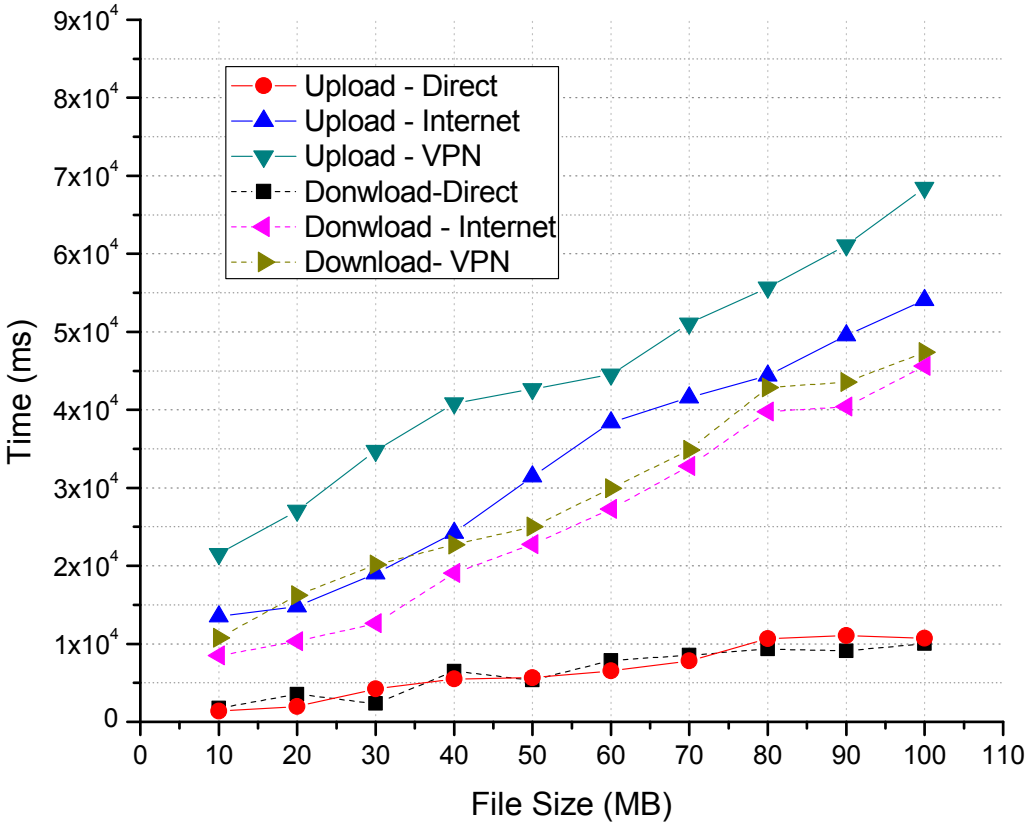


Figure 4.7: Performance by Upload and Download Time [116] .

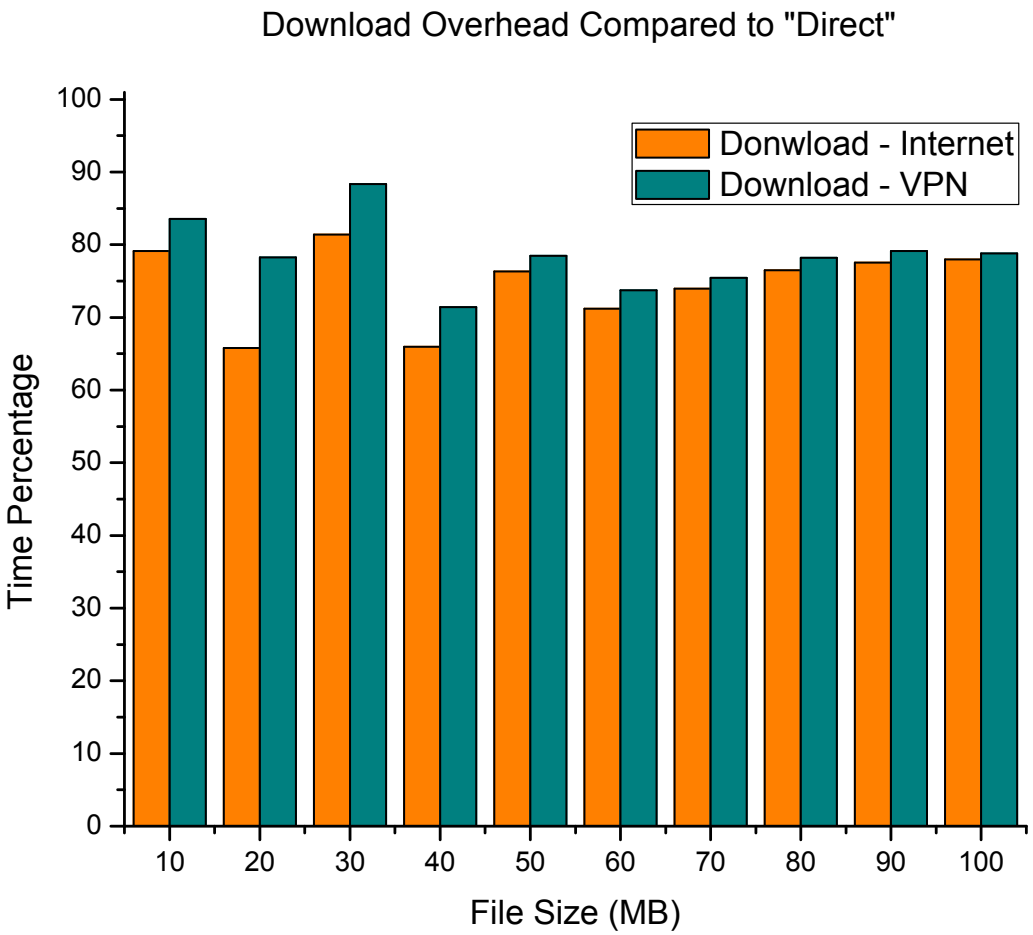


Figure 4.8: Download Time Overhead [116] .

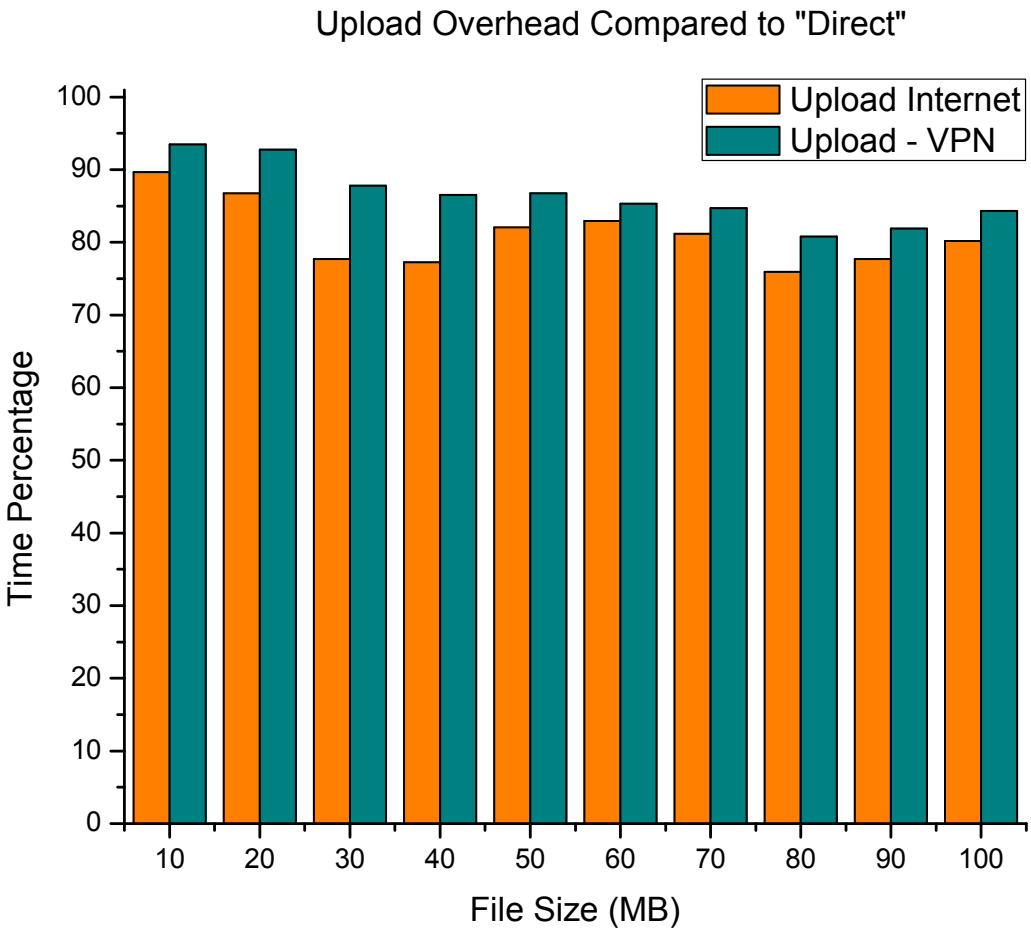


Figure 4.9: Upload Time Overhead [116] .

# Part II

# Part: How Cloud Brokerage Can Create Benefits for all Stakeholders

# Chapter 5

# J2CBROKER as a Service: A Service Broker Simulation Tool Integrated in OpenStack Environment

## 5.1 Brief Introduction to the Problem

Cloud computing is radically enhancing enterprises' productivity thanks to its elasticity, flexibility, efficiency, and on-demand and pay-as-you-go nature. Today, it is possible to benefit from the Cloud by deploying it in different service models, such as *Infrastructure* (IaaS), *Platform* (PaaS), and *Software* (SaaS). Cloud services may be offered by *Cloud Service Providers* (CSPs) in private Data Centers (DCs), i.e., *private Clouds*, or they can be commercially offered to customers, which is known as *public Clouds*. In addition, public and private Clouds may be combined to form *hybrid Clouds*. The market needs for timely, repeatable, and controllable methodologies that evaluate the conceived Cloud algorithms and policies before their actual development and deployment. Simulation-based environments play a fundamental role in this direction. First of all, they allow to set environment variables and parameters easily, define models, reproduce tests, and analyze the obtained results (textual and graphical). More importantly, the use of simulation-based approaches in Cloud environments is often a necessity, since the access to the actual infrastructure would incur payments in real currency (pay-

as-you-go service model). Thus, simulation tools can significantly benefit Cloud customers by allowing them to test their services in a repeatable and controllable environment, without paying for access to the Cloud. On the other hand, simulators can enable CSPs to evaluate, e.g., where to allocate computational resources according to varying performance, workload conditions, and monetary cost distributions. Consequently, in the absence of such simulation-based environments, both Cloud customers and providers risk to make severe mistakes of assessment or to refer to non-objective evaluations, thus resulting in inefficient service performance and economic losses.

## 5.2 The Goal of the Study

The last few years have seen the success of the Cloud computing paradigm and the steadily increasing number of service providers and available services.

Within the Digital Single Market strategy [1] (*European Commission* priority in order to achieve better online access to digital goods and services) the Cloud computing plays a key role through the *data-driven innovation* initiatives, *data ownership, access* and *usability ownership, portability of data* and *switching* of service providers. In this complex context, customers' discovery of the services and selection of the one which best suits their needs is not a trivial issue and might be very time-consuming and ineffective:*Cloud service brokerage* might help to overcome this problem. According to *MarketsandMarkets* [2], a market research firm, the Cloud service brokerage and Enablement market size is estimated to grow from USD 7.44 billion in 2016 to USD 26.71 billion by 2021.

**Cloud Service Broker (CSB)** is an additional computing layer that acts as an intermediary between service customers on one side and service providers on the other. *Gartner*, the world-leading information technology research and advisory company, identifies three areas (i.e., aggregation, integration, customization) in which Cloud brokerage might play an essential role toward service customer, but also service providers. *Aggregation* gives the possibility to manage multiple services, possibly from different providers, and present them as a unified service. However, it is not always easy because of the complex relationships and agreements among providers. The *integration* purpose is to make applications, which are independent at first, work nicely together, and cooperate to fulfill the customer's needs. *Customization* consists of the tweaking of services to best suit users' needs.

Other applications of CSB, which still come under the concept of service selection, are the *ranking* of services according to parameters provided by users (e.g., services ordered by cost) and the *selection* of the best data center (or site), among the $N$ available, to execute a certain job. For example, CSB plays an important role in legislation compliance and QoS management of Cloud services [35]. Some of the most important CSB companies are, in alphabetical order: *Appirio*, *ComputeNext*, and *Dell Boomi*.

## 5.3 Background of the Study

Due to the high interest and importance played by Cloud Service Brokerage, several works have been carried out in this field, surveying the possible approaches and algorithms for the service selection [122], [123], [110], [25] but also the Cloud simulators that can be used to evaluate CSBs performance [87], [113].

Probably, **CloudSim** [34] is the most popular and complete framework for modeling and simulating Cloud environments. It was developed at the *CLOUd computing and Distributed Systems (CLOUDS) Laboratory*, in the *University of Melbourne, Australia*. It is open-source, entirely written in Java and provides basic classes for modeling data centers, users, brokers, computational resources, policies, and virtual machines.

CloudSim is built on top of another open-source framework, namely **GridSim** [33], which was also developed at the CLOUDS Laboratory. GridSim is written in Java and mainly presents the same functionalities as CloudSim but with the difference that it is used for large scale Grid systems and P2P networks.

Thanks to its success, CloudSim has been extended by researchers, and thus other products using it as their core have been developed. The most important example of these is **CloudAnalyst** [135], which is a Java-based simulation tool. The main feature of CloudAnalyst is the presence of an intuitive Graphical User Interface (GUI), which makes it easy to set up and run the simulation. The results of the simulation are then returned in the form of charts and tables, which is very important considering their complexity and variety.

**GreenCloud** [92] is an open-source simulation environment built as an extension of the Ns2 network simulator. What distinguishes this environment

from all the others is the focus on the energy consumed by all the components of a DC in a simulated Cloud environment. Indeed, DCs require a great amount of energy, which greatly impacts the overall operational costs. GreenCloud is a packet-level simulator, meaning that protocol processing is performed whenever a packet is to be transmitted. On the other hand, CloudSim and CloudAnalyst are event-based simulators; hence, they do not individually process packets but capture the overall effect of interactions instead. The result is that GreenCloud is slower in simulating, but it is more accurate as well.

**iCanCloud** [103] is an open-source simulation platform entirely written in C++ and developed as an extension of the *OMNET* network simulator. The main purpose of iCanCloud is to estimate the trade-off between costs and performance, thus to help users make their decisions to optimize it. Besides, it provides a very complete and user-friendly GUI. Finally, iCanCloud has a feature that the environments mentioned above do not have. If there is a cluster of nodes available to experiment, it is possible to perform a parallel simulation among them. The only requirement is for the nodes to have MPI installed.

However, even if many simulator tools that can be used for studying Cloud systems, it is not possible to establish what is the best simulator to use because the evaluation depends upon the actual requirement.

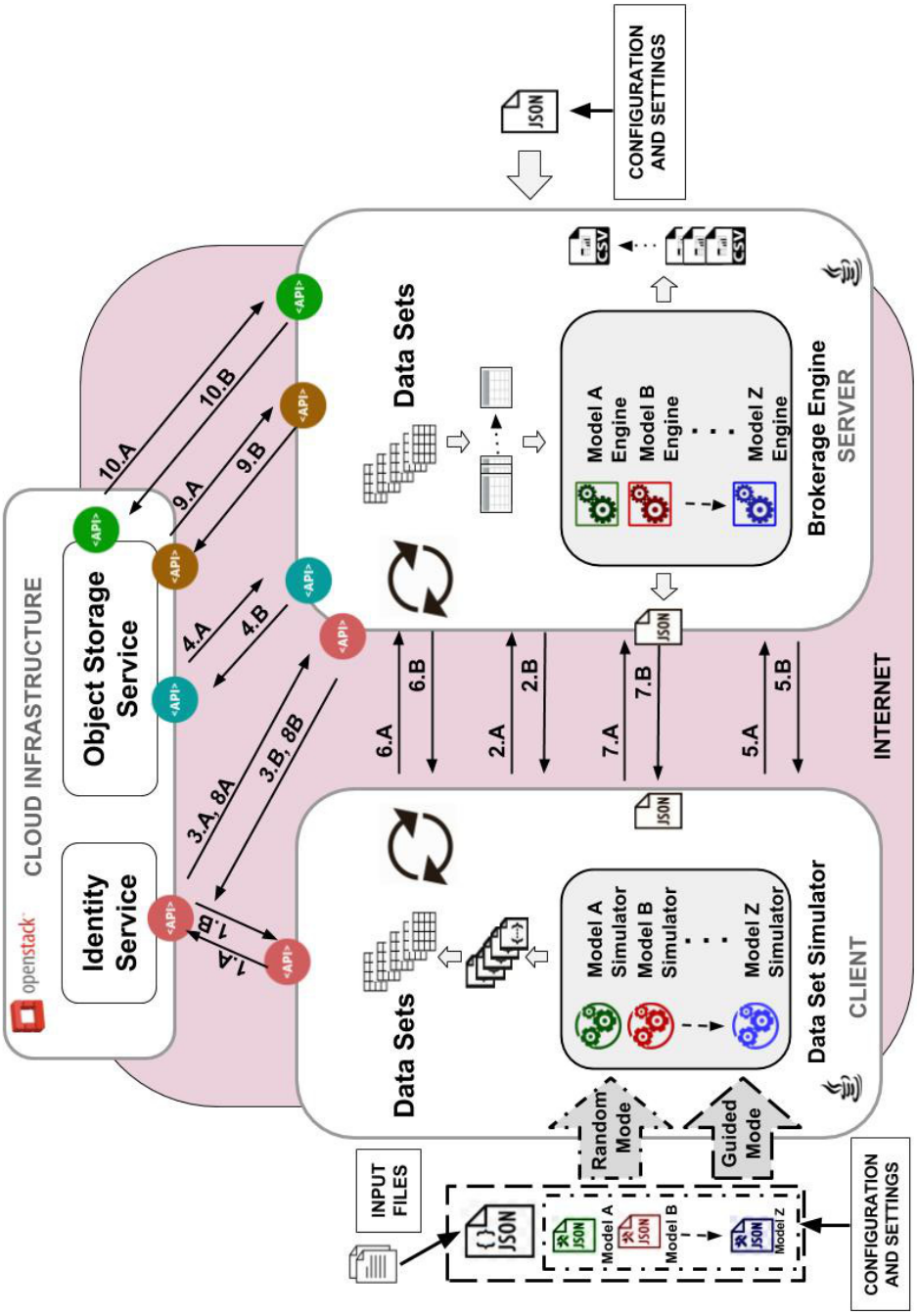


Figure 5.1: A general architecture of the *J2CBROKER* Simulation Tool [46].

## 5.4 The J2CBROKER Simulation Tool

Based on the above considerations, the decision was made to create a tool capable of simulating different cooperative Cloud Brokerage scenarios across different metrics and evaluation criteria included in *models* by using "multi-criteria" strategies. As the name implies, *"Java Json Cloud BROKER"* [60] is written by using JAVA language and JSON documents. More specifically, the main goal is to provide a simulation-based tool that: (I) dynamically manages JSON documents as inputs simulating both requests and offers by CSPs; (II) calculates the best choices (i.e., offers) based on specific parameters through different multi-criteria engines (i.e., multi-criteria algorithms implemented in JAVA language); (III) provides the resulting best offers of its calculation as outputs, both in the form of JSON documents and *on-screen* (results can also be provided in many other forms, such as *CSV* files and diagrams). J2CBROKER service is based on a JAVA client-server architecture integrated into an OpenStack Cloud infrastructure. This integration allows J2CBROKER to be owned and hosted by a service provider and to be offered to consumers on-demand. More details about the integration between J2CBROKER and OpenStack services are introduced later. J2CBROKER uses a stateless RESTful approach for its communication. Moreover, as proposed in [47], the communication protocol used between client and server uses a data protection mechanism that combines both symmetric (AES256) and asymmetric encryption (RSA) smartly. Figure 5.1 shows the general architecture of the proposed *J2CBROKER* Simulation Tool.
J2CBROKER incorporates the concept of **Model**. A Model is essentially a JSON file that contains all the *input* metrics describing the essential characteristics of a given scenario.

More specifically, a Model identifies what needs to be simulated and how. Figure 5.2 shows several metrics and *Service Measurement Index (SMI) - Key Performance Indicators (KPIs)* that are possible to be considered to have different multi-criteria Models. For each Model, the architecture provides two specific components: the related Model Simulator at the client-side, and the related Model Engine at the server-side. The architecture has been developed to simulate several possible scenarios, each one defined by a Model. To this end, referring to Figure 5.1, two blocks can be distinguished: the **Data Set Simulator** at the client-side and the **Brokerage Engine** at the server-side, both containing several Models (respectively Simulators and

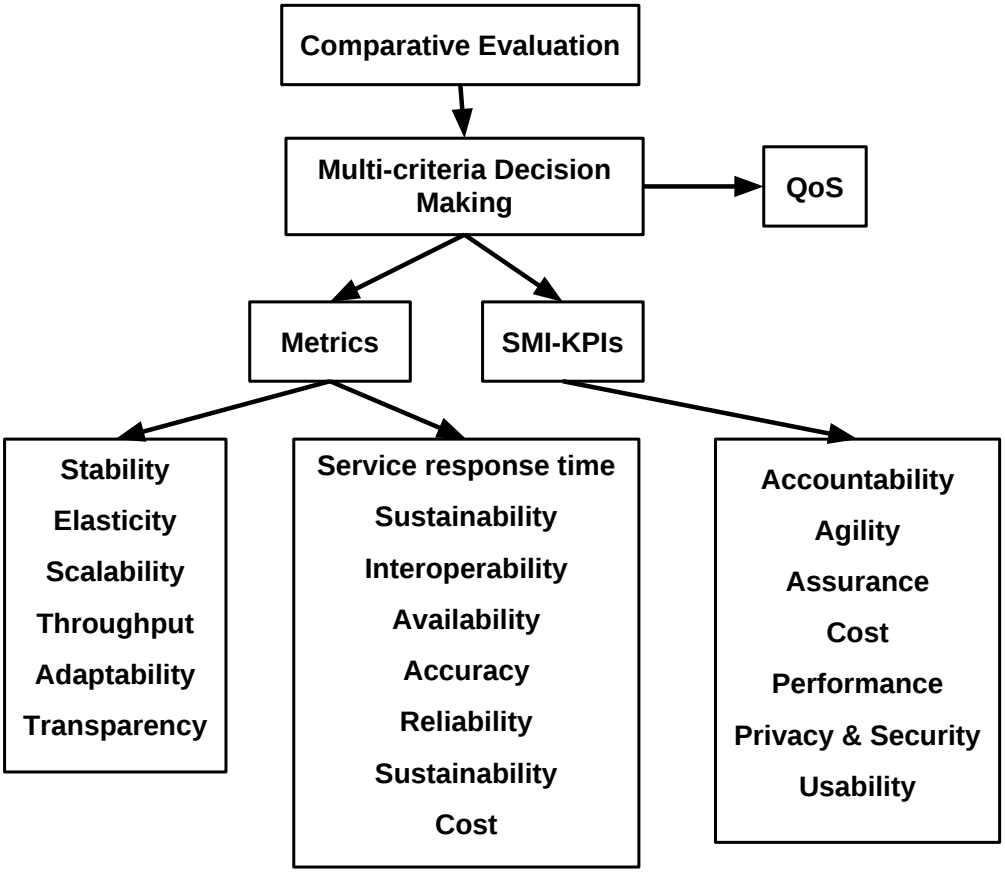


Figure 5.2: Metrics and SMI-KPIs to realize different multi-criteria Models [46].

Engines). These blocks will be discussed below.

### SaaS deployment model

In opposition to the traditional model of software deployment, the term *"SaaS deployment model"* refers to the installation and delivery of *Software as a Service*. *Software as a Service (SaaS)* is a software distribution model where the application and services are run in a centralized environment in which users access it through the network, almost always via the Internet, by using a client (e.g., Web browser or GUI) as an interface. SaaS model is characterized by a **multi-tenant** architecture; that is, there is only one

application that serves multiple users while keeping separate data and operating environments. SaaS deployment is similar to the establishment phase of a utility service, which is followed by metering and billing at regular intervals, for the services that have been delivered.
SaaS model is considered to be the winning one by all major vendor software. For this reason, the essential software vendors in the world are delivering *"as a service"* versions of their software applications, and they are offering them through an ad hoc proprietary Cloud infrastructure or relying on other Cloud service providers.

### SaaS Benefits

There are several practical and economic benefits pushing the SaaS Cloud model. From the user point of view, the big benefit is that he does not have to face a large cash outlay for software purchase, implementation, and maintenance. SaaS is used in subscription and requires lower costs at defined time intervals, and maintenance is performed directly by the software vendor. Another key benefit is that SaaS environments are based on infrastructures that can increase the amount of computing and storage offered to customers according to their needs, even momentarily and not regularly.
For the above reasons, J2CBROKER was implemented as a Service, thus to test and to deploy it in a real scenario.
The next subsection contains the explanation of the usefulness of integrating J2CBROKER in the OpenStack environment.

### OpenStack Integration

OpenStack is a set of software tools for building and managing open Cloud computing platforms for public and private Clouds. Maintained and supported by the most significant vendors in software development and hosting, and counting on the support of thousands of individual community members, today, OpenStack represents the present and the future of open Cloud computing. OpenStack is managed by the "OpenStack Foundation" [3], a non-profit organization that deals with following, supporting and influencing both the development and the ecosystem-building around the project. At the moment, the OpenStack project consists of nine main components, which represent the *"core"* of the project itself. The J2CBROKER service integrates the main components related to "Identity" and "Object Stor-

age" functionalities, which are implemented by the projects *Keystone* and *Swift*, respectively. *Keystone* is an OpenStack service that provides API client authentication and authorization by implementing OpenStack's Identity API [105]. *Swift* is an OpenStack service that provides highly available and distributed object/blob storage by implementing OpenStack's Object Storage API [106].

### 5.4.1 J2CBROKER Description

#### The Client

To work, the client needs the presence of a necessary JSON configuration file called *json-conf-file*. For the correct functioning of the J2CBROKER service, the *json-conf-file* file must be compiled correctly. As the name implies, the *json-conf-file* file contains information about the configuration of the client. In particular, it contains its internal settings (including the symmetric encryption key used); the metrics of the Model used for that specific simulation; the configuration and setup of the communication with the server (including the public key of the server); the configuration and setting of the connection with the Identity Service of the Cloud infrastructure. As shown in Figure 5.3, J2CBROKER can work in two different modes: the *Random Simulation Mode* and the *Guided Simulation Mode*. Both the above simulation modes are part of the setting at the client-side.

#### The Data Set Simulator

The Data Set Simulator represents the "core" of the client application. It consists of a modular structure that contains different Model Simulators. It is essential because it makes this *service* a general-purpose tool, which allows anyone to create and connect his Model Simulator. However, each Model Simulator implements different specific behaviors. Those latter depend on both the directives received from the *json-conf-file* file and the characteristic dedicated for the simulation scenario. If the user uses the *service* according to the directives of the *"Random Simulation Mode"*, the Data Set Simulator creates specific Data Sets according to a dedicated Model [60]. Otherwise, if the user uses the *service* according to the directives of the *Guided Simulation Mode*, the Data Set Simulator gathers and parses the information from the

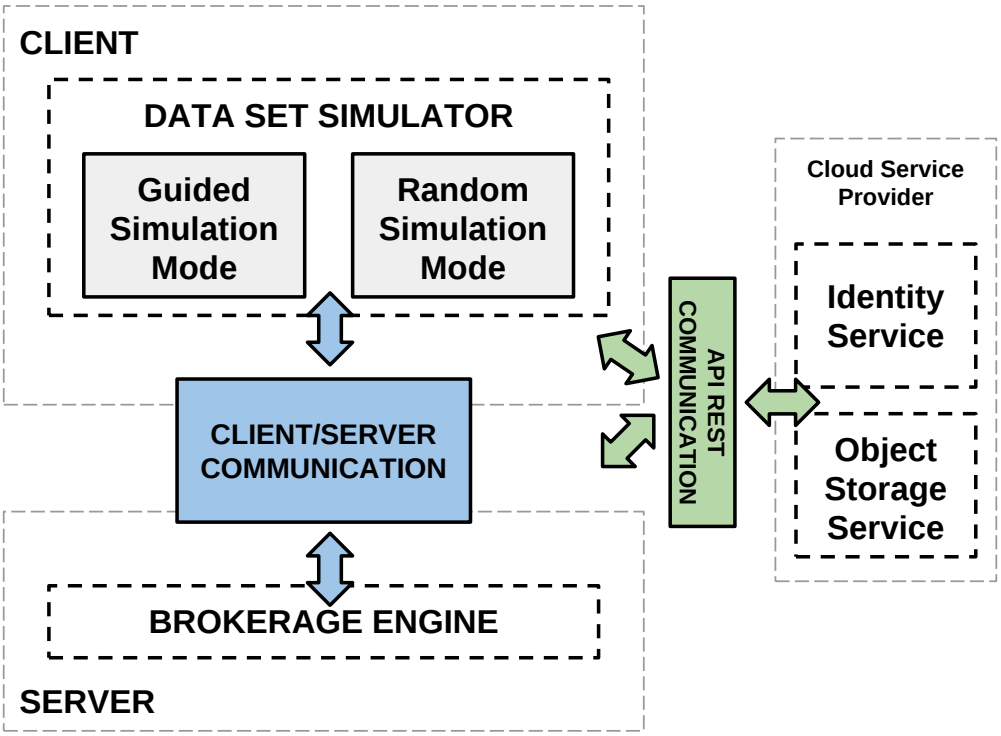


Figure 5.3: The *J2CBROKER* Simulation Modes [46].

JSON absolute paths listed inside the *json-input-list-file* file [60].

In any case, the expected behavior at the server-side will be the same. It will store all the Data Sets and, when it will receive the `active request`, it will elaborate the Data Sets according to the Model Engine predetermined in the simulation. Finally, the server will send a JSON file to the client with the result of the calculation at the Brokerage Engine. In such a context, each Data Set represents a simulated offer by a CSP at a specific Cloud site.

### The Client/Server Communication

The communication phase between the client and the server is done in six different steps (see Figure 5.1):

1. First of all, to be authenticated by the Cloud, the client sends an `authentication request` to the Identity Service of the Cloud Infrastructure. If the Identity Service returns the token **(1.B)**, which means

that it is successfully authenticated, the client can move to the next step **(2.A)**.

2. In order to verify if the server is alive, the client sends to it a `test request` **(2.A)**. Therefore, the server needs to verify if the referenced Cloud services are active:

   (a) Firstly, the server sends an `authentication request` to the Identity Service of the Cloud Infrastructure **(3.A)**. If the Identity Service returns the token **(3.B)**, then the server can move to the next step **(4.A)**.

   (b) Then, the server sends a `test request` to the Object Storage Service of the Cloud Infrastructure **(4.A)**. If the `test request` is successfully done **(4.B)**, then the server can move to the next step **(2.B)**.

   When the test is successfully done **(2.B)**, then the client can move to the next step **(5.A)**.

3. To set several environment parameters at the server-side, the client sends a `set-environment request` with some encrypted parameters inside its Headers **(5.A)**. If `set-environment request` is successfully done, then the client can move to the next step **(6.A)**.

4. To transfer all the Data Sets to the server, the client uses one `dataset request` for each Data Set. Each request will contain all the information about a particular Data Set that the client wants to transfer at that moment. All this information represents encrypted parameters stored as Headers inside the `dataset request`. If the `dataset request` was successfully done **(6.B)**, then the client will send another `dataset request`, and so on, until the end. Then the client can move to the next step **(7.A)**.

5. To start the server-side processing phase, the client sends to the server an `active request` **(7.A)**. When the server completes the processing with success, it will return a response with the JSON file containing the output of the processing phase **(7.B)**.

6. At the end of the processing phase, the server permanently stores all the data created during the processing phase:

(a) Firstly, the server sends an `authentication request` to the Identity Service **(8.A)**. If the Identity Service returns the token **(8.B)**, then the server can move to the next step **(9.A)**.

(b) Then, the server sends a `container-creation request` to the Object Storage Service **(9.A)**. If the container creation is successfully done **(9.B)**, then the server can move to the next step **(10.A)**.

(c) In order to put on the Cloud all the CSV files created during the processing phase, the server sends to the Object Storage a `create-object request` for each file **(10.A)**. If the `create-objetc request` is successfully done **(10.B)**, then the server will send another `create-object request`, and so on, until the end.

### The Server

To work, the server needs the presence of a necessary JSON configuration file called *json-conf-file*. For smooth functioning of J2CBROKER service, the *json-conf-file* file must be filled in the proper way. As the name implies, the *json-conf-file* file contains the information about the configuration of the server, in particular: its internal settings; the configuration and setting of the communication with the client; the configuration and setting of the communication with the Identity service and the Object Storage of the Cloud infrastructure. When the server starts the communication with a client, it receives `set-environment request`. From this latter, the server acquires guidelines about the Model Simulator that characterizes the Simulation Scenario, and the successive actions to do. After the server receives the Data Sets from the client and stores them, it keeps listening for an `active request` to begin the processing phase through the related Brokerage Engine. This processing phase will be different depending on which Model Engine will be used during the simulation scenario. Regardless of the type of the Model Engine used, the processing result of any simulation scenario is formalized in the form of an output JSON file, which is forwarded back to the client.

### The Brokerage Engine

The "Brokerage Engine" represents the "core" of the server application. It consists of a modular structure that contains different Model Engines. This

is important because it makes this *service* a general purpose tool, which allows anyone to create and connect his own Model Engine. However, each Model Engine implements different specific behaviors. Those latter depend on both the directives received from the client through the `set-environment request`, and on the characteristic of the dedicated brokerage scenario.

## 5.5 Case Study: Sustainability-Cost Model

In this section, a *case study* to prove the goodness of the proposed methodology is presented. Figure 5.4 introduces a **sustainability-Cost Model** to make the best choice in *resource allocation*. Sustainability is expressed through several sub-metrics which are generally used to define *"how green is a datacenter"*:

- the *Information Technology Equipment Utilization* (*ITEU*);
- the Information Technology Equipment Efficiency (*ITEE*);
- the Power Usage Effectiveness metric (*PUE*);
- the Green Energy Coefficient (*GEC*);
- the Data center Performance Per Energy (*DPPE*).

Aside from the sustainability metric, the case study includes **availability** and **monetary cost** criteria in a *multi-criteria approach*. The aim is to demonstrate the qualità of the methodology used in the sustainability model and not of the only multi-criteria approach that is well known in the literature. Availability is the degree to which a system, product or component, is operational and accessible when required for use. It is usually expressed as a percentage quota. The product quality model defined in ISO/IEC 25010 [4] comprises availability as a quality characteristic. As already reported in Figure 5.2, it is also an important *Key Performance Indicator (KPI)*. It is generally computed as a function of the total service time, the Mean Time Between Failure (MTBF), and the Mean Time to Repair (MTTR) as follows:

$$av = (MTBF/(MTBF + MTTR)) * 100 \tag{5.1}$$

The physical interpretation of availability is the percentage of time during which a system correctly operates. Monetary Cost is a quantifiable criterion

that addresses customers and organizations in their business. By specifically referring to the IT services, it is generally expressed in $\$/h$ (i.e., dollars-per-hour) or $\$/GB$ (i.e., dollars-per-GigaByte). Usually, providers offer instance placement services with a fixed price in the service maintenance time at a site. Therefore, the cost can expressed for an instance $i$ at a site node $s$, as follows:

$$cost_{s,i} = service\_price_{s,i} * \Delta_t \tag{5.2}$$

where $\Delta_t$ is the *running time* of the $i-th$ instance at the $s-th$ site node.

J2CBROKER was deployed as Service in an OpenStack infrastructure hosted on a *IBM BladeCenter LS21* at the *Cloud Laboratory Data Center - University of Messina* [98]. The service was tested by running several application clients, hosted in several machines at the *High-Performance Computing and Application - University of Messina* [82].

### 5.5.1 Scenario

J2CBROKER simulates a scenario where the main goal is to reduce *carbon dioxide emissions* (i.e. the *CO2*) through a Cloud brokerage ecosystem, where *Cloud Service Providers (CSPs)* cooperate in a *centralized brokerage environment* to run *instance workloads* at the most convenient Cloud sites. The term "instance" indicates a temporary virtual server that must be allocated to run services. The scenario considers a Sustainability-Cost Model Data Set at the client-side and a Sustainability-Cost Model Engine at the server-side.

#### The Sustainability-Cost Model Simulator

The proposed simulation environment presents the modeling of both service and Cloud site, i.e., the **Model Data Set**, thus to provide *input* data for the related Model Engine. Each offer is modeled by a *JSON* document (i.e., the Model Data Set) that includes two main collections: the first one defines a *service data set* describing the service parameters and among these availability and cost; the second one identifies the sustainability metrics and factors at each Cloud site. The service data set, in particular, is obtained from a survey on several "top" providers of IT technologies (e.g., Dell blade servers), Cloud services and solutions (e.g., Amazon Web Services (AWS)).

The second one, instead, is based on the measurement results of a real scenario, from the METI Japan project [5] on enhancing the energy efficiency and the use of green energy in data centers. Figure 5.4 shows the Model Data Set: the simulator selects a random value between the range set for each metric to characterize each offer by its sustainability, availability, and monetary cost values. The detailed description of the tabulated parameters and the multi-criteria algorithm implemented at the related Model Engine is part of our previous work [63].
Figures 5.5 shows an example of the Data Set created by the Data Set Simulator used for the proposed case study.
In particular, it identifies an *offer* in terms of:

1. "simulationName": the parameterized name of the simulation itself;
2. "providerName": the name of the simulated provider;
3. "providerNumber": the id of the simulated provider;
4. "datasetName": the name of the simulated Data Set which represents the *offer* (according to the **Sustainability-Cost Model**);
5. "datasetNumber": the id of the simulated Data Set (each provider can present different offers);
6. "Site": the information about the metrics that describe the site of the *offer* [63];
7. "Service": the information about the metrics that describe the service of the *offer* [63].

### The Sustainability-Cost Model Engine

The Brokerage Engine consists of a modular structure that contains different Model Engines and among these the Sustainability-Cost Model Engine. This one is mainly based on a decision-making algorithm from the perspective of an energy- and cost-aware instance allocation in a centralized brokerage Cloud environment. It results in the *opt* value, which is an "optimum" index computed by the algorithm to weigh each offer in terms of carbon dioxide

| Sustainability-oriented Model Data Set | |
|---|---|
| **Service Data Set** | |
| **Parameter** | **Range** |
| **Instance workload (watts)** | 200-300 |
| **Power basic (watts)** | 100 |
| **Running time (hours)** | 10, 24, 360, 750 |
| **Number of instances in each request** | 1, 10, 20, 50 |
| **Number of instances in each offer** | 12, 14, 16, 18, 20 |
| **Availability (%)** | 99.90-99.99 |
| **Service price ($/h)** | 0.007-0.112 |
| **Sustainability Data Set** | |
| **Parameter** | **Range** |
| **ITEU** | 0.3-0.6 |
| **ITEE** | 0.1-3.9 |
| **PUE** | 1.4-2.3 |
| **GEC** | 0.0-0.003 |
| **CDIE (kgCO2/kWh)** | *source: https://www.ipcc.ch* |

Figure 5.4: The Sustainability-Cost Model Simulator Data Set [60], [46].

emission (sustainability in gCO2), cost ($-per-hour), and availability (%) as follows:

$$\begin{aligned} opt_{nProvider,nDataset} = \\ A * (gCO2_{nProvider,nDataset}/gCO2_{worst} + \\ B * (cost_{nProvider,nDataset}/cost_{worst}); \end{aligned} \tag{5.3}$$

```
{
"Simulation": {
        "simulationName": "J2CBROKER-Sim"
        },
"Provider": {
        "providerName": "PROVIDER#1",
        "providerNumber": "1"
        },
"Dataset": {
        "datasetName": "R5KL3",
        "datasetNumber": "1"
        },
"Site": {
        "ITEU": "0.4",
        "ITEE": "3.0",
        "GEC": "0.0",
        "PUE": "2.1"
        },
"Service": {
        "instanceNum": "1",
        "running_time": "10",
        "t_start": "03-05-2017 16:46",
        "availability": "99.22",
        "workLoad": "212",
        "servicePrice": "0.099"
        }
}
```


Figure 5.5: An example of Data Set created by the Data Set Simulator [46].

where $A + B = 1$. More specifically, A and B represent the *weights* respectively assigned to "sustainability" and "monetary cost-saving" in the simulation. In that formula, *nProvider* is a unique number which identifies the CSP site; *Dataset* is a unique number identifying the offer of that CSP;

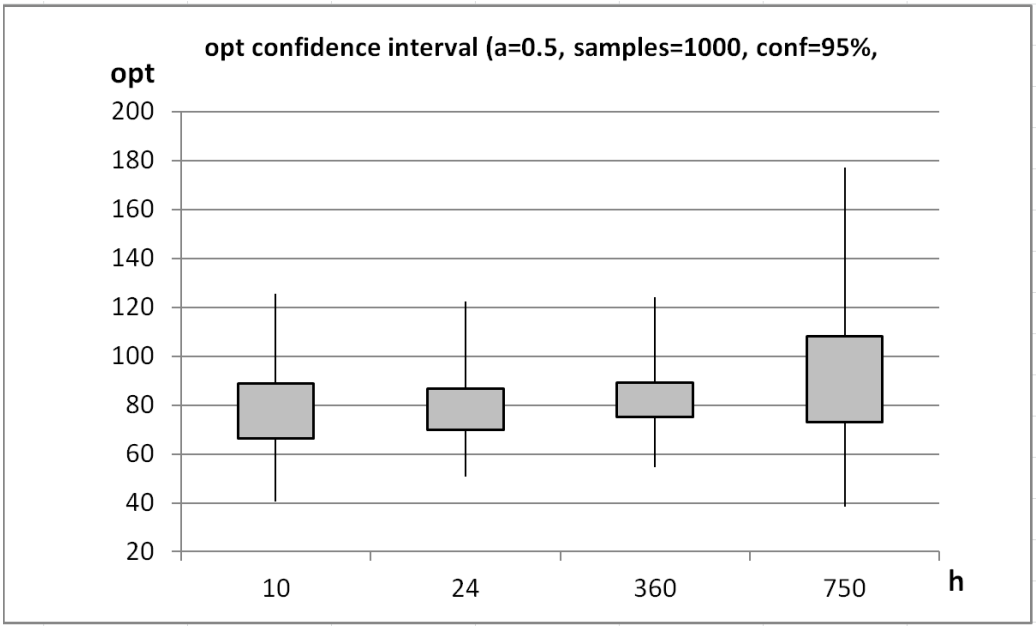


Figure 5.6: Confidence interval of the *opt* index for the allocation of 10 instances [46].

*gCO2* quantifies the related CO2 emission to run an instance workload; *cost* is the service price in $-per-hour. Both CO2 emission and cost are normalized to the respective worst-case calculated at each iteration on all the offers by all the CSPs.

### 5.5.2 Experimental Results

Figure 5.7 shows several *opt* values calculated on the basis of Formula 5.3. The experimental results are distinguished by *running time* "h" (10, 24, 360, 750 hours), as reported in the dataset at Figure 5.4, and the test considers a number of 1000 samples (iterations) for each h with a 95% confidence. This latter is an observed interval, in principle, different from sample to sample, that frequently includes the value of an unobservable parameter of interest if the experiment is repeated. The desired level of confidence is set (i.e., not determined by data). If a corresponding hypothesis test is performed, the confidence level is the complement of the respective level of significance, i.e., a 95% confidence interval reflects a significance level of 0.05. For each *running time* the simulator calculates the *minimum (MIN)*, the *maximum (MAX)*,

| opt range (a=0.5) | | | | |
|---|---|---|---|---|
| h | 10 | 24 | 360 | 750 |
| MIN | 40,483 | 50,777 | 54,660 | 38,549 |
| MAX | 125,38 | 122,195 | 124,219 | 177,089 |
| AVG | 77,53 | 78,275 | 82,064 | 90,456 |
| CONF | 22,686 | 16,943 | 14,009 | 35,095 |
| AVG-(CONF/2) | 66,194 | 69,803 | 75,060 | 72,909 |
| AVG+(CONF/2) | 88,880 | 86,747 | 89,0697 | 108,004 |

Figure 5.7: Experimental results [46].

the *average (AVG)*, the *confidence (CONF)*, and the last two confidence interval limits. Figure 5.6 provides the reader with a quick visual feedback about the confidence interval of the above mentioned *opt* index to allocate a number $N$ of 10 instances, with $A = 0.5$, $B = 0.5$, a number of 1000 samples, and a 95% *confidence*. The values reported in Figure 5.6 are the result of a post-processing phase, by getting as input all the best *opt* values calculated at each run step. Considering that for each run in our simulation, the worst case results in an *opt* index close to 1000, the energy-aware algorithm at Broker can select offers with an *opt* index much lesser when compared with the worst case. The result is a good compromise between sustainability and cost since it is as better as it is closer to zero.

Figure 5.8 and Figure 5.9 show a graphical representation of the results that are obtained in 5.5.1, respectively in terms of *sustainability* and *cost saving* metrics. This latter are indicative of the *goodness* of the CSP offers.

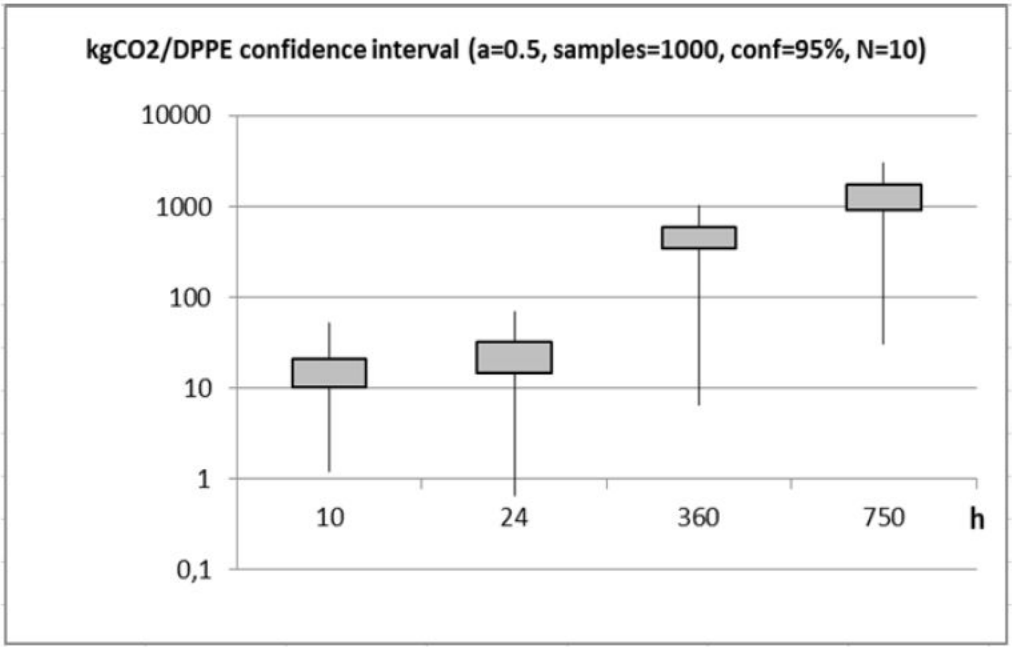


Figure 5.8: Confidence interval of the sustainability (kgCO2/DPPE) for the allocation of 10 instances [46].

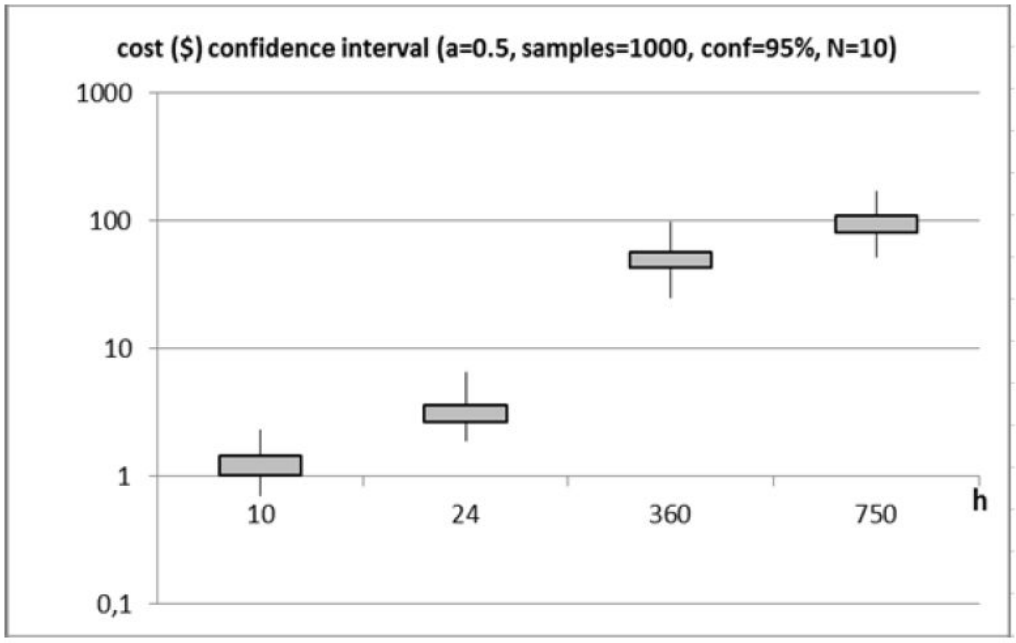


Figure 5.9: Confidence interval of the cost saving for the allocation of 10 instances [46].

# Chapter 6

# The Internet Of Things In Oil And Gas Industry: A Multi Criteria Decision Making Brokerage Strategy

## 6.1 Brief Introduction to the Problem

The proliferation of a wide variety of Internet-connected and low-cost devices is leading to the development of the revolutionary communication paradigm known as the Internet of Things (IoT) [75]. It allows both public and private organizations to combine always-connected, non-invasive, smart objects (Things) [50] to improve everyday human activities. Moreover, the combination of IoT and Cloud computing is pursuing new levels of efficiency in delivering services [58]. The emerging business perspectives coming from IoT are pushing private, public, and hybrid *Cloud Service Providers (CSPs)* to integrate their systems with embedded devices (including sensors and actuators) to provide new services. As a consequence: i) new types of providers have been rising that combine the traditional Cloud computing paradigm with IoT; ii) a new type of distributed system has been designed. They consist of a set of smart devices interconnected with remote Cloud infrastructure, platform, or software through the Internet, to provide Sensor and Actuator as a Service (SAaaS).

Furthermore, *Community Cloud* is an emerging topic. It is built and

provisioned by its members, and it can be owned and managed by the Community itself, by a third party, or a combination of both. A small or medium-size CSP which receives IoT service requests from customers and is unable to satisfy all the demands by allocating suitable and affordable resources at its data-centers can take part in a "community" [101]. The resulting benefits, costs (i.e., money), and responsibilities are shared among the Community CSPs. These latter will be able to offer IoT-Cloud services through private data-centers, i.e., by *private Clouds*, they can be commercially provided for customers, i.e., by *public Clouds*, or yet both public and private Clouds may be combined forming *hybrid Clouds*.

In any case, the IoT-Cloud union may require a wide range of new "big data" technologies and services capable of managing both semi-structured and unstructured data. About this "big data problem", the total amount of data created (and not necessarily stored) by any device will reach 600 ZB per year by 2020 [6].

In this scenario, the IoT-Cloud union is creating a new digital agenda for Oil and Gas (O&G), thus changing the way to conceiving distributed systems to serve this business.

To be a leader, an O&G company must innovate its industrial control systems by using IoT as a new model to integrate information from data collected, employees, and industrial processes across its supply chain. This progress can result in new business opportunities, mainly to keep human safety in the industrial plant. The novelty element introduced by this research is the proposed ***multi-criteria approach*** which adapts the need of Chief Information Officers (CIOs) and IT leaders to plan a strategic perspective to acquire business value from the IoT-Cloud union. *Insirio SpA* pays serious attention to the above needs. The firm develops IT solutions for the Owner Operators, the *Engineering Procurement Construction (EPC)* Companies, and their suppliers and sub-contractors. The experience of *Insirio SpA* consists of a hundred projects for the O&G industry that addressed the presented work to understanding the feasibility of the IoT application in O&G. In this regard, the use of IoT in O&G Industry is feasible in the construction phase of the site (e.g., traceability of specific modules for pipelines, human safety devices), and not in its executive and maintenance phases. The reason is due to the current know-how and the difficulties in using "open" hardware and software (e.g., *Arduino*) by suitable low-cost smart objects for executive and maintenance, in compliance with safety conditions.

## 6.2 The Goal of the Study

O&G companies can control their processes by using intelligent objects and distributed services able to act both locally on the data they generate and using the Cloud for data management, analytics, and durable storage.

This scenario requires timely, repeatable, and controllable methodologies for the evaluation of algorithms, applications, and policies before the development of IoT-Cloud services, especially to reach a good compromise between several *heterogeneous* indicators. The use of real testbeds would be the best choice. However, since the use of real testbeds limits the experiments to the scale of themselves, thus making the reiteration of results a challenging undertaking, alternative approaches need to be considered. Among these possible alternatives, **simulations tools** allow researchers and practitioners to evaluate the working hypothesis before software development. The simulation approach is fundamental for IoT-Cloud environments because access to the infrastructure incurs payments in real currency (*pay-as-you-go* system). Simulation tools can offer significant benefits for both customers and CSPs.

### Customer-side Benefits

Customers can test their IoT and Cloud services in a repeatable and controllable environment, at no additional cost. Moreover, a customer can evaluate the number of smart objects to use and raw data to transmit to the Cloud to contemporary reduce costs and to increase the quality of the data.

### CSP-side Benefits

On the CSP-side, simulators can allow the assessment of different scenarios where to allocate IoT-Cloud resources based on performance, workload conditions, and monetary cost distributions. A Community CSP can proceed in its *business analysis*, thus optimizing the cost to access to resources with particular attention on improving profits.

In the absence of such simulation-based environments, both IoT-Cloud customers and CSPs risk making severe errors of assessment, or to refer to non-objective evaluations, with the consequent inefficient service performances and economic losses. Therefore, both for customers and CSPs, simulation-based environments allow evaluating the hypothesis before the

software development, thus reducing the risk of economic losses and low *Quality of Service*. In this context, providing CIOs and IT leaders with guidelines for managing their investment objectives in digital transformation and implementation strategies is a priority.

For the above reason, this chapter introduces a simulation-based *Multi Criteria Decision Making (MCDM)* brokerage strategy. The goal is to allow small and medium-size cooperative IoT Cloud providers to satisfy the demand for IoT-Cloud services, with a good compromise between service level and business for the O&G industries.

## 6.3 Background of the Study

This Section introduces the background concerning the use of IoT in industry and the management of massive sensing data through the Cloud according to the technical needs.

In the past, *Wireless Sensor Networks (WSN)* was not the preferred choice for offshore monitoring in the O&G sector due to many reasons, such as reliability and security. Today, however, progress in terms of reliability, security, and affordability in the constraints of frequency allocation allow O&G companies to take full advantage of WSN for challenging industrial environments. On this topic, [88] authors propose a WSN-based industrial rack safety monitoring system that uses the ISA100.11, which consists of a standard for industrial field instruments.

Standardization efforts on IoT and machine-to-machine communication are addressing *industrial-strength networking* in multiple forums: a viewpoint about *Intelligent Systems* as a new industrial revolution is described in [24].

In [27], the authors address massive sensing data management in the Cloud. It includes a framework supporting parallel storage and processing of enormous sensor data in Cloud manufacturing systems, based on Hadoop MapReduce.

A context-oriented data acquisition and integration platform for the IoT over a Cloud is presented in [37]. The platform collects sensor data from different types of sensor devices and integrates them into semantic contexts, which can be easily shared and reused among various mobile applications. As a consequence, the context information can enhance mobile app usability by adapting to conditions that directly affect their operations.

In [138], the authors propose a *Sensor-Cloud Infrastructure*, which can manage physical sensors on IT infrastructure to improve the usability through virtual sensors on Cloud computing. The Missouri S&T (Science and Technology) sensor Cloud [97] connects different networks over a large geographical area, to be employed simultaneously by multiple users according to an on-demand service model.

The *Mobile and Distributed Systems Lab (MDSLab)* at the *University of Messina, Italy,* developed *Stack4things*: an OpenStack-based and Cloud-oriented horizontal solution providing IoT object virtualization, customization, and orchestration [99].

The aforementioned scientific contributions clarify that it is not possible to postpone the correct planning and management of the IoT-Cloud services in industrial plants. In such a complex context, it is necessary to use optimized systems aimed to control a multitude of variables, thus avoiding low productivity and economic losses.

## 6.4 The IoT-Cloud Brokerage Scenario

This section introduces the description of the real IoT-Cloud Brokerage scenario this study refers to. It is useful to understand better the proposed strategy and how the simulation tool presented in the next Sections works.

In particular, the study refers to a dynamic scenario in which CSPs can share their IoT-Cloud resources among the related Cloud sites (i.e., datacenters) within a cooperative Cloud environment, thus forming a *Community* to satisfy what the O&G industries need in terms of *Quality of Service (QoS)*.

In such a dynamic critical service scenario, an automated *Service Level Agreement (SLA)* negotiation process, involving both customers and cooperative CSPs, facilitates bilateral negotiation between **Community Cloud broker** and multiple providers to achieve goals for the O&G industries. Therefore, relating each other heterogeneous parameters, which are usually considered "individually" by the IoT-CSPs and to balance them, it's a challenge. To this end, a Cloud brokering approach can simplify procedures in making the best choice to achieve a good compromise.

The proposed brokerage scenario is shown in Fig. 6.1. It is representative of a large-scale IoT-Cloud platform with a centralized brokerage scheme.

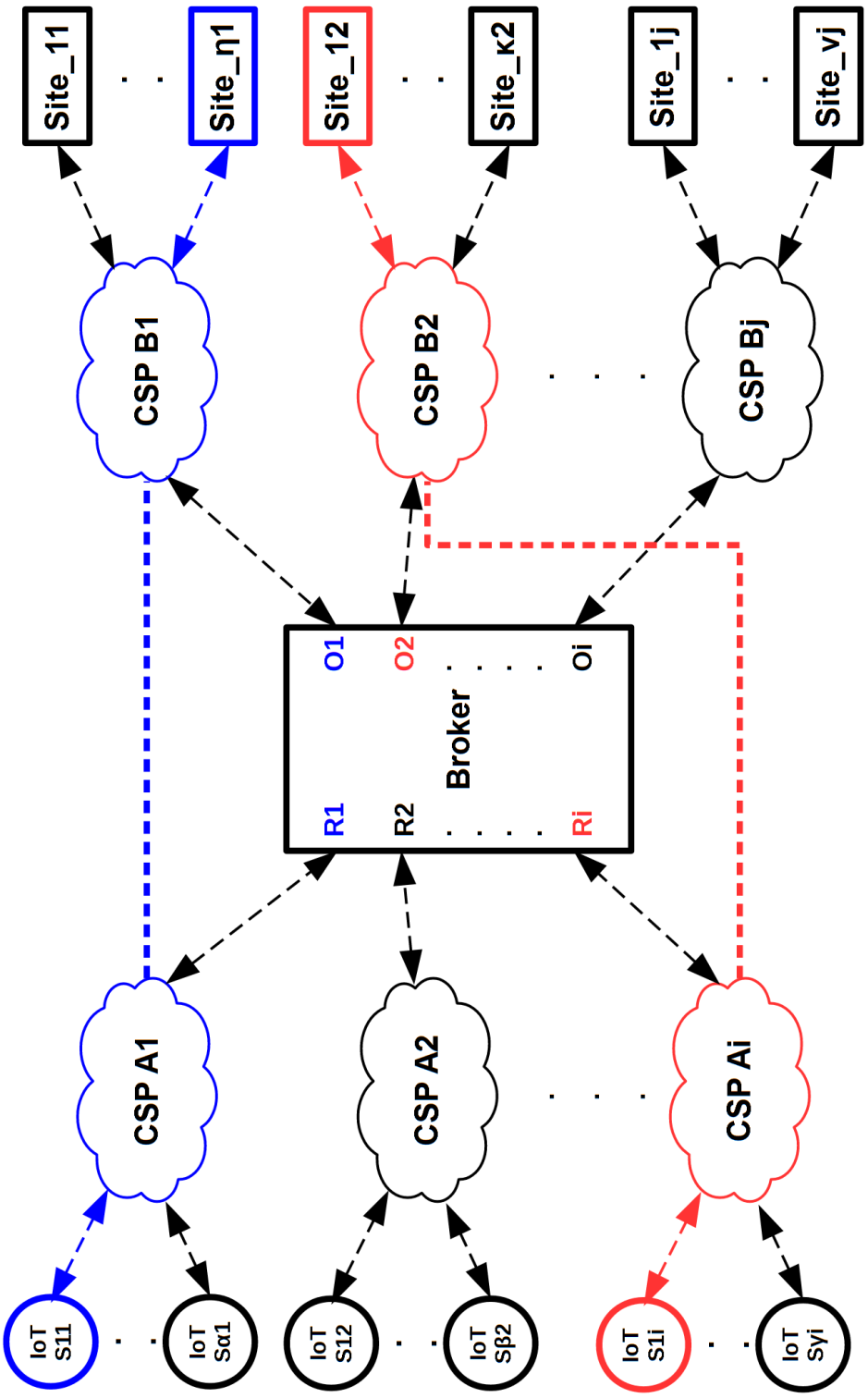


Figure 6.1: Exemplifying IoT-Cloud Brokerage Scenario [61].

More specifically, on the left, each *Applicant CSPs*, that is unable to serve the IoT services (i.e., in terms of QoS) to a customer through its Cloud resources, makes a request R to the Broker to receive the best offer O from the *Bidder CSPs* on the right. The 'blue' connection int the exemplifying schema reports the CSP $A_1$ makes a request $R_1$ to the Broker, which, in turn, calculates the best offer among the Bidder CSP. It results in the $O_1$ from the CSP $B_1$ which is available to run the IoT services $S_{11}$ at its own $Site\_\eta 1$. Therefore the $O_1$ is turned by the Broker to the CSP $A_1$ which attempts to contact the CSP $B_1$. Finally, the CSP $A_1$ provides the IoT services $S_{11}$ to the customer, such as the O&G industry, through the CSP $B_1$ at the Cloud $Site\_\eta 1$. The same approach is for the 'red' connection between the CSP $A_i$ and the CSP $B_2$. In this case, the request $R_i$ for the IoT services $S_{1i}$ can be satisfied by the offer $O_2$.

## 6.5 The MCDM Strategy

The main objective of the proposed Cloud broker decisional system is to pick out a set of offers that meets specific complex requirements. To this end, the proposed strategy implements a Multi-Criteria Decision Making (MCDM) algorithm that has been adapted to address the requirements of the proposed IoT-Cloud scenario using *multi-criteria* JSON Data Sets. This section presents and discusses the design of the multi-criteria decisional system.

The MCDM algorithm allows the Cloud broker to solve a decisional problem in which, according to O alternatives (i.e., offers) and $\Gamma$ decisional criteria, the goal is to identify the best alternative or a set of $A$ alternatives so that $2 \leq \mathrm{A} \leq \Gamma$. The algorithm dynamically manages the alternatives in the form of JSON documents as inputs, i.e., each offer is modeled by a JSON document.

The proposed strategy consists of three phases: **preliminary, applicant** and **brokerage**.

### 6.5.1 Preliminary phase

A preliminary phase is under the responsibility of the customer and consists of four main steps.

The **first step** is to choose the set $\Gamma$ of decisional criteria. The choice implies the following AND condition:

$$AND(c_1, ..., c_N) = true; with\ N \geq 2 \tag{6.1}$$

Starting from an in-depth analysis of the IoT-Cloud commercial platforms and services by several "top" leader (e.g., AWS, Teradata, Deloitte US, Google) the study guided the choose on the following criteria:

- **Operational Availability**;
- **Storage Capacity Service Price**;
- **Data Analytics Service Price**;
- **Cybersecurity Level**;
- **Support Level**;

The **second step** was to quantify, for each criterion, the basic needs of the IoT-Cloud service to be required.

Once criteria have been selected, at the **third step**, a weight $w$ is associated with each criterion $c$, so that:

$$\forall c \in \Gamma \Rightarrow \exists w \in W : \sum_{k=1}^{N} w_k = 1 \tag{6.2}$$

The **fourth step** was the transmission of the results of the previous three steps to the Community CSP for the next evaluation phase.

### 6.5.2 Applicant phase

The applicant phase is under the responsibility of the Community CSP, which receives the request from the customer. As a *Applicant CSP* (Fig. 6.1) it sends a request to the Broker containing all the weighted criteria in the form of *JSON Request Data Set*.

### 6.5.3 Brokerage phase

This phase allows the entire Community CSPs to evaluate which is the best Community CSP, able to satisfy the request both quantitatively and in terms of quality. The Broker transmits the request received from the Applicant CSP to the Community CSPs, which, in turn, will be able to present one or more offers in terms of quality of the Bidders CSPs. Therefore, the Broker: i) collects the offers for each request, ii) evaluates them by calculating a *score* through the Formula (6.3), iii) classifies the offers based on the scores archived, iv) share the rank with all CSPs in the Community.

$$score = F_w(w, g, N) = \frac{\sum_{k=1}^{N}(w_k * g_k)}{N} \tag{6.3}$$

More specifically, the score is calculated by multiplying the weight $w$ and the evaluation *grade (g)* assigned to each *k-th* criterion and normalizing it to a *zero-to-one* ranking. The brokerage phase takes place dynamically at the Broker, thus allowing it to provide almost real-time rankings for each request.

### 6.5.4 Criteria

#### The Operational Availability ($A_0$) Criterion

Operational Availability is the percent of the time the IoT-Cloud equipment is available for use; that is, it works when it is required. Essentially, the $A_0$ represents the *uptime* of the offered service by the CSP and considers the effect of *reliability, maintainability*, and of the *Mean Logistics Delay Time (MLDT)*. It may be calculated by dividing the *Mean Time Between Maintenance (MTBM)* by the sum of the *MTBM*, the *Mean Maintenance Time (MMT)*, and the *MLDT* as follows:

$$A_o = \frac{MTBM}{MTBM + MMT + MLDT} \tag{6.4}$$

#### The Storage Capacity Service Price ($S_{price}$) Criterion

Monetary cost is a quantifiable criterion that is aimed at customers and organizations in their business. Usually, a company that must control the production cycle of a plant through a *smart objects network* requires an

IoT-Cloud service that considers required number of messages to manage (i.e., millions of messages) and its service price generally expressed in $\$/M$ (i.e., dollars-per-million of messages). In particular, O&G company needs the management of several TeraBytes (TB)-per-year of data due to the complexity and magnitude of its plants in general. Therefore, it should make sure of CSPs ensure adequate *storage capacity*, also with a supportable monetary cost. For example, *Amazon Web Services (AWS)* has created IoT specific services (i.e., AWS Greengrass and AWS IoT) based on the above specifications. On the other hand, Bidder CSPs generally offer a "customized volume pricing", which takes into account both the quantity and the type of data to be stored and managed.

With specific reference to IoT services, the Storage Capacity Service Price (labeled $S_{price}$) is generally expressed in $\$/TB$ (i.e., dollars-per-Tera Byte). A customer (e.g., O&G company) who wishes to quantify the basic needs in Tera Bytes, before making a service request to a Community Cloud member (e.g., via the Web interface), should consider the following Formula:

$$\begin{aligned} TB &= F_b(Obj_{num}, M, p_{size}, \Delta_t) = \\ &= Obj_{num} * M * p_{size} * \Delta_t \end{aligned} \tag{6.5}$$

where $Obj_{num}$ is the number of smart objects providing data through the Internet. $M$ is the number of messages-per-hour to manage. The $p_{size}$ parameter indicates the payload size for each message (i.e., byte-per-message). For example, the use of the *MQ Telemetry Transport (MQTT) protocol* [7] by several "top" CSPs (e.g., AWS) in their offered IoT services results *512 byte* is the maximum payload size for each message. The *running time* $\Delta_t$ can be generally expressed as a function of the service start and stop dates, and of the number of years $yy$, months $mm$, days $dd$ and hours $hh$ in the service "start-stop" time period, as follows:

$$\Delta_t = F_t(date_{start}, date_{stop}, yy, mm, dd, hh) \tag{6.6}$$

A yearly (i.e., by considering 365 days) *full-time* maintenance at a capable destination results $\Delta_t$ equals 8760 hours.

Once a customer knows its basic TB necessity $B$, it can make its request.

### The Data Analytics Service Price ($A_{price}$) Criterion

Data Analytics is the "added value" to the Storage Capacity: as a service, it processes raw data and converting it into information useful for decision-

making by customers. Bidder CSPs generally offer a "customized volume pricing" ($/TB), which takes into account both the amount and the typology of data to process by using a specific software framework (e.g., the Hadoop MapReduce).

#### The Cybersecurity Level ($C_l$) Criterion

The growing threat of increasingly sophisticated cyberattacks (i.e., cybercrime) threatens the development of safe Information and Communication Technologies globally. This is a problem of the increasing use of IoT, especially in complex environments such as O&G plants. Safeguarding IoT-Cloud provides for a reliable environment critical for organizations and individuals to conduct business and freely communicate. Based on the above considerations, the proposed strategy includes the Cybersecurity Level as criterion, by referring to the geographical *Global Cybersecurity Index (GCI)* annual report by both ITU and ABI Research [14].

#### The Support Level ($S_l$)Criterion

Support Level is indicative of the CSP's *problem solving capability*, that is how much time (hours) the CSP needs to solve a problem communicated by the customer, once the support terms specified in the service agreements are activated.

## 6.6 Case Of Study

To demonstrate the goodness of the proposed MCDM strategy, the study considers a simulated IoT-Cloud brokerage scenario using the *J2CBROKER* tool [60] on a *Virtual Machine* equipped with *Ubuntu Server 14.04* and hosted in a *IBM BladeCenter LS21* at the *Cloud Laboratory Data-center - University of Messina*.

J2CBROKER is mainly designed on a JAVA client-server architecture, which models both the IoT-Cloud service requests and offers (i.e., *Data Sets*) through JSON documents. It executes the calculations introduced in Section 6.5 and shows the results (i.e., the Community Bidder CSPs offers) in the form of rankings.

| **Scenarios** | | | | | |
|---|---|---|---|---|---|
| **Weights →** | $\mathbf{w}_1$ | $\mathbf{w}_2$ | $\mathbf{w}_3$ | $\mathbf{w}_4$ | $\mathbf{w}_5$ |
| **Scenario 1 →** | **0.2** | **0.2** | **0.2** | **0.2** | **0.2** |
| **Scenario 2 →** | **0.1** | **0.1** | **0.1** | **0.6** | **0.1** |
| **Scenario 3 →** | **0.1** | **0.4** | **0.3** | **0.1** | **0.1** |

| **Requests (Eq.(6.5))** | | | |
|---|---|---|---|
| **Smart Objects ($Obj_{num}$)** | **M [mess/h]** | **$p_{size}$ [B]** | $\Delta_t$ [hours] |
| 100, 500, 1000 | 720, 1800 | 500 | 8760 |

| **Offers** | |
|---|---|
| **Bidder CSPs$_{num}$** | **Offers$_{num}$** |
| 3÷5 | 5÷10 |

Table 6.1: Simulation-based Scenarios [61].

### 6.6.1 Simulation Environment

As reported in Table 6.1: i) the study modeled three scenarios based on different weights distributions for the selected five criteria and various typologies of requests; ii) each request includes the number of smart objects $Obj_{num}$, the message-per-hour $M$ to manage, the payload size $p_{size}$ and the the service start-stop time period $\Delta_t$; iii) the Bidder CSPs ranges from 3 to 5 in number and it is able to present offers ranges from 5 to 10 in number. Tab. 6.2 shows the selected range and grade for each MCDM criterion.

### 6.6.2 Results

Table 6.3 shows a ranking of samples resulting from the *Scenario 2* (Table 6.1). The ranking includes the offers matching the *request* for 500 *Smart Objects* which produce 1800 message-per-hour with a $p_{size}$ of 500 Byte. The "best offer" is the number 4 by the CSP number 1 at a Cloud site, which is located in the United States (US). The "total" monetary cost for the proposed IoT-Cloud service is 32 k€/TB due to the sum of the $S_{price}$ and the $A_{price}$, i.e., 114.56 k€, to manage 3.58 TB in a period of 8760 hours.

Figure 6.2 shows a linear diagram which represents a set of 30 offers referred to the Scenario 1. For each offer, it reports the *Score* and the quality index related to the Storage and Data Analytics Service Prices. This

| MCDM Simulation Environment | | | | | |
|---|---|---|---|---|---|
| **Criteria →** | $\mathbf{c_1}$ | $\mathbf{c_2}$ | $\mathbf{c_3}$ | $\mathbf{c_4}$ | $\mathbf{c_5}$ |
| **Labels →** | $\mathbf{A_0}$ | $\mathbf{S}_{price}$ | $\mathbf{A}_{price}$ | $\mathbf{C}_l$ | $\mathbf{S}_l$ |
| **Units →** | **%** | **$/TB** | **$/TB** | **GCI index** | **hh** |
| **grade=1** | 99.95 | 41÷50 k | 5 k | 0.000÷0.199 | 19÷24 |
| **grade=2** | 99.96 | 31÷40 k | 4 k | 0.200÷0.499 | 13÷18 |
| **grade=3** | 99.97 | 21÷30 k | 3 k | 0.500÷0.699 | 5÷12 |
| **grade=4** | 99.98 | 11÷20 k | 2 k | 0.700÷0.749 | 2÷4 |
| **grade=5** | 99.99 | 1÷10 k | 1 k | 0.750÷1.000 | ≤1 |

Table 6.2: Simulation Environment. Selected range and grade for each MCDM Criterion [61].

| Scenario 2; Obj=500; M=1800; $\mathbf{p}_{size}$=500 B; TB=3.58 | | | | | | | |
|---|---|---|---|---|---|---|---|
| **CSP** | **Off** | $\mathbf{A_0}$ | $\mathbf{S}_{price}$ | $\mathbf{A}_{price}$ | $\mathbf{C}_l$ | $\mathbf{S}_l$ | **Score** |
| **id** | **id** | **%** | **$/TB** | **k$/TB** | **GCI** | **hh** | **0÷1** |
| **1** | **4** | 99.97 | 31 k | 1 k | 0.824 US* | 18 | **0.84** |
| **3** | **2** | 99.95 | 35 k | 1 k | 0.735 NO* | 7 | **0.7** |
| **3** | **1** | 99.95 | 31 k | 1 k | 0.706 IN* | 20 | **0.66** |
| **2** | **4** | 99.98 | 22 k | 4 k | 0.559 MA* | 11 | **0.6** |
| **4** | **1** | 99.96 | 39 k | 2 k | 0.500 RU* | 24 | **0.54** |

**US=USA; NO=Norway; IN=India; MA=Morocco; RU=Russian Federation*

Table 6.3: A list of samples resulting from the J2CBROKER simulation [61].

quality index is calculated as follows:

$$Q = F_q(S_{price}, A_{price}) \qquad = 1 - \frac{S_{price} + A_{price}}{S_{priceMax} + A_{priceMax}} \tag{6.7}$$

The graph allows for clear visualization of the best offers. Based on the Score value, the best offer is the number 5 by the CSP 4 at a destination site, which is located in Australia. Based on the Q value, the best offer is the number 1 by the CSP 3 at a destination site, which is located in Canada.

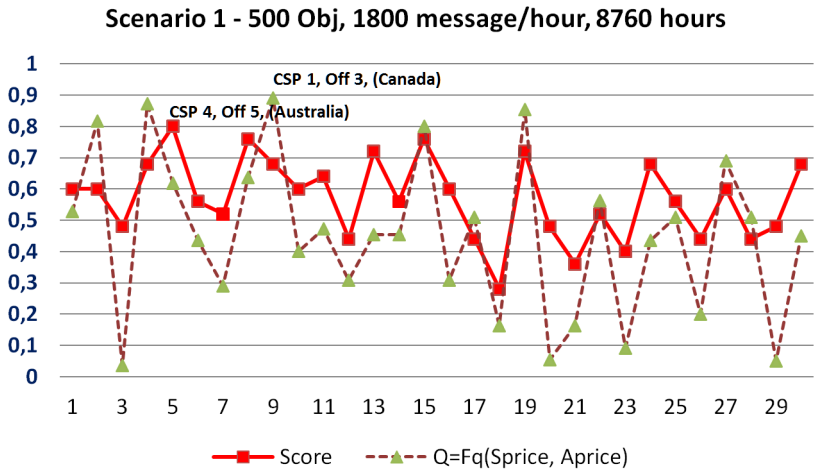


Figure 6.2: Comparison between Score and Q for a set of 30 offers in the Scenario 1 [61].

Considering the same set of offers previously considered, Figure 6.3 reports, for each offer, the related *Score* and the Cybersecurity Level. Based on the Score value, the best offer is the number 2 by the CSP 2 at a destination site, which is located in Canada. Based on the Cybersecurity Level, the best offer is the number 3 by the CSP 1 at a destination site, which is located in the United States (USA).

Figure 6.4 reports the related *Score* and the Cybersecurity Level for 30 offers with reference to the Scenario 2. Based on the Score value, the best offer is the number 1 of the CSP 4 in the USA. The graph shows how the correspondence between the Score and the Cybersecurity lines is more accurate than Scenario 1. This result occurs because the weight requested for the Cybersecurity criterion is the highest one (0.6) among all the other weights of the criterion (0.1).

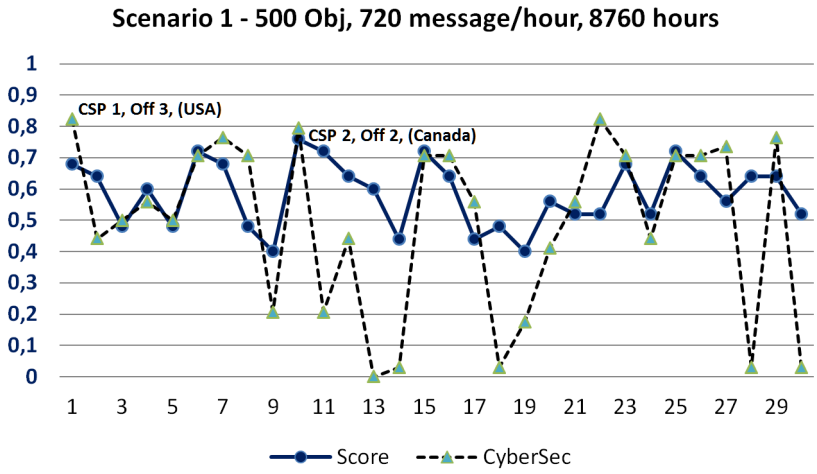


Figure 6.3: Comparison between Score and Cybersecurity Level for 30 offers in the Scenario 1 [61].

Moreover, the evaluation of the presented offers only based on the criterion 4, the Cybersecurity line highlights the two worst offers. Otherwise, weighting the latter through all the heterogeneous criteria, they are not the worst cases.

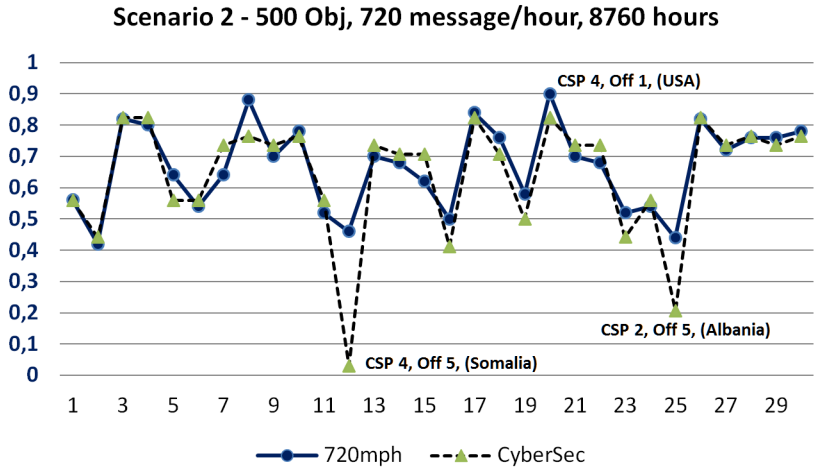


Figure 6.4: Comparison between Score and Cybersecurity Level for 30 offers in the Scenario 2 [61].

# Chapter 7

# An Energy-Aware Brokering Algorithm To Improve Sustainability In Community Cloud

## 7.1 Brief Introduction to the Problem

Today, companies around the world are increasingly sensitive to the environmental sustainability issue. Their products and services are empowering customers, both people and organizations, to satisfy their requests in different contexts, where improving efficiency and reducing pollution are two essential goals. Several companies operate in *Information and Communication Technology (ICT)*. For example, the 2015 Global 100 Most Sustainable Corporations in the World index [13] reports *Accenture (Ireland)* is the first in *IT Services* (54th overall position). Meanwhile, the ranking reports *Nokia (Finland), Lenovo Group (China), and EMC (United States)* are the most sustainable companies in *Technology, Hardware, Storage & Peripherals*.

Community Cloud is an emerging topic in ICT. It consists of a deployment model in which Cloud infrastructure is built and provisioned to be used by a specific community of consumers with shared concerns, goals, and interests [101]. Community Cloud can be owned and managed by its members, by a third party, or a combination of both. In particular, the benefits, costs (i.e., money) and responsibilities (e.g., sustainability) are shared among the

Community Cloud members.

A network of Cloud providers can provide the deployment environment to meet specific community requirements and conditions. Cloud providers can be interconnected according to open standards to offer a universal decentralized Cloud computing environment.

## 7.2 The Goal of the Study

This study addresses medium and small Cloud providers towards solutions allowing them to compete with large Cloud providers in a more sustainable service marketplace. This study refers to a dynamic scenario in which Cloud providers share their IT resources among their respective Community Cloud sites (i.e., data centers) intending to reduce costs and energy-efficiency gap compared to the *top* Cloud computing service providers (e.g., Amazon, Google, Rackspace, etc.). An automated *Service Level Agreement (SLA)* negotiation process for customers (i.e., multiple Cloud service providers in such dynamic critical service-based scenario) facilitates bilateral negotiation between **Community Cloud broker** and various providers to achieve the different goals for the community members. However, for these purposes, balancing the above goals with performance is a challenge. To this end, a Cloud brokerage approach can simplify the procedures in making the best choice.

A brokerage scenario is showed in Fig. 7.1: an *Authority in charge* evaluates both the *Organization* and the *Cloud service provider (CSP)* requirements (e.g., SLA) to determine if they are satisfactory to become *Community members*. The same for a candidate broker, who consequently can evaluate *service requirements* and classify the offers among those presented by the members of the Community.

For example, *service requirements* may be based on the fact that the use of electricity on a Cloud site can change at certain times of the day and specific periods of the year, even different by geographic area and energy source. In addition, a Cloud site can use a solar energy source, and the efficiency of its Photo-Voltaic system (PV) can change at different moments of the day (i.e., morning, afternoon, evening, night), and to varying periods of the year (i.e., spring, summer, autumn, winter).

A sustainable approach, such as the one presented in this Chapter, can help Cloud providers to receive funds to build new *green* plants, therefore to

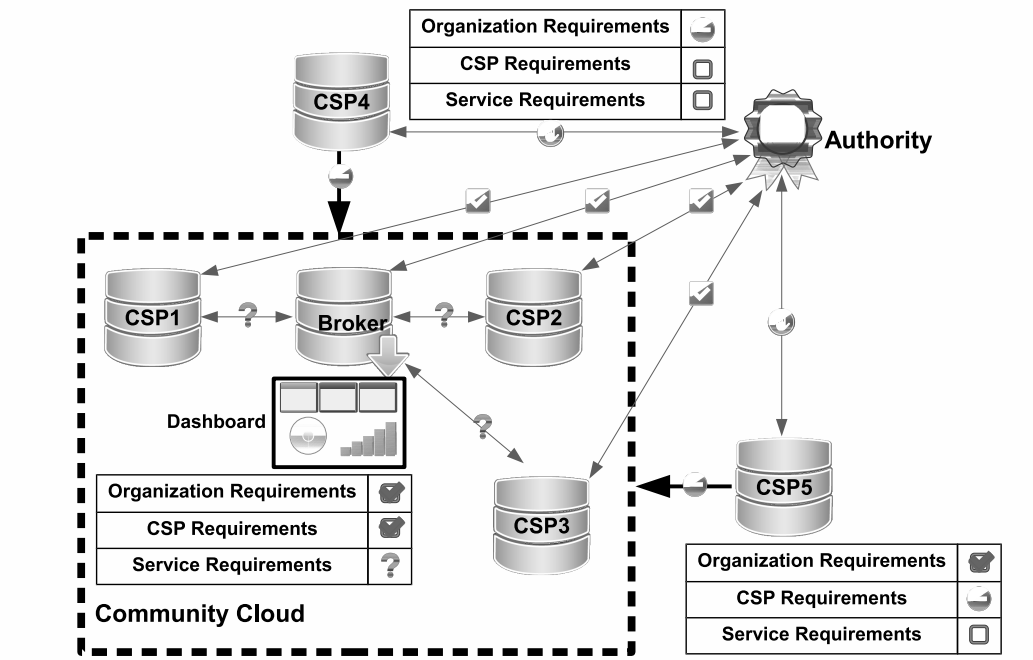


Figure 7.1: Brokerage in a Community Cloud environment [63].

produce clean energy and to receive *Renewable Energy Certificates (RECs)*.

The proposed approach is also intended for companies that want to approach the Cloud paradigm combining an environmentally sustainable choice with adequate levels in performance (e.g., availability), as well as maintaining costs low.

For example, companies that want to prototype new applications such as startup projects that initially require low resources but may need extra processing resources for short periods, with the evolution of the project.

The rest of the Chapter is organized as follows. Section 7.3 discusses related work. Section 7.4 presents the main sustainability metrics that have to be considered in the proposed strategy. Section 7.5 presents the proposed approach to make the best choice in resource allocation. Section 7.6 presents a simulation environment for the analytic evaluations considering real parameters, thus proving the goodness of proposed approach.

## 7.3 Background of the Study

Currently, most of the energy-aware management strategies are explicitly focus on independent Cloud providers, and less are starting to look at Community Cloud. The Scientific literature presents numerous contributions to the planning of "green" Cloud and intra-Datacenters resource scheduling to reduce energy consumption. Still, less attention has been paid to the "sustainable" community, cooperative, or federated Clouds. This study, instead, presents a decision-making approach, which focuses on the allocation of resources in locations where their use is more sustainably. The Community Cloud model helps its implementation.

In [89], the authors present a multi-objective genetic algorithm, named *MO-GA*, to optimize energy consumption, carbon dioxide emissions, and generated profit of a geographically distributed Cloud computing infrastructure. Differently from the *MO-GA* approach, the study presented in this chapter mainly focuses on *brokerage*, i.e., on *Cloud Brokerage Services*, to achieve a Community Cloud ecosystem which advantages all its members.

In [36], the authors present a review of literature on Cloud Brokerage Services, but Community Cloud is not considered.

In [77], the authors present a fascinating study to minimize the cost while satisfying Network as a Service (NaaS) and Infrastructure as a Service (IaaS) QoS constraints. Their framework addresses resource allocation according to an end-to-end SLA established between a Cloud service user and several Cloud service providers in a Cloud networking environment. Compared to that study, the proposed study mainly looks at sustainability, taking into account significant parameters by type of service/application.

A survey of the significant contributions dealing with energy sustainability and cost-saving strategies aimed at Cloud computing and the Federation is presented in [62]. The survey helps researchers to identify the future trends of energy management in the Cloud Federation. Its authors present a taxonomy useful to analyze the current state-of-the-art, thus to highlight possible directions for future research efforts.

In [134], the authors present two complementary energy-efficiency optimization approaches, each one of them covered in the scope of the two European *CoolEmAll* and *Eco2Clouds* projects. They describe metrics applied to assess and optimize energy-efficiency, even if in a different environment than Community Clouds. The *Eco2Clouds* project, in particular,

focuses on energy-efficient Cloud-application deployment in federated Cloud-environments, showing as groups of researchers have developed methods, guidelines, and technology to reduce carbon dioxide emissions (i.e., carbon footprint).

In [128], the authors propose a work based on a multi-criteria optimization technique for a better selection of a service provider, by using a Pareto-based approach to decide the Cloud service provider which satisfies the *Quality of Service (QoS)* requirements for the user. However, that work does not cover the dynamic composition of services based on the migration of data.

In [21], a management algorithm is proposed that allows us to decide about the reallocation of Virtual Machines in Cloud contexts characterized by a large number of hosts. Such an algorithm simply identifies some critical instances and makes decisions without resorting to typical thresholds.

## 7.4 Sustainability Metrics

The study considers several sustainability metrics, which are generally computed based on real-time monitoring of the consumptions of the electrical loads in each *Data Center* (DC).

### 7.4.1 The Power Usage Effectiveness Metric

The *Power Usage Effectiveness (PUE)* is a sustainability metric recommended by the *Green Grid consortium* to characterize the DC infrastructure efficiency. PUE is generally defined as follows:

$$PUE = \frac{P_{DC}}{P_{IT}} \tag{7.1}$$

PUE indicates how much the internal power consumption $P_{DC}$ of a DC exceeds the Information Technology power consumption $P_{IT}$ at the same DC, mainly due to electrical equipments and cooling systems. It is one of the four sub-metrics useful to compute the *Data center Performance Per Energy*.

### 7.4.2 The Data center Performance per Energy Metric

The *Data center Performance Per Energy (DPPE)* is a sustainability metric introduced by the *Japan's Green IT Promotion Council* in order to improve

on the PUE. DPPE is defined by the following Formula (7.2):

$$DPPE = ITEU * ITEE * 1/PUE * 1/(1 - GEC) \tag{7.2}$$

and it is essentially based on four sub-metrics:

- the *Information Technology Equipment Utilization* ($ITEU$);
- the Information Technology Equipment Efficiency ($ITEE$);
- the Power Usage Effectiveness metric ($PUE$);
- the Green Energy Coefficient ($GEC$).

The $DPPE$ is defined in such a way that a higher value in DPPE indicates a greater energy efficiency [11].

In the following, the study assumes that each Cloud service provider dynamically computes the $DPPE$ in each site measuring in real-time the relative sub-metrics, without exposing functionalities deemed too sensitive or risky for its own business. This value is indicated for $n$-th site at time $t$ with $DPPE_n(t)$.

### 7.4.3 The Carbon Dioxide Intensity Of Electricity Factor

The *Carbon Dioxide Intensity Of Electricity* ($CDIE$) is a measure of the quantity of carbon dioxide emitted by an IT infrastructure concerning the used energy, and it is measured in $kgCO_2/kWh$. It depends on the region where the Cloud site is located, and it is based on the government's published data for that region of operation for that year. In particular, the study refers to the *Intergovernmental Panel on Climate Change* ($IPCC$) database. IPCC is the leading international body for the assessment of climate change [12]. Since $CDIE$ changes year to year, it is a time-dependent quantity; thus it denotes the admitted quantity into the generic cloud site $n$ at time $t$ with $CDIE_n(t)$.

The proposed strategy foresees this factor takes into account the impact of operational carbon usage.

### 7.4.4 The Sustainability Impact Factor

Based on the definitions above, the study introduces the *sustainability impact factor* of site $n$ at time $t$ as the ratio of the above-mentioned $CDIE$ and $DPPE$ metrics:

$$k_n(t) = \frac{CDIE_n(t)}{DPPE_n(t)} \tag{7.3}$$

It is expressed in *KgCO2/kWh*. If it is multiplied for the energy consumption (kWh) resulting from running a service at the related $n$-th site, it represents the "weight" in terms of carbon dioxide emission (kgCO2), i.e., the *workload footprint*, which is correlated with that energy consumption. The higher it is, the greater the pollution due to run that service. The values of $CDIE$ are published yearly and that the $DPPE$ for a given site changes only when structural modifications are done. Due to these reasons, both of them could be considered constant in time, depending on the period analyzed. In such case the value of sustainability impact factor is written as:

$$k_i = \frac{CDIE_i}{DPPE_i} \tag{7.4}$$

assuming it constant over the time.

## 7.5 Energy-aware Resource Allocation Approach

This section introduces a new approach to make the best choice in *resource allocation* to push down environmental pollution. The section mainly refers to reducing carbon dioxide emissions through the Community Cloud ecosystem, running a specific instance workload at the most convenient DCs, thus contemporary taking into account sustainability, availability, and monetary cost criteria.

More specifically, the study starts from considering resource allocation in terms of *instance* $i$ at a node $n$, and the related workload $w_{n,i}$. In this context, an instance is a temporary virtual server that needs to be allocated to run services. The instance is distinguished from the classical static virtual server due to its dynamism: an instance that is allocated on a specific node can be easily moved to other nodes, thus to be better managed according to real needs. The workload is expressed in terms of power consumption (kW) needed to run a particular instance.

### 7.5.1 The Availability Criterion

Since the study wants to develop a method to quantify how the workload submitted to a Community Cloud impacts on its sustainability, it consider the availability of DCs to take into account operating periods during which systems can produce pollution.

Availability is the degree at which a system, product, or component is operational and accessible when required for use. The product quality model defined in ISO/IEC 25010 [9] comprises availability as a quality characteristic. Moreover it is an important *Key Performance Indicator (KPI)* generally computed as a function of the total service time, the Mean Time Between Failure (MTBF), and the Mean Time to Repair (MTTR); it is well known that the availability at stable conditions is given by the following equations:

$$av = (MTBF/(MTBF + MTTR)) * 100 \tag{7.5}$$

when expressed as percentage quota. The physical interpretation of availability is the percentage of time during which a system correctly operates. The study will assume that during some not operational periods, a computational node does not produce any useful work, but it could waste power to perform other kinds of maintenance activities.

### 7.5.2 The Service Price Criterion

Service price is a quantifiable criterion that addresses customers and organizations in their business. Generally, it is expressed in $\$/h$ (i.e., dollars-per-hour) or $\$/GB$ (i.e., dollars-per-GigaByte). Starting by identifying several profiles of service requests, for example in terms of required running time $\Delta_r$ of an instance $i$ at a node $n$, it is possible to determine the total cost for that service as follows:

$$cost_{n,i} = \int_{t_{start}}^{t_{start}+\Delta_r} service\_price_{n,i}(t)\, dt \tag{7.6}$$

Usually providers offer instance placement services with a fixed price in the maintenance time. Therefore, eq. (7.7) becomes:

$$cost_{n,i} = service\_price_{n,i} \cdot \Delta_r \tag{7.7}$$

### 7.5.3 Analytic Aspects in a Cloud-to-Cloud Comparison for an Eco-Sustainable Community Cloud Environment

The study assumes that an instance workload has to be moved from a Cloud source node $a$ to a Cloud destination node $b$ in the Community to have management focusing on both sustainability, reducing the carbon footprint ($kgCO_2$), and cost-saving goals.

More specifically, an instance uses electricity to run at any node, and this power consumption generally changes from a node to another due to different technological choices. Moreover, each node can change the power consumption distribution in time. As a consequence, the carbon footprint differs at each node. Therefore, the study mainly distinguishes two different phases during which an instance can be managed, i.e., the **running** at a specific node and the **migration** from the source to a possible destination.

1. *Running phase*: To evaluate the carbon footprint an execution instance $i$ has at a generic Cloud node $n$ at time $t$ for a $\delta$ long period is used the following function:

$$F_r(n, i, t, \delta) = \\ k_n * \int_t^{t+\delta} (av_N * w_{n,i}(\tau) + (1 - av_n) * p_n)\, d\tau \tag{7.8}$$

where $k_n$ is the sustainability impact factor at Cloud node $n$, $w_{n,i}(t)$ is the power consumption to run the $i$-th instance workload at Cloud node $v$. The availability $av_n$ at Cloud node $n$ is used to take into account the real usage of the infrastructure when the instance $i$ runs at that node. The 'idle' condition at the same node, instead, is taken into account through the $p_n$ basic power consumption factor. Based on eq. (7.8), the carbon footprint of a given load $l$ when it runs on Cloud source node $a$ at time $t_a$ for $\Delta_r$ time instants is:

$$co2_{a,l} = F_r(a, l, t_a, \Delta_r) \tag{7.9}$$

Generally speaking, when two different Cloud nodes $a$ and $b$ are considered, their footprints over the same time interval are different ($co2_{a,l} \neq co2_{b,l}$) because both their sustainability impact factor and availability

are different; on the contrary, if the two Cloud nodes $a$ and $b$ are in the same bladecenter (or datacenter) they are characterized by similar sustainability impact factors and availability ($k_a \approx k_b$ and $av_a = av_b$), resulting in the same footprint ($co2_{a,l} = co2_{b,l}$).

2. *Migration phase*; To characterize the carbon footprint to move an instance $i$ from a Cloud node $c_1$ to a Cloud node $c_2$ within a time $\delta$, is used the following function:

$$F_m(c_1, c_2, i, t, \delta) = k_{c_1} * \int_t^{t+\delta} w_{c_1,i}(\tau)d\tau + k_{c_2} * \int_t^{t+\delta} w_{c_2,i}(\tau)d\tau \quad (7.10)$$

$F_m()$ takes into account the fact that during the migration phase two copies of the instances exist in the source and in the destination node thus the footprint is affected by the power consumption of both of them.

The carbon footprint to move load $l$ from a Cloud node $a$ to a Cloud node $b$ within a time $\Delta_m$ starting at $t_m$, is thus computed as:

$$co2_{a\to b,l} = F_m(a, b, l, t_m, \Delta_m) \quad (7.11)$$

In the proposed Community Cloud ecosystem, the formulas above are used to characterize resources of two providers through their footprints with respect the instance $i$ under examination in order to determine the best sustainability. Footprints of both source and destination nodes are computed and they are exploited to choose whether it is convenient to run the instance $i$ on the infrastructure of the original service provider, the source, or at destination. The carbon footprint due to the running and migration phases are:

$$co2_{source,i} = co2_{a,i} + co2_{a\to b,i} \quad (7.12)$$

$$co2_{dest,i} = co2_{a\to b,i} + co2_{b,i} \quad (7.13)$$

When $\Delta_m \ll \Delta_r$, eq. (7.12) and (7.13) simplify into the following:

$$co2_{source,i} \cong co2_{a,i} \quad (7.14)$$

$$co2_{dest,i} \cong co2_{b,i} \tag{7.15}$$

Therefore, it is possibile to compare the $co2_{source,i}$ with all the possible $co2_{dest,i}$, to determine what is the best carbon footprint choice. However, when Quality of Service (QoS) requires to take into account how much time is needed to migrate a service through one or more critical parameter (e.g., latency), $\Delta_m$ can not be negligible, and must be consider the equations (7.12) (7.13) for a comparison.

### 7.5.4 The Algorithm

This paragraph presents a proposal for a decision-making algorithm, named *energy-aware Brokering Algorithm* (*eBA*), in the perspective of an energy-aware instance allocation in a centralized brokerage Community Cloud. The decision making process is detailed through the algorithms 1 and 2 using pseudo-code where were used the symbol $h$ instead $\Delta_r$ to simplify the notation.

Line to line descriptions are provided in Table 7.1 and 7.2. The entire process, which includes the algorithm, mainly consists of three roles (*applicant (A), offering (O)* and *brokering (B)*) executed in a distributed way on Cloud provider infrastructures and the Broker node.

**Role A** When a Cloud provider receives some instances to run, it generates a request (REQ) for computation and cooperates with the Community to determine a set of best offers (OFFs) where the load should run. It is done as follows:

- A.1 Determines the application/service to be supported by Cloud resources in terms of typology and requirements;
- A.2 Asks the Broker for OFFs to allocate several instances for that service among Community Cloud providers;
- A.3 Formulates its REQ including the data related to A.1 in a recognized format;
- A.4 Sends REQ to the Broker;
- A.5 Receives a set of best offers OFFs from the Broker;
- A.6 Chooses the best OFF among the set.

```
1: nosql_db = newNoSQL()
2: use nosql_db
3: define nosql_db.REQs_collection
4: define nosql_db.OFFs_collection
5: define nosql_db.reqsTags
6: define nosql_db.resulting
7: define reqsTags = nosql_db.REQs_collection.tags( )
8: define hMapReq = nosql_db.REQs_collection.gather(reqsTags.h)
9: define hMapOff = nosql_db.OFFs_collection.gather(reqsTags.h)
10: while true do
11:    define reqs_status = hMapReq.trigger( )
12:    define offs_status = hMapOff.trigger( )
13:    if (reqs_status is true) then
14:       hMapReq = hMapReq.update( )
15:    else
16:       if (offs_status is true) then
17:          hMapOff = hMapOff.update( )
18:       end if
19:    end if
20:    nosql_db.resulting = hMapOff.calc()
21:    nosql_db.resulting.find( )
22:    nosql_db.resulting.sort(co2_footprint, cost, opt, N)
23: end while
```

**Algorithm 1:** The energy-aware Brokering Algorithm (eBA) [63].

1: $define\ h\_num = hMapOff.count(reqsTags.h)$
2: $define\ worst\_cf[h\_num]$
3: $define\ worst\_cost[h\_num]$
4: **for** $j = 1$ to $h\_num$ **do**
5: $off\_num = hMapOff.count(reqsTags.h.j)$
6: $define\ co2\_footprint[h\_num][off\_num]$
7: $define\ cost[h\_num][off\_num]$
8: $define\ opt[h\_num][off\_num]$
9: **for** $i = 1$ to $off\_num$ **do**
10: $define\ av,\ price,\ iteu,\ itee,\ pue,\ gec,\ cdie$
11: $define\ w,\ p_{basic},\ t0,\ h,\ dppe,\ k,\ a,\ N$
12: $av = hMapOff.j.i.availability$
13: $price = hMapOff.j.i.price$
14: $iteu = hMapOff.j.i.iteu$
15: $itee = hMapOff.j.i.itee$
16: $pue = hMapOff.j.i.pue$
17: $gec = hMapOff.j.i.gec$
18: $cdie = hMapOff.j.i.cdie$
19: $w = hMapOff.j.i.workload$
20: $p_{basic} = hMapOff.j.i.p_{basic}$
21: $t0 = hMapOff.j.i.tstart$
22: $h = hMapOff.j.i.h$
23: $dppe = dppe\_func(iteu, itee, pue, gec)$
24: $k = k\_func(cdie, dppe)$
25: $co2\_footprint[j][i] = integral(t0, h, w, k, av, p_{basic})$
26: $cost[j][i] = cost\_func(price, h)$
27: **end for**
28: $worst\_cf[j] = max(co2\_footprint, j)$
29: $worst\_cost[j] = max(cost, j)$
30: **for** $i = 1$ to $off\_num$ **do**
31: $opt[j][i] =$
$a * (co2\_footprint_{j,i}/worst\_cf_j) + (a - 1) * (cost_{j,i}/worst\_cost_j)$
32: $hMapOff.j.i.update(co2\_footprint, cost, opt)$
33: **end for**
34: **end for**

**Algorithm 2:** The *calc*() method of the energy-aware Brokering Algorithm (eBA) [63].

**Role O** As a consequence of a Broker REQ, each Community Cloud provider creates an OFF based on its sustainability factors and sends it to the coordinating Broker as follows:

- O.1 Receives REQ from the Broker;
- O.2 Computes the value for each service parameter reported in REQ at each Cloud site;
- O.3 Computes the value for each sustainability parameter at each Cloud site;
- O.4 Formulates its OFF in a recognized format;
- O.5 Sends its OFF to the Broker.
- O.6 If its OFF is the best for that REQ, it is contacted by the corresponding applicant.

| **Algorithm 1** | | | |
|---|---|---|---|
| **Line** | **Description** | **Line** | **Description** |
| 1-2 | The eBA uses NoSQL database because relational schemata are hard to change incrementally in order to accommodate new content types without impacting performance. | 10 | true condition for the while loop |
| 3 | *REQs_collection* is the collection of the requests at Broker. | 11 | *trigger*() is a function which returns a true (or false) condition in case of a change (or not) in $hMapReq$. The *reqs_status* denotes related value. |
| 4 | *OFFs_collection* is the collection of the offers at Broker. | 12 | *trigger*() returns a true (or false) condition in case of a change (or not) in $hMapOff$. The $offs\_status$ denotes related value. |
| 5 | *reqsTags* is the collection of tags characterizing requests at Broker. | 13-19 | The $hMapReq$ and $hMapOff$ collections are updated in case of change on related statuses, by using the *update*() function. |
| 6 | resulting is the collection where to save the results calculated by the eBA, and following queries. | 20 | Once $hMapOff$ has been updated, the function *calc*() is used to compute all the metrics on the submitted requests. |
| 7 | The function *tags*() extracts tags from *REQs_collection* to insert in *reqsTags*. | 21 | The $find$() applied to the resulting collection allows to make queries based on $co2\_footprint$, *cost* or the weighted opt values. |
| 8-9 | The *REQs_collection* is gathered according to the $h$ tag in the new $hMapReq$ collection. Likewise, the *OFFs_collection* is gathered in the new $hMapOff$ collection. | 22 | The *sort*() function applied to the resulting collection allows to sort the collection based on $co2\_footprint$, *cost* or the weighted opt values. |

Table 7.1: Line to line description of *eBA* Algorithm 1

| Algorithm 2 | | | |
|---|---|---|---|
| **Line** | **Description** | **Line** | **Description** |
| 1 | Once $hMapOff$ has been updated, $h$ is assigned to the *count*() function thus to determine the number of $h\_num$ groups of collections. | 9-27 | For each $i$-th offer of the $j$-th $h$ group, the eBA calculates the parameters to be processed. |
| 2-3 | The $worst\_cf$ and $worst\_cost$ arrays are defined. They will be fulfilled by computing the respectively functions at line 28 and 29. | 23 | $DPPE$ calculation using eq. (7.2) for each $i$-th offer of the $j$-th $h$ group. |
| 4 | The $h\_num$ value is used as maximum count limit in a FOR loop to calculate results that are distinguished for each one of the $h\_num$ groups of collection. | 24 | Sustainability impact factor $k$ calculation using eq. (7.4) for each $i$-th offer of the $j$-th $h$ group. |
| 5 | For each $j$-th $h$ group, the *count*() function returns the number of offers ($off\_num$) from the $hMapOff$ collection. | 25 | Carbon dioxide footprint calculation using eq. (7.14) for each $i$-th offer of the $j$-th $h$ group. |
| 6 | The $co2\_footprint$ matrix is defined. It will be fulfilled with the carbon dioxide emission values (see line 25) | 26 | Cost calculation using eq. (7.7) for each $i$-th offer of the $j$-th $h$ group. |
| 7 | The *cost* matrix is defined. It will be fulfilled with the cost values (see line 26). | 28-29 | For each $j$-th $h$ group, the maximum costs in terms of carbon footprint and money are computed. |
| 8 | The *opt* matrix is defined. It will be fulfilled by computing the function at line 31. | 30-32 | The *opt* matrix returns weighted values useful for a comparison among the offers for each $h$ group. Each offers is updated with the $co2\_footprint$, *cost* and *opt* values. |

Table 7.2: Line to line description of $eBA$ Algorithm 2

**Role B** : The OFFs sent by the providers are evaluated by the Broker, which can select the best OFFs where an instance has to run in the following way:

- B.1 Receives REQs from the applicant Community Cloud providers;
- B.2 Broadcasts REQs to the whole Community;
- B.3 Receives OFFs from the bidder Community Cloud providers;
- B.4 Computes the **eBA Algorithm** providing results as a set of best OFFs;
- B.5 Communicates results to the applicant Community Cloud providers.

## 7.6 Experiments

In order to evaluate the *eBA* algorithm behavior, the study considered a scenario by using the **J2CBROKER** tool [60] developed at the *University of Messina*.

### 7.6.1 Performance and Sustainability Datasets

This paragraph presents the modeling of both services and Cloud sites, thus to provide *input* data for the proposed *eBA* Algorithm. Each offer is modeled by a $json$ document which includes two main collections (TABLE 7.3): the first one refers to a *Service Dataset*, to specify workload and performance parameters, and the second one to a *Sustainability Dataset* to calculate the $carbonfootprint$ (Algorithm 2, line 25), the *cost* (Algorithm 2, line 26) and the *opt* evaluation index (Algorithm 2, line 31).

The Service Dataset is obtained from a survey on several "top" providers of IT technologies [8], Cloud services, and solutions (e.g., Amazon Web Services (AWS)).

The Sustainability Dataset results from the METI project [10] on characteristics and energy efficiency of several monitored Asian DCs. The simulator selects a random value between the range set for each metric, and each offer is characterized by its sustainability, cost, and availability values.

| **Service Dataset** | |
|---|---|
| **Parameters** | **Values** |
| Workload (watts) | 200-300 |
| Power basic (watts) | 100 |
| Running Time (hours) | 10,24,360,750 |
| Number of Instances in each Offer | 12,14,16,18,20 |
| Number of Instances in each Request | 1,10,20,50 |
| Availability (%) | 99.90-99.99 |
| Service Price ($/h) | 0.007-0.112 |
| **Sustainability Dataset** | |
| **Parameters** | **Values** |
| ITEU | 0.3-0.6 |
| ITEE | 0.1-3.9 |
| PUE | 1.4-2.3 |
| GEC | 0.0-0.003 |
| CDIE (kgCO2/kWh) | (*) |

Table 7.3: Service and Sustainability Datasets [63].

(*) source:https://www.ipcc.ch

### 7.6.2 Simulation Environment

In the J2CBROKER simulator, both the client and server sides use their own mandatory $json$ configuration files, respectively called *client* $json\_conf$ and *server* $json\_conf$, to dynamically set features and behavior during the simulation steps. The first one contains information about the server application and several fields which are used for the dataset simulation phase. The second one contains information about the server-side elaboration phase. If the client/server applications are launched, and they cannot find the JSON configuration files, these last are automatically generated at run-time as empty templates. The client has two modes of operation: the *Random Simulation Mode* (used in this simulation), which allows the random creation of the datasets to send at the server-side broker algorithm for computation, and the *Guided Simulation Mode*, which enables the user to specify the list of the dataset files for the server-side broker algorithm. The communication between client and server is made through the HTTP POST requests exchange. The output of the simulation is a JSON file that contains the results of the elaborations done by the server-side *eBA* Algorithm.

### 7.6.3 Experimental Results

This paragraph reports, in a graphical form, the results produced by the simulations, based on 1000 samples. Figures 7.2 and 7.3 show distinct results for the different running time $h$ (as reported in service dataset) and based on the established parameters (i.e., weight $a$, confidence in terms of percentage and number $N$ of instances to allocate). In particular, the weight $a$ is a value in the [0,1] range and it is part of the *'opt'* Formula at the line 31 of the Algorithm 2. It is used to assign a weight for each offer in terms of sustainability and cost (the sum of the attributed weight equals one). It was chosen a weight parameter $a$ equals 0.5 (that means to assign the same weight for sustainability and cost confidence to achieve a good compromise between them) and a 95% in confidence intervals for the selected $kgCO2/DPPE$ (i.e., the carbon dioxide emission compared with the DPPE expressed by the Formula (7.2)) and *cost* indexes.

In particular, Figure 7.2 shows four scenarios based on the four different number $N$ of instances in each request (Service Dataset in TABLE 7.3, line 11 in the Algorithm 2) and two typologies of graphs (i.e., kgCO2/DPPE vs $h$ and *cost* (expressed in $) vs $h$. The chosen graphical representation

allows the reader for quick visual feedback about the confidence interval of $kgCO2/DPPE$ and *cost* to allocate a different number of $N$ of required instances. The purpose of these graphs is to give a clear indication of the amount of carbon dioxide emission-per-DPPE and the cost (i.e., money) varying the running time $h$ at each Cloud site. For example, if compared with the others through the y-axis reading, the fourth graph on the left shows that the carbon dioxide emission-per-DPPE confidence interval is more restricted. It means that the proposed algorithm encourages the broker to select the 'best' offers in the presence of a high number of instances to allocate for each request. The same by reading the related *cost* graph (on the right).

Furthermore, even if both carbon footprint and cost grow with $N$ and $h$, their relative $kgCO2/DPPE$ and *cost* confidence intervals are below the 67% in the most expensive of all ($N$=50, $h$=750), that is a 23% less in wasteful among the Community Clouds.

Figure 7.3 shows the confidence interval of the *opt* index for one instance allocation.

The values reported in Figure 7.3 are the result of a post-processing phase, by getting as input all the best *opt* values calculated at each run step. Considering that for each run in the simulation, the worst case results in an *opt* index closer to one, the *eBA* Algorithm at Broker can select sets of offers with an *opt* index lower than 0.12, that is very low if compared with the worst case. It means that the algorithm can select sets of offers with an *opt* index closer to zero (the least possible), taking into account not only sustainability but also the service price criterion.

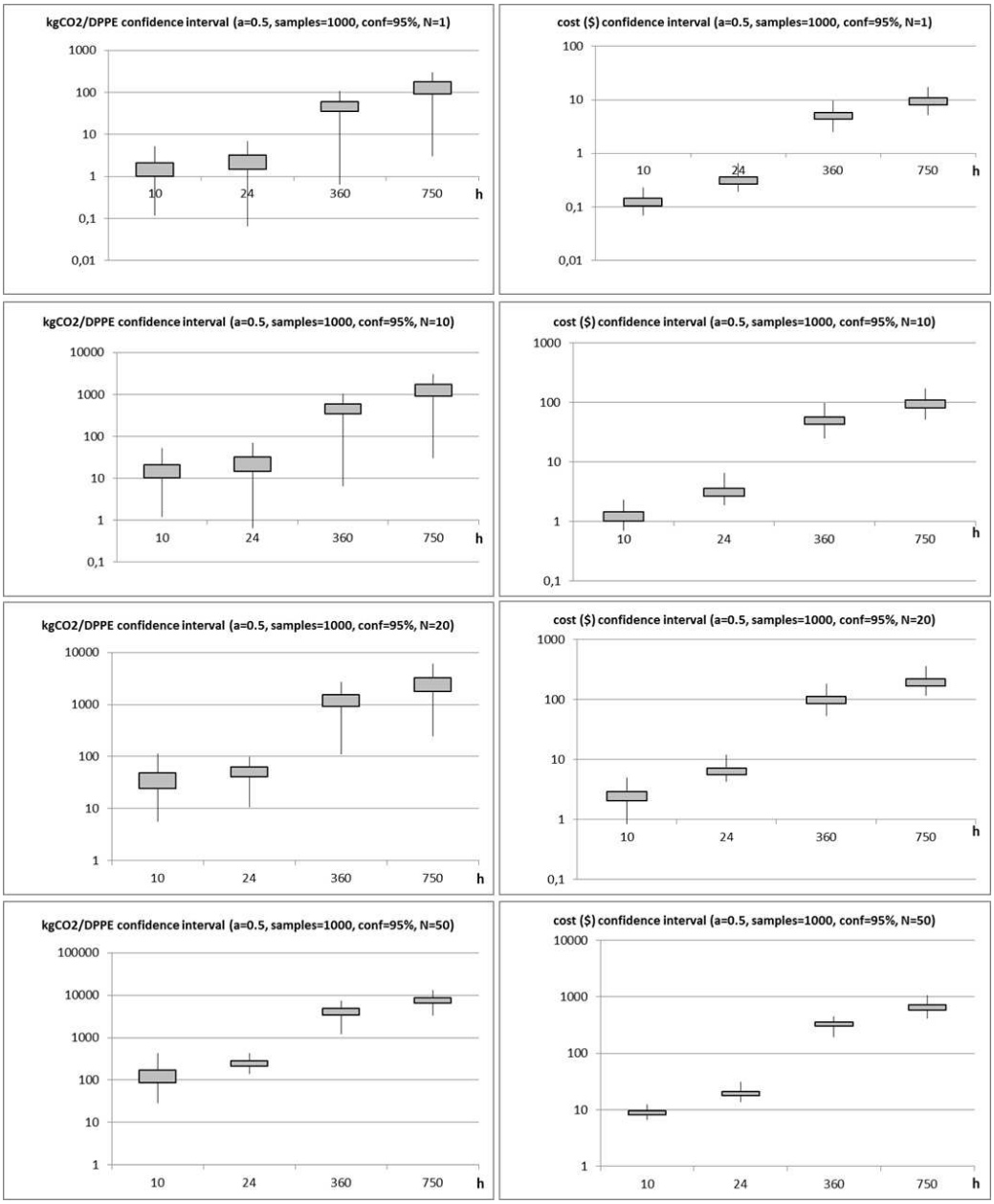


Figure 7.2: Confidence interval of *kgCO2/DPPE* and *cost* for different number of instances to allocate [63].

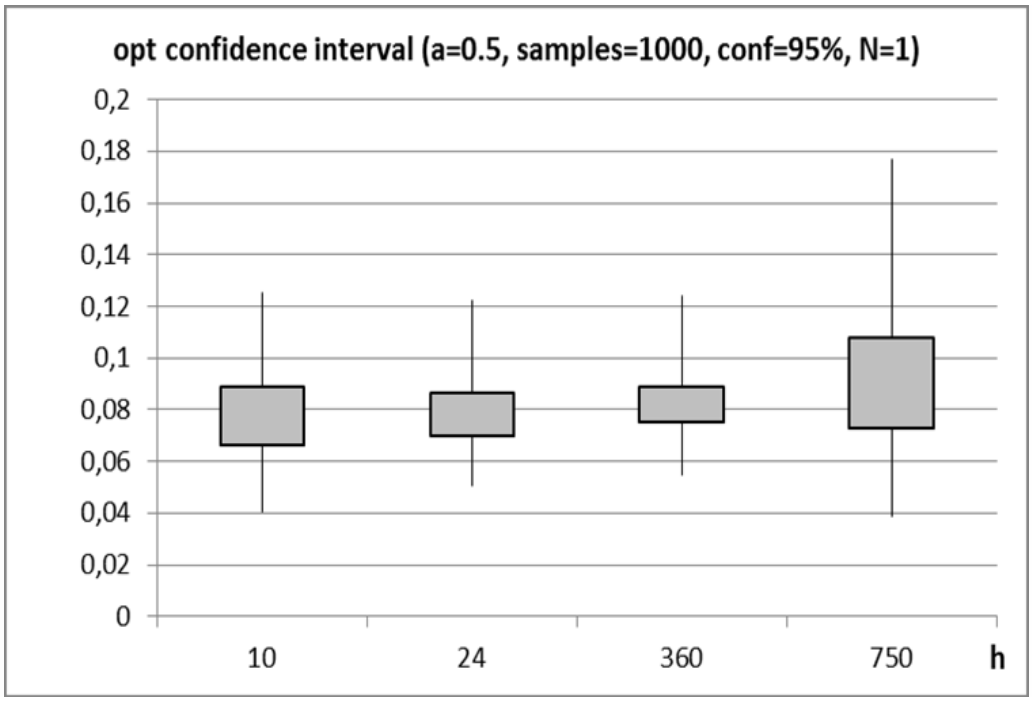


Figure 7.3: Confidence interval of the *opt* index for one instance allocation [63].

# Part III

# Part: How Technology Can Improve Education

# Chapter 8

# VSP: A Cognitive Training Tool in Education

## 8.1 Brief Introduction to the Problem

We live in a society increasingly influenced by technology. Technology is changing the way we live, learn, and work. Computers, cell phones, and smart devices are greatly influencing our lifestyle. Information grows at a high rate, but, at the same time, both its fragmentation and inaccuracy level are growing, making its perception difficult by humans. The use of the Internet has also significantly changed the way we communicate. With a simple *"click"*, it is now possible to get in touch with people who are in different parts of the world, subscribe to social networks or blogs, share news posted by others, create and share new knowledge. These new habits have also influenced forms and language, the way to deal with studies, and the method of approaching the world of knowledge of our time. Furthermore, technological progress also has had a strong influence on politics, market dynamics, educational processes, and more. We have come to a moment in the history of humanity in which man risks losing control over his creativity in favor of existing technology. In other words, technology is not just changing society. It is changing what it means to be human.
Of course, the new generations have a great opportunity to benefit from these changes. The possibility of having all human knowledge available at the *"click"* of a button is something incredible if we make a comparison with the past. Unfortunately, the new generations have revealed difficulties navi-

gating this new world. They appear disoriented. All the problems stem from the fact that the social, economic, and educational systems have not evolved with the same speed as technological progress. When we talk about new generations, we refer to *digital native*. In [112] the author used to describe as *digital native* the generation of people who grew up in the digital age. He motivated this statement by explaining how digital natives are comfortable with technology and computers at an early age and how they consider technology to be an integral and necessary part of their lives. Children and teenagers naturally belong to this category of people. Other people who have simply been *"infected"* by technology are defined as *digital immigrants*.
Today the digital natives are the weak link in this new world that is taking shape on the horizon. From an educational point of view, unlike in the past, natives digital today have free access to all knowledge. But this confuses them, and the school system is still not able to respond to new educational needs. From a sociological point of view, today, digital natives are statistically more and more alone. This situation has its roots in the economic crisis that forces both parents to work away from home. Consequently, teenagers and children are followed less by their family than in the past. Unfortunately, this growing condition of loneliness has negative repercussions on school productivity. The lack of constant feedbacks prevents the construction of the correct study method in the first years of study, and this contributes negatively to the flattening down of the level of preparation of the new generations.
This Chapter proposes a technological solution that aims to overcome these problems.

The remainder of this Chapter is organized as follows. Section 8.2 describes the motivation of this proposal also by introducing the technological choices made. Section 8.3 gives a brief overview of some existing mobile application tools and technology which positively contribute to learning and achievements. The *"Virtual Study Partner"* application and scenario are described in Sect. 8.4.

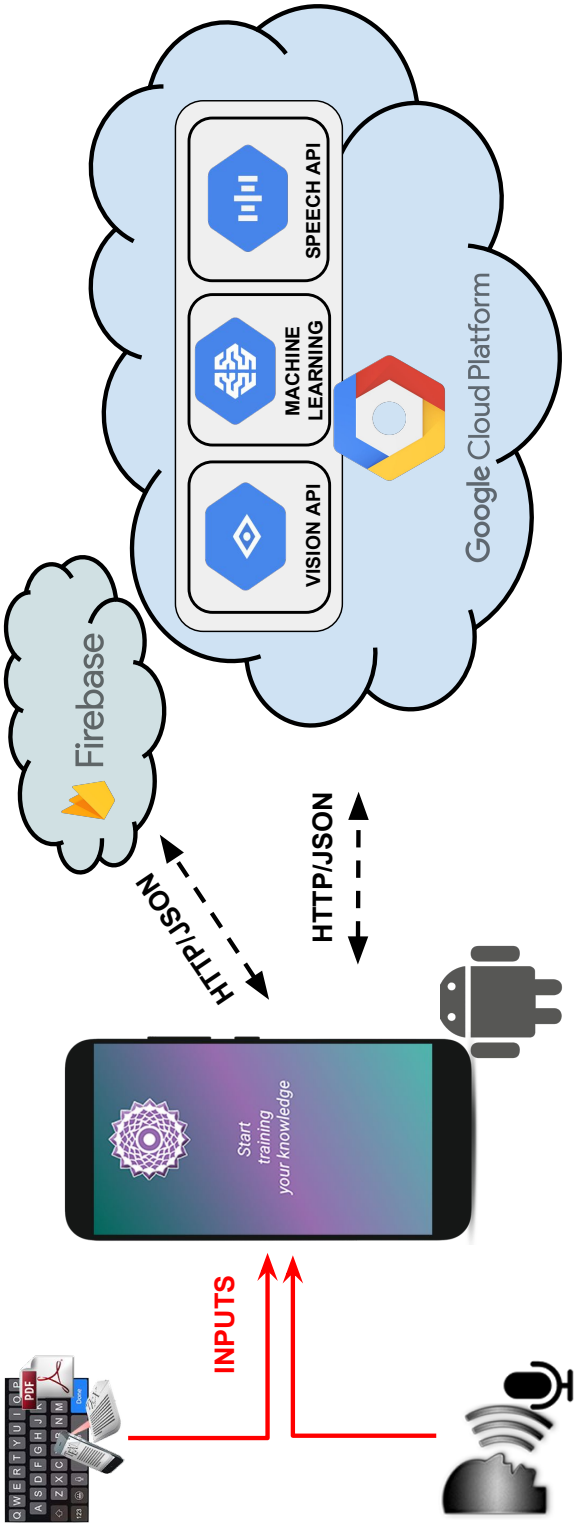


Figure 8.1: A general architecture of the *Virtual Study Partner* scenario [41].

## 8.2 The Goal of the Study

Starting from the premises introduced in Sec. 8.1, it is clear that there is a need to provide adequate support to the school-age children in dealing with the homework activities during the early school years. This research began by focusing on this simple idea. Even in this scenario, technology must play the fundamental role it is playing in today's world. To provide a study support platform to young people in school-age, the first step was the creation of the *"Virtual Study Partner"*. It consists of an Android mobile app that integrates both machine learning concepts and Cloud technologies. This mobile app has to be considered as the first output of a wider research activity carrying out at the *Mobile and Distributed System Laboratory - MDSLab* [98] and at the *High Performance Computing and Application Laboratory* [82] of the University of Messina [127].
The ultimate goal of this research is to create a Learning Platform able to detect and process data obtained from a user's teaching activities during the learning of a theoretical concept taken from a written text and, therefore, to return assessments on acquired skills and predictive analysis. With this aim, the research started with a rigorous analysis of all the technological solutions available on the market that could help achieve the goals. It was decided to focus all the efforts exclusively on Google software products and services because they represent the most mature and consolidated solutions on the market. Paragraph 8.2.1 gives a brief introduction to the technologies used in the reference scenario.

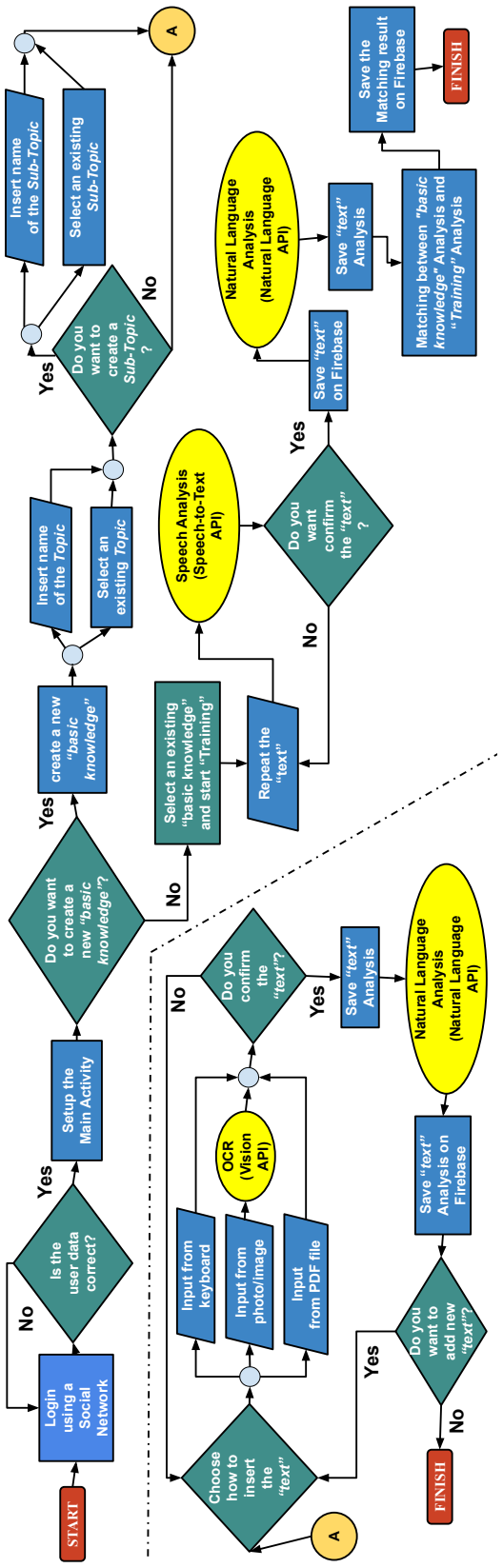


Figure 8.2: Flow-Chart of the *Virtual Study Partner* app [41].

### 8.2.1 Technological choices

**Android** [22] Android is a mobile operating system developed by Google. It is based on a modified version of the Linux kernel and other open-source software and is designed primarily for touchscreen mobile devices such as smartphones and tablets. Android has been the best-selling OS worldwide on smartphones since 2011 and tablets since 2013. As of May 2017, it has over two billion monthly active users, the largest installed base of any operating system, and as of December 2018, the Google Play store features over 2.6 million apps. [23]

**Firebase** [55] is a Web and mobile application development platform developed by Firebase Inc. in 2011 and subsequently acquired by Google in 2014. The success of Firebase is linked to its unique characteristics:

- *data synchronization and storage capacity*, Firebase is capable of update data instantly making it available to web and mobile apps who use it;
- *availability of client libraries*, to integrate Firebase into all the most common ones Web and mobile technologies there are libraries ready to be imported into own projects;
- *REST API*, make the Firebase features available for each technology for which there are no special libraries or in the case of non-operations contemplated in them;
- *Security*, data stored in Firebase is replicated and backed up continuously. Communication with clients is always in encrypted mode using SSL with 2048-bit certificates.

**Google Cloud Platform (GCP)** [67] is a suite of Cloud computing services that runs on the same infrastructure that Google uses internally for its end-user products, such as Google Search and YouTube. Alongside a set of management tools, it provides a series of modular Cloud services, including computing, data storage, data analytics, and machine learning. Among the APIs made available by the platform, in the scenario, were used the following:

- ***Google Cloud Speech-to-Text*** API [68] allows developers to convert audio to text by applying effective neural network models in an easy-to-use API. The API recognizes 120 languages and variants to support

the global user base. It's possible to activate the recognition of controls and voice commands, transcribe the call center audio, and much more. The API can also process live streaming or pre-recorded audio, thanks to Google's machine learning technology.

- ***Google Cloud Vision*** API [68] allows developers to understand the contents of an image by integrating effective machine learning models into an easy-to-use REST API. Quickly rank images into thousands of categories, detect individual objects and faces within images, and read the printed words contained in the images. It's possible to create metadata about the image catalog, moderate offensive content, or enable new marketing scenarios through image sentiment analysis.

- ***Google Cloud Natural Language*** API [38] reveals the structure and meaning of the text through advanced models of pre-trained machine learning in a simple to use REST API and through easy-to-create custom templates with AutoML Natural Language BETA. Thanks to this API, it is also possible:

  – to use Cloud Natural Language to *extract information* about people, places, events and much more mentioned in text documents, articles or blog posts;

  – to use the API to perform *sentiment analysis* on the product in social media or analyze the intent emerged from customer conversations in a call center or a messaging app;

  – to *analyze the text* loaded in your request or integrate it with your document storage space in Google Cloud Storage.

## 8.3 Background of the Study

Several studies have demonstrated that mobile application tools and technology positively contribute to learning and achievement.

In [93] authors state that the smartphone apps thanks to their independence on time and place may serve as good intervention tools for cognitive training of individuals.

In [121], authors state that today's students experience increased anxiety around the school and have difficulty keeping track of course assignments.

The authors conceptually develop and empirically test a model of the impact of a homework reminder mobile app on executive function skills and learning outcomes with undergraduate business students.

In [95], authors sought to understand whether the use of mobile apps had an impact on students' learning of new statistical concepts. They demonstrate that incorporating mobile apps into lectures has the potential to affect student learning positively.

In [86], authors investigated the effects of a mobile app tool for identifying species on biology students' achievement and well-being. They hypothesized that the mobile app, compared to a textbook, would enhance feelings of competence and autonomy and, in turn, intrinsic motivation, positive affect, and achievement because the mobile app' built-in functions provide students with choice and volition, informational feedback, and optimal challenges. Their results indicated that the mobile app produced higher levels of students' perceived competence, perceived autonomy, and intrinsic motivation. Further, they state that the mobile app had indirect effects on positive affect through independence, expertise, and intrinsic motivation, and on achievement through competence.

In [83], the authors studied the students' attitudes towards the use of the book-app instead of a printed book as well as their opinions and suggestions about the content formulation and app's features. At the end of their study, they emphasize the opportunity for students to use book-apps for their course literature and it suggests design and content features that will enhance the usability and students' satisfaction with the book-app.

In [32], the authors provide insights regarding the current use of mobile technologies for learning. Their research indicates that the students are very willing to use mobile learning, and there is a relationship between the use of mobile devices (mainly the use of Tablet) and the use of apps with the global evaluation of mobile learning by students. They bring into light that it is necessary to support and promote the use of these technologies with a curricular and educational purpose by institutions and universities.

## 8.4 Virtual Study Partner APP and Scenario

Fig. 8.1 shows a high-level outline of *"Virtual Study Partner"* scenario, mainly identifying the mobile app, the GCP Services [67] and the Firebase database [55] communicating via HTTP through JSON files.

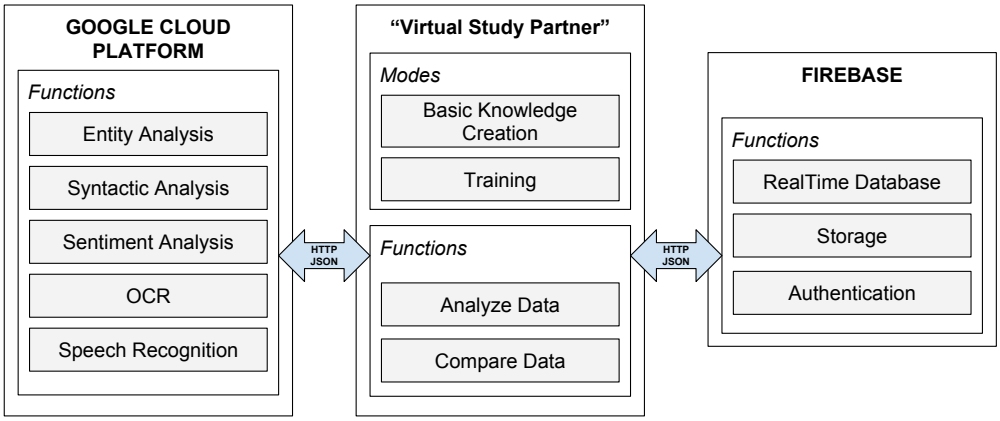


Figure 8.3: *"Virtual Study Partner"* architecture general scheme [41].

To collect as many suggestions, feedbacks, and data as possible, which can help continue the activities, it was decided to publish the Android app on Google Play Store [73] in beta version [133] and it is freely available. Users who want to use it must only register using their Google of their Facebook [53] account. Fig. 8.1 shows the general architectural scheme of the *"Virtual Study Partner"* scenario. Fig. 8.2 shows the Flow-Chart of the *"Virtual Study Partner"* app. As shown in Fig. 8.1, the app offers two modes of use. The initial one, also called *"Basic Knowledge Creation Mode"* and the next one, also called *"Training Mode"*. The description of the two modes follows in Paragraphs 8.4.1 and 8.4.2.

### 8.4.1 Basic Knowledge Creation Mode

In the *Basic Knowledge Creation Mode*, after the login phase (Fig. 8.4), the app asks the user to enter a "textual content". This content will be classified by linking it to a topic and sub-topic during its creation phase. This textual content represents what was called *"basic knowledge"*, that means the reference text on which to carry out all future analyzes and considerations (more details later).
For example, it was decided to create a new *"basic knowledge"* called it *"Alexander the Great"*. We gave it the topic *"history"* (Fig. 8.5 and the sub-topic *"biography"* (Fig. 8.6.

The app allows the input of textual content in three different ways (Fig.

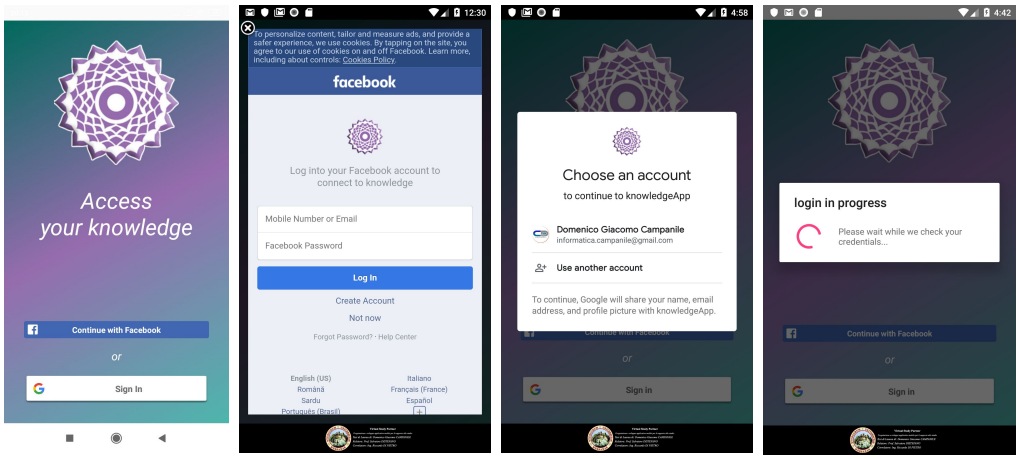


Figure 8.4: *"Virtual Study Partner"*: Login [41].

8.7):

- writing it using the Android O.S. keyboard;
- capturing it from an image (or from a photo) (Fig. 8.7);
- extracting it from a PDF file.

The file from which to extract the text can reside both on the file system of the device or external Cloud storage service.
Once the text is acquired and confirmed by the user, it is sent and processed by Google's Cloud services [69] [38], which extracts text and metrics that are then stored on Firebase [55]. These metrics will be used for future comparisons at the end of each session of *Training Mode* that the user will launch on that specific *"basic knowledge"*. More details about these metrics in Section 8.5.

### 8.4.2 Training Mode

In the *Training Mode*, after the login phase (Fig. 8.4), the app asks the user to *"train"* a specific *"basic knowledge"* which was previously inserted. This training consists of acquiring an audio recording containing the user's oral exposure regarding a specific *"basic knowledge"*.

The app allows acquiring the user's oral exposure using the integration with the Cloud service [68]. The app is, therefore, able to understand the

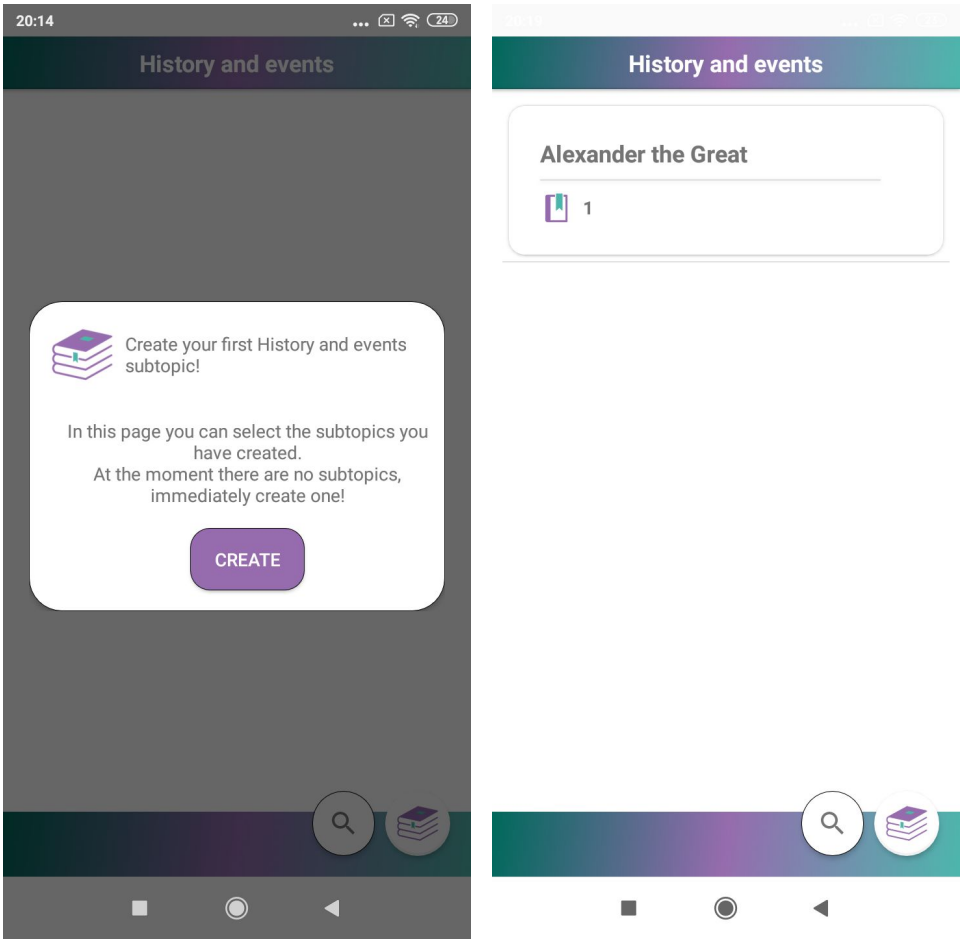


Figure 8.5: ListSubTopics [41].

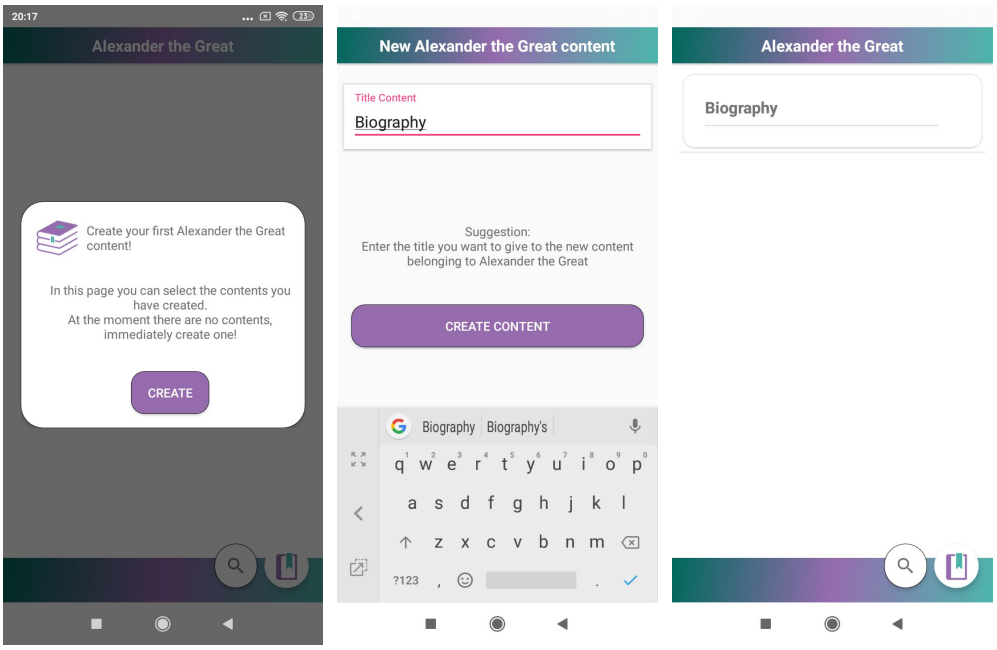


Figure 8.6: ListContents [41].

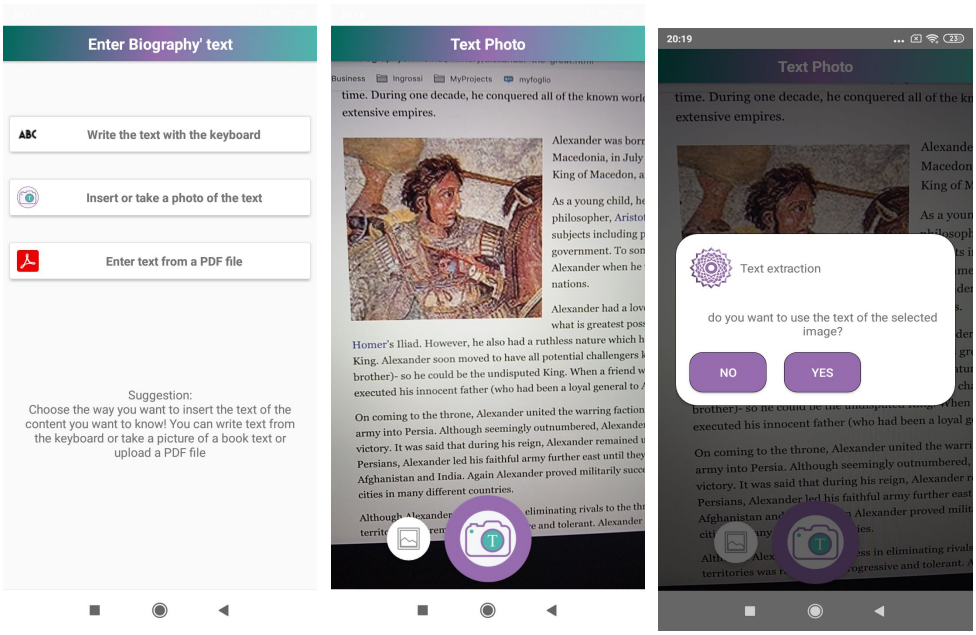


Figure 8.7: *"Virtual Study Partner"*: The input of the textual content for the "Alexander the Great" basic knowledge taken from a photo [41].

natural language of man (Fig. 8.9). It means that the users can repeat their *"basic knowledge"* in a very personal way and not strictly identical to the text they have previously loaded in the system. Thanks to the Cloud service [38], it is possible to analyze the content of the vocal produced by the user. The analysis deals with examining the equivalence, the conformity, the oral exposition, and the same meaning between the text inserted during the *Basic Knowledge Creation Mode* and the text coming from the transcription of the received vowel during the *Training Mode*. The two texts are compared and analyzed structurally, syntactically, and semantically. The system analyzes each part of the speech by detecting the morphology, the dependence on other words present and the taxonomy of the text, to match arguments, concepts, and words present in both texts. At the end of the analysis, the app stores the metrics related to this training session on Firebase [55]. At this point, the app makes a comparison between the text and metrics related to the *"basic knowledge"* considered with the text and metrics of the current training session. Results are both stored on Firebase [55] and displayed to the user on the app (Fig. 8.8) .
The score is expressed in hundredths and in relative percentages which refers to:

- oral exposure;
- equivalence of the texts;
- similarity of the subject dealt with;
- percentage of knowledge of the acquired text;
- time dedicated to oral repetition.

## 8.5 Metrics Details

The Natural Language API [102] returns a JSON file for each type of analysis required. Each type of analysis is, in turn, characterized by specific metrics. Below is a description of the individual metrics broken down by type of analysis. For more details, readers can refer to the official documentation of the Google Cloud Natural Language Service [38].


Figure 8.8: *"Virtual Study Partner"*: Global Learning Statistics for the "Alexander the Great" basic knowledge [41].

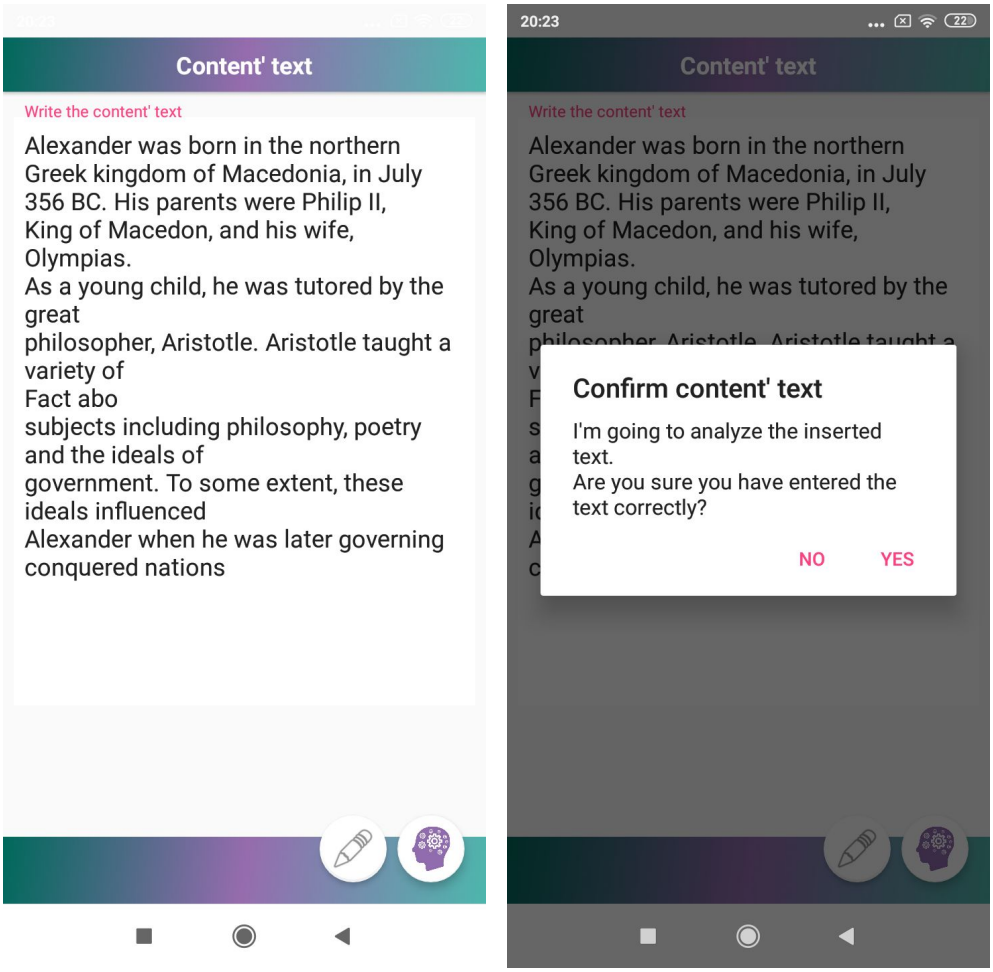


Figure 8.9: *"Virtual Study Partner"*: Training Session for the "Alexander the Great" basic knowledge [41].

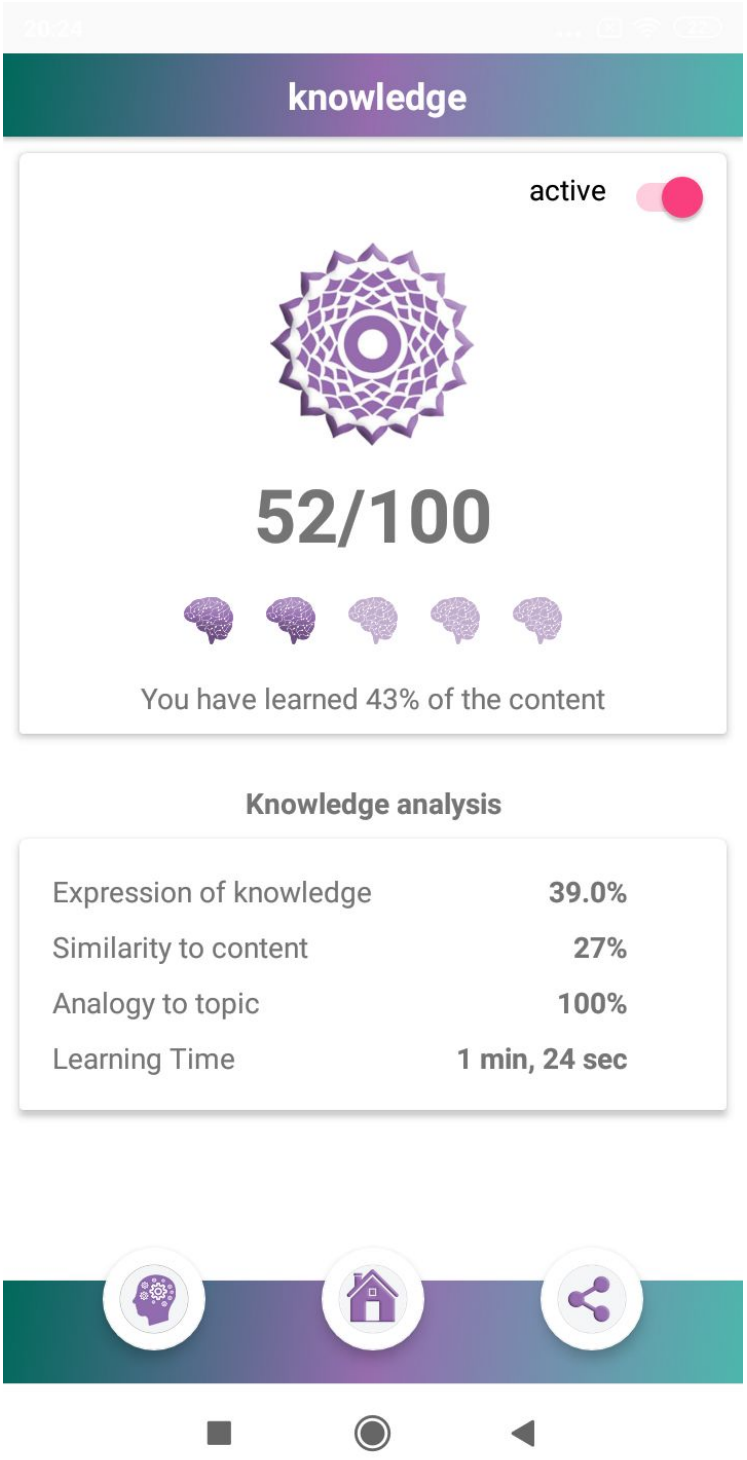


Figure 8.10: *"Virtual Study Partner"*: Results for the "Alexander the Great" basic knowledge [41].

### 8.5.1 Sentiment Analysis

Sentiment analysis inspects the given text and identifies the prevailing emotional opinion within the text, primarily to determine a writer's attitude as positive, negative, or neutral [102]. The response JSON file of a *sentiment analysis request* contains the following metrics:

- *documentSentiment* contains the overall sentiment of the document, which consists of the following fields:
  - *score* of the sentiment ranges between -1.0 (negative) and 1.0 (positive) and corresponds to the overall emotional leaning of the text [102].
  - *magnitude* indicates the overall strength of emotion (both positive and negative) within the given text, between 0.0 and +inf. Unlike score, magnitude is not normalized; each expression of emotion within the text (both positive and negative) contributes to the text's magnitude (so longer text blocks may have greater magnitudes) [102].
- *language* contains the language of the document, either passed in the initial request, or automatically detected if absent [102].
- *sentences* contains a list of the sentences extracted from the original document, which contains:
  - *sentiment* contains the sentence level sentiment values attached to each sentence, which contain score and magnitude values as described above [102].

The *score* of the *documentSentiment* field indicates the overall emotion of a document [102]. The *magnitude* of the *documentSentiment* indicates how much emotional content is present within the document, and this value is often proportional to the length of the document [102]. It is important to note that the Natural Language API indicates differences between positive and negative emotion in a document, but does not identify specific positive and negative emotions [102]. A document with a neutral score (around 0.0) may indicate a low-emotion document, or may indicate mixed emotions, with both high positive and negative values which cancel each out [102].

### 8.5.2 Entity Analysis

Entity analysis provides information about entities in the text, which generally refer to named "things" such as famous individuals, landmarks, common objects, etc [102]. Entities broadly fall into two categories *proper nouns* that map to unique entities (specific people, places, etc.) or *common nouns* (also called "nominals" in natural language processing) [102]. Entity analysis returns a set of detected entities, and parameters associated with those entities, such as the entity's type, relevance of the entity to the overall text, and locations in the text that refer to the same entity [102]. Entities are returned in the order (highest to lowest) of their *salience* scores, which reflect their relevance to the overall text [102]. The response JSON file of a *entity analysis request* contains the following metrics:

- *type* indicates the type of this entity (for example if the entity is a person, location, consumer good, etc.) This information helps distinguish and/or disambiguate entities, and can be used for writing patterns or extracting information [102].
- *metadata* contains source information about the entity's knowledge repository. This field may contain the following sub-fields:
  - *wikipedia_url*, if present, contains the Wikipedia URL [136] pertaining to this entity [102] .
  - *mid*, if present, contains a machine-generated identifier (MID) corresponding to the entity's Google Knowledge Graph entry [70] [102].
- *salience* indicates the importance or relevance of this entity to the entire document text. This score can assist information retrieval and summarization by prioritizing salient entities. Scores closer to 0.0 are less important, while scores closer to 1.0 are highly important [102].
- *mentions* indicate offset positions within the text where an entity is mentioned [102].

### 8.5.3 Syntactic Analysis

Syntactic analysis extracts linguistic information, breaking up the given text into a series of sentences and tokens (generally, word boundaries), providing

further analysis on those tokens [102]. Syntactic analysis consists of the following operations:

- *Sentence extraction* breaks up the stream of text into a series of sentences [102]. The response JSON file of a *syntactic analysis request* returns an array of sentences extracted from the provided text, with each sentence containing the following fields within a text parent:
  - *beginOffset* indicating the (zero-based) character offset within the given text where the sentence begins [102].
  - *content* containing the full text of the extracted sentence [102].
- *Tokenization* breaks the stream of text up into a series of tokens, with each token usually corresponding to a single word [102]. The set of token fields returned from a *syntactic analysis request* appears below:
  - *text* contains the text data associated with this token, with the following child fields:
    - *beginOffset* contains the (zero-based) character offset within the provided text [102].
    - *content* contains the actual textual content from the original text [102].
  - *partOfSpeech* provides grammatical information, including morphological information, about the token, such as the token's tense, person, number, gender, etc [102].
  - *lemma* contains the "root" word upon which this word is based, which allows you to canonicalize word usage within your text. As well, plural and singular forms are based on lemmas: "house" and "houses" both refer to the same form [102].
  - *dependencyEdge* fields identify the relationship between words in a token's containing sentence via edges in a directed tree. This information can be valuable for translation, information extraction, and summarization [102]. Each dependencyEdge field contains the following child fields:
    - *headTokenIndex* provides the (zero-based) index value of this token's "parent token" within the token's encapsulating sentence. A token with no parent indexes itself [102].

* *label* provides the type of dependency of this token on its head token [102].

### 8.5.4 Content Classification

Content classification analyzes text content and returns a content category for the content [102]. Formore details, readers can refer to the complete list of content categories returned for the *classifyText* [71].

# Chapter 9

# An Intelligent Tutoring System Tool Combining Machine Learning and Gamification in Education

## 9.1 Brief Introduction to the Problem

The spread of Information and Communication Technologies (ICT) has significantly impacted our day life: on the one hand, new opportunities have been created; on the other, the relationships have changed. This phenomenon has undoubtedly led to benefits and improvements in the way of communicating, allowing to overcome time and space barriers, for example, giving access to almost infinite information sources. E-learning, webinars, and online educational institutions are now becoming increasingly common, indeed. Today it is possible to learn almost everything thanks to online courses and MOOC; everything is just a *"click"* away.
The current trend in online learning of different disciplines can naturally lead to better educational opportunities and consequently to work. Concerning textbooks, on the other hand, e-books are becoming increasingly common, leading to faster and easier availability of texts and a reduction in the production of printed paper. Furthermore, the very way we think about textbooks is now completely different. No longer just words and images, but specific websites, evaluations, animations, additional materials, and whatever else

allows the assimilation of new content are offered alongside the more traditional reading methods. No less important is the speed with which today we can find information in a few seconds: up to a decade ago, it was necessary to spend hours in the library to see what we were looking for. Given these premises, it seems that technological development has created a perfect world within reach of learners.

The ***Intelligent Tutoring System (ITS)*** term refers to a computer system used to support the student by a learning system that performs functions similar to those of a human tutor. These technologies are designed to interact with human learners in a naturally adaptable way. The final goal of an ITS is to analyze the competencies and behavior of the student framed within a digital learning system that is linked to a specific field of knowledge. The ITS is able to assess the difference between the learner's educational situation and the educational goals to be achieved. Normally, during the training activity, the ITS gives learners proper comments and suggestions by selecting the most appropriate content and types of activities to help them correct themselves. Doing so, it fills learner gaps and allows them to progress in the scheduled training process. Despite the fact that literature shows us that the ITSs have improved the learner achievement and enhance learning, there are problems in their systemic use. On the one hand, ITSs still have problems of a technical and organizational nature, of which we will give some consideration in Sect. 9.2; on the other hand, ITSs have shown problems of *"longevity"* in their use. These problems can be summarized both in the lack of interest and in the boredom of performing repetitive actions by learners. For this reason, the research is going in the direction of studying how to use and integrate gamification concepts and mechanisms with traditional ITSs.

The ***Gamification*** term consists of the use of game mechanics to influence performance and create accountability. These game mechanics satisfy some basic human psychological needs like a sense of competence, autonomy, and relatedness. Gamification uses the *"intrinsic motivation"*, which is the most reliable driver of long term engagement. Gamification uses sophisticated game mechanics and takes a long term approach to behavioral changes and students' work-habit creation. Through its power to communicate goals and give real-time feedback about learners' achievement, gamification is an ideal *tool* for the creation of a new identity of being a study participant, enabling smooth structural change. This Chapter briefly introduces the idea on an ITS that acts as a *cognitive tool*, that can help learners in the development of

the correct study method by interacting with a digital virtual study partner that can assess and suggest them how to improve their performance, also learning by such interactions.

The remainder of this Chapter is organized as follows. Section 9.2 describes motivations, some choices, theoretical and technological considerations of our proposal. Section 9.3 gives a brief overview of existing works in ITS and gamification. The *"Virtual Study Buddy"* system, its features, and usage scenario are introduced in Sec. 9.4.1.

## 9.2 The Goal of the Study

As stated in [90], ICT technologies can help create an education system based on the principles that help teachers, students, and administration to be effective in what they do, improving the quality and relevance of teaching-learning process. With this goal in mind, we started the design and development of our idea of ITS, we called it *"Virtual Study Buddy"* or simply *"VSB"*. Initially, we focused on *digital native* [112] needs, namely the generation of people who grew up in the digital age, comfortable with technology and computers at an early age. Digital natives, from an educational point of view, have nevertheless shown difficulties navigating this new world. As the first milestone of this research, we have released a prototype that is freely available in beta version [133]. This prototype can detect and process data obtained from a user's teaching activity during the learning of a theoretical concept taken from a written text and, therefore, to return assessments on acquired skills and predictive analysis. Currently, a class of a primary school in the province of Messina (Italy) uses experimentally *VSB*. In this phase, we are collecting user experience feedback to improve the system setup, the gamification process elements we adopted, and the self-provisioning mechanism of the educational contents we implemented.

### 9.2.1 Choices and Theoretical/Technological considerations

**Is it worth using ITS? Why didn't they prospered? Does it still make sense to talk about it in 2019?** Although in the last 30 years there has been a significant research activity related to the development of ITS. Despite a large number of projects funded, a lot of money spent, experiments

well underway, the use of these systems has never been started concretely, it never became systemic. Why? The lack of use of ITS systems in the real world outside the university research labs certainly does not depend on the lack of results obtained in the various experimentation experiences described in the literature. All research suggests that ITSs can achieve remarkable increases in student learning over the traditional education community. From a historical point of view, research on ITS has the main aim to provide an excellent tutoring experience comparable with that obtainable with a human tutor rather than the one achieved by conventional computer-aided instruction (simple check on the correctness of the answer given). From a strictly operational point of view, many ITSs were not definitively adopted by the education system because it was challenging to manage them from an educational point of view. Often it was not so fast and straightforward for instructors to create new teaching materials or to update the existing ones. In most systems, the *"knowledge maintenance"* had to be done by skilled programmers at great expense. In our opinion, this fact has led to an increase in costs and time to be taken into account by the instructors and by the educational institutions, effectively blocking their diffusion and actual use. Reducing the costs of all aspects of implementation and management of ITSs is the only way to make them systemic. In our proposal, we decided to automate the phase of the *"knowledge maintenance"* by using artificial intelligence techniques. *VSB* accepts input teaching materials in a digital format that does not require prior processing (e.g., the standard textbooks recommended in class). It makes our *"knowledge maintenance"* economically sustainable. Furthermore, considering this automatism together with those provided for return assessments on acquired skills and predictive analysis, it appears clear how *VSB* is easy to manage even for non-technical personnel.

**Why Gamification?** According to [57], good videogames are *"machines for learning"* since they incorporate some of the most crucial learning principles postulated by today's cognitive science. In [139], the authors explain how a good gamification process needs the presence of two essential components: the application of effective dynamics and the use of the right technologies. Moreover, they declare that *"gamification is 75% Psychology and 25% Technology"*.

From the psychological point of view, thanks to the model proposed in [56], it is possible to identify three fundamental phases to involve the participants in the game:

effectively *Provide a motivation*. The starting point of each gamification activity is to give people a reason to participate. The mechanism of the game and the challenge is deeply rooted in the human mind and is a potent stimulus. Still, for it to work at its best, the players must have a prize in front of them, a goal, an objective that attracts attention and increases determination. The choice of benefits and prizes is critical because the more accurate it is, the higher the drive to compete for that will be generated in the group.
*Provide tools to participate*. For the gamification to work, all the subjects involved must have, at least at the outset, the same possibilities, and the same tools to scale the rankings. The adoption of gamification for positive results is necessary to include one or more training and preparation moments to avoid the possibility of insinuating among the participants that the organizer could have favored someone.
*Offer a starting point*. Every gamification activity needs a start-up moment (also called Kickoff) that acts as a zero moment from which to start the challenge. It means, for example, in creating a dedicated event, a team-building activity, an official communication, and so on. In the case of long-term competitions, intermediate stages must be planned in which to check the progress of the activity, deliver special prizes, celebrate who is achieving results and motivate participants in difficulty.
But the most crucial thing in gamification activity is the *timing*: if all the mechanics of the game are not activated simultaneously and in a coordinated manner, the risk is that the participants quickly lose interest in what they are doing. Section 9.3 gives an overview of some experimental gamification tools and technology which have positively contributed to learning and achievements. Section 9.4.2 describes techniques and strategies discussed in [56] that we are using in *VSB* experimentation with the aim to involve our participants.

## 9.3 Background of the Study

Some work in literature adopted gamification techniques to ITS solutions.

In [79], the authors present some empirical results on teaching basic Mandarin as a second language to college students using a gamification approach. This study shows some evidence that gamification outperforms non-gamification teaching method in related to learning concentration, skills,

feedback, and immersion.

In [54], the authors examine the benefit of an RPG (Role-Playing Game) to learn other languages and their complicated letters, in this case, Japanese kanji. Moreover, the paper provides some suggestions with particular reference to how gamification can bridge learning outcomes as well as a game-play experience.

In [126], the authors define the concept of gamification and introduce its elements. They describe how the gamification model and how the connection between motivation and gamification works. They give some examples of applied gamification in the focus of smartphone applications.

In [17], the authors describe and analyze some gamification methods used by the Zagreb School of Economics and Management (ZSEM) in different courses related to technologies and legal discipline. The results showed students' satisfaction and an increase in their motivations in their studies.

In [114], the authors analyze how the application of gamification strategies in MOOCs on energy sustainability affects participants' engagement and motivation in students. The results show the achievement of high levels both of engagement and student motivation.

In [48], the authors present the integration 'Gamification' instructional strategy along with traditional teaching modes for the final year of Computer Science and Engineering students for the course of Information and Cyber Security. The results show that problem-solving among students increased significantly.

In [115], the authors applied gamification in mobile learning for memorizing Alquran to increase the fun factor. The test results showed that there were significant differences in learning outcomes between the experimental group and the traditional group.

In [100], the authors carried out an exploratory study assessing the effect of using the gamification of interactive digital storytelling on classroom dynamics and students' interaction. The results showed an increase in classroom discussions and students' engagement.

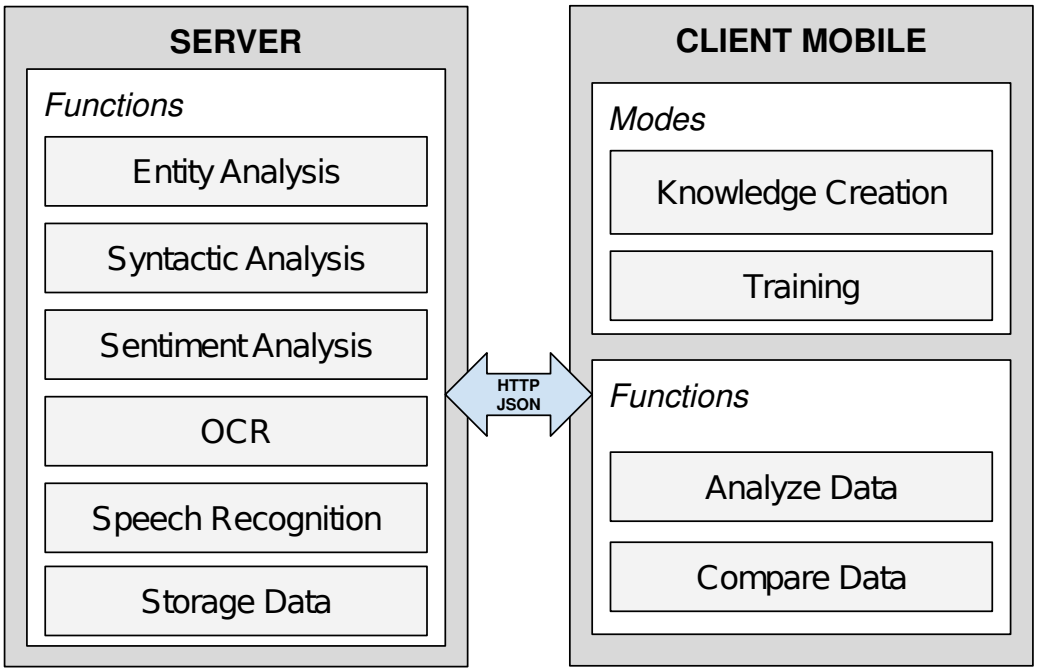


Figure 9.1: *"Virtual Study Buddy"* architecture general scheme [45].

## 9.4 The Virtual Study Buddy Tool

### 9.4.1 Scenario

Fig. 9.1 shows the general architectural scheme of the *Virtual Study Buddy* scenario. It consists of a client-server architecture that communicates via HTTP through JSON files. In [41] we detailed the technological choices behind *VSB*. The system offers two main ways to interact, one in which it is possible to *"create knowledge"* and another one in which it is possible to *"exercise a particular knowledge"* previously initialized in the system. As already mentioned, *VSB* accepts input educational materials in digital format, labeled in topic and sub-topic, which does not require previous processing. In particular, the educational contents can be inserted using the keyboard, capturing it from an image (or from a photo), or extracting it from a PDF file. After entering new knowledge into the system, in a completely automatic and transparent way to the user, the system proceeds with the extraction of the text and some *"metrics"* using machine learning services. Texts and metrics will be stored in the system and used for future comparisons during the users' learning actions. As detailed in [41], the metrics that the system

can extract come from the following types of analysis: Sentiment Analysis, Entity Analysis, Syntactic Analysis, and Content Classification. The user can decide at any time to exercise his knowledge on a topic by selecting one of those available from the system and starting a training session. During a training session, thanks to the features provided by his mobile device, the user records his speech on the topic he wants to train and then sends it and to the system. Thanks to Machine Learning, the system can understand the natural language of man. It means that the users can present the topic orally in a very personal way and not strictly identical to the text stored in the system. Once the speech is acquired, the system extracts text and metrics in the same way as it does in the knowledge creation process. The final step is to examine the equivalence, the conformity, the oral exposition, and the similar meaning between the text inserted during the knowledge creation and the text coming from the transcription of the received vowel during the training session. The two texts are compared and analyzed structurally, syntactically, and semantically. The system analyzes each part of the speech by detecting the morphology, the dependence on other words present, and the taxonomy of the text, to match arguments, concepts, and words present in both texts. The training session ends with a *score* expressed in hundredths and in relative percentages, which refers to the oral exposure, the equivalence of the texts, the similarity of the subject dealt with, the percentage of knowledge of the acquired text, the time dedicated for oral repetition.

### 9.4.2 Techniques we used to involve our participants

*VSB* was designed to use gamification dynamics and mechanics in different mixes and modes depending on the goal you want to achieve. Each user is associated with a profile in which it is possible to view its "carrier" in the system, for example, points, levels, badges, rankings, missions, achievements, and so on. *VSB* offers three learning methods: *autonomous*, *cooperative* and *guided*. The autonomous one was explained in Sec. 9.4.1. The *"cooperative"* and *"guided"* methods are based on the same operating principle as the autonomous one; it only changes learners' organization and educational goals. In the *"cooperative method"*, the learners are grouped in open peer groups where all the participants can share educational materials, results, and goals to reach. All the learning activities are public. In the *"guided method"*, the user who has the privilege of a "teacher" can create closed groups, invite

learners, associate different educational material and goals to learners of the same group, evaluate and reward progress achieved by the learners with the choice of publishing the results.

# Chapter 10

# Internet of Things Network Infrastructure for The Educational Purpose

## 10.1 Brief Introduction to the Problem

Internet of Things (IoT) became one of the critical branches of the modern digital era. It is possible to observe a vast number of opening positions related to the IoT on the professional market in both hardware and software, research, development, and implementations. Gartner's report dated 2019 predicts 5.8 bln IoT end nodes by the end of 20201. There is an observable need for qualified engineers and technical staff [1] related to the IoT, including hardware and software developers, IoT network, energy efficiency, and IoT security specialists as well as IoT solution designers. Since IOT-OPEN.EU started to provide standardized IoT training for bachelor's level, masters level, professionals who are already beyond their regular education but are about to dive into the IoT world because they're willing to or are required by their commercial needs. Parallel and independently to the IoT development, it is possibile to observe how classical teaching changes, from traditional lectures, classes and laboratory exercises towards self-paced learning, using online resources. This process was suddenly speed-up along with the outbreak of the SARS-COV-2 virus and related COVID-19 pandemic that

[1] https://www.gartner.com/en/newsroom/press-releases/2019-08-29-gartnersays-5-8-billion-enterprise-and-automotive-io

forced universities, school teachers, and trainers to switch from classical into online learning rapidly. Also, from the social point of view, distant learning is no longer the domain of amateurs, enthusiasts, and hobbyists providing some information on their blogs, and vlogs: development of the massive online learning platforms (MOOCs) like, i.e. Coursera, EDX or Instructables, shows future development of training and teaching methodologies. Nowadays, it is not unusual to use, i.e. Youtube, Github, or Wiki (i.e., Dokuwiki) for authoring and delivery of the teaching material. Of course, their credibility can be in doubt, so selecting trustworthy, up-to-date and credible material is a challenge. Interestingly, thanks to the WEB 2.0 development, and the possibility to comment on the content, one can find early symptoms of incredible ones.

## 10.2 The Goal of the Study

This chapter summarises the results of our IOTOPEN.EU, an Erasmus+ funded project, that was intended to deliver high-quality study materials within the IoT scope. In particular, it focuses only on one aspect of the project: VREL - virtual, remote access IoT laboratory nodes that everyone can access using the web browser only. However, it is also presented briefly other components of the project to present how VREL IoT laboratory relates to it.

## 10.3 Background of the Study

Distance learning has become a popular approach to providing knowledge in modern university education.

In addition to primary teaching aids such as e-books and other electronic teaching materials, video lectures, tests and quizzes that are available on most e-learning platforms laboratories with remote access are particularly helpful in teaching technical subjects. In the literature, it is possible to find descriptions of many examples of distance laboratories created to support teaching in many various areas of professional study.

In [28], authors presented some real cases in distance learning courses in the Industrial Engineering School at the Universidad Nacional de Educacion (The Spanish Open University), Madrid, Spain. Their approach combines

individual and collaborative learning in remote and local laboratories, in a distance learning context, and proposes the use of a Web based experimental environment called Active Document [131] to improve the development of reasoning skills in practical work. The learning environment they offered is used to organize and invoke the different computer tools that make up a virtual chemistry lab. Experiments may be structured in a way that enables students to perform lab work with colleagues, which is both more motivating than doing it alone and also allows students to learn to collaborate.

In [39], the authors described their Practical Experimentation by Accessible Remote Learning (PEARL) system, which allows students to work together while at a distance from the laboratory site, using a range of synchronous and asynchronous communications tools. They illustrated some experiments as a demonstration of the potential and validity of their approach. Experiments developed include an implementation of a remote electron microscope, a spectrometer, visual inspection of printed circuit boards and a digital electronic bench.

In [15], the authors present IoT Rapid Proto labs designed as authentic, productive learning environments. Their approach is based on three design principles: 1) Realistic, complex task situations, 2) Multidisciplinarity, and 3) Social interaction. The laboratory proposed by authors is not a pure remote lab. However, rather blended (virtual as well as real), user-driven, and productive learning environment supported also by Project Arena (a web-platform), which enables learners to effectively collaborate on rapid-prototyping of IoT products/services stimulating the flow of knowledge and innovation between higher education, enterprises, and other stakeholders.

In the paper [65] authors present an approach that aims to transform traditional computer labs into virtual lab environments. The work recognizes two categories of experimental setups, where slightly different approaches are needed. The first category is software-based experimentation. The second category is hybrid experimentation, where software and hardware experimentation need to be conducted within the same experience. The proposed design relies on the concept of 'virtual presence' whereby the students and their home computers appear as if located inside the lab.

In [18], the authors discuss the disadvantages of software simulation. They claim that while simulation packages have a significant place in Distance Learning (DL), they can never replace the need for real labs where students can construct their knowledge and put their theory and practice to

a real test. Therefore, they argue that a Remote Laboratory (RL) expands the efficacy of a DL. Moreover, they present an alternative to simulation as developed two prototype laboratories for electrical engineering and physics.

In [132], the authors present a study, carried out in a Higher Education Institution in Brazil, where a remote lab (VISIR), addressing electric and electronic topics, was implemented, yielding 471 students' academic results and opinions. The results reveal some factors teachers may tackle to foster student learning and motivation. Teachers' involvement plus their ability to brief students on VISIR's usefulness have a significant influence not only on students' performance but also on their perception of learning and satisfaction with the tool.

In the paper [109] authors present remote access for a laboratory experiment that involves measurement of a volt-ampere characteristic of a semiconductor diode. The remote laboratory assumes using real equipment with setup controlled over the Internet, and with a video camera to display readings from real instruments to the learner.

Paper [94] presents the description of some examples of remote laboratories created in Australia and some European countries. They are mainly electric, microelectronic, control or computer laboratories. Still, there are also realizations in the disciplines of physics, mechanical and mechatronic engineering including gasoline motors, pneumatics, material testing, plasma diagnostics, and radio-physics.

An IoT remote access laboratory idea appears in [49] in the context of the resource sharing between rich and poor schools in South Africa. In this particular example, authors present a closed system with a remote interface to manipulate a robotic arm (controlled with Arduino) to perform various chemical experiments. Distant students can remotely manage physical devices and observe results through the Internet with means of the video stream and sensor outputs.

In the paper [91], authors present a RAL (Remote Access Laboratory) for the Queensland (Australia) primary school, where approximately 76% of pupils study remotely. It is because of the specific inhabiting conditions in Australia, where many people (including children) live in distant locations and cannot send children to school daily. The project applied to children aged between 7 and 12. In this laboratory, Meccano SpyKee robots were used (a humanoid form) that were controlled using PC computers and connected wireless using WiFi. In particular, relating to the aforementioned

Queensland case, authors in [94] present in-depth analysis of more, selected cases, where they tell a short story of RALs in various regions and universities across the world. Authors pay attention to the different reasons for driving RAL development in various regions of the world. In the case of large countries with small populations like Canada, Australia, and Russia, the development of RALs, according to authors, was driven mostly by physical distances and lack of access to the educational resources. In the case of Europe, this one was technology-driven and introduced with various EU funded grants to lower the difference between developing and developed countries with providing access to the research and educational infrastructure across educational bodies and also to optimize setup and maintenance costs.

An interesting case study of the distant learning process using remote access laboratories is presented in [130]. Authors point out differences between classical (on-site) and on-line, distant learning and propose a pedagogical approach with the goal-oriented approach and three-phase educational process including 'pre-lab', 'lab-phase' and 'post-lab' steps to achieve best results.

In the [78], authors start from the same conclusion as authors of the IoT-OPEN.EU project, where students are entering STEM education on any level face lack of IoT courses. Authors propose a learning framework on IoT, integrating hardware, software, and communication, however mostly using available components (i.e., ThingSpeak, Google Cloud Web Services) and base on existing embedded systems course, extending it towards IoT education. One of the key assumptions of the IoT-OPEN. EU project was the placement of laboratory equipment among different partners. A similar approach is presented in [84], where the HVAC laboratory has been created in the cooperation between the US and Switzerland universities. The authors created a laboratory in which US students could conduct research on a heat recovery system that was physically located in Switzerland. Swiss students also had access to a variety of equipment located in the US.

Authors of the paper [40] describe a virtual laboratory designed for teaching the Internet of Things course. Students can create a model of the IoT system using provided sensors and actuators. Components of the system are created with the use of popular Arduino and Raspberry PI microcomputers; students can use them only with API provided. In our approach, students can not only utilize functions for reading or sending data to the IoT nodes but can reprogram the firmware using C language remotely. The base environment that our distance laboratory uses has been created at Tallinn

University of Technology. It is described in the paper [119], and outcomes of the approach with case study and impact on the teaching results are presented in [120]. This teaching tool is created as a rich Internet platform, where different remote and virtual labs are integrated. The DistanceLab is designed to enable programming and controlling the connected devices via the web interface. It is done with a web-based programming environment, an automatically invoked compiling process and the possibility of flashing programs directly to the connected devices. In [118], the extension of this lab towards robotics is described.

## 10.4 IoT devices and Platform towards Education

As the Internet of Things emerged rapidly, many STEM (Science, Technology, Engineering and Mathematics) educators found themselves with a lack of curriculum on the IoT. On the other hand, technical universities, VETs (Vocational Education and Training), and professional training centers had already vast experience in automation control training. This experience usually concerns:

- homogeneous technology/manufacturer oriented courses for professionals to earn a certificate on some technology;
- embedded system courses for university students;
- digital circuits and electronics courses, networking and mobile devices programming curriculum and modules and others, related to the fundamentals of the IoT.

All that constitutes a substantial and concrete basis for the introduction of IoT in education. A common challenge in the case of the IoT courses is the need to provide to the audience an experimental part, such as a laboratory. It can be a form of laboratory activities or projects, and this need is growing with the introduction of project-based and experiential-based learning. Many vendors currently deliver environments for the IoT labs in the form of the development toolkits that are usually associated with a particular software development kit, proprietary software, and closed solutions.

Training centers typically are offered with a free or discounted solution (including hardware and software samples and even full sets). They are unable to implement their laboratory environments from scratch due to the lack of resources. This type of approach follows the patterns in which vendors want to 'tie' trainers and students to their technology exclusively, and ultimately, it causes tailoring of the curriculum to fit one specific IoT solution provider. Although this is acceptable in the case of the VETs training willing to get education and certification for a particular product or system, it is wholly unacceptable for universities and full STEM education. In any case, the implementation of laboratory rooms is costly, time-consuming, and of low flexibility. In addition, large market players provide free (or limited) services like, i.e., Azure, Watson, Google IoT services, delivering de-facto flexible, yet, software-only IoT frameworks. Of course, those are useful in the education process as access is virtual, and it is quite easy to use any of them. Still, without IoT hardware, it is only one piece of the IoT puzzles needed for comprehensive engineering education nowadays. In many cases, IoT systems are provided to students with simulators means. While it is an essential part of the teaching and training, simulation cannot replace the real hardware interface with their vulnerabilities, failures, timings, and other physical phenomena, usually not simulated. An ideal approach to the IoT devices and platforms should, therefore, promote IoT laboratory solutions that are:

- easy to implement and maintain;
- provide touch with real hardware (not simulated one);
- include a variety of devices (platforms, sensors, actuators);
- integrate easily with other services;
- provide the ability to set up heterogeneous IoT networks;
- simplify user access, possibly over the web, without the need (or with scope) of software installation and configuration;
- ensure security for both users and infrastructure;

## 10.5 IOT-OPEN.EU Project

IOT-OPEN.EU is an educational project within the Erasmus+ Key Action 2 framework, oriented towards Strategic Partnership between Higher Education (HE) and also commercial bodies. In 2015, once the project idea was developed, there was no standardization in IoT teaching and training, and not so many universities had courses related to the IoT. On the other hand, the IoT idea was quickly accepted by the industry, together with Industry 4.0 and the development of 'Smart' devices. The commercial market expected universities to offer IoT courses to their students to make them became well-trained engineers, ideally with practical experience in IoT systems and devices. This situation presented a gap between HE (Higher Education) and the expectations of the European digital market. Those key facts lead to the shape of the grant, in which 6 partners (5 HE bodies and one SME) decided to prepare a standardized solution for IoT teaching and training on various levels of education, starting from those who have never heard of IoT, ending up with the R&D opportunists, looking for research ideas. Moreover, materials prepared within the IoT-OPEN.EU was classified, and a track for non-HE was identified, such as,, for example for hobbyists that are willing to play with IoT and VETS, ready to extend or adapt their positions to the labor market requirements. As part of the project, 3 significant results were planned and implemented:

- A set of materials for classical courses held at the university within the IoT fields, consisting of an IoT coursebook, many DLP presentations to be provided to students with classical, auditory based lectures and on-site laboratories accompanied by a practical laboratory manual.

- A purely online, self-paced courses in the form of MOOCs, available via edX.org and local platforms, kept by HE consortium participants.

- Several heterogeneous IoT laboratory nodes with remote access (VRELs): virtual and distant access laboratory nodes implemented in different countries yet able to integrate additional services and cross-cooperate between different physical locations. Nodes present different hardware and provide an opportunity to implement different IoT scenarios. Classical courses and online one can be treated as stand-alone or supplemental. In any case, VRELs bring practical, laboratory opportunities

to interact with real, physical hardware, whether students choose to study on-site or online.

The components mentioned above compose the IoT educational framework, introduced and implemented within the scope of the IOT-OPEN.EU. The following sections focus on the VREL part. VRELs are a key component in the IoT training enabling students to be able to interact with real devices during their study track even if they're unable to access them physically because they have no technical background or cannot afford to buy one. MOOCs and VRELs together provide a robust solution for all those that are unable to attend university for the regular course, whether because of lack of resources, living in a remote area, being disabled, or because of any other reason, but are still willing to participate in the IoT revolution.

## 10.6 Distant Laboratories Model

Concluding requirements presented in section 10.5, it was identified that the most problematic part in introducing IoT modules into the curriculum are laboratories using end nodes layer and fog layer devices. Other resources are usually readily available as a variety of them is already present on the Internet or can be provided in this way without any particular obstacles. Our VREL laboratory solution provides IoT infrastructure, that tackles IoT end nodes and fog layer services. VREL laboratory implemented within the frame of IOT.OPEN.EU project has distributed hardware and software resources across 3 European countries: Estonia, Poland, and Italy. Usually, laboratories with remote access provide limited API that end-users (here students and supervisors) can use, implementing a closed number of scenarios. In the case of VRELs, there is low-level programming support that enables a virtually unlimited number of scenarios, that can be implemented. This approach, however, requires that all components of the VREL infrastructure are safe to operate on this level of access thus require detailed and careful design, in particular regarding hardware and mechanical parts as real IoT solutions virtual laboratory nodes should contain both sensors and actuators. Users should be able to access the system with means of a single, universal and standardized interface, regardless of their location and physical location of the laboratory infrastructure. Moreover, all tasks should be implementable using a web interface; thus, users only need an Internet connection and a web

browser to use it 2. Users should have the ability to book a device(s) in exclusive mode. Low-level programming usually also means that development requires dedicated libraries to interface hardware. Seamless library management is a complex challenge, as there are typically many libraries supporting particular devices. Because of it, library management should be simplified and consistent across all laboratory components, constituting a solution that includes versatile possibilities like automatic updates or locking on a particular library version. Last but not least, users implementing networking should be able to create consistent communication solutions among nodes located in different locations (even countries), access other components like, i.e., cloud resources. On the other hand, users using in particular wireless interfaces should be limited to access hosting infrastructure, to limit vulnerabilities, implemented either voluntary or involuntary while performing exercises.

## 10.7 VREL Implementation

Distant, remote access IoT laboratories implemented within the scope of the IOT-OPEN.EU project was settled in three European countries: Estonia, Poland, and Italy. Other grant partners performed their End Node tests and integration with the IOT-OPEN.EU VREL Central Server platform successfully and is awaiting future integration, possibly during the following extension of the grant. The location of the components across Europe and grant partners are presented in figure 10.1.

On the hardware level, two platforms were chosen as presenting current trends for popular End Node MCUs:

- Arduino Uno (Atmel) MCUs: ATmega328P;
- Espressif NodeMCUs: ESP 8266 (ESP-12E);

Additionally, RPi (Raspberry Pi) v.2 and v.3 boards were used to implement router/MCU remote programming interfaces and video streaming for the laboratory nodes. The software of choice for video streaming was open source Motion server. RPi platforms were also used to implement some of the integrated networking services along with PCs. Regular IP networking (IPv4) were used; however, other radio interfaces may be connected to the laboratory platforms with ease, to extend networking capabilities, and to implement pure IoT networking, like, i.e., 6LowPAN. The user interacts

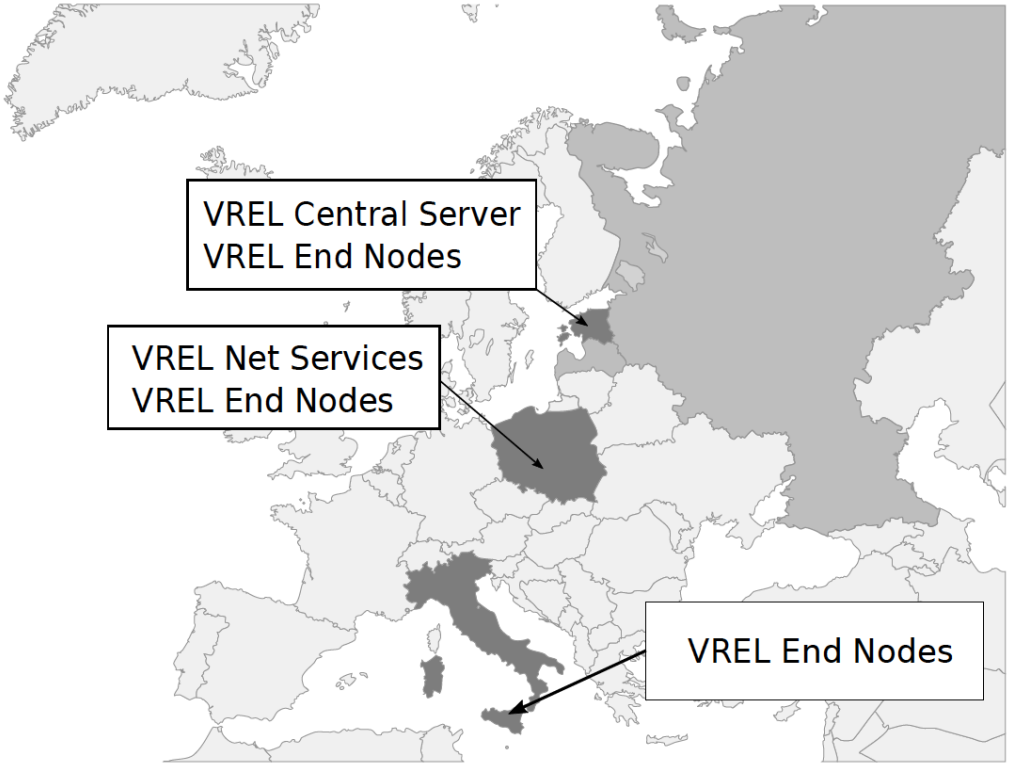


Figure 10.1: IOT-OPEN.EU VREL infrastructure across Europe ( [125]).

with the system using a web browser and receives feedback via text messages (regarding compilation, upload and flashing) and visual input from the integrated web cameras. Each End Node contains at least one web camera while some of them are equipped with more, i.e., to present precise visual effects like, i.e., LCD display. The sample user interface is shown in figure 10.2.

The general schema of the VREL infrastructure is visualized in figure 10.3. End nodes are grouped physically in remote destinations related to the grant partners. However, within the scope of the local resources, they are or may be distributed among different locations. SUT (Silesian University of Technology) uses devices (end nodes) located in various areas, including i.e., building roofs for environmental measures, simulating heating and cooling of the smart house (figure 10.4) but also nodes located indoors (figure 10.5).

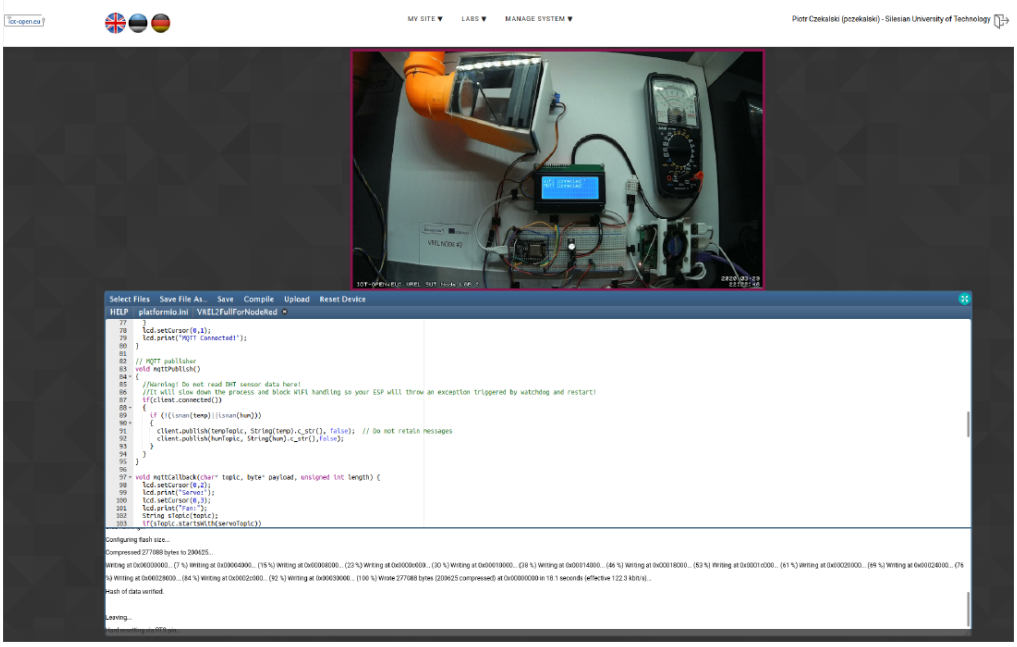


Figure 10.2: User interface for End Node programming in C++ ( [125]).

### 10.7.1 VREL Management System and Front-side Services

The primary services for the solution are located in Tallinn/Estonia, implementing user front-side with rich web interface that includes source code editor and file manager, user management system with roles, device booking features, and remote communication center. These services also integrate source code storage, source code compilation tool-chain for various platforms (including two aforementioned), and development library management services. Part of those features is implemented with means of popular PlatformIO framework, which in detail is being used for library management and source compilation. PlatformIO also handles libraries and framework/-compiler tool-chain updates, performed on demand (not automatically). As the VREL system is used by partner universities (users), a separate user management system was introduced and integrated: any user willing to use VREL laboratory nodes must register an account. That is a requirement to enable exclusive device booking to let users do not interrupt one another during experiments. Users can book more than one device at a time. Thus it can create a complex networking solution itself or can cooperate with other users

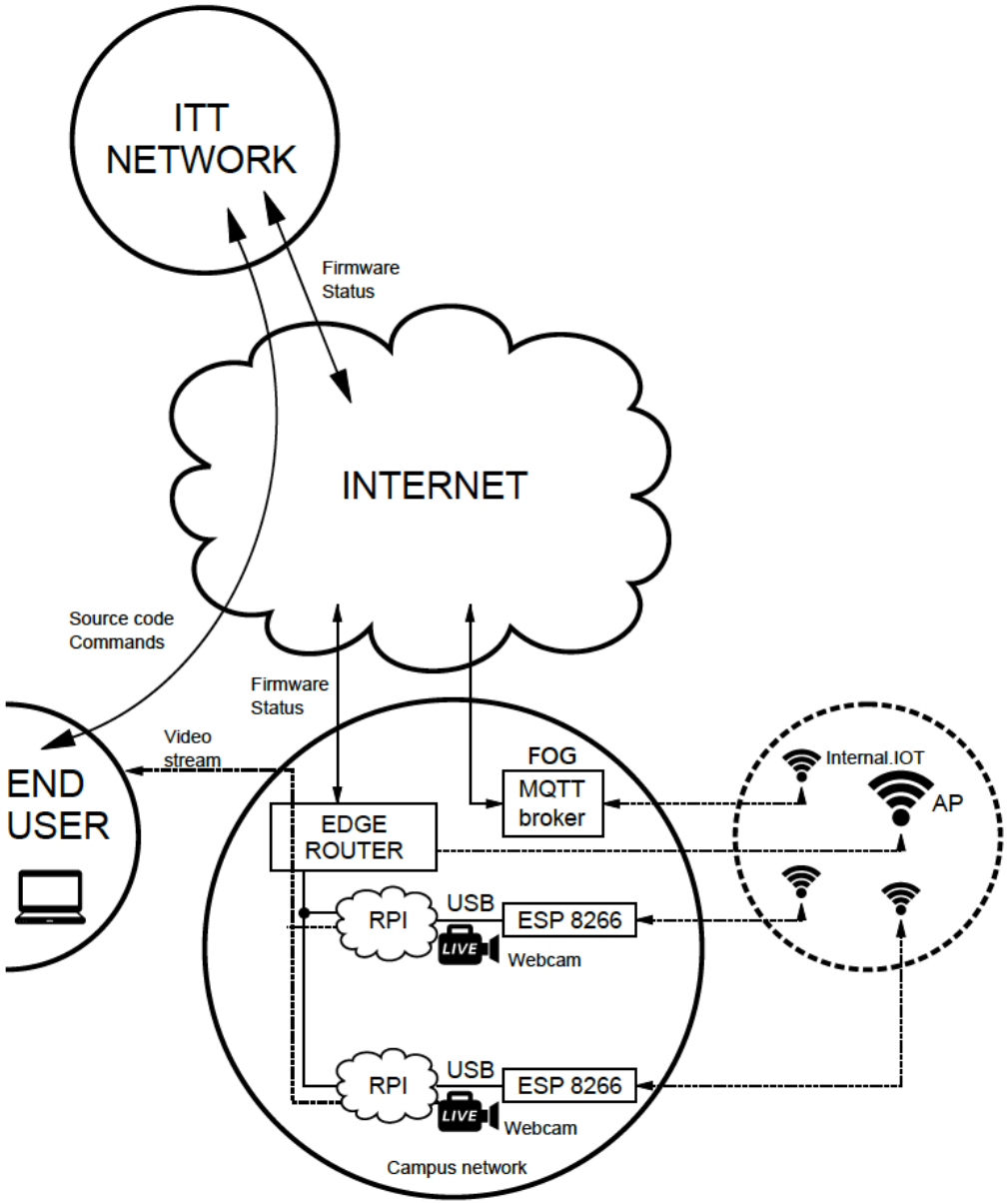


Figure 10.3: VREL infrastructure ( [125]).

and services using local networking and Internet services. Central management system stores user files (per user) as well as templates (per End Node), so users can temporarily suspend their work and return later. That enables teaching scenarios with continuous laboratory work, where students develop


Figure 10.4: VREL SUT roof top thermal smart house laboratory End Node ( [125]).

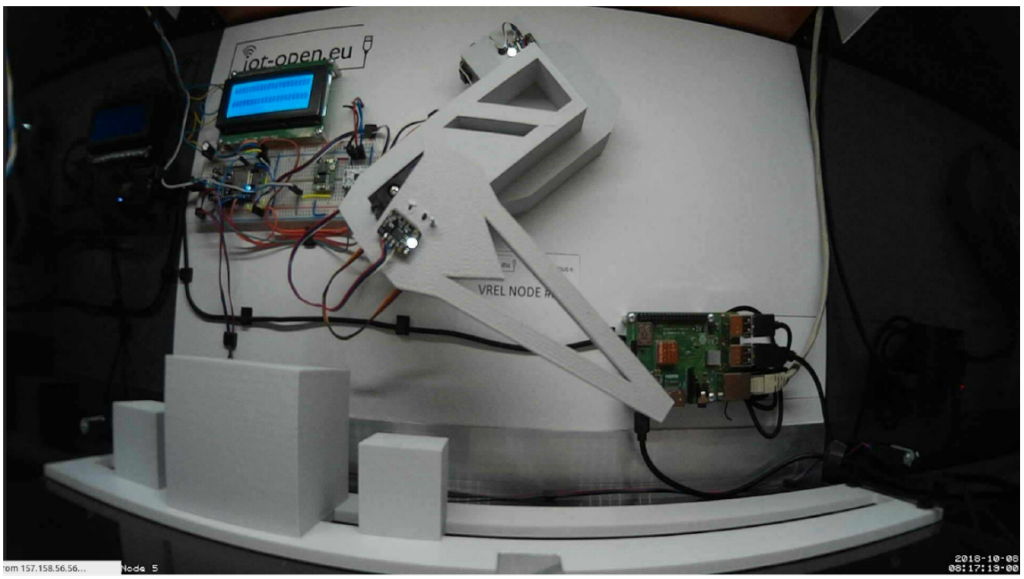


Figure 10.5: VREL SUT indoor laboratory End Node ( [125]).

their solution over several meetings, implementing a project-based learning model. Finally, the central VREL management server provides compiling features for C++ code, concerning the specific requirements given by plat-

forms, i.e., memory limits, memory mapping, and adding resources. Once the firmware package with source code is created, it is then injected through the SSH channel (additionally secured with a VPN connection) into the proxy/routing/programming devices (here RPi). Then RPi flashes MCU using an integrated programmer via USB interface. Use process is presented in figure 10.6.

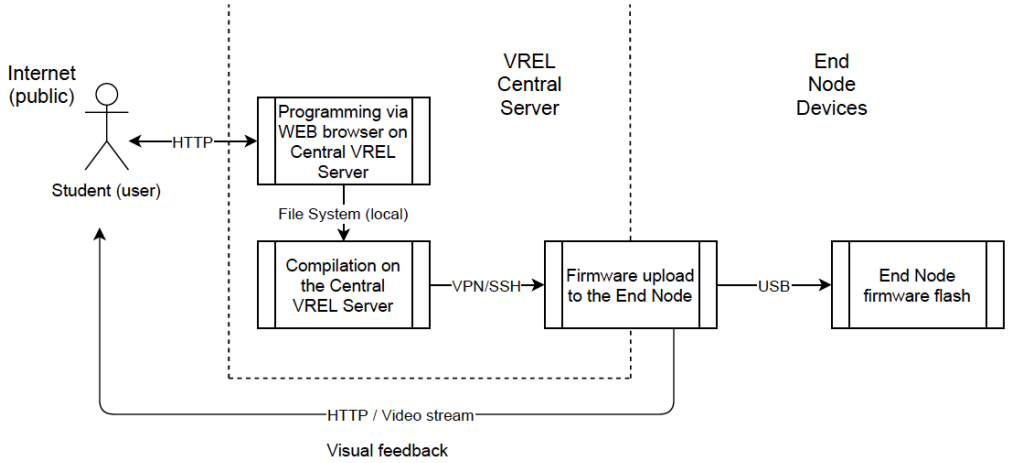


Figure 10.6: User interaction ( 125 ).

### 10.7.2 End Nodes

End nodes represent various hardware, including at the moment aforementioned: Arduino and ESP 8266 MCUs, temperature and humidity sensors, light level sensors (here for measuring the reflection of the light to detect flap movement), 6DOF IMUs (Gyro + Accelerators), servos, stepper motors, high power LEDs, LCD displays and colour RGB sensors. It is assumed that every single End Node should be able to access the network directly or indirectly. Also, local networks (among nodes) should present to limits to let students be able to implement various communication scenarios, even 'dangerous ones', like i.e., IoT security-related ones (i.e., hacking). Whilst devices are located in a separate network, and this is not considered to be a serious vulnerability. VREL nodes provide the ability to let students practice interaction with various sensors and actuators using different, low-level protocols. That covers in particular:

- digital inputs and outputs;
- analog inputs using A/D converters (built-in into the MCUs);
- communication with external sensors using SPI, I2C, OneWire, and Serial protocols;
- simulating analog output using PWM, directly over GPIO and indirectly through I2C expanders;
- controlling servos;
- controlling step motors;

The aforementioned list is non-exclusive but presents the core of the IoT technologies. It prepares students to implement IoT devices from scratch, in most real-life scenarios, even if laboratory nodes seem to be synthetic.

### 10.7.3 Network Integration and Services

As experimentation with IoT systems requires network connectivity, natural choice to bind distributed laboratories is the Internet network. Because of the diversity of hardware platforms, there were two implementations of layer 1 and 2 chosen:

- IEEE 802.3 for Arduino Uno with Ethernet Shield (implemented in Italy and Estonia),
- IEEE 802.11 for Espressif ESP8266 with integrated WiFi 2.4 GHz (implemented in Poland and Estonia).

Arduino Uno based end nodes constituted de-facto a sensor network, so in most scenarios, data are transferred from End Node to the Internet and cloud services thus, devices are connected into the sub-networks, hidden behind NAT and firewall and physically connected to the Internet. Similar way, ESP8266 based end nodes are provided with a dedicated, private, separated from the Internet (no packet routing), WiFi access point (AP), to implement various training scenarios. That also includes peer-to-peer communication among nodes as well as implementing mesh networks using ESPs' capability to act simultaneously in access point and station mode (AP+STA). Those devices include both sensors and actuators and require bi-directional data

transfer. As the aforementioned AP network is physically separated from the Internet, to provide connectivity to the Internet and cloud services, there is an application-level MQTT broker, binding AP and Internet, using two interfaces and Node-Red server. This service is implemented using the RPi 3B+ device, using its Ethernet interface for Internet connectivity and wireless one to connect to the private AP.

### 10.7.4 Security considerations

During the design of the system, its security was one of the critical aspects. This kind of distributed solution with several devices spread across Europe and different networks, if misused or compromised due to the vulnerabilities, would rise hard to track and trace cases. Access to the system requires account registration and exclusive booking of the device. That enables tracking of the voluntary acts of attacks and identification of the attacker. Access to the system is logged; thus, backward identification is possible. The system uses secure VPN connections among distant laboratory nodes, and the central management server and firmware are injected to the End Node via proxies (RPi devices, located on the End Node side) using SSH connections (over VPN) with authorization using certificates. This channel is considered secure and along with best practices. On the other hand, a model where users are enabled to have access to the network-connected devices on the low, programming level (firmware) like in case of our VREL nodes, raises serious considerations and possibly of a virtually unlimited number of vulnerabilities, i.e., DoS/DDoS attacks using VREL devices, MAC address fooling, etc. In the case of the open development environment, it is impossible to create a fully secure solution. Yet, the riskiest is enabling users to connect to the Internet network freeway. In such a case, the standard approach is to create an internal, separate wireless network that, when compromised, won't impact public network nor affect many resources. On the other hand, separating devices from the web critically limits the number of actors participating in scenarios, i.e., students won't be able to send their data to the cloud, analyze, store nor visualize it using external tools. That also breaks the idea of a distributed system that integrates devices across their physical locations (across participating countries). Here comes a solution with FOG layer devices that act as application protocol level routers. In the case of VREL labs, an MQTT (Message Query Telemetry Transport)

protocol broker was used. Access to the broker requires providing credentials that are distributed in the publicly available documentation. This approach raises the question about implemented security model (de-facto "security by obscurity") however our tests show that for over 2 years if running, we didn't note a single issue regarding miss-use of the broker, nor unauthorized access, thus proving that chosen approach is keeping this vulnerability on the reasonable and acceptable level. The idea of the separated network with message-level routing is presented in figure 10.7.

## 10.8 Acknowledgment

This material is based upon work supported by the European Union under the Erasmus+ Key Action 2 (Strategic Partnership) project IOT-OPEN.EU (Innovative Open Education on IoT: improving higher education for European digital global competitiveness), reference no. 2016-1-PL01-KA203-026471. The European Commission support for the production of this publication does not constitute the endorsement of the contents, which reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein. This work was partially supported by the research project (RAU-6, 2020) of the Silesian University of Technology (Gliwice, Poland), via Statutory Research funds of Faculty of Automatic Control, Electronics and Computer Science, Silesian University of Technology, Gliwice, Poland.

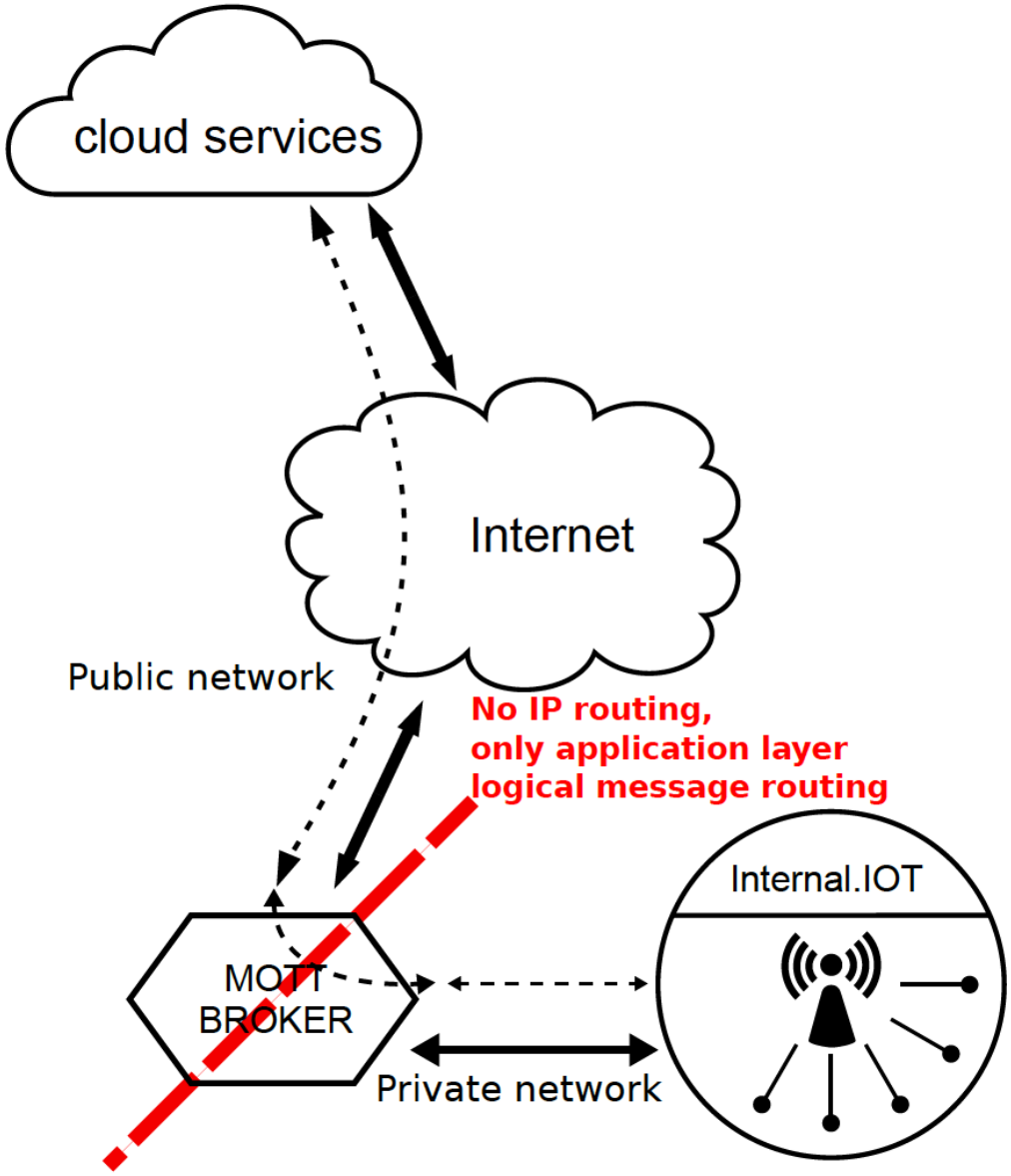


Figure 10.7: VREL Security: MQTT routing model ( [125]).

# Conclusions

This dissertation proposed an overview of some of my recent research developments in the field of smart computing and its applications.
As explained in the introductory chapter, to facilitate the readers' understanding of my study path, I have organized my thesis into three main sections. Each section contains chapters related to a macro-area of issues that I have studied in-depth and which I dealt with during my study activities.
Each chapter reported a single research activity, highlighting the purpose of the study, the motivations of the technological choices, the description of the new resolution strategy, taking into account the background and scientific works already present in the literature.
The underlying theme of all the research activities presented in this manuscript was to foster technological innovation and the exploitation of science, technology, and engineering for the benefit of human society.
As outlined below, these premises permitted me to work on a wide range of systems, applications and services, ranging in the field of smart computing.

Conclusion and some future developments of the contents proposed in this thesis are as follows.

Section I presented unsolved challenges and issues in the management of `confidentiality` and `integrity` concerning `data` stored on Cloud storage services, which are part of `Multi-Cloud` Environments.
Cloud computing is still one of the hottest topics, and everything related to its security aspects is always current. In Cloud computing, user's data is stored on remote servers, and users may access data through the Internet connection. This method of delivery involves high chances the data may be compromised. Hence, confidentiality and integrity of data remain a severe issue. Confidentiality refers to the prevention of unauthorized access to the

data and thus making sure that only the user who has permission can access the data. Integrity refers to the accuracy and consistency of data over its entire life-cycle. Moreover, the use of a Multi-Cloud strategy today allows organizations to overcome vendor lock-in, which, in turn, permits them to use the best solutions on the market from time to time.
Chapters from 1 to 4 report some steps of my research activity aimed at proposing an innovative approach called ***``ARIANNA approach"***, mainly oriented to guarantees protection against insider attackers.

- Chapter 1 presented the ***SSME service*** aimed to provide ***Secure Storage in a Multi-Cloud Environment***. The reasons for the design choices summarized in 1.2 were motivated in depth in the chapter. In summary, the architecture of the service proposed to combine symmetric and asymmetric cryptography smartly by offering a dynamic fragmentation schema to users, which guarantees protection against insider attacks. The symmetric cryptography was directly applied to the data on the client-side, while the asymmetric cryptography was used to encrypt sensitive service information exchanged in the communication steps trough the HTTP requests Headers, as described in Section 1.4.3.
  After the design phase, the SSME architecture was developed, and finally, the secure storage service was deployed at the *Cloud Data Center of the University of Messina* [98]. The service has been made available on an experimental basis. The extended test period has proven its *correctness*, *robustness*, and *reliability* as a secure storage service in a true Multi-Cloud Environment scenario.
  The study was enriched by a service performance analysis in terms of the *"overall response time"* of the system 1.6. The analysis focused on the different processing phases intending to isolate the parts with the highest impact in terms of processing load, to minimize and limit this processing impact.
  With these results, future development will affect the design level with the implementation of architectural optimizations, which can speed up and therefore improve the performance of the service itself, while maintaining its characteristics of efficiency and reliability.

- Chapter 2 presented the Android ***ARIANNA mobile app*** that in-

tegrated the *"ARIANNA approach"* with the mobile world. The study describes how the ARIANNA app addresses confidentiality and integrity issues concerning data stored in mobile devices to protect them against insider attacks. Moreover, the description of the ARIANNA scenario 2.1 went deep in the features and motivations of the design. Section 2.4.3 highlighted how the ARIANNA app effectively enhances the confidentiality and integrity of data. To do this, a comparative table a comparative table 2.2 created considering the requirements introduced in SubSection 2.4.2 was used. The ARIANNA app was developed and released for experimental purposes for free on the Google Android App Store.
The extended test period has proven its *correctness*, *robustness*, and *reliability* as the software *enabler* extending the experimental multi-Cloud system, discussed in Chapter 1, towards the mobile world represented by the *smart devices*.
This study was completed with an analysis of the performance (2.6) of the mobile app considering the *"overall response time"* of the system 2.5.
Even in this case, the analysis aimed to focus on the different processing phases intending to isolate the parts with the highest impact in terms of processing load, to minimize and limit this processing impact. However, an additional level of analysis has been added. The analysis considered time variation occurring when the mobile app was used in different mobile networks conditions such as "ADSL" and "4G".
With these premises, a future development is to investigate some architectural changes that, considering the response time of the Cloud storage services and the features of the network environment, can manage a load balancing of the processing data in a dynamic and real-time way.

- Chapter 3 evaluated ***"how much" the overall time overhead introduced by the ARIANNA approach costs*** rather than using some row commercial Cloud storage services such as *Google Drive* [66], *Dropbox* [51] and *OpenStack Swift* [106].
The study presented a deep quantitative performance analysis (3.3.2) comparing the *"overall response time"* of the system produced by the row commercial Cloud storage services with that one produced by the

multi-Cloud approach enabled by the ARIANNA app.
Performance evaluation study considered different mobile connections ("ADSL" and "4G") and pushes to make further considerations also considering the intrinsic characteristics of the Cloud storage services (3.3.3) used by ARIANNA in the dynamic building of the multi-Cloud environments used for the simulations presented in Chapters 1 and 2.
The study finally made the comparison *"ARIANNA vs Cloud storage services"* (3.3.4) to quantify the differential introduced by the ARIANNA app. To do this, the study introduced two indicators of *Percentage Difference*, one one to quantify the upload (*U-Diff %*) and another one to quantify download (*D-Diff %*) phases.
The experimental values showed encouraging results. As shown in 3.7 and 3.6, the overhead introduced by the ARIANNA approach per file of 100 $MB$ considering a *"4G"* and a *"ADSL"* connection, can already be considered acceptable in some application domains strongly oriented to persistent storage.
It should be keep in mind that the overhead introduced of the ARIANNA approach enriches the data with those characteristics of confidentiality and integrity that the use of the individual Cloud storage services would not have guaranteed. Also, the features of the ARIANNA approach protect against insider attacks at the system administrator level.
Future developments will go in the direction of improving the differential introduced by the ARIANNA approach to broaden the adoption of this storage service in further application domains.

- Chapter 4 proposed a ***Web Framework compliant with the SSME-middleware*** policies and protocol. The Framework was implemented at the *Cloud Data Center of the University of Messina* [98] and it has been made available on an experimental basis 4.1.
  The extended test period has proven its correctness, robustness, and reliability as *enabler*. This, in consequence allows extending the experimental multi-Cloud system, already discussed in Chapter 1, towards the Web for the benefit of possible integration with Web applications and systems. The purposes and characteristics of the Framework were motivated in depth in chapter 4.1.
  This study ended with an analysis of the performance (4.2) of the

Framework in terms of the *"overall response time"* of the system (4.7). The analysis added an additional level of investigation considering the time variation occurring when the Framework was run in different networks conditions such as"public Internet" and "VPN service" (4.8,4.9).

Section II explored the systematic adoption of tools, even simulation tools, that enable more streamlined and flexible `flexible decision-making processes` allowing profiting from the most varied sources of data available. The decision-making process helps decision-makers to solve problems by examining choices and deciding on the best route to take. In a business context, a decision-making process consists of a set of steps taken by managers in an enterprise to determine the planned path for business initiatives and to set specific actions in motion. The wrong choices are equivalent to economic losses that can seriously compromise the business. Since performance evaluation in a real Cloud environment is too cost and time-consuming, simulation tools can help researchers, Cloud Service Providers, and customers to evaluate their need and their proposals.
This Section presented some aspects of the Cloud Brokerage challenge, highlighting the lack of adequate decision support simulation tools in the Cloud Service Providers (CSPs) context. The reasons why it was necessary to propose a new *Simulation Tool*, as well as the peculiarity of a new multi-criteria approach and a new algorithm, were outlined in Chapters 5 to 7.

- Chapter 5 proposed an evolution of the **J2CBROKER** *Simulation Tool* [60]. This new tool was redesigned according to the Cloud *Software as a Service* (SaaS) model and integrated into the OpenStack environment. From a technological point of view, to integrate J2CBROKER in the OpenStack environment was the best decision possible because OpenStack represents the present and the future of the open Cloud computing. This integration in the Cloud allows J2CBROKER to be owned and hosted by service providers and to be offered to consumers on-demand by following the *"utility model"*. From a simulation point of view, modeling different Cloud sites and economic offers on several criteria, the proposed case demonstrated how approaches could accommodate different scenarios characterized by a different number of instances to allocated and based on both performance and business parameters. These parameters may also come from real Data Sets, and thanks to an "optimum" balance between them, and demonstrated in

the proposed simulations, it is possible to analyze scenarios where a cooperative Cloud ecosystem can reduce the gap in competition with larger providers. J2CBROKER service was deployed at the *Cloud Data Center of the University of Messina* [98] for experimental uses. The results provided by its Brokerage Engine are available through a JSON file, CSV file. Future improvements go in the direction of developing a Web GUI to graphically show the simulation results, thus providing both service customers and providers with a better user experience.

- Chapter 6 proposed and evaluated a new ***multi-criteria approach*** that fitted the strategic perspective to capture business value from the IoT-Cloud union, with a good compromise between service level and business for the ***O&G industries***. The MCDM algorithm allowed the J2CBROKER to model a typical IoT-Cloud Brokerage scenario (6.4) and to solve the proposed decisional problem.
  In summary, starting from O alternatives (i.e., offers) and Γ decisional criteria, the goal was to identify the *"best option"* or a set of alternatives $A$, so that $2 \leq A \leq \Gamma$.
  The MCDM strategy was presented in 6.5. The motivations underlying each of the five criteria that characterize the multi-criteria approach were discussed in 6.5.4. The *case of study* that proved the goodness of the proposed MCDM strategy was detailed in 6.6.

- Chapter 7 put forward the new ***energy-aware Brokering Algorithm (eBA)***. It consists of a ***low carbon strategy*** designed to make the ***best choice in resource allocation***, based on *sustainability*, *availability*, and *costs*, in Community Cloud ecosystems. The study demonstrated how the proposed approach could discover the most convenient offers delivered by the Community Cloud service providers through a balance between sustainability and cost-saving requirements at a Cloud Broker level. Modeling different Cloud sites, and the related economic offers on the sustainability criterion using J2CBROKER, the simulation of the algorithm demonstrated how it could be possible to manage different scenarios characterized by a different number of instances to allocate. The proposed approach allows characterizing offers based on the geographic area where the offered Cloud resources are available, the energy-efficiency of the Cloud site, and service parameters. Future improvements will go in the direction of investigating

a strategy to balance sustainability with several other performance metrics smartly. Thanks to an optimum balance between sustainability, cost, and service parameters, a Community Cloud ecosystem can reduce the gap in competition with larger providers, towards an encouraging *"green"* resource sharing among Community Clouds.

Section III focused on the need for `enhancement of human capital` through the design and implementation of new software solutions that integrate the best of the new technologies such as Cloud, Mobile, Artificial Intelligence, Edge systems, Big Data, and Internet of Things, aiming at the development of Society. The Section presented some systems, applications, and new services that use *Machine Learning* and *Gamification* concepts along with *Cloud and IoT Technologies*. These new tools meet the emerging needs of a new generation of students that are facing new social scenarios and new emotional conditions.
This Section closes with the description of the activities and objectives achieved by an Erasmus+ project called IOT-OPEN.EU. This project has been awarded as *Best Practices* from the National Agency for the Erasmus+ program in Poland.
IOT-OPEN.EU combines good practices and new innovative technologies that make it possible to carry out remote laboratory activities for technology students in the field of IoT and embedded systems.
This experience shows how it is possible to put technology at the service of society to overcome barriers and guarantee rights. Specifically, in this case, the right to education for a generation of students affected by the lockdown due to the COVID-19 pandemic.

- Chapter 8 introduced the *"Virtual Study Partner - VSP"* Android app which consists of a cognitive training tool that integrates both machine learning concepts and Cloud technologies.
  The VSP app provides study support to young people in school-age, such as *digital native* [112], in dealing with their homework activities. The VSP app was developed and released for experimental purposes for free on the Google Android App Store.
  The study described the design choices of the app 8.2.1, its software architecture 8.4 and its operating principles (8.4.1 and 8.4.2). Currently, VSP is used experimentally in a class of a primary school in the province of Messina.

Future improvements will go in the direction of making changes and improvements to the Android app. In this regard, the app could automatically ask questions to users to better understand and learn the user's preparation. Another point is to adopt the use of the app for blind people through the use of messages and voice commands. Moreover, it is at an advanced stage of design and development of the Restful Web service that will allow further and more in-depth analysis of the data collected from the users. This Web service will lay the foundations for planning strategies for smarter study activities, and it will enable the predictive analysis of the learning progress of users.

- Chapter 9 briefly presents the new *Intelligent Tutoring System - ITS* called *"Virtual Study Buddy - VSB"*. The goal of the study was to overcome technical maintenance problems and well-known longevity problems, which were the leading causes that prevented ITS from being permanently incorporated into the education systems.
  The study argued how, finally, these unsolved technical problems represent the real motivation that has prevented the adoption of ITS systems in educational systems despite the fact that research in the past 30 years has shown that their use brings tangible benefits. The VSB is currently under development.
  A trial of the system is used in a class of a primary school in the province of Messina. However, it has not yet been released publicly.
  The choices and the theoretical and technological reasons that are guiding this project are in 9.2.1. The VSB scenario is underlined in 9.4.1. The techniques used to involve the participants are summarised in 9.4.2. Future improvements will go in the direction to make changes and improvements to the pilot version. The data collected during the trial will indicate the direction to follow to make the *VSB* more user-friendly, economically sustainable, easy to maintain even by non-technical personnel, to encourage its adoption on a larger scale.

- Chapter 10 presented ***IOT-OPEN.EU***, the educational project within the ***Erasmus+ Key Action 2 framework*** representing an ***excellent and timely solution*** that may be adopted worldwide by schools and universities which were forced by their governments to timely showdown due to the recent outbreak of the SARS-COV-2 virus and the related ***COVID-19 pandemic***.

The recent outbreak of the SARS-COV-2 virus and the related COVID-19 pandemic throughout the world has caused governments across the world to shut down schools and universities from one day to another. The lockdown represents the only strategy against the spread of the coronavirus that is causing the disease. This situation has forced a lot of universities and schools to switch from the traditional classrooms to virtual classrooms. However, this mode of learning is not working well for laboratory subjects and courses as it is not straight forward to handle laboratory subjects and practices that require access to hardware resources remotely. We have presented current advances in distance learning and have discussed remote laboratory models. The Chapter presented the IOT-OPEN.EU remote laboratory infrastructure and IoT courses, which were designed and implemented as part of the IOT-OPEN.EU ERASMUS+ project. Moreover, it discussed IoT related technologies and platforms that can be leveraged for IoT training. The presented solution has been introduced into the participating universities' curriculum on the Internet of Things. Pilots testing performed in the Silesian University of Technology, covering classical, online courses and use of VREL labs and IOT-OPEN.EU project-created content in the years 2017- 2020, present and prove usability and a reasonable approach to distance learning with this kind of tool. They also indicate the growing popularity of the mixed learning model, where students use on-site and online materials parallel. At the moment, over 500 students are studying or have already studied using IOT-OPEN.EU materials and tools on-site and close to 8000 students enrolled for IOT-OPEN.EU online courses, including the use of the VREL system. Future improvements go in the direction of developing a framework for remote laboratory courses in IoT, AI, Big data, and automation where the data captured by the IoT nodes will be analyzed, and the results used to control cyber-physical systems.

# Bibliography


[1] The Digital Single Market. https://ec.europa.eu/digital-single-market/en/digital-single-market.

[2] MarketsandMarkets. http://www.marketsandmarkets.com/.

[3] The OpenStack Foundation. https://www.openstack.org/foundation/.

[4] The ISO/IEC 25010. http://iso25000.com/index.php/en/iso-25000-standards/iso-25010.

[5] The Ministry of Economy Trade and Industry (METI) Japan Project - Enhancing the Energy Efficiency and Use of Green Energy in Data Centers. http://home.jeita.or.jp/greenit-pc/sd/pdf/ds2.pdf.

[6] Cisco Global Cloud Index: Forecast and Methodology, 2015/2020.

[7] The MQ Telemetry Transport (MQTT) Protocol. http://mqtt.org/.

[8] 2010, power Efficiency Comparisonof Enterprise - Class Blade Servers and Enclosures, A Dell Technical White Paper. http://www.dell.com/.

[9] 2011, the ISO/IEC 25010. http://iso25000.com/index.php/en/iso-25000-standards/iso-25010.

[10] 2012, the Ministry of Economy Trade and Industry (METI) Japan Project - Enhancing the Energy Efficiency and Use of Green Energy in Data Centers. http://home.jeita.or.jp/greenit-pc/sd/pdf/ds2.pdf.

[11] "New data center energy efficiency evaluation index. dppe (datacenter performance per energy) measurement guidelines. (ver. 205)," Green IT Promotion Council, Tech. Rep., 2012.

[12] 2014, the Carbon Dioxide Intensity Of Electricity, Intergovernmental Panel on Climate Change (IPCC) Report. http://www.ipcc.ch/.

[13] 2015, the 2015 Global 100 Most Sustainable Corporations in the World index. http://www.corporateknights.com/reports/global-100/2015-global-100-results/.

[14] ABIresearch, *Global Cybersecurity Index & Cyberwellness Profiles Report*, 2015.

[15] W. Admiraal, L. Post, P. Guo, N. Saab, S. Makinen, O. Rainio, J. Vuori, J. Bourgeois, G. Kortuem, and G. Danford, "Students as future workers: Cross-border multidisciplinary learning labs in higher education," *International Journal of Technology in Education and Science*, vol. 3, no. 2, pp. 85–94, February 2019. [Online]. Available: https://www.learntechlib.org/p/207262

[16] K. Akherfi, H. Harroud, and M. Gerndt, "A mobile cloud middleware for data storage and integrity," in *2015 International Conference on Cloud Technologies and Applications (CloudTech)*, 2015, pp. 1–7.

[17] K. Aleksic-Maslac, M. Rasic, and P. Vranesic, "Influence of gamification on student motivation in the educational process in courses of different fields," in *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2018, pp. 0783–0787.

[18] B. Alhalabi, D. Marcovitz, K. Hamza, and M. Larrondo Petrie, "Remote labs: An innovative leap in the world of distance education," 10 2000.

[19] H. S. Alqahtani and P. Sant, "A multi-cloud approach for secure data storage on smart device," in *2016 Sixth International Conference on Digital Information and Communication Technology and its Applications (DICTAP)*, 2016, pp. 63–69.

[20] Amazon. (2016) Cloud is the New Normal AWS Government, Education, & Nonprofits Blog. [Online]. Available: https://aws.amazon.com/it/blogs/publicsector/cloud-is-the-new-normal/

[21] M. Andreolini, S. Casolari, M. Colajanni, and M. Messori, "Dynamic load management of virtual machines in a cloud architecture," in *ICST CLOUDCOMP*, 2009.

[22] Android. (2019) The world's most popular mobile OS - From phones and watches to cars and TVs, customize your digital life with Android. [Online]. Available: https://www.android.com/

[23] Android-Statista. (2019) Number of available applications in the Google Play Store from December 2009 to December 2018. [Online]. Available: https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/

[24] G. Arnold, "Intelligent systems: A new industrial revolution [viewpoint]," *IEEE Electrification Magazine*, vol. 4, no. 1, pp. 64–63, 2016.

[25] E. Badidi, "A framework for software-as-a-service selection and provisioning," *arXiv preprint arXiv:1306.1888*, 2013.

[26] V. R. Balasaraswathi and S. Manikandan, "Enhanced security for multi-cloud storage using cryptographic data splitting with dynamic approach," in *2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies*, 2014, pp. 1190–1194.

[27] Y. Bao, L. Ren, L. Zhang, X. Zhang, and Y. Luo, "Massive sensor data management framework in cloud manufacturing based on hadoop," in *Industrial Informatics (INDIN), 2012 10th IEEE International Conference on*, 2012, pp. 397 –401.

[28] B. Barros, T. Read, and M. F. Verdejo, "Virtual collaborative experimentation: An approach combining remote and local labs," *IEEE Transactions on Education*, vol. 51, no. 2, pp. 242–250, 2008.

[29] K. Bicakci, D. Yavuz, and S. Gurkan, "Twincloud: A client-side encryption solution for secure sharing on clouds without explicit key management," 06 2016.

[30] K. Bicakci, D. D. Yavuz, and S. Gurkan, "Twincloud: A client-side encryption solution for secure sharing on clouds without explicit key management," *CoRR*, vol. abs/1606.04705, 2016.

[31] D. Bradshaw, G. Cattaneo, R. Lifonti, and J. Simcox, "Smart 2013/0043 - uptake of cloud in europe," IDC EMEA, Tech. Rep., 2015.

[32] L. Briz, A. Pereira, J. Juanes-Méndez, and F. García-Peñalvo, *Evaluation of M-Learning among students according to their behaviour with apps*, 05 2017, pp. 37–48.

[33] R. Buyya and M. Murshed, "Gridsim: A toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing," *CONCURRENCY AND COMPUTATION: PRACTICE AND EXPERIENCE (CCPE*, vol. 14, no. 13, pp. 1175–1220, 2002.

[34] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, and R. Buyya, "Cloudsim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Softw. Pract. Exper.*, vol. 41, no. 1, pp. 23–50, 2011. [Online]. Available: http://dx.doi.org/10.1002/spe.995

[35] E. Casalicchio, V. Cardellini, G. Interino, and M. Palmirani, "Research challenges in legal-rule and qos-aware cloud service brokerage," *Future Generation Computer Systems*, vol. 78, pp. 211 – 223, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X16306641

[36] S. CHANDRASEKAR, "A review of literature on cloud brokerage services," *International Journal of Computer Science and Business Informatics*, vol. 10, no. 1, 2014.

[37] Y.-S. Chen and Y.-R. Chen, "Context-Oriented Data Acquisition and Integration Platform for Internet of Things," in *Technologies and Applications of Artificial Intelligence (TAAI), 2012 Conference on*, 2012, pp. 103 –108.

[38] Cloud-Natural-Language. (2019) Cloud Natural Language - Extract significant information from unstructured text thanks to Google's learning machine. [Online]. Available: https://cloud.google.com/natural-language/

[39] C. Colwell, E. Scanlon, and M. Cooper, "Using remote laboratories to extend access to science and engineering," *Computers & Education*, vol. 38, no. 1, pp. 65 – 76, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S036013150100077X

[40] M. Despotovic-Zrakic, A. Labus, Z. Bogdanovic, M. Labus, and S. Milinovic, "A virtual laboratory for teaching internet of things," in *10th International Conference on Virtual Learning ICVL 2015*, vol. 2005, 2005, pp. 259–264.

[41] R. Di Pietro, D. G. Campanile, and S. Distefano, "Virtual study partner: A cognitive training tool in education," in *2019 IEEE International Conference on Smart Computing (SMARTCOMP)*, 2019, pp. 192–197.

[42] R. Di Pietro, M. Scarpa, M. Giacobbe, and F. Oriti, "Wip: Arianna: A mobile secure storage approach in multi-cloud environment," in *2018 IEEE International Conference on Smart Computing (SMARTCOMP)*, 2018, pp. 273–275.

[43] R. Di Pietro, M. Scarpa, M. Giacobbe, and A. Puliafito, "An approach to enhancing confidentiality and integrity on mobile multi-cloud systems: The

arianna experience," in *2018 Fifth International Conference on Internet of Things: Systems, Management and Security*, 2018, pp. 244–249.

[44] R. Di Pietro, M. Scarpa, and A. Puliafito, "How much enhancing confidentiality and integrity on data can affect mobile multi-cloud: The "arianna" experience," in *2019 IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, 2019, pp. 346–350.

[45] R. Di Pietro and S. Distefano, "An intelligent tutoring system tool combining machine learning and gamification in education," in *Software Technology: Methods and Tools*, M. Mazzara, J.-M. Bruel, B. Meyer, and A. Petrenko, Eds. Cham: Springer International Publishing, 2019, pp. 218–226.

[46] R. Di Pietro, M. Giacobbe, C. Puliafito, and M. Scarpa, *J2CBROKER as a Service: A Service Broker Simulation Tool Integrated in OpenStack Environment*. Cham: Springer International Publishing, 2019, pp. 269–285. [Online]. Available: https://doi.org/10.1007/978-3-319-92378-9_17

[47] R. Di Pietro, M. Scarpa, M. Giacobbe, and A. Puliafito, "Secure storage as a service in multi-cloud environment," in *Ad-hoc, Mobile, and Wireless Networks*. Cham: Springer International Publishing, 2017, pp. 328–341.

[48] R. Dixit, M. Nirgude, and P. Yalagi, "Gamification: An instructional strategy to engage learner," in *2018 IEEE Tenth International Conference on Technology for Education (T4E)*, 2018, pp. 138–141.

[49] N. Dlodlo and A. Smith, "The internet-of-things in remote-controlled laboratories," 09 2011.

[50] K. Dolui, S. Mukherjee, and S. Datta, "Smart device sensing architectures and applications," in *International Computer Science and Engineering Conference (ICSEC), 2013*, 2013, pp. 91–96.

[51] Dropbox-2018. (2018) Dropbox. [Online]. Available: https://www.dropbox.com

[52] Dropbox-2019. (2019) Dropbox API for Developers. [Online]. Available: https://www.dropbox.com/developers/

[53] Facebook. (2019) Facebook - Social Network. [Online]. Available: https://www.facebook.com

[54] A. F. C. A. Fathoni and D. Delima, "Gamification of learning kanji with "Musou Roman" game," in *2016 1st International Conference on Game, Game Art, and Gamification (ICGGAG)*, 2016, pp. 1–3.

[55] Firebase. (2019) A comprehensive mobile development platform. [Online]. Available: https://firebase.google.com/

[56] B. Fogg, "Persuasive technology," in *Persuasive Technology*, ser. Interactive Technologies, B. Fogg, Ed. San Francisco: Morgan Kaufmann, 2003.

[57] J. Gee, "What video games have to teach us about learning and literacy," *Computers in Entertainment*, vol. 1, p. 20, 10 2003.

[58] M. Giacobbe, A. Celesti, M. Fazio, M. Villari, and A. Puliafito, "A sustainable energy-aware resource management strategy for iot cloud federation," in *2015 IEEE International Symposium on Systems Engineering (ISSE)*, 2015, pp. 170–175.

[59] M. Giacobbe, R. Di Pietro, A. Longo Minnolo, and A. Puliafito, "Evaluating information quality in delivering iot-as-a-service," in *2018 IEEE International Conference on Smart Computing (SMARTCOMP)*, 2018, pp. 405–410.

[60] M. Giacobbe, R. D. Pietro, C. Puliafito, and M. Scarpa, "J2CBROKER: A service broker simulation tool for cooperative clouds," in *10th EAI International Conference on Performance Evaluation Methodologies and Tools (Valuetools 2016)*, 2016.

[61] M. Giacobbe, R. D. Pietro, A. Zaia, and A. Puliafito, "The internet of things in oil and gas industry: A multi criteria decision making brokerage strategy," in *Special Issue, 4th International Conference on Automation, Control Engineering and Computer Science (ACECS 2017), Proceedings of Engineering and Technology - PET*, vol. 21, 2017, pp. 47–52.

[62] M. Giacobbe, A. Celesti, M. Fazio, M. Villari, and A. Puliafito, "Towards energy management in cloud federation: A survey in the perspective of future sustainable and cost-saving strategies," *Computer Networks*, vol. 91, pp. 438 – 452, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128615002911

[63] M. Giacobbe, M. Scarpa, R. D. Pietro, and A. Puliafito, "An energy-aware brokering algorithm to improve sustainability in community cloud," in *Proceedings of the 6th International Conference on Smart Cities and Green ICT Systems*, 2017, pp. 166–173.

[64] K. Giannakouris and M. Smihily, "Cloud computing - statistics on the use by enterprises," Nuffield College, Oxford, UK, Discussion paper, Tech. Rep., 2016.

[65] G. Gokhan and N. Saleem, "Transforming traditional labs into virtual computing labs for distance education," *International Journal of Online Engineering*, vol. 4, 02 2008.

[66] Google. (2018) Google Drive cloud storage & file backup for photos, docs & more. [Online]. Available: https://www.google.com/drive/

[67] Google-Cloud. (2019) Google Cloud Platform products and services. [Online]. Available: https://cloud.google.com/

[68] Google-Cloud-Speech. (2019) Google Cloud Speech-to-Text - Conversion from voice to text based on machine learning technology, available for short or long duration audio. [Online]. Available: https://cloud.google.com/speech-to-text/

[69] Google-Cloud-Vision. (2019) Google Cloud Vision - Extract meaningful information from your images thanks to our powerful pre-trained API models or easily train custom artificial vision models with AutoML Vision. [Online]. Available: https://cloud.google.com/vision/

[70] Google-Knowledge-Graph. (2019) Google Knowledge Graph Search API. [Online]. Available: https://developers.google.com/knowledge-graph/

[71] Google-Natural-Language-Content-Categories. (2019) Cloud Natural Language API - Content Categories. [Online]. Available: https://cloud.google.com/natural-language/docs/categories

[72] Google-Nexus. (2018) Nexus tech specs Check out the tech specs for these Nexus devices. [Online]. Available: https://support.google.com/nexus/answer/6102470?hl=en

[73] Google-Play-Store. (2019) Google Play Store. [Online]. Available: https://play.google.com/store

[74] GoogleDrive-API. (2019) Google Drive APIs. [Online]. Available: https://console.developers.google.com/apis/

[75] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future Generation Comp. Syst.*, vol. 29, no. 7, pp. 1645–1660, 2013.

[76] T. Gunasekhar, K. T. Rao, and M. T. Basu, "Understanding insider attack problem and scope in cloud," in *2015 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2015]*, 2015, pp. 1–6.

[77] M. Hamze, N. Mbarek, and O. Togni, "Broker and federation based cloud networking architecture for iaas and naas qos guarantee," in *2016 13th IEEE Annual Consumer Communications Networking Conference (CCNC)*, 2016, pp. 705–710.

[78] J. He, Dan Chia-Tien Lo, Y. Xie, and J. Lartigue, "Integrating internet of things (iot) into stem undergraduate education: Case study of a modern technology infused courseware for embedded system course," in *2016 IEEE Frontiers in Education Conference (FIE)*, 2016, pp. 1–9.

[79] Y. Heryadi and K. Muliamin, "Gamification of m-learning mandarin as second language," in *2016 1st International Conference on Game, Game Art, and Gamification (ICGGAG)*, 2016, pp. 1–4.

[80] Hesiod, *Theogony 947*.

[81] Homer, *Odyssey 11.320*.

[82] HPCALab. (2019) High Performance Computing and Application Laboratory (HPCALab) - University of Messina. [Online]. Available: http://hpca.unime.it/

[83] E. Hulthén and P. Papadopoulou, "Book-app as course literature in cdio-based project courses- students' perspectives," 2018.

[84] W. Hutzel and R. Furter, "International partnership using remotely accessed labs," vol. 2005, 11 2005, pp. T4D – 1.

[85] IBM. (2017) IBM cyber security intelligence index 2016. [Online]. Available: http://www.ibm.com/security/data-breach/cyber-security-index.html

[86] L. M. Jeno, P. Adachi, J. A. Grytnes, V. Vandvik, and E. Deci, "The effects of m-learning on motivation, achievement and well-being: A self-determination theory approach," *British Journal of Educational Technology*, pp. 1–15, 08 2018.

[87] F. Jrad, "A service broker for intercloud computing," Ph.D. dissertation, Karlsruhe, Karlsruher Institut für Technologie (KIT), Diss., 2014, 2014.

[88] J. Jung and B. Song, "The possibility of wireless sensor networks for industrial pipe rack safety monitoring," in *2014 47th Hawaii International Conference on System Sciences*, 2014, pp. 5129–5134.

[89] Y. Kessaci, N. Melab, and E.-G. Talbi, "A pareto-based metaheuristic for scheduling hpc applications on a geographically distributed cloud federation," *Cluster Computing*, vol. 16, no. 3, pp. 451–468, 2013.

[90] B. Keswani, D. Banerjee, and P. Patni, "Role of technology in education: A 21st century approach," *Journal of Commerce and Information Technology*, vol. 8, pp. 53–59, 06 2008.

[91] A. Kist, A. Maxwell, P. Gibbings, R. Fogarty, W. Midgley, and K. Noble, "Engineering for primary school children: learning with robots in a remote access laboratory," 01 2011, pp. 586–591.

[92] D. Kliazovich, P. Bouvry, and S. U. Khan, "Greencloud: a packet-level simulator of energy-aware cloud computing data centers," *The Journal of Supercomputing*, vol. 62, no. 3, pp. 1263–1283, 2012. [Online]. Available: http://dx.doi.org/10.1007/s11227-010-0504-1

[93] B. Klimova and M. Valis, "Smartphone applications can serve as effective cognitive training tools in healthy aging," *Frontiers in aging neuroscience*, vol. 9, pp. 436; 436–436, 2018. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/29379432

[94] H. Ku, T. Ahfock, and T. Yusaf, "Remote access laboratories in australia and europe," *European Journal of Engineering Education*, vol. 36, no. 3, pp. 253–268, 2011. [Online]. Available: https://doi.org/10.1080/03043797.2011.578244

[95] C. Ling, D. Harnish, and R. Shehab, "Educational apps: Using mobile applications to enhance student learning of statistical concepts," *Human Factors and Ergonomics in Manufacturing and Service Industries*, vol. 24, 09 2014.

[96] L. Liu, O. D. Vel, Q. L. Han, J. Zhang, and Y. Xiang, "Detecting and preventing cyber insider threats: A survey," *IEEE Communications Surveys Tutorials*, vol. 20, no. 2, pp. 1397–1417, 2018.

[97] S. Madria, V. Kumar, and R. Dalvi, "Sensor Cloud: A Cloud of Virtual Sensors," *IEEE Software*, vol. 31, no. 2, pp. 70–77, 2014.

[98] MDSLab. (2018) Mobile and Distributed Systems Laboratory at University of Messina. [Online]. Available: http://mdslab.unime.it/

[99] G. Merlino, D. Bruneo, S. D. Stefano, F. Longo, and A. Puliafito, "Stack4Things: Integrating IoT with OpenStack in a Smart City context," in *Proceedings of the 2014 International Conference on Smart Computing Workshops (SMARTCOMP Workshops)*. Hong Kong, China, 5 November 2014: IEEE Computer Society, 2014, pp. 21–28.

[100] A. Molnar, "The effect of interactive digital storytelling gamification on microbiology classroom interactions," in *2018 IEEE Integrated STEM Education Conference (ISEC)*, 2018, pp. 243–246.

[101] S. Murugesan and I. Bojanova, *Community Clouds*. Wiley-IEEE Press, 2016, pp. 744–. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7493782

[102] Natural-Language-API. (2019) Natural Language API Basics. [Online]. Available: https://cloud.google.com/natural-language/docs/basics)

[103] A. Núñez, J. L. Vázquez-Poletti, A. C. Caminero, G. G. Castañé, J. Carretero, and I. M. Llorente, "icancloud: A flexible and scalable cloud infrastructure simulator," *J. Grid Comput.*, vol. 10, no. 1, pp. 185–209, 2012. [Online]. Available: http://dx.doi.org/10.1007/s10723-012-9208-5

[104] OpenStack, "Official Web Site," https://www.openstack.org/, online; accessed 28 June 2017.

[105] Openstack. (2018) Keystone the OpenStack Identity Service. [Online]. Available: https://docs.openstack.org/keystone/latest/

[106] OpenStack. (2018) Swift the OpenStack Object Storage Service. [Online]. Available: https://docs.openstack.org/swift/latest/

[107] Openstack. (2019) Keystone the OpenStack Identity Service. [Online]. Available: https://docs.openstack.org/keystone/latest/

[108] OpenStack. (2019) Swift the OpenStack Object Storage Service. [Online]. Available: https://docs.openstack.org/swift/latest/

[109] E. Otoakhia, T. Jenmanachaiyakun, A. Afaneh, S. Alzebda, M. Mani, O. Sonbul, and A. N. Kalashnikov, "Embedded web server for remote laboratory access for undergraduate students studying electronic engineering," in *2011 IEEE International Symposium of Circuits and Systems (ISCAS)*, 2011, pp. 337–340.

[110] I. Patiniotakis, Y. Verginadis, and G. Mentzas, "Pulsar: preference-based cloud service selection for cloud service brokers," *Journal of Internet Services and Applications*, vol. 6, no. 1, pp. 1–14, 2015. [Online]. Available: http://dx.doi.org/10.1186/s13174-015-0042-4

[111] R. Pottier and J. M. Menaud, "Trustydrive, a multi-cloud storage service that protects your privacy," in *2016 IEEE 9th International Conference on Cloud Computing (CLOUD)*, 2016, pp. 937–940.

[112] M. Prensky, "Digital natives, digital immigrants part 1," *On the Horizon*, vol. 9, no. 5, pp. 1–6, 2001. [Online]. Available: https://doi.org/10.1108/10748120110424816

[113] M. Radi, "Efficient service broker policy for large-scale cloud environments," *CoRR*, vol. abs/1503.03460, 2015. [Online]. Available: http://arxiv.org/abs/1503.03460

[114] L. M. Romero-RodrÃguez, M. S. RamÃrez-Montoya, and J. R. V. GonzÃ¡lez, "Gamification in moocs: Engagement application test in energy sustainability courses," *IEEE Access*, vol. 7, pp. 32 093–32 101, 2019.

[115] Y. Rosmansyah and M. R. Rosyid, "Mobile learning with gamification for alquran memorization," in *2017 International Conference on Information Technology Systems and Innovation (ICITSI)*, 2017, pp. 378–383.

[116] M. A. B. Sahbudin, R. Di Pietro, and M. Scarpa, "A web client secure storage approach in multi-cloud environment," in *2019 4th International Conference on Computing, Communications and Security (ICCCS)*, 2019, pp. 1–7.

[117] D. Schwab, L. Yang, K. Winters, M. Jallouk, E. Smith, and A. Claiborne, "A secure mobile cloud photo storage system," in *2017 26th International Conference on Computer Communication and Networks (ICCCN)*, 2017, pp. 1–5.

[118] R. Sell, T. Ruutmann, and S. Seiler, "Inductive principles in engineering pedagogy on the example of remote labs," 09 2013.

[119] R. Sell and S. Seiler, "Improvements of multi-disciplinary engineering study by exploiting design-centric approach, supported by remote and virtual labs," *International Journal of Engineering Education*, vol. 28, pp. 759–766, 01 2012.

[120] R. Sell, S. Seiler, and D. Ptasik, "Embedded system and robotic education in a blended learning environment utilizing remote and virtual labs in the cloud, accompanied by robotic homelab kit," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 7, pp. 26–33, 10 2012.

[121] L. Simmons, A. Crook, C. Cannonier, and C. Simmons, "There's an app for that: The impact of reminder apps on student learning and anxiety," *Journal of Education for Business*, vol. 93, no. 5, pp. 185–195, 2018. [Online]. Available: https://doi.org/10.1080/08832323.2018.1441120

[122] L. Sun, H. Dong, F. K. Hussain, O. K. Hussain, and E. Chang, "Cloud service selection: State-of-the-art and future research directions," *Journal of Network and Computer Applications*, vol. 45, pp. 134 – 150, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S108480451400160X

[123] S. Sundareswaran, A. Squicciarini, and D. Lin, "A brokerage-based approach for cloud service selection," in *Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on*, 2012, pp. 558–565.

[124] J. Tang, Y. Cui, Q. Li, K. Ren, J. Liu, and R. Buyya, "Ensuring security and privacy preservation for cloud data services," *ACM Comput. Surv.*, vol. 49, no. 1, pp. 13:1–13:39, 2016. [Online]. Available: http://doi.acm.org/10.1145/2906153

[125] K. Tokarz, P. Czekalski, G. Drabik, J. Paduch, S. Distefano, R. Di Pietro, G. Merlino, C. Scaffidi, R. Sell, and G. Suila Kuaban, "Internet of things network infrastructure for the educational purpose," in *2020 IEEE Frontiers in Education Conference (FIE)*, 2020.

[126] Ã. Tóth and S. Tóvölgyi, "The introduction of gamification: A review paper about the applied gamification in the smartphone applications," in *2016 7th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, 2016, pp. 000 213–000 218.

[127] UniME. (2019) Università degli Studi di Messina - Piazza Pugliatti, 1 - 98122 Messina, Italy (UE). [Online]. Available: https://international.unime.it/

[128] M. Usha, J. Akilandeswari, and A. Fiaz, "An efficient qos framework for cloud brokerage services," in *International Symposium on Cloud and Services Computing (ISCOS)*, 2012, pp. 76–79.

[129] M. B. Vaidya and S. Nehe, "Data security using data slicing over storage clouds," in *2015 International Conference on Information Processing (ICIP)*, 2015, pp. 322–325.

[130] M. Verdejo, B. Barros, R. Antón, and T. Read, "The design and implementation of experimental collaborative learning in a distance learning context," 04 2020.

[131] M. F. Verdejo, B. Barros, T. Read, and M. Rodriguez-Artacho, *Designing a CSCL environment for experimental learning in a distance learning context.* Boston, MA: Springer US, 2007, pp. 139–153. [Online]. Available: https://doi.org/10.1007/978-0-387-71136-2_9

[132] C. Viegas, A. Pavani, N. Lima, A. Marques, I. Pozzo, E. Dobboletta, V. Atencia, D. Barreto, F. Calliari, A. Fidalgo, D. Lima, G. Temporão, and G. Alves, "Impact of a remote lab on teaching practices and student learning," *Computers & Education*, vol. 126, pp. 201 – 216, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0360131518301878

[133] Virtual-Study-Partner. (2019) Virtual Study Partner - Istruzione. [Online]. Available: https://play.google.com/store/apps/details?id=com.knowledgepkg.domenicogiacomocampanile.knowledgeapp

[134] E. Volk, A. Tenschert, M. Gienger, A. Oleksiak, L. Sisó, and J. Salom, "Improving energy efficiency in data centers and federated cloud environments," in *CGC*, 2013, pp. 443–450.

[135] B. Wickremasinghe, A. Prof, and R. B. Contents, "Cloudanalyst: A cloudsim-based tool for modelling and analysis of large scale cloud computing environments," 2009.

[136] Wikipedia. (2019) Wikipedia, the free encyclopedia. [Online]. Available: https://www.wikipedia.org/

[137] G. Wu, J. Chen, W. Bao, X. Zhu, W. Xiao, J. Wang, and L. Liu, "Meccas: Collaborative storage algorithm based on alternating direction method of multipliers on mobile edge cloud," in *2017 IEEE International Conference on Edge Computing (EDGE)*, 2017, pp. 40–46.

[138] M. Yuriyama and T. Kushida, "Sensor-cloud infrastructure - physical sensor management with virtualized sensors on cloud computing," in *13th International Conference on Network-Based Information Systems (NBiS), 2010*, 2010, pp. 1–8.

[139] G. Zichermann and C. Cunningham, *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps.* O'Reilly Media, Inc., 01 2011.

[140] K. Zkik, G. Orhanou, and S. E. Hajji, "Secure scheme on mobile multi cloud computing based on homomorphic encryption," in *2016 International Conference on Engineering MIS (ICEMIS)*, 2016, pp. 1–6.