COPPE

Instituto Alberto Luiz Coimbra de
Pós-Graduação e Pesquisa de Engenharia

UFRJ

DIRECTOR:
A CLOUD MICROSERVICE SELECTION FRAMEWORK

Marcelo de França Costa

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia de Sistemas e Computação.

Orientadora: Cláudia Maria Lima Werner

Rio de Janeiro
Junho de 2019

DIRECTOR:
A CLOUD MICROSERVICE SELECTION FRAMEWORK

Marcelo de França Costa

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

____
Prof. Cláudia Maria Lima Werner, D.Sc.

____
Prof. Toacy Cavalcante de Oliveira, D.Sc.

____
Prof. Rodrigo Pereira dos Santos, D.Sc.

____
Prof. Márcio de Oliveira Barros, D.Sc.

____
Prof. Eduardo Santana de Almeida, D.Sc.

____
Prof. Elisa Yumi Nakagawa, D.Sc.

RIO DE JANEIRO, RJ – BRASIL
JUNHO DE 2019

Costa, Marcelo de França

DIRECTOR: A Cloud Microservice Selection Framework/ Marcelo de França Costa. – Rio de Janeiro: UFRJ/COPPE, 2019.

XIV, 185 p.: il.; 29,7 cm.

Orientadora: Cláudia Maria Lima Werner

Tese (doutorado) – UFRJ/ COPPE/ Programa de Engenharia de Sistemas e Computação, 2019.

Referências Bibliográficas: p. 101-116.

1. Software Ecosystems. 2. Software Architecture. 3. Cloud Computing. 4. Cognitive Computing. 5. Microservices. 6. Software Engineering. I. Werner, Cláudia Maria Lima. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

"I have fought the good fight, I have finished the race, I have kept the faith.
Now there is in store for me the crown of righteousness, which the Lord,
the righteous Judge, will award to me on that day – and not only to me,
but also to all who have longed for his appearing."

(2 Timothy 4:7-8, Holy Bible, New International Version)

Acknowledgments

First, from an academic perspective, I would like to acknowledge the trust that my adviser, Professor Claudia Werner, has put in me – her first part-time PhD candidate. She has given me a lifetime opportunity to learn and to contribute to the Software Engineering community, being part of one of Brazil's most prestigious universities. I was very fortunate to count with her assistance during these five years. Thank you, Professor.

I would like also to recognize my former Master's adviser, Professor Márcio Barros, for all the encouragement and support during these challenge years. Professor Rodrigo Santos for the immense availability of time. Professor Klaus Schmid for the free advices regarding the evolution of my thesis' proposal. Professor Oswaldo Barcia for helping me with the evaluation analysis. My sincere thanks to you all.

Then, from a professional perspective, I would like to start by thanking Matheus Americano, Renata Proença, Luciana Povoa and Bruno Martins (plus our CTO Raymundo Barros), from Grupo Globo, for all the assistance (financially and motivationally speaking) regarding my PhD course. I would like also to thank Professors Rosymere Cersosimo, João Claudio and Maura Luiza, from FAETEC, for understanding the challenges of getting a PhD while teaching. Finally, I would really like to thank Enio Moura, Argemiro Lima and Thiago Guimaraes, my colleagues from IBM, for helping me pave the way for this PhD research, since the beginning of this journey. I cannot stress enough how your support was pivotal to this conquest. For that, I am truly grateful.

Lastly, from a personal perspective, I would like to thank my friends André Farzat, Natacha Pragana and Anderson Paulo for all the good moments that, somehow, have alleviated the pressure of this endeavor. Very special thanks to my dear friend Fábio Farzat, for giving me the "final push" to start this long-term project.

Then, obviously, I want to thank those who helped me in a way that nobody else could. My parents Darcy Costa and Fátima França, besides my brothers and cousins, for all the love and unconditional aid. My wife, Amanda Barcia, for just about everything and for being the best part of my days. Additionally, my little precious, my daughter Rafaela França, for giving me an additional motivation for concluding this quest. I love you all from the bottom of my heart.

Thank You, Lord, for the all the unique opportunities, challenges, gifts and ordeals… For allowing me to feel Your presence along the way and for my faith in You!

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

DIRECTOR:
UM FRAMEWORK PARA SELEÇÃO DE MICROSSERVIÇOS NA NUVEM

Marcelo de França Costa

Junho/2019

Orientadora: Cláudia Maria Lima Werner

Programa: Engenharia de Sistemas e Computação

O campo de pesquisa de Ecossistemas de Software tem recebido cada vez mais atenção da academia e da indústria, já que organizações os tem adotado como plataforma colaborativa para alcançar inovações mais rapidamente. Mais recentemente, com o advento da Computação em Nuvem, modernos ecossistemas passaram a ser ofertados como serviço, permitindo que atores contribuam, mas também comercializem suas próprias soluções, reutilizando ativos de software disponíveis, popularmente, no formato de microsserviços, isto é, uma funcionalidade bem específica, normalmente exposta através de tecnologias Web. Com a atual proliferação de plataformas e microsserviços, um desafio relevante para os arquitetos de software é adquirir o componente mais adequado, frente a um conjunto de requisitos e prioridades. Neste contexto, propomos DIRECTOR: Um framework para seleção de microsserviços na nuvem, baseado em perspectivas complementares (técnica, social e semântica), ou seja, utilizando análise objetiva, reputação e inteligência artificial. Os resultados obtidos mediante a uma prova de conceito, e de um estudo de viabilidade conduzido com especialistas da indústria, indicam que ele pode apoiar a aquisição de software por meio da descoberta, avaliação e comparação de microsserviços, sendo capaz de recomendar o mais apto dentre centenas de candidatos em múltiplas plataformas de nuvem.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

DIRECTOR:
A CLOUD MICROSERVICE SELECTION FRAMEWORK

Marcelo de França Costa

June/2019

Advisor: Cláudia Maria Lima Werner

Department: Computer Science and System Engineering

The Software Ecosystem research field has been receiving an increasing amount of attention from both academia and industry, as many organizations have been adopting them as a collaborative platform to achieve innovation faster than before. More recently, with the advent of Cloud Computing, modern ecosystems have been offered as a service, allowing actors to contribute, but also commercialize their own solutions, by reusing available software assets, popularly in the shape of microservices, i.e., very specific functionality, usually exposed through Web technologies. With the current proliferation of platforms and microservices, an open and relevant challenge for software architects is to find and acquire the most adequate component, given a set of requirements and priorities. In this context, we propose DIRECTOR: A cloud microservice selection framework, based on complementary technical, social and semantical perspectives, i.e., by relying on objective analysis, reputation and artificial intelligence, respectively. The results obtained through a proof-of-concept (PoC), and a feasibility study conducted with industry experts, indicate that it can support software acquisition via discovery, evaluation and comparison of microservices, being able to recommend the fittest among hundreds of candidates in multiple cloud platforms.

SUMMARY

LIST OF ILLUSTRATIONS

LIST OF TABLES

ABBREVIATIONS

ADR	Architectural Decision Record
AI	Artificial Intelligence
API	Application Programming Interface
CAMS	Cloud Computing, (Big Data) & Analytics, Mobile & Social (Networks)
CBD	Component-Based Development
CBSE	Component-Based Software Engineering
CF	Cloud Foundry
COTS	Commercial Off-The-Shelf
GQM	Goal-Question-Metric
IaaS	Infrastructure as a Service
IoT	Internet of Things
IT	Information Technology
MCDA	Multi-Criteria Decision Analysis
MCDM	Multi-Criteria Decision-Making
MSA	Microservice Architecture
NLP	Natural Language Processing
OSS	Open Source Software
PaaS	Platform as a Service
PoC	Proof of Concept
RQ	Research Question
SE	Software Engineering

SECO	Software Ecosystem
SME	Subject Matter Expert
SMI	Service Measurement Index
SOA	Service Oriented Architecture
SoIS	System-of-Information Systems
SaaS	Software as a Service
SoS	System-of-Systems
SPL	Software Product Line
TAM	Technology Acceptance Model
µSaaS	Microservice as a Service

Chapter 1 – Introduction

This chapter presents the motivation for the development of this research, as well as the problem and questions related to this PhD thesis. Additionally, the general goals are listed, and a brief description of the text organization is given.

1.1 Context

According to BOEHM (2006), the increasing pace of change in the global industry is driving organizations towards increasing levels of agility in their software development methods, i.e., they need to respond fast to the changes to keep and grow their market share. At the same time, their products and services are becoming more software-intensive, strongly depending on information systems. In this sense, SANTOS & WERNER (2011) ratify the importance of software, which has been a crucial element for most of the existing systems, since it affects functions, resources, and risks in different industry sectors. Software-intensive systems have also become increasingly ubiquitous, large and complex, with considerable dissemination in several application domains and tightly dependent upon different technologies (BOSCH, 2012).

Such systems are usually centered in a software platform. Their modern versions might rely on cloud-based platforms, such as IBM Cloud[1], Google Cloud Platform[2], Microsoft Azure[3] and Amazon AWS[4]. In those platforms, various elements create a socio-technical network[5], an interplay between the social system and the technical system (HANSSEN & DYBÅ, 2012). For instance, suppliers, outsourcing companies, independent developers, acquirers, technology providers, clients, end-users, software applications, and technologies interact and change the development process whenever they exchange information (IANSITI & LEVIEN, 2004). As a consequence, besides technical aspects, the treatment of economic and social issues has been pointed out as a

[1] https://www.ibm.com/cloud/

[2] https://cloud.google.com/

[3] https://azure.microsoft.com/en-us/

[4] https://aws.amazon.com/

[5] We define a socio-technical network as a set of actors and artifacts, including their relationships, commonly represented as a graph where nodes are actors and artifacts, and relationships are edges.

challenge for Software Engineering (SE) research and practice over the last decade (BOEHM, 2006; CUKIERMAN et al., 2007; JANSEN et al., 2009; SANTOS et al., 2012; FINKELSTEIN, 2014; MENDES et al., 2015; MANIKAS, 2016). Thus, when approaching challenges in such context, one could benefit from taking a holistic view, leveraging complementary perspectives to address the different aspects of the problem (economic, technical, social, among others).

According to JANSEN & CUSUMANO (2012), software-intensive system engineering requires a better analysis regarding those cloud platforms. Thus, increasing attention has been paid to the influence and interdependency in relationships between players within a competitive shared market. It means that organizations that produce software no longer function as isolated units that deliver separate products. They have become dependent on each other for vital components and infrastructures, such as operating systems, programming languages, libraries, and component stores (JANSEN et al., 2009). Additionally, one could see an exponential proliferation of such software components because of the adoption of cloud-based platforms with greater numbers of participants, which in turn brings additional challenges related to the discovery, comparison, and acquisition of those components.

For most organizations, large-scale software development is interconnected, expensive, slow, and unpredictable (BOSCH & BOSCH-SIJTSEMA, 2010a), which in turn triggered a trend that accelerates complexity in SE industry: the emergence of software ecosystems (SECOs) with the challenge of opening up the platform architecture to get contributions from external players. One should bear in mind that the concept of an ecosystem has its origins in Ecology, where a community of living organisms, in conjunction with the non-living components of their environment, interacts as a system (MOLLES, 1999). In that regard, a SECO would add the software as an abstract component into those interactions. Hence, it can be defined as the interaction of a group of actors on top of a common technological platform that results in several software solutions or services (MANIKAS & HANSEN, 2013).

Organizations started to adopt a SECO approach to achieve innovation by fostering collaboration between employees and external contributors. In the past, they used to rely on Software Product Lines (SPL), a set of software-intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way (SEI, 2019). However, to further reduce time-to-market whilst responding to an increasing amount of customer requests, companies started to move from a traditional approach, such as SPL, to a more open and collaborative environment, such as a SECO (BOSCH, 2009), leading to the fourth generation of software reuse (SANTOS, 2016).

It seems natural for a SECO actor to reuse software as a common practice. Such actors take advantage of the platform's built-in features, as well as assets produced by others. Thus, SECOs are considered as the most recent generation of software reuse, where one should consider not only a technical view but also the social and business dimensions (CAMPBELL & AHMED, 2010). Though the "technical" sense is straightforward, it is worth mentioning that "social" relates to the community's feedback and contributions influencing the evolution of the software asset and the platform itself. Similarly, economic factors (e.g., the cost for acquiring a software component or service) drive business in a SECO.

With the advent of Cloud Computing, the underlying technological platform of modern SECOs started to be offered as a service in the cloud, i.e., a platform as a service (PaaS). One should notice that Cloud Computing is an element of the so-called CAMS (Cloud, Analytics, Mobile and Social): the third model of computing platform, as depicted in Figure 1.1. They support innovation enablers for the new digital economy, according to the marketing firm IDC (2016), as well as to Gartner (2014a).

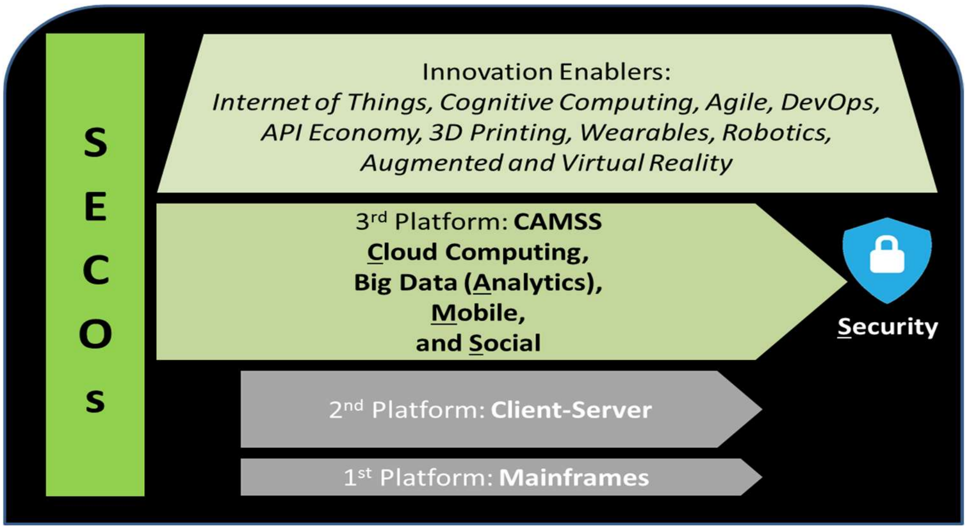

S E C O s
Innovation Enablers:
Internet of Things, Cognitive Computing, Agile, DevOps,
API Economy, 3D Printing, Wearables, Robotics,
Augmented and Virtual Reality
3rd Platform: CAMSS
Cloud Computing,
Big Data (Analytics),
Mobile,
and Social
Security
2nd Platform: Client-Server
1st Platform: Mainframes

Figure 1.1: The third platform supporting digital transformation.

In this new scenario, where modern SECOs are being adopted by organizations, the SE community has been facing new challenges, especially from an architectural point of view. To produce good architectures that fit well in the depicted situation, one must take into consideration the multiple dimensions of the SECO nature, as well as the technologies and strategies used by its community members. One of the main strategies applied by SECO participants is to reuse the platform's assets. The motivation for this research is closely related to that, as explained in the following subsection.

1.2 Motivation

The novelty of reusing software artifacts in a SECO context might also be associated with the evolution of the Service Oriented Architecture (SOA) to a more granular version of itself, named Microservice Architecture (MSA). Whilst nowadays MSA is a de facto standard in cloud-based platforms, in 2005, one of the key goals of SOA was the integration between multiple corporate IT systems, written in different technologies, each having its own proprietary interface (OPEN GROUP, 2011). The intra-organization interoperability need was addressed by providing a way to interact with applications using open standards, such as SOAP, HTTP and XML, encapsulating the underlying application's technology and consolidating the web service model.

However, challenges started to be handled outside the company's boundaries as part of the transition to a SECO strategy (BOSCH, 2009). Whereas Cloud Computing further contributed to SOA expansion, it has also reshaped it. While SOA is strongly business-oriented and its services "should be" closely related to a business process, *web services* became less company-specific to maximize their reuse within a SECO. Instead of business process-size services, microservices with a single responsibility and independently deployable started to appear as a new architectural style (HILWA, 2015). While interoperability is promoted in coarse-grained services, reusability is fostered with finer-grained services (ERL, 2005). In addition, one can argue that MSA became a popular approach in cloud-based SECOs due to developers' influence (O'GRADY, 2017). Instead of a top-down strategy, MSA started to be adopted by programmers who, in turn, became "technology evangelists" for such architectural style, building a critical mass of support and establishing it as a technical standard among their peers.

Besides an increase in the number of available components in SECO platforms deployed in the cloud, the popularity of the MSA style based on Web standards made possible to an application on a specific platform to reuse a microservice on a different platform, resulting in a plethora of possibilities from a composite solution standpoint. In that sense, finding the fittest microservice to be used in a new cloud solution became an endeavor that has motivated this research.

This work was built upon the idea of complementary perspectives using multi-criteria decision analysis (MCDA) for microservice selection, first presented in the IEEE International Conference on Software Architecture in 2018 (FRANÇA & WERNER, 2018). A survey, which was followed by an interview with a senior IT architect working for a global consulting firm, gathered the ideas that shaped the proposed complementary perspectives. Later, a second survey was conducted with software architects and ratified the assumptions used for the selection mechanism. Lastly, despite the popularity of the theme, there is still relatively little research in the microservices field, if compared to other SE topics. This is, in part, due to its novelty, since the term was coined less than ten years ago (LEWIS, 2012).

1.3 Problem

The main goal of this research if to provide means for finding the fittest microservice provided by SECO platforms deployed on the cloud to be used in a new software system. The main idea is to adopt complementary perspectives using MCDA, a sub-discipline of operations research that explicitly evaluates multiple conflicting criteria in decision-making. Applying MCDA to microservice selection allows for a flexible way of addressing the challenge of selecting such microservices while taking into consideration SECO's multiple dimensions.

It is valid to point out that the problem of selecting reusable software assets is not new. The work of Mohamed et al. (2007) on "Commercial Off-The-Shelf (COTS) Selection" discusses practices that date back to 1995. The paper confirms the importance of selecting appropriate COTS in software development (typically referred to as "component-based development"). According to the authors, COTS selection is the process of determining the fitness-of-use of COTS products in a new context, and then selecting one or more products with the highest fitness. In a cloud-based SECO, a microservice available on the PaaS can be considered as a "COTS". In that sense, as in the past, a software architect designing a new software solution could benefit from having a catalog of available assets, as well as means to select the most appropriate one, especially when the number of choices grows beyond the grasp of the architect.

Despite several efforts made during the last decade to model the COTS selection process, the authors argue that none of the existing methods could be considered as a "silver-bullet" to solve this problem, therefore leaving room for innovative approaches targeting different contexts. Though one could still claim that this is an old problem, which has been already solved, according to NIST (2015), with Cloud Computing in the mainstream there is a preponderance of cloud-based services in the market, creating new choices for consumers daily. Comparing a plethora of service offerings, made available by various cloud providers, is not a trivial and straightforward exercise.

Despite being an old problem, it should be stressed out that the novelty of this scenario is part due to the fact that microservices have made the reuse unit smaller while increasing its numbers. Additionally, it is due to the high rates of adoption of cloud computing (especially PaaS) since it has a considerably reduced CAPEX[6] for software developers and solution providers. Both facts directly influence the abundance of microservices, making their selection a challenging endeavor.

To address such challenge, the proposed framework relies on three complementary perspectives capable of comparing and recommending a microservice among all available in SECO platforms. The first perspective is based on a technical evaluation, leveraging information extracted from the platform's microservice catalog, using quality metrics inferred from that metadata. The second one is based on the SECO social dimension, measuring the community engagement on a microservice's technology. The third is based on a semantic analysis of the textual description of the user's main goal, providing an unbiased recommendation.

1.4 Objectives

With the expansion of COTS offers in modern SECO platforms, a recommendation tool becomes a must when it comes to software acquisition activities. From the architect's point of view, such tool could even support documenting the rationale behind a selection process by describing the possible choices and the characteristics (pros and cons) of the COTS being considered – again, alternatives and the justification for the chosen one.

[6] Capital expenditure or capital expense (capex or CAPEX) is the money a company spends to buy, maintain, or improve its fixed assets, such as buildings, vehicles, equipment, or land.

Thus, from the challenges pointed out in Section 1.3, our **hypothesis** is defined as: *"as a recommendation system, DIRECTOR affects the effectiveness, efficiency and confidence of architectural decision activities regarding the choice of microservices to reuse in a cloud-based SECO (PaaS)"*. From this hypothesis, two research questions (RQ) were established and addressed by this work.

Inspired by the ideas from the interview with IT architects, an exploratory study on popular cloud platforms and standards was conducted to answer **RQ1** – *What quality attributes, from a technical perspective, could be extracted from the metadata available on microservice catalogs, in cloud platforms?* Though we detail this study in Chapter 3, it could be summarized that it is possible to capture technical metadata related to the microservices in a platform-agnostic way, unbound to a given cloud provider, by leveraging an open standard. A survey with practitioners from the industry was conducted to corroborate the inferences regarding the quality of service (QoS) metrics derived from such metadata. Therefore, a QoS mapping based in a standard measurement framework for cloud microservices was formulated.

Since architects face challenges on making microservice selection decisions in a SECO context, a framework to support practitioners regarding the architectural decision activities was developed: **RQ2** – *Can* ***DIRECTOR*** *complementary perspectives help in the architectural decision of choosing a microservice for a PaaS-based solution?* A feasibility study, presented in Chapter 5, was conducted with practitioners, using real-life scenarios, to evaluate the proposed framework according to effectiveness, efficiency, confidence and utility.

In short, this research aims to support, on a SECO context and from a Software Architecture perspective, decisions regarding selection of reusable assets in the form of microservices provided by cloud platforms. Current methods and tools for COTS selection do not take into consideration a multi-cloud strategy[7] (a popular pattern in industry nowadays) nor the microservice nature, e.g., their granularity and price models.

Therefore, the main research goal can be stated as: given a set of requirements describing the features needed from a microservice (and priorities), to find the most

[7] A multi-cloud strategy is the use of two or more cloud computing services. While a multi-cloud deployment can refer to any implementation of multiple software as a service (SaaS) or platform as a service (PaaS) cloud offerings, today, it generally refers to a mix of public infrastructure as a service (IaaS) environments, such as Amazon Web Services and Microsoft Azure.

adequate microservice available in cloud platforms known by the framework. The objective is to provide a SE practitioner, such as a software architect, with a tool for microservices discovery and evaluation, producing recommendations from complementary perspectives: technical, social and semantic. The results described in Chapter 5 indicate that all three strategies can identify valid choices, accordingly to the criteria supplied by the SE practitioner, as well as present the rationale behind each of the recommendations. Some specific objectives of this work include:

- To *investigate* the microservices field on how they differ from SOA services and, leveraging an open standard, how to discover and query information about them in a SECO context;
- To define a *QoS attributes mapping* from the metadata captured from the SECO platforms' microservice catalog;
- To survey and interview experts from industry to *identify concrete problems related to architectural decisions in a SECO* and to *confirm the technical evaluation criteria* for microservice selection based on collected data;
- To design and develop a *framework* for recommending microservices in cloud-based platforms;
- To *ensure* that this framework helps SE practitioners to acquire software for new cloud-based solutions.

1.5 Methodology

This work's research methodology was inspired by Design Science paradigm (HEVNER et al., 2004), which is a problem-solving paradigm based on some guidelines to create and evaluate artifacts developed to cope with real, organizational problems. Some real cases and interviews were used to help us to develop our framework in the form of a supporting tool due to its practical nature.

Figure 1.2 shows the research methodology adopted in this PhD Thesis, composed of five phases, adapted from (NUNES, 2014) and inspired by SANTOS (2016). In the first phase, Problem Perception and Definition, the basics of SECO field were investigated while some exploratory analysis was performed on popular cloud platform to identify critical problems and, then, to derive relevant questions related to architectural decision activities in this context.

In the second phase, Theory Analysis, SECO architecture literature was examined. Our investigations helped identifying open challenges related to the software acquisition and software reuse process in cloud-based SECOs. It was also identified that MSA is a novel research field, though being a very popular format nowadays.

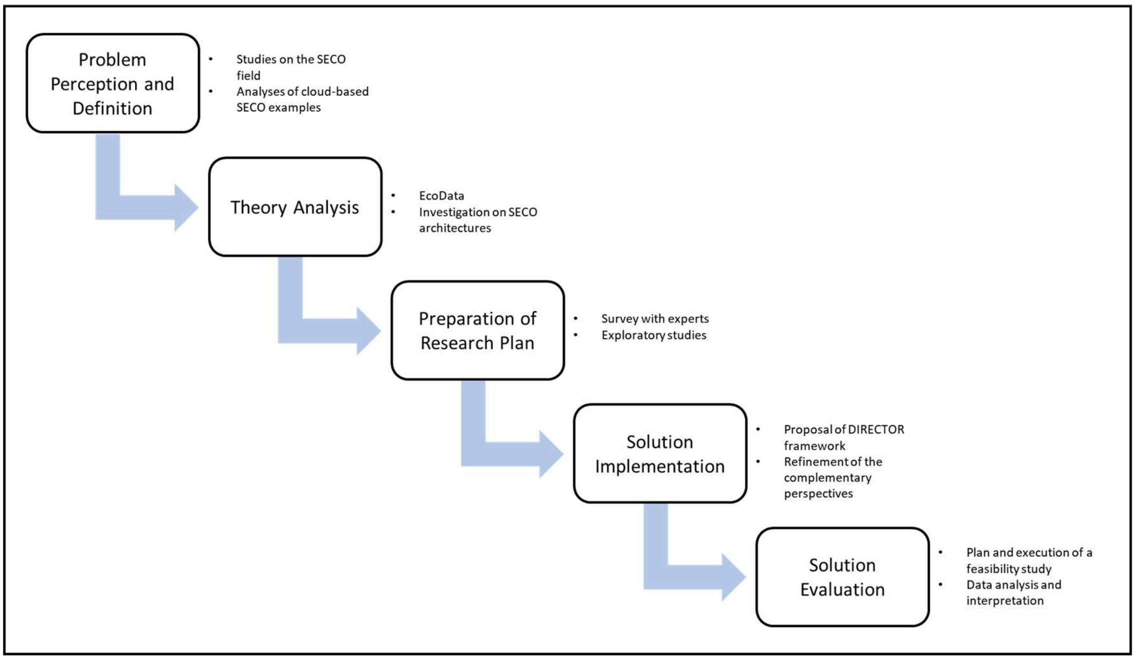

Problem Perception and Definition
Studies on the SECO field
Analyses of cloud-based SECO examples
Theory Analysis
EcoData
Investigation on SECO architectures
Preparation of Research Plan
Survey with experts
Exploratory studies
Solution Implementation
Proposal of DIRECTOR framework
Refinement of the complementary perspectives
Solution Evaluation
Plan and execution of a feasibility study
Data analysis and interpretation

Figure 1.2: Research methodology.

In the third phase, Preparation of Research Plan, architectural decisions were reviewed in a SECO context, focusing on COTS selection and composite solutions. Surveys with experts were planned and executed o collect information related to IT architecture decision activities. Exploratory studies were also conducted with popular cloud platforms, to identify a common approach or strategy that could be used to retrieve information about the available microservices in a platform-agnostic fashion.

Motivated by the challenges identified in the third phase, a framework capable of discovering and evaluating microservices in multiple cloud providers was developed. In this research, much effort was put on the complementary perspectives with the objective of providing a recommendation system that would be flexible enough to accommodate different viewpoints and priorities. Additionally, the strategies were evolved so that they could even include the understanding of natural language, which makes the framework available to users who are not technically versed, such as business analysts and product owners.

Finally, in the fifth phase we verified DIRECTOR framework with practitioners in a real-life scenario through a proof of concept (PoC) and a feasibility study.

1.6 Outline

This PhD Thesis is structured in six chapters. This chapter presented the context of our work and the motivation for this research. The problem identified as a gap in theory and practice, the objective and originality of this thesis were explained, as well as the research questions, the goals of this work, besides the methodology that guided us towards our scientific contribution.

Chapter 2 discusses the background of this research. As such, we introduce ecosystem in SE area, briefly investigating the SECO field, and then characterizing "modern SECOs", describing the challenge of dealing with the expansion of COTS alternatives in such platforms, and how software acquisition and architectural decision activities relate to that challenge.

Chapter 3 describes the technical perspective of the proposed framework for microservice selection and the surveys that were conducted with SE practitioners from industry to ratify the technical criteria used for evaluating cloud microservices and get feedback about what could help software architects when it comes to architectural decisions related to software acquisition.

Chapter 4 explains DIRECTOR as a selection framework for the cloud era. DIRECTOR is a conceptual framework, comprising the discovery, selection (filtering) and ranking of microservices to be reused, including the technical, social and semantic evaluation strategies. We conclude this chapter with the preliminary results from a PoC.

Chapter 5 presents a feasibility study we executed to evaluate DIRECTOR. We focused on how feasible DIRECTOR is to aid software architects to execute architectural decision activities in a SECO context, more specifically those related to the reuse and acquisition of microservices exposed in cloud platforms. It also discusses the main findings we observed while conducting the study. Strengths and weaknesses are summarized, and threats to validity of the experiment are presented.

Chapter 6 concludes this document. We present some closing considerations, contributions of the thesis, and limitations of this research. Lastly, we propose some future work.

Chapter 2 – Modern Software Ecosystems

In this chapter, a brief discussion of modern SECOs is presented. Then, some literature perspectives on the contemporary evolution from SOA to MSA are discussed. Later, the classical COTS selection problem is revisited, discussing the nuances of cloud microservices and the importance of recording such architectural decisions.

2.1 Software Ecosystems

From an organizational perspective, a Software Ecosystem (SECO) is an environment where internal and external actors, mainly developers, construct large software systems by reusing components, such as services, supported by a common platform (BOSCH, 2009). As a relative new topic, where SE practice is not limited to a single software company anymore, but it is expanded across a group of organizations and external individuals, SECO research has received more attention since 2007 (MANIKAS & HANSEN, 2013). Notice, though, that SECO differs from more traditional outsourcing strategies in that the initiating actor does not necessarily own the software produced by contributing actors (independent developers or *free-lancers*) nor hires them. Among the players in a SECO, it is common the existence of a keystone, a dominant actor that owns the platform and establishes rules for the others. Such dominant actors include Microsoft in the Windows SECO, Apple in the iOS SECO, Google in the Android SECO, and even Facebook, which is a SECO platform by itself.

At the same time, one could claim that there is an everlasting demand for innovative solutions, since competitors are struggling to differentiate themselves by delivering more value to their customers nowadays. Additionally, industry has to deal with the so called "digital transformation" (STOLTERMAN & FORS, 2004), where changes associated with the utilization of digital technology are affecting all aspects of human society, while creating demands for a new set of applications that can leverage modern devices, from smart watches to smart cars. Thus, in order to reduce time-to-market whilst responding to such challenges, companies are moving from a traditional, independent strategy, such as Software Product Lines (SPL), to a more open, collaborative, and even agile approach, such as a SECO (BOSCH, 2009).

A SECO is usually characterized according to three dimensions or central pillars: business, architecture (technology), and social aspects (CAMPBELL, P. R., AHMED, F., 2010), as can be seen in Figure 2.1. In a SECO, software reuse is fostered by the collaboration between organizations, third party and independent developers, besides feedback from end-users. Note that reuse may refer to software components such as libraries, classes, services and, more generically, APIs (Application Program Interfaces). Despite the modern context, the notion of reuse is old and originated from the search for consistent solutions to problems that could also be applied to new ones, and whose repetition would make them accepted, generalized and standardized, as in Mathematics and Physics (WERNER, 1992).

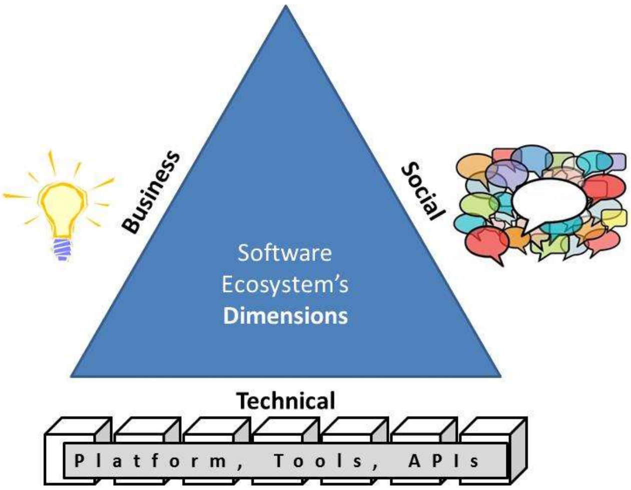


Figure 2.1: The three dimensions of a software ecosystem.

As academy, industry is also interested in SECO, since one could observe companies moving to a SECO strategy after adopting approaches such as crowd-sourcing, by opening their products and platforms, in some degree, for external contributors. Even today, building and using software products are still considered complex challenges in modern technology development, where the main evidence is the increasing number of failures in projects, programs and businesses (BOEHM, 2006), some of them related to late schedules. Thus, companies are adopting SECO because of the possible reduction in time-to-market, since they may count with the contribution of external developers.

Other interesting aspect is the fact that an organization may adjust the employee "head count" accordingly to the demand, since it may include external collaborators who are not hired. The creation of social-technical networks between the stakeholders involved in the development process and the different artifact types being produced brought to the software industry the need to open organizational borders. That way, external actors, known in advance or not, are included and compose a new way for structuring business models.

Lastly, organizations are interested in their end user's feedback, which is facilitated by the collaboration environment of a SECO. Although Google Android and Apple iOS application "stores" are popular examples of how feedback from end-users is done by rating "apps", SECO platforms are not limited to the mobile world.

2.1.1 Software Ecosystems and Software Reuse

Because of open-innovation business trends (CHESBROUGH, 2003), a transition from the traditional SE to SECO started to take place. Researches began to investigate the social and economic issues intrinsic to the life cycle of reuse-based software (SANTOS & WERNER, 2010). Starting from this context, one can define a path for SE to the SECO approach from a software reuse perspective, illustrated by the "generations" (SANTOS, 2016) presented in Figure 2.2 – though such "evolutionary" view is not the only way of achieving a SECO environment.

As depicted in Figure 2.2, the first type of reuse was the equivalent of today's "copy and paste", where code fragments were duplicated in different programs, while the concept of modularity, applied at a program level, made possible for a set of lines of code, also known as a "routine", to be called from different places in the same application. That was common in monolithic systems, where all the required features were assembled in a single deployable software program.

Then, program subroutines started to be put together as a set of reusable functions, being called "libraries". Those libraries were a set of procedures that sometimes were not thought a priori to be put together, but were very found useful by the programmers. Programmers had their own set of libraries that they applied repeatedly while developing different applications. With the emergence of the Object-Oriented paradigm, "libraries" somehow evolved to a higher level of abstraction, taking the shape of components. Components could be also used in different applications, sometimes even being written in a different programming language. Differently from "libraries", components could run in a different process, using strategies like Remote Procedure Call (RPC), as in the case of the CORBA standard, and also the web services, a popular format nowadays.

Finally, the unit of reuse took an even higher level of abstraction, where common-parts of an application or a system are reused for composing or customizing new applications. The concept of SPL, as already discussed, was the highest level of software reuse at this point, considering an intra-organizational context. After that, the way to achieve more, escalating reuse, was by opening the organizations "gates", allowing for external parties to reuse organization assets (not only software), as well as receiving contribution, accelerating innovation, and reducing the deployment cycles.

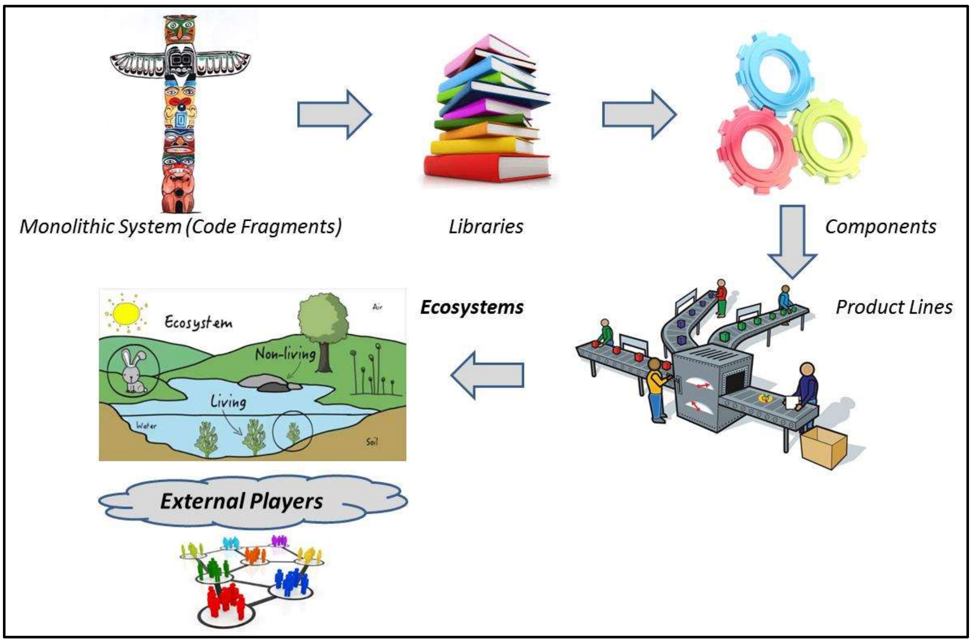


Figure 2.2: Generations of software reuse.

With the advent of SECO and cloud computing, one can now reuse a software asset in an unknown location and at scale. Even better, one could reuse an asset free (open-source software) or pay-per-use instead of paying a full amount in advance. Such disruptive technologies modified the reuse strategies being applied by organizations around the world, which in turn started a new generation of software reuse.

2.1.2 Systems-of-Information Systems Ecosystems

The concept of System-of-Information Systems (SoIS), a set of interoperable Information Systems that exchange data and services to achieve some major business goal (MAJD et al., 2015; SALEH & ABEL, 2015), was raised from the need of information systems to support interoperability, creating complex business processes, and opening up to new business chains (TOMICIC-PUPEK et al., 2012; ARAUJO & MAGALHÃES, 2015). According to Neto et al. (2017), SoIS can be dynamic, enabling new constituents to join the SoIS to contribute with their specific functionalities in order to achieve complex behaviors. In this sense, SoIS can be seen as a SECO subtype.

The concept of SECO has helped researchers and practitioners to model and analyze several existing relations among software elements that compose a technological platform, as well as their internal and external players, such as Apple SECO or Eclipse SECO (MANIKAS, 2016). In this context, Neto et al. (2017) claim that SECO can foster the comprehension about SoIS by exploring the existing relations among constituent information systems within a SoIS, as well as the nature of such relations. This concern may lead to the concept of SoIS Ecosystems, a SECO that involves the development and interoperable activity within a set of information systems working together to support business and social goals.

Especially in the web and mobile eras, information system research is facing challenges, such as how to establish and control information systems' borders and how to govern the software supply network formed over them considering the social web environment (NETO et al., 2017). Such systems can be combined to form what is termed a System-of-Systems (SoS). SoS are alliances of independent systems that are combined to interoperate, resulting in a more complex behavior. Such behaviors could not be obtained from those independent systems working separately (MAIER, 1996). In this sense, SoIS are also a particular type of SoS composed by information systems (GRACIANO NETO et al., 2017a).

Moreover, SoS should exhibit an opportunist nature, i.e., a system should be able to join other systems to form a SoS that accomplishes a mission, leaving the SoS when the mission finishes (NETO et al., 2017). Dynamic architectures have also been considered a remarkable SoS characteristic. In the context of social web and human aspects research, SoS is still barely explored as methods, techniques, and tools largely focused on the technical aspects. On the other hand, organizational aspects emerge when, for instance, developers and users interact with those complex systems in order to accomplish their missions and the social web environment aid those stakeholders to communicate and collaborate to evolve such systems with new requirements (GRACIANO NETO et al., 2017b).

From a more general point of view, software vendors co-evolve their market capabilities around innovation: they work cooperatively and competitively to support and develop new products, to satisfy customer needs and to innovate continuously (MOORE, 1996). In this sense, SECO is an effective way to achieve that goal, by constructing software on top of a common technological platform, by composing applications and technologies developed by multiple actors (ARAUJO & MAGALHÃES, 2015). Furthermore, a SECO comprises a foundation technology or set of components used beyond a single company that brings multiple parties together for a purpose such as to solve a common business problem. In this context, the ecosystem platform can be seen as a broker that supports a social web based on the interaction among organizations, developers and users (NETO et al, 2017).

According to Neto et al (2017), SECO is also characterized by both software production and consumption relations – including components, e.g., services available on the platforms. These relations can be established with third-party developers, communities and/or other organizations to foster components development, supply and evolution in a large ecosystem created over the common technological platform. In the social and business dimensions, a SECO provides a complementary, organizational view to SoS development, which defines roles, rules of interaction, collaboration and synergistic capabilities for its constituent systems.

Finally, there are many similarities between SoS characteristics (SANTOS et al., 2014) and SECO technical challenges (BOSCH, 2009), e.g., how to ensure platform stability, simplicity, security, reliability, and evolution. In this sense, Neto et al (2017) claim that SECO and SoIS may also hold intrinsic and synergistic relations that can be explored. They glimpse that the association among distinct software intensive information systems creates a SECO comprising the emergent behavior resulting from the association of their different business goals into a new and common one SoIS Ecosystem. According to them, this phenomenon creates an entire SECO that surrounds the entire SoIS and involves other inner SECOs that are inserted in that context.

2.2 The Emergence of Microservices

Contemporary to the appearance of modern types of SECO in the cloud, a new architectural style, strongly influenced by SOA, emerged - microservice architecture (MSA). Though MSA and SOA both rely on services as the main unit of reuse, they vary in terms of service characteristics. SOA and MSA are considered two types of higher order, web service architectures. While SOA defines four basic coarse-grained types of services (business, enterprise, application and infrastructure services), MSA has a more limited taxonomy, usually consisting of just two fine-grained service types: functional and infrastructure services[8].

[8] https://es.atlassian.com/continuous-delivery/microservices, accessed 05-25-2019.

To understand the contrast between MSA and SOA, one should start from the definition of SOA. The standard definition from The Open Group (2011) states that service-orientation is a way of thinking in terms of services and service-based development and the outcomes of services. According to them, a service (i) is a logical representation of a repeatable business activity that has a specified outcome (e.g., check customer credit, provide weather data, consolidate drilling reports, among others); (ii) is self-contained; (iii) may be composed of other services; and (iv) is a "black box" to the consumers of the service.

The TOGAF standard[9] states that an *architectural style* is the combination of distinctive features in which an architecture is performed or expressed. They might differ in terms of focus, form, techniques, materials, subject, and period. In this sense, The Open Group definition for SOA identifies the following distinctive features:

- It is based on the design of services that mirror real-world business activities comprising the enterprise (or inter-enterprise) business processes;
- Service representation uses business descriptions to provide context (i.e., business process, goal, rule, policy, service interface, and service component) and provides concrete implementation of services using service orchestration;
- It places unique requirements on infrastructure: it is suggested that implementations use open standards to allow interoperability and location transparency;
- Implementations are environment-specific, as they are constrained or enabled by context and must be described within that context;
- It requires strong governance of service representation and implementation;
- It requires a "service litmus test"[10], which determines a "good service" – the metaphor is used to denote a set of tests that, when applied, will determine if a given candidate service should be eligible for exposure using a service description.

In SOA, a service may be composed of other services. In MSA, we define a service as independent and self-contained, which implies that it cannot be composed of other

[9] https://pubs.opengroup.org/architecture/togaf9-doc/arch/chap17.html

[10] In Chemistry, it is a test used to figure out whether a chemical solution is acidic or basic. The Service Litmus Tests are included in SOMA, a de facto end-to-end SOA Development Method. Ref.: https://www.ibm.com/developerworks/webservices/library/ws-soa-design1/, retrieved in 08-03-2019.

services. Herein lies one of the main differences between SOA and MSA architectural styles. In further examining each aforementioned feature, we find that, for the most part, the frame of a microservice will align with that of a service of the SOA architectural style, with the exception of how much of a business process it encapsulates, as many business processes contain many services to complete their task. In a MSA this would be a conflict in purpose. This implies that MSA is a subset or special architectural form of SOA. MSA provides an approach to delivering SOA in an effective manner for the right set of business drivers.

The term microservice appears to have been coined by 2012, when James Lewis from ThoughtWorks, a private global technology company, was one of the first IT professionals to present some of these ideas as a case study (LEWIS, 2012). In summary, MSA is an approach for developing applications on top of a set of small and independently deployable services, easier for a developer to understand. Those services are individually scalable since each one runs on its own process, relying in a lightweight inter-process communication mechanism, commonly via an HTTP API. They typically use web technologies and standards, such as the Atom protocol[11], even not being necessarily on the web. Notice that besides SOA, other concepts seem to have influenced MSA, such as UNIX pipes & filters (LEWIS, 2012) and even compilers and their phases (MARTIN, 2015).

As already mentioned in Section 1.2, Cloud Computing contributed to SOA expansion, but also reshaped it. SOA is strongly business-oriented, where services “should be” closely related to a business process. However, web services started to become less company-specific in order to maximize their reuse within a SECO. Additionally, business process-sized services gave place to finer-grained ones, with a single responsibility and independently deployable (HILWA, 2015). While coarse-grained services promote interoperability of a larger amount of functionality, finer-grained services foster reusability (ERL, 2005). The prefix “micro” obviously refers to the granularity of the internal components. Service components within a MSA are generally single purpose ones, which do just one thing (higher cohesion). Conversely, services usually include much more business functionality in SOA, and they are often implemented as complete subsystems.

[11] The Atom Publishing Protocol, available at https://www.ietf.org/rfc/rfc5023.txt, retrieved in 12-05-2015.

MSA started to become a popular trend in 2015, due to developers' influence (O'GRADY, 2017), i.e., by playing a decisive role on what would be adopted and what would not. Instead of a top-down strategy where consulting firms "sold" the SOA approach (and tools) to executives, MSA started to be adopted by programmers who considered microservices easier to develop and maintain, rather than "monolithic" alternatives – besides being decoupled from vendor standards and tools.

The "hype" of MSA, together with the emergence of Cloud Computing, resulted in a proliferation of services being offered in cloud-based platforms, especially of the infrastructure kind, e.g., database as a service (DbaaS). With so many alternatives, the selection of microservices became a relevant challenge, as discussed in the next section.

2.2.1 Architecting with Microservices

Several companies, such as Amazon, Netflix, LinkedIn, Spotify, and SoundCloud, have evolved their applications towards a MSA (FOWLER & LEWIS, 2014; VILLAMIZAR et al., 2015; YAHIA et al., 2016). According to Di Francesco et al. (2019), while there has not been a wide acceptance of a specific definition of the microservices architectural style, the most acknowledged one is provided by Fowler and Lewis (2014). This definition describes the microservices architectural style as an approach for developing an application as a suite of "small" services, each running in its own process and communicating through lightweight mechanisms, often as an HTTP resource API. Recurrent characteristics of the microservice architectural style are: (i) organization of the system around business capability, (ii) automated deployment, (iii) intelligence in the endpoints, and (iv) decentralized control of languages and data. Those characteristics allow designing architectures that should be flexible, modular and easy to evolve (DI FRANCESCO et al., 2019).

While it is well known that microservices have their roots in the industry, there are research groups focusing on them from an academic standpoint (DI FRANCESCO et al., 2019). However, a very small number of publications have been produced until 2014, which is the first year in which microservices started to attract the interest of large organizations, and the term microservice as architectural style was consistently used (PAHL & JAMSHIDI, 2016). Di Francesco et al. (2019) also state that year 2015 signaled a booming in the research field of designing with microservices, with the trend increasing in 2016 and still growing in the first months of the year 2017.

The fact that evaluation research on MSA is rarely performed has a negative impact on the potential for transferring current research results in industry (DI FRANCESCO et al., 2019). This suggests a gap that should be filled by future research, especially if practitioners want to either solve real problems coming from industrial scenarios or push further the technology transfer of research results in industry. This Thesis' research contributes to the evaluation research on microservices by presenting an evaluation with practitioners from industry.

It is worth to point out that if MSA allows for team and business agility, on the other hand, adopting a microservice-based architecture may bring higher complexity (DI FRANCESCO et al., 2019). By analyzing the MSA literature, one finds problems related to system-level aspects such as time to market, low testability, low portability, and security being discussed. These aspects have been extensively investigated in the software architecture area, but are still new to microservice architectures. This indicates a potentially relevant research gap needing attention, i.e., old problems that require investigation due to a new scenario (DI FRANCESCO et al., 2019).

The recent work from Di Francesco et al. (2019), a systematic mapping study, states that research on designing with microservices is still in its initial phase for what concerns transferability of the developed technologies to industry. The results also indicate the need to support knowledge-based tools with more software-based tools in order to demonstrate how effective knowledge-based tools are and how they can be compared to the others. This would help researchers and practitioners to improve the overall quality of microservice-based systems. In this sense, the approach proposed in this thesis' is supported by a tool, which is presented in Chapter 4.

From a literature perspective, the number of microservices used for evaluation is also an aspect worth mentioning. In their mapping, Di Francesco et al. (2019) say most of the primary studies have only used a relatively small number of microservices (i.e., less than 10). Only three primary studies have used a relatively significant number of microservices using a total of 27, 28 and 67 microservices, respectively. To put this result into context, a recent industrial survey (DI FRANCESCO et al., 2018) showed that the expected number of microservices deployed after migrating towards the microservices architectural style varies between 5 and 250, with an average of 59. Lastly, no work on microservices selection or comparison was included in their literature mapping, though some work discussed architecting activities including decision-making and documentation, the later especially regarding communication between architects and other stakeholders.

The work by Eisa et al. (2016) about trends and directions in cloud service selection clearly articulates the differences between classical SOA webservices and cloud microservices, while ratifying that there are potential pros and cons of such a large number of providers. The authors state that providing more choices improves the overall service quality, due to the competition and the savings to consumers who want to move their computing infrastructure to the cloud. Reciprocally, choosing the right cloud microservice becomes a much harder task due to a greater number of candidate services. Moreover, users are not always technically skilled enough to estimate and understand their non-functional requirements. Thus, a recommendation system capable of understanding their needs could help. Different from a SOA service selection process, initially restricted to an organization or a business partner network, the scope drastically increases when one shifts to a cloud environment. Thus, we revisit the problem of reusable component selection in the next section.

2.3 The COTS Selection Problem

The use of COTS software has been an economic and strategic need for many organizations. The prospect of reducing the time and cost associated with software development has led organizations to an increasing interest in acquiring and integrating commercial products instead of developing systems from scratch (ALVES, 2005). The idea of developing systems from software components is not new. The principle of mass-produced software components has been suggested back in 1968 by McIlroy (1968). Since then, the software industry has moved progressively towards a coarse-grain component-based paradigm, though MSA has recently shifted this trend. This trend can be verified by the large availability of off-the-shelf components, such as software development environments, operating systems, database management systems, and business specific applications (ALVES, 2005).

Organizations expect to gain a number of benefits, including faster system development time, lower development costs, and continual product improvement by using COTS products. Given that the number of customers using off-the-shelf products is likely to be wide and diverse, the opportunities to surface problems increase and ultimately lead to a more stable and mature product. Moreover, the costs to acquire COTS products are expected to be lower than to develop customized systems because the product development costs can be shared among many users (ALVES, 2005).

Conversely, the development of complex COTS-based systems is known to be an intricate and risk prone process (WALLNAU et al., 2002; FINKELSTEIN et al., 1996). Thus, to deal with such challenges, the evaluation and selection of COTS products have to be carefully performed (ALVES & CASTRO, 2001; NCUBE & MAIDEN, 1999, MEYERS & OBERNDORF, 2001). In addition, according to Alves (2005), the functionality and quality of COTS candidates have to be assessed against the requirements of the acquirer organization as part of the selection activity.

Given that COTS products are developed to satisfy the requirements of an entire market, e.g., one or more SECO platforms, instead of the specific requirements of the buyer organization, mismatches may occur between what is desired by the stakeholders of the organization and what is possible to achieve with the COTS product (ALVES & FINKELSTEIN, 2003). An additional difficulty is that the degree of confidence in the way COTS candidates satisfy a particular requirement may be low because of the lack of trusted and complete information about COTS capabilities (ALVES, 2005). In this sense, a social-based strategy might increase the confidence, by assessing other user's opinions over the COTS under interest part of the decision process. Furthermore, when it comes to infrastructure microservices, e.g., a relational database, these might be considered “commodity” components, offering lower functionality gaps risks or mismatches related to the expectation if compared to functional microservices.

Much effort has been devoted trying to agree on a definition of what constitutes a COTS product (CLARK & TORCHIANO, 2004). In the scope of this Thesis, the definition given by MEYERS & OBERNDORF (2001) will be considered. Such definition states that COTS is a product that is: sold, leased, or licensed to the general public; offered by a vendor trying to profit from it; supported and evolved by the vendor, which retains the intellectual property rights, available in multiple, identical copies; used without internal modification by a consumer (MEYERS & OBERNDORF, 2001). Lastly, different from the work of Alves (2005), this Thesis focuses on microservices a particular, finer-grained class of COTS.

2.3.1 COTS Selection Approaches

According to Alves (2005), the typical steps involved in the selection of COTS include the identification of COTS candidates available in the marketplace, evaluation of products, decision to select/reject products, and finally acquisition of the best COTS product. It has been widely agreed that requirements engineering is a core activity to ensure the success of such selection process (MAIDEN & NCUBE, 1998; CHUNG & COOPER, 2002; ROLLAND, 1999; CARNEY, 1998).

The Off-The-Shelf Option (OTSO) method (KONTIO, 1995; KONTIO, 1996) was one of the first COTS selection methods proposed in the literature. OTSO is a well-defined method that covers the whole selection process. The definition of hierarchical evaluation criteria is the core task of this method. The criteria consist of a set of functionalities, architectural constraints, and organizational needs. The selection activity identifies four different subprocesses: search criteria, definition of the baseline, definition of evaluation criteria, weighting of criteria.

A controversial feature of OTSO is the way it deals with quality aspects (e.g., reliability, portability, performance), since the methods assume that these are extra factors that may influence the decision but do not necessarily need to be included in the evaluation criteria (ALVES, 2005). This position has been contested by other researchers who argue that properly assessing quality requirement is a fundamental step to ensure the successful selection and integration of COTS (CARVALLO et al., 2003; BEUS-DUKIC, 2000). OTSO uses the Analytic Hierarchy Process (AHP) technique (SAATY, 1980) to conduct the decision to select or reject COTS products – the same technique implemented in this Thesis' proposed framework.

Alves (2005) also point out that Tender Management[12], from Telelogic, is one of the few commercial tools aimed at supporting the COTS procurement process. The methodology behind Tender Management has been developed based on best practice across industry and government sectors. Tender Management is integrated with the Doors requirements management tool[13], hence allowing useful traceability links between requirements and COTS evaluation criteria. Alves explains that the tender assessment

[12] Ten (2005). Tender Management. http://www.telelogic.com.

[13] https://www.ibm.com/support/knowledgecenter/en/SSYQBZ_9.6.0/com.ibm.doors.requirements.doc/topics/c_welcome.html

process consists of four phases: preparation, scoring, decision, and completion. For each phase, the roles of people involved in the evaluation process are explicitly defined, facilitating the assignment of responsibilities and the tracking of decisions made through the process. The preparation includes the definition of hierarchical evaluation criteria. Each criterion has a weight associated in order to identify core and distinguishing criteria. In this sense, the tool developed to support this Thesis' proposed framework also defines a hierarchical evaluation criterion, while allowing for different weights for each quality attribute, as will be detailed in Chapter 3.

2.3.2 Decision Making Techniques

The evaluation of COTS products is considered a form of decision-making where COTS candidates are assessed and ranked according to their relative importance to meet the customer requirements. The generic decision-making process involves the following steps: (i) identify the alternatives; (ii) define the evaluation criteria; and (iii) rank the alternatives against criteria.

The evaluation of COTS can be characterized as a decision problem involving multiple objectives. This type of decision is known in the literature as multi-criteria decision analysis (MCDA). This is a well-established research area that aims to provide quantitative models to support complex decisions. MCDA techniques provide well defined strategies to evaluate and score alternatives. These techniques are based on the notion of underlying preferences. One of the most widespread MCDA approaches to support the evaluation of COTS products is AHP (SAATY, 1980). This technique has been adopted by OTSO and this Thesis' microservice selection framework. AHP is a multicriteria decision technique that can combine qualitative and quantitative factors in the overall evaluation of alternatives. AHP enables decision makers to understand complex decisions by decomposing the problem in a hierarchical structure (ALVES, 2005). Decision makers then make simple pairwise comparison judgments throughout the hierarchy to arrive at overall priorities for the alternatives. Frair (1995) suggests the following generic steps to apply AHP to solve a decision problem:

1) Build a decision hierarchy by decomposing the problem into individual criteria;
2) Obtain relational data for the decision criteria and alternatives and encode using the AHP relational scale;
3) Estimate the relative priorities of the decision criteria and alternatives;

4) Perform a composition of priorities for the criteria that gives the rank of the alternatives.

To answer questions related to the comparison of candidates, decision makers must have a complete understanding as concerns the performance of both products and components. According to Alves (2005), given the high level of uncertainty regarding COTS product capabilities, performing such comparisons may be difficult if not impossible. Unless, one could leverage some form of information catalog about such candidates. That is the approach of this Thesis' proposal, i.e., to query a catalog of microservice metadata on several platforms to get information that would allow such automated comparisons as described in Chapter 3.

2.3.3 The COTS Selection Problem Revisited

As stated in Section 1.3, it is known that the problem of selecting a reusable software asset is not new. The work of Mohamed et al. (2007) discusses practices that date back to 1995. The authors confirm the importance of selecting appropriate COTS, in a "component-based development", as the most crucial phase. According to them, COTS selection is the process of determining the fitness-of-use of COTS products in a new context, and then select the product with the highest fitness. In a modern SECO, a microservice available on the PaaS can be considered a COTS component. In that sense, a SE practitioner could benefit from having a catalog of available software assets, as well as means to select the most appropriate one.

Despite several efforts made during the last decade to model the COTS selection process, the same authors argue that none of the existing methods could be considered a "silver-bullet" to solve this problem, thus leaving room for new approaches targeting different contexts. On the other hand, they have defined a general selection process, as follows:

1) Define the evaluation criteria based on stakeholders' requirements and constraints;
2) Search for COTS products;
3) Filter the search results based on a set of "must have" requirements. This results in a short list of promising COTS candidates, which are to be evaluated in more detail;
4) Evaluate COTS candidates on the short list;

5) Analyze the evaluation data (i.e., the output of Step 4) and select the COTS product that has the best fitness with the criteria. Usually, decision-making techniques, e.g., AHP, are used for making the selection.

Though one could still claim that this is an old problem, which has been already solved, according to NIST (2015), with Cloud Computing in the mainstream, there is a preponderance of cloud-based microservices in the market, making the choices for consumers to increase daily. This is known as *choice overload* (WHITMORE, 2001), a cognitive process in which a person has a difficult time making a decision when faced with too many equivalent options. This phenomenon has been associated with unhappiness (SCHWARTZ, 2004), decision fatigue, going with the default option, as well as choice deferral – avoiding making a decision altogether, such as not buying a product or acquiring a service (IYENGAR & LEPPER, 2000). In this case, making a decision becomes overwhelming due to the many potential outcomes and risks that may result from making the wrong choice. Having too many approximately equally good options is mental draining because each option must be weighed against alternatives to select the best one. Thus, comparing a plethora of service offerings, between various cloud providers, is not a straightforward exercise.

The work of Guinard et al. (2010), which focuses on device integration, shares some similarities with this research, such as relying on service metadata information. From a technology point of view, their key challenge is how to discover, assess, and integrate real-world services into business applications. While their work is IoT specific, the research described here may include, but it is not limited to, IoT microservices. On the other hand, their proposal encompasses the support for on-demand service provisioning. Though DIRECTOR could also allow that (via PaaS' APIs), it is not the focus nor included in this research. Whilst their implementation supports only two quality of service attributes (service health and latency), this research's technical perspective includes seven attributes, as will be discussed in Chapter 3. Finally, their key contribution is a process based on a single strategy of network discovery of embedded devices and types query (keywords describing the type of service wanted), whereas the main contribution of our research is a framework adopting complementary perspectives (technical, social and semantic).

When it comes to requirements, constraints and priorities, previous work have identified and addressed, at least partially, the challenge of ranking and selecting based on multiple criteria (TRIANTAPHYLLOU, 2000). In the literature, it is defined as multiple-criteria decision-making (MCDM) problem, e.g., finding the most suitable microservice among a list of candidates by ranking them based on several quality metrics. As in related work (MENZEL et al., 2013; NIE et al., 2011; NIZAMANI, 2012; GARG et al., 2011; GODSE & MULIK, 2009; KARIM et al., 2013), DIRECTOR's technical and social perspectives rely on a dynamic ranking algorithm based on AHP (SAATY, 1990). AHP is one of the most widely used mechanism for solving problems related to MCDM when the number of choices is known beforehand (SAATY, 2000). The semantic perspective does not make use of such algorithm, as it relies on an Artificial Intelligence technique explained in Chapter 4.

2.4 Service Measurement Index

This research focuses on microservices as a subtype of COTS. In that regard, a framework named Service Measurement Index (SMI) provided the basis for the quality of service model proposed to compare microservices in technical terms. The seven categories of quality attributes and the proposed model will be described in Chapter 3.

As the US government mandates to move substantial amounts of data storage, software deployment, and other services from individual agencies to cloud providers (KUNDRA, 2010), the pressing need for development of publicly available measures of service quality from different cloud providers has taken on an urgency that researches at Carnegie Mellon University recognized. To ensure that the work would be globally relevant and useful for the public and private sector users, the creation of a research consortium seemed advisable. Thus, in 2010, the Cloud Services Measurement Initiative Consortium (CSMIC) was formed to address the need for industry-wide, globally accepted measures for calculating the benefits and risks of cloud-computing services.

A team in CSMIC has developed SMI: a standard measurement framework (SIEGEL & PERDUE, 2012). SMI involves the application of consistent, meaningful measures that are designed to enable comparison of current cloud-based services with non-cloud services or cloud services available from multiple providers. Cloud services characteristics whose measures were documented and tested include *accountability*, *agility*, *assurance*, *financials*, *performance*, *security and privacy*, and *usability*. Fifty-one quality attributes were designed and distributed among these seven categories. For instance, the *agility* category indicates the impact of a service upon a client's ability to change direction, strategy, or tactics quickly and with minimal disruption. It consists of the following six attributes:

- *Adaptability*: the ability of the cloud service provider to adjust to changes in client requirements;
- *Elasticity*: the ability of a cloud service provider to adjust its resource consumption for a service at a rapid enough rate to meet client demand;
- *Extensibility*: the ability to add new features or services to existing services;
- *Flexibility*: the ability to add or remove predefined features from a service;
- *Portability*: the ability of a client to easily move a service from one cloud service provider to another with minimal disruption; and
- *Scalability*: the ability of a cloud service provider to increase or decrease the amount of service available to meet client requirements and agreed SLAs.

SMI is a hierarchical framework. The top level divides the measurement space into seven categories and each category is further refined by four or more attributes. Then, within each attribute, a set of Key Performance Indicator (KPI) are defined that describe the data required to measure the related attribute. Some of these KPIs will be service specific, while others will apply to all services (BPaaS, IaaS, PaaS, and SaaS)[14]. SMI defines a framework and method for the calculation of a relative index, which may be used to compare IT Services against one another or to track services over time.

During the exploratory phase of this research, the metadata extracted from PaaS' catalogs was inspected and associated to SMI categories, since SMI in its current form does not account for microservices as a type of cloud service – microservice as a service or **µSaaS**. From a technical perspective, it was possible to identify quality attributes in six out of the seven SMI categories. *Accountability* was the only one left out since all metadata available was related to microservices and not the providers. The inferences drawn from the PaaS' microservice metadata were further ratified by a survey with software architecture experts from the industry, as detailed in Chapter 3.

[14] Business Process as a Service (BPaaS) is any type of horizontal or vertical business process that is delivered based on the cloud services model. These cloud services, which include Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS), are therefore dependent on related services.

2.5 Architectural Decisions Regarding Software Acquisition

Once the choice of which microservice to be acquired has been made, the IT architect should document the decision. In this context, organizations have been increasingly interested in software acquisition (WEBER et al., 2007), due to all recognized benefits from reuse, such as quality and productivity gains. A related reference model, ISO 12207:2008 (ISO, 2008), considers the acquisition of a software product or service as a fundamental process, where a recommendation tool can be used to support the associated activities, avoiding a judgmental decision.

Regarding the solution being designed, one should note that decisions as such should be documented in an Architectural Decisions artifact (FOWLER, 2003) which describes design decisions addressing architecturally significant requirements perceived as hard to make and/or costly to change. They are also known as Architecture Decision Records (ADRs), a text file in a format similar to an Alexandrian pattern[15] that describes a set of forces and a single decision in response to those forces (NYGARD, 2011). They communicate the rationale behind decisions made in the past, e.g., software components reused in existing solutions, the alternatives considered and criteria used for selecting one or another. Moreover, one should notice that the use of tools for supporting design rationale is a common practice, as discussed by Kruchten et al. (2009).

In summary, as acquiring a software component for a new solution might be considered as an architectural decision, it should be easy to communicate to different stakeholders, including other architects. The decision view of Software Architecture considers the architecture as the set of design decisions (DUEÑAS & CAPILLA, 2005), which affect the non-functional characteristics of a system and that should be documented in an architectural decision document containing the alternatives and the justification for the chosen one. In this regard, the proposed framework can help not only by providing more alternatives (some that could be missed or discarded due to the architect's bias), but also for informing the rationale behind the perspectives' recommendations, such as the level of community engagement of a given microservice.

[15] Christopher Alexander's patterns consist of a short name, a rating (up to two '*' symbols), a sensitizing picture, the context description, the problem statement, a longer part of text with examples and explanations, a solution statement, a sketch and further references. This structure and layout are sometimes referred to as the "Alexandrian form".

2.6 Final Remarks

Although there are many research works focusing on this challenge, to the best of this research's knowledge, none of them addressed the microservice granularity level, nor applied an Artificial Intelligence technique to provide recommendations based on the understanding of the requirement described by the practitioner, in his/her own words. In addition, according to Sun et al. (2014), cloud service selection still has open challenges, as most of the work in the literature is too general, or focusing exclusively on the IaaS (infrastructure), PaaS or SaaS (software) layers. Conversely, it is reasonable to presume that developers tend to be more interested in the available microservices that can be reused, than on the PaaS itself – actually, the amount of PaaS' resources might ultimately drive their choice of platform or provider. Finally, also according to them, a large proliferation in the number of cloud services on the Internet brings new challenges in cloud service discovery and selection.

In this chapter, the concept of SECO was introduced to contextualize this research. We started with an introduction to this topic, providing definitions and its relation to software reuse. We briefly described the three dimensions of a SECO: technological, social and economic, as well as how it depicts the most recent generation of software reuse. Modern SECOs and MSA were both strongly affected by the emergence of Cloud Computing. As Cloud Computing enabled cloud-based platforms of modern SECOs, it has also expanded the proliferation of fine-grained services, named microservices, which have specific purposes and can be deployed independently. Finally, this chapter tried to stress the challenge of choosing the right microservice among many alternatives, as even an experienced software architect might face *choice overload* in this respect. That is the reason why we revisited the COTS selection problem in the light of cloud-based SECOs. We discussed previous work on this topic, though none has addressed this problem in a SECO/MSA context.

With an understanding about SECO and its evolution, the distinction between SOA and MSA, the effects of the emergence of Cloud Computing, and with the challenge of software acquisition through COTS selection, the next chapter presents the findings from two surveys conducted with practitioners from industry regarding extracting quality attributes from a cloud platform's microservice catalog metadata.

Chapter 3 – Surveying Industry Experts

This chapter describes two surveys with software architects from industry. First, we present a survey that sparked the ideas about the recommendation of cloud microservices. To address technical aspects of microservice selection, a quality of service model is designed based on the metadata available from cloud platforms supporting the Cloud Foundry standard. Then, a second survey was designed and executed to evaluate whether the set of propositions associating platform metadata with the proposed technical model made sense for experts in the field.

3.1 Introduction

In February 2017, a survey was carried out with 23 subject matter experts from the industry regarding software architectures in SECOs – especially considering the cloud-based ones described in the previous chapter. Back at that time, this research was still in its exploratory phase, where the scope was broader than cloud microservices' discovery, evaluation, and recommendation. The goal of that survey was to evaluate the relevance of non-functional requirements for SECO application architectures and get feedback on how to document and evaluate a reference architecture, from the point of the view of experts in Software Architecture. The sources used for specifying the set of items to be evaluated were described and consolidated through a technical report and a systematic literature mapping[16].

The survey was run from February 1st, 2017 to February 22th, 2017. Thirty invitations were sent, out of which 23 invitees responded. The average participant's age was 42 and they were from the United States, Germany, Hungary, Australia, and Brazil. Most participants had large experience in the Software Architecture area – around 50% had more than ten years of experience. They had significant experience with IT and most worked at the private and industrial sectors. Their job roles were IT Architect, Application Architect, Solution Architect, Enterprise Architect, Software Architect, Data Architect,

[16] "Software architecture for Software Ecosystems: A systematic literature mapping and research agenda", May, 2019, Technical Report at COPPE/UFRJ and submitted to Technological Forecasting and Social Change, an International Journal by ELSEVIER in 05-11-2019.

IoT Software Engineer, and even one Architect Director – 65% had an “architect-like” job role.

From an academic viewpoint, 50% of the participants had a Bachelor’s degree, and 30% a Master. Finally, from a SECO context, 70% were currently participating in at least one ecosystem as platform evangelist, developer, customer, or lead organization. Therefore, one could assume some degree of familiarity with the survey’s background. To reduce bias, the strategy was to invite practitioners from different geographies, i.e., architects from different countries that could provide different types of feedback based on their own experience with ecosystems.

During the interviews that followed that survey, one respondent (an IT architect working for a global consulting firm) mentioned that choosing a microservice for a cloud-based application was a concrete and relevant problem that, most of the time, was addressed arbitrarily. He mentioned that microservices were usually selected because they were already known by the practitioner or by the development team. When other respondents were questioned about this scenario, many shared a similar perspective. While it is largely acknowledged that “familiarity” is one of the drivers for acquiring a specific software component or technology, if this is the sole reason being considered, then one could be losing sight of potential candidates which, in turn, could lead to improvement opportunities or cost savings. This was the seminal motivation for this research, which intends to provide the means (and tools) for practitioners to make informed architectural decisions, instead of judgmental ones (FRANÇA, 2017).

Inspired by the findings of this survey, a technical quality of service model was designed to capture the technical aspects of microservice selection. The basis for such a model was the metadata available in cloud platform providers. As this metadata varies among providers, the Cloud Foundry (CF) standard for cloud platforms was selected as the basis for the quality of service model. CF is a dominant open-source framework for PaaS. Then, a second survey was conducted to address the selection of cloud microservices and to validate the inferences about the technical relationships between the metadata provided by cloud microservice providers and the SMI-based faceted scheme presented in Section 2.4.

Forty-four participants were selected from the researches’ professional network of contacts. Respondents included enterprise architects, cognitive computing specialists, IT executives, solution architects, software engineers, among other job roles, from Australia, Brazil, Portugal, Germany, Hungary, and the United States. More than half of them (68.1%) had more than 5 years of experience with IT/software architecture, 6.8% having more than 20 years of experience. The vast majority, about three-quarters of them (72.7%), has been responsible for making architectural and/or design decisions in their organizations.

3.2 A Microservices Technical Quality of Service Model

As mentioned in Chapter 1, the DIRECTOR framework here proposed for microservice selection from cloud platform providers includes three complementary perspectives. The first one, the Technical Perspective, ranks a set of candidate microservices adopting a set of technical criteria. An objective criterion is required for such a ranking and it depends on the information available about the microservices.

To establish a quality (QoS) taxonomy, a search for quality metrics for cloud services was performed, which led to the SMI standard presented in Section 2.4. At least two other standards were considered: Cloud Computing Service Metrics Description from NIST[17], and ISO/IEC 25000 (ISO, 2005). However, SMI seems to be closer to the cloud microservices context, due to its focus on cloud services and providers. Therefore, SMI was selected as the basis for the quality attributes that were evaluated for each microservice based on the information available about it.

The information available for each microservice varies according to the cloud provider where it is deployed. Some of these providers attend to frameworks that were developed to reduce dependence on given providers and to standardize the information available on the microservices that they provide. According to the Gartner Group (2014b), digital business is driving interest in PaaS and Cloud Foundry (CF) and OpenShift[18] are the dominant open-source frameworks. These frameworks have amassed the strongest development communities. Both make possible to write code in a variety of languages and deploy applications to public or private clouds.

In the context of this research, CF was chosen due to having a larger support community (Cisco, Dell EMC, IBM, Pivotal, SAP, SUSE, VMware, among many others) and more PaaS offerings (Anynines, AppFog, Atos CF, IBM Cloud, devpack, Heroku,

[17] https://www.nist.gov/sites/default/files/documents/itl/cloud/RATAX-CloudServiceMetricsDescription-DRAFT-20141111.pdf

[18] https://www.openshift.com/

HPE Helion Stackato, MoPaaS, Pivotal CF, Pivotal Web Services, Swisscom Application Cloud), as reported by PaaSfinder[19]. Therefore, the technical perspective relies on information that the CF framework requires from providers.

The ranking of microservices is calculated from metadata gathered from the CF-compliant cloud platform offering the microservice that is used according to a quality of service (QoS) model adapted from SMI. The proposed model is based on inferences between SMI quality attributes and data available in the microservices' catalogs, regarding the microservice itself and its commercial plans. The relative ranking values for all candidates are calculated using AHP, by comparing their "scores" with each other and sorting them down. Thus, to calculate the rank of each QoS attribute, the value for that attribute is compared with the values for the same attribute on all other candidate microservices, taking into consideration the user assigned weights (priorities) for each QoS category. Then, for each candidate, the rank of all criteria is aggregated to generate the global ranking for all candidates. For instance, considering N candidate microservices, the best would score N points, the second best, N-1, the next, N-2, and so on, until the worst classified on that attribute would score a single point.

Some attributes are "negative", i.e., the lower the absolute value, the better. For example, adaptability, where the difference in days between the current date and the last update is calculated with the assumption that clients continue to request new features. Thus, one candidate having a more recent version (for instance, released last week) might be better (for instance, it has incorporated the latest requested features) than one whose last version is a year old. Additionally, there is no attribute being currently used for the *Accountability* category because this category is related to microservice providers instead of microservices. The complete list of currently mapped attributes, resulting from adapting SMI for microservices, is displayed in Table 3.1.

Next, we introduce each attribute comprising the quality attribute model and the rationale for its calculation from the cloud platform metadata.

- **Scalability**: Scalability is the property of a system to handle a growing amount of work by adding resources to the system. Here, each microservice has at least one commercial plan, which charges for a certain amount of resources. Many platforms

[19] https://paasfinder.org/vendors

allow the upgrade of microservices to bigger commercial plans, paying an extra amount of money for more resources. Thus, to evaluate if a microservice is scalable, it is checked whether it allows the user to change from one commercial plan to a better one, i.e., if the "s.plan_updateable" field equals to "true";

Table 3.1: SMI QoS attribute mapping.

Category	**Attribute**	**Microservices Metadata[a] Field**
Agility	Scalability	s.plan_updateable
Agility	-(Portability)	s.requires
Agility	Flexibility	# of service plans
Agility	-(Adaptability)	s.updated_at
Assurance	-(Availability)	s.active
Assurance	Stability	s.created_at
Financial	On-going cost	sp.free
Performance	Functionality	s.tags
Security & Privacy	Access control & Privilege management	s.bindable
Usability	Accessibility	sp.public
Usability	Learnability	s.documentation_url

[a.] The metadata resource (entity): "s" for "Service" and "sp" for "Service (commercial) Plan".

- **Portability**: Portability is the ability to use the same software in different environments. Here, each microservice could have a set of dependencies. The more the dependencies, the less the portability. Thus, to evaluate the degree of portability of a microservice, we count the number of dependencies in the "s.requires" field;
- **Flexibility**: Software flexibility is one of the properties that indicate if the software is easy to change. Here, it means that one can change the microservice itself, for instance by choosing a commercial plan that allows higher computational power or network bandwidth. The more commercial plans a microservice has, the higher is its flexibility. Thus, to measure the flexibility of a microservice we count the number of service plans through which it is offered;
- **Adaptability**: From an architecture standpoint, adaptation of software systems is an almost inevitable process, due to changes in customer requirements, the need for faster development of new, or maintenance of existing, software systems, among other factors (SUBRAMANIAN & CHUNG, 2001). Thus, if a microservice was adapted, for any of those reasons, it needed to be rebuilt and updated in the platform.

Thus, adaptability (a negative attribute) is calculated by looking into the "s.updated_at" field that contains the date of the last time the microservice was updated. Recent dates are preferable to older ones. Therefore, the difference in days from the current system date to the "s.updated_at" times -1 is calculated;

- **Availability**: High availability means a software system that is running and available most of the time. We check the degree of a microservice's availability by counting how many inactive days it had during a period. Thus, availability (a negative attribute) is measured by counting the number of days on which the "s.active" field was equal to "false". The higher this number, the worst is the availability of a given microservice;
- **Stability**: Stability in a software system is closely related to reliability. We assume that the older the microservice, the more stable it is, i.e., the older the date in the "s.created_at" field, the better. Therefore, the difference in days from the current system date to the "s.created_at" times -1 is calculated);
- **On-going cost**: Software on-going costs could include licenses fees and support. In the case of a µSaaS, it is the price (usually monthly charged) of the chosen microservice's commercial plan. However, some microservices might not charge at all, which could be of interest to companies with restricted budgets. Thus, it is checked whether a microservice has at least one free commercial plan, which would mean that it could be consumed without any cost at all. For that, it should be checked if any record of "service plans" with the field "sp.free" equals to "true" exist, which would be good;
- **Functionality**: In information technology, functionality is the sum or any aspect of what a product, such as a software application or computing device, can do for a user. Here, it is checked how many "tags" a microservice has, i.e., a count of the values in the list from the "s.tags" field, where higher values means better;
- **Access control & privilege management**: While privilege management encompasses many strategies, a central goal is the enforcement of least privilege, defined as the restriction of access rights and permissions for users, accounts, applications, systems, devices (such as IoT) and computing processes to the absolute minimum necessary to perform routine. In that sense, a popular pattern is to store

configurations in the environment[20], instead of hard coding them in the source code. Modern cloud platforms account for that by allowing a microservice to be directly bound (provided it possesses such feature) to an application, receiving the configurations directly from it – all the management is through the platform itself (secret or parameter vaults). This could be done by checking whether the "s.bindable" field is equal to "true", which would be good;

- **Accessibility**: The goal of accessibility is to ensure that application software is available to and usable by the widest possible audience. Here, it is checked whether a microservice is publicly available for anyone on the platform, i.e., by looking into the "sp.public" field of any record of "service plans", which would be good;
- **Learnability**: Learnability usually means how quickly a new user can begin efficient and error-free interaction with a system. Here, it is checked whether a microservice provides a website, e.g., a wiki or forum, so that new users might easily find supporting documentation. This could be done by looking into the "s.documentation_url" field, where the existence of a URL would be positive.

3.3 Surveys

A survey is a research method used for collecting data from a pre-defined group of respondents to gain information and insights on a topic of interest. As a field of applied statistics of human research surveys, survey methodology studies the sampling of individual units from a population and associated techniques of survey data collection, such as questionnaire construction and methods for improving the number and accuracy of responses to surveys (GROVES et al., 2011).

The method could be summarized as: (i) defining an audience that should consist of a group of survey respondents; (ii) defining the survey questions; (iii) administering the survey by querying the audience; and (iv) analyzing the responses to gain insights. According to GROVES et al. (2011), the most important methodological challenges of a survey methodologist include making decisions on how to:

- identify and select potential sample members;

[20] https://12factor.net/config

- contact sample individuals and collect data from those who are hard to reach (or reluctant to respond);
- evaluate and test questions;
- select the mode for posing questions and collecting responses;
- train and supervise interviewers (if they are involved);
- check data files for accuracy and internal consistency;
- adjust survey estimates to correct for identified errors.

While there are various types of surveys, they are categorized into two broadly dimensions: according to instrumentation and according to the span of time. The types of survey according to instrumentation include the questionnaire and the interview, while cross-sectional surveys and longitudinal surveys are related to the span of time through which the survey was conducted (SINCERO, 2012).

In survey research, the instruments that are used can be either a questionnaire or an interview, either structured or unstructured. In this research, both were used and the interviews always followed the questionnaire. Nowadays, on-line questionnaires (or web surveys), such as the ones provided by Google Forms[21], are a digital instrument that is typically administered to respondents. The usual questions found in questionnaires range from closed-ended questions, which are followed by response options, to open-ended questions, aimed at exploring the respondents' answers.

On the other hand, interviews are more personal and probing. Questionnaires do not provide the freedom to ask follow-up questions to explore the answers of the respondents, but interviews do (SINCERO, 2012). An interview includes two persons – the researcher as the interviewer, and the respondent as the interviewee. One could conduct personal or face-to-face interviews, phone interviews, and, more recently, online interviews – the type adopted in this research due to the geographic diversity of the respondents' locations.

Additionally, the span of time needed to complete the survey originates two different types of surveys: cross-sectional and longitudinal (SINCERO, 2012). Cross-sectional surveys collect information from the respondents at a single period in time. They usually use questionnaires to ask about a particular topic at one point in time. This was

[21] https://www.google.com/forms/about/

the type of the survey described in this chapter. Conversely, when the researcher attempts to gather information over some time or from one point in time up to another, he or she is doing a longitudinal survey. Longitudinal surveys aim to examine the changes in the data gathered. They are used in cohort studies, panel studies, and trend studies (SINCERO, 2012).

3.4 A Survey about Microservices Quality Attributes

Inspired by the feedback received after executing the first survey, the goal of the second survey was to validate a set of assumptions regarding the quality attributes for microservices available in cloud-based SECO platforms that were chosen to drive the technical perspective of the microservice selection framework we propose. We have inquired subject matter experts whether they envisage a tool capable of supporting the architectural decision process when it comes to selecting SECO microservices.

3.4.1 Planning

The survey consisted of a questionnaire comprised of three types of questions: (i) characterizing questions, for collecting participant's profile; (ii) relevance degree, for statements about microservice quality attributes; and (iii) general comments about a microservice recommendation tool. The estimated time of response was 20 minutes.

A pilot study was executed with two participants to improve the questionnaire as regards its structure, questions, and instructions. After some adjustments, the survey was sent via e-mail to potential participants from our sample. Participants were chosen from personal contacts of the researcher, industry practitioners with experience in Software Architecture. In a "snowball-like fashion", they were asked to indicate other practitioners, especially software architects, who could participate in the survey.

As discussed in Section 2.4, CSMIC was formed in 2010 to address the need for industry-wide, globally accepted measures for calculating the benefits and risks of cloud-computing services. As part of the effort, a global team developed the SMI standard measurement framework (SIEGEL & PERDUE, 2012) to evaluate the characteristics of cloud services. The characteristics for which measures were documented and tested are *Accountability*, *Agility*, *Assurance*, *Financials*, *Performance*, *Security and Privacy*, and *Usability*. Fifty-one attributes were distributed among these seven categories.

This Thesis' framework for technically evaluating the characteristics of microservices infers QoS metrics by mapping eleven SMI attributes to the available platform metadata – PaaS such as IBM Cloud (Bluemix), Pivotal Cloud Foundry and SAP Cloud Platform. The objective of this survey, therefore, was to verify if such assumptions are reasonable and sound, by adopting a five-point scale as follows: totally agree; agree; neutral/do not know; disagree; and strongly disagree.

3.4.2 Execution

The survey was made available to respondents from September 1st, 2018 to October 31th, 2018. We sent 50 invitations for SE practitioners from industry, out of which 44 invitees responded. The response rate (88%) can be considered positive (very high) for this kind of survey – an average response rate is reported to be around 55% (BARUCH, 1999). One should take into consideration the fact that the majority of participants were known by the researcher, which might have increased the response rate, with the cost of also bringing some bias to the results.

The average participant's age is 38 and they are from several countries, such as United States, Germany, Hungary, Portugal, Australia and, mainly, Brazil. Figures 4.1 and 4.2 summarize the survey's data regarding participants' workplace and experience, respectively. Regarding experience, we further investigate the degree of work experience in general IT and Software Architecture.

Most participants have large experience in software architecture: around 30% have more than ten years of experience. They have significant experience with IT and mostly work at the private and industrial sectors. Their current job roles include IT Architect, Application Architect, Solution Architect, Enterprise Architect, Software Architect, Data Architect, IoT Software Engineer, and even one Architect Director – around 50% have an "architect-like" job role, e.g., Software Engineer.

One should bear in mind that, in the context of this survey, influencing design decisions was considered as experience with software architecture. This means that other SE practitioners, besides software architects, could reply informing that he or she had that sort of experience, depicted by almost one-third (31.8%) reporting having from zero to five years of experience with IT/Software architecture – though, in future surveys, we should add a "no experience at all" option.

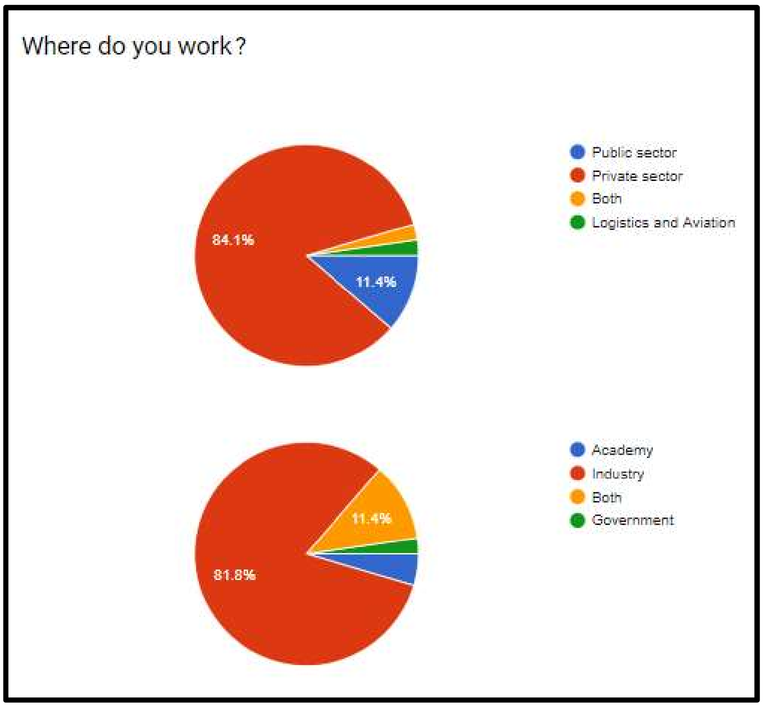


Figure 3.1: Survey participant's profile - workplace.

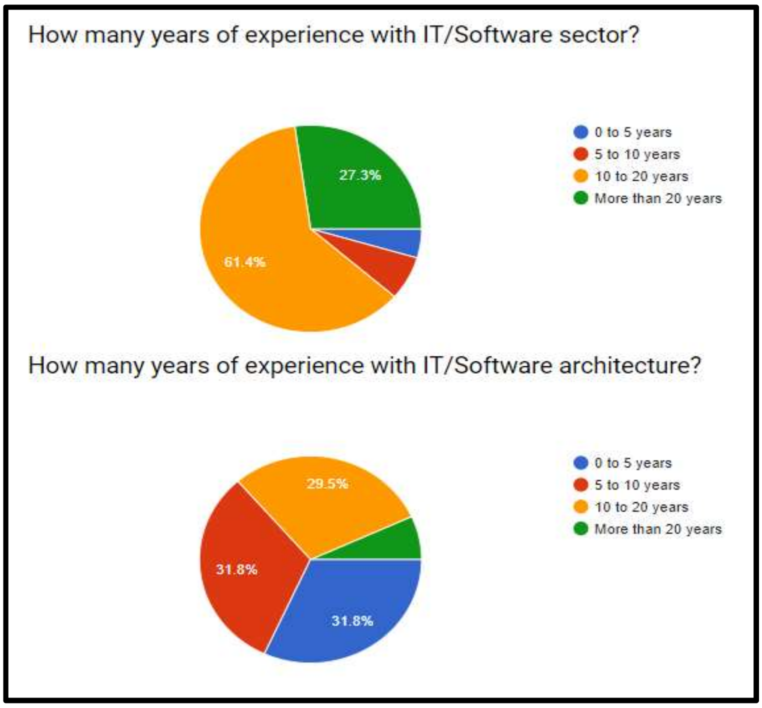


Figure 3.2: Survey participant's profile – experience.

From an academic perspective, 30% of the participants have a Bachelor degree, and 40% a Master. Finally, from an architectural perspective, 70% are currently responsible for making architectural and/or design decisions in their organizations (for instance, deciding or influencing the choice of a microservice for a new solution). Therefore, one can assume a degree of familiarity with the survey's context. Lastly, our intent by inviting architects from different countries that could provide different types of feedback based on their own reality was to reduce the aforementioned bias.

3.4.3 Analysis

Once the survey execution was completed and the data collected, some tasks to extract information were established. Three steps were performed to analyze the data gathered through the survey: data transformation and formatting; drawing response distribution plots; and manual analysis of the open questions.

Most answers were collected through a rating of propositions, such as "*If a microservice has several commercial plans, and it is possible to upgrade/downgrade to one more adequate to a given application's necessity/capacity/load, then this microservice has the quality of scalability*". So, the final distribution of the responses was used for the analysis of the technical perspective of microservice evaluation, to the extent of the participant's agreement or disagreement.

In this section, the main findings of this survey are discussed, while providing a summary of the responses. The survey itself was divided into two parts: the first part composed of closed propositions, and the second one composed of open questions – not all questions from the second part were mandatory.

3.4.3.1 Analyzing the Closed Propositions

Eleven propositions have been defined to ratify the validity of the mapping between SMI and the metadata provided by cloud microservice platforms, one for each QoS attribute. Nine of such propositions were ratified by the majority of the respondents, with an "agree" or "totally agree" answer. For the tenth attribute, 43.2% of the respondents agreed with proposition Q07, 27.3% disagreed, and 29.5% had nothing to say. As for the eleventh attribute, 36.3% of the respondents agreed with proposition Q08, while 27.3% disagreed and 36.4% had nothing to say. The eleven propositions are reproduced below, followed by Figure 3.3 showing a summary of the answers.

Q01) If a microservice has several commercial plans, and it is possible to upgrade/downgrade to one more adequate to a given application's necessity/capacity/load, then this microservice has the quality of scalability.

Q02) The more a microservice has requirements/dependencies for running, the less its quality of portability (moving it to another cloud provider).

Q03) The more commercial plans (different prices for different capacities) a microservice has, the more its quality of flexibility to accommodate one's demands efficiently.

Q04) If a microservice is constantly updated (new versions are released), then it keeps continually taking into consideration the customer's feedback about defects and requests for improvements (new features), having the quality of adaptability.

Q05) If a microservice is continually active, while being normally used, during an extended period of time (no outages), then it has the quality of availability.

Q06) Considering only the software's age, the longer a microservice has come into existence (has been created), and has been regularly used, the higher its quality of stability (maturity).

Q07) If a microservice has a free commercial plan, then its on-going cost is better than another that only has paid plans (considering only the financial aspect).

Q08) Considering only the functional aspect, the more features (classifications tags) a microservice has, the more its quality of functionality (it has more uses).

Q09) If a microservice has the ability to be bound/linked to an application, in a way that credentials do not need to be stored in the application itself, the better its access control and privilege management compared to another that requires the application to manage such information.

Q10) If a microservice is public, i.e., it can be used/provisioned by anyone, the better its accessibility quality (it can be more easily acquired), compared to another that is not publicly available.

Q11) If a microservice has documentation (URL) available, then it has the quality of learnability, i.e., one could learn how to use/consume it by going over its examples and API definition.

By analyzing Figure 3.3, one can argue that the majority of subjects agreed, at some degree, with the propositions related to the metadata extracted from the platforms (SECO PaaS). Although there were cases where replies such as “neutral” and “disagree” were given, those are fewer when compared to the others. We then claim that such propositions might be used to evaluate or to compare microservices by inferring QoS metrics from the metadata available in the platform’s microservices catalog.

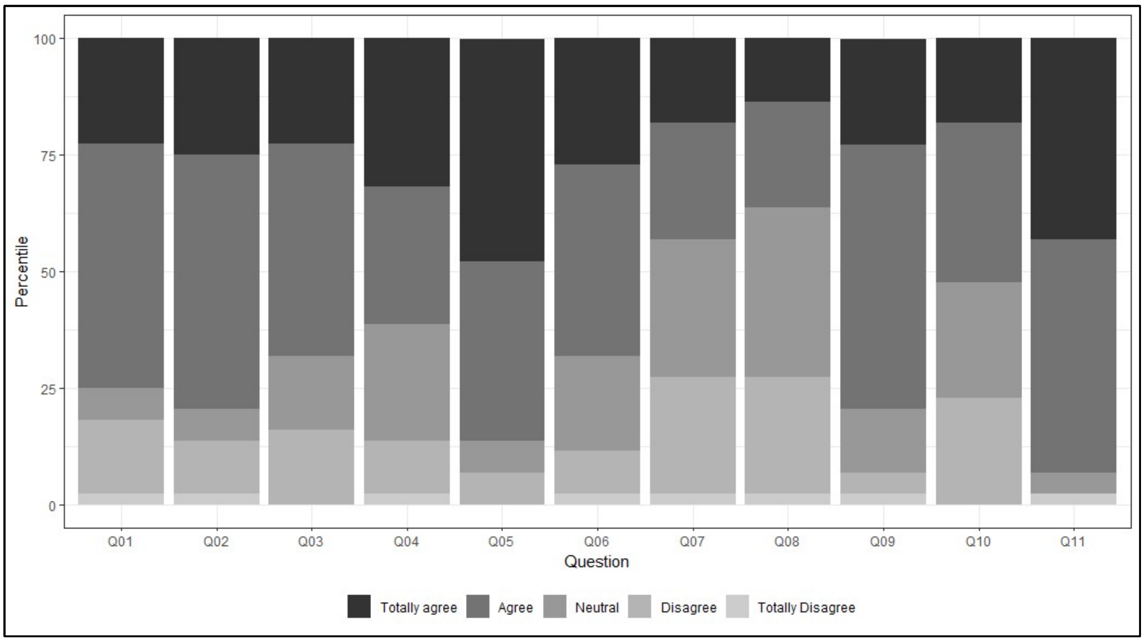


Figure 3.3: Summary of the answers to each question in the survey.

As already stated, the propositions Q07 and Q08 were the ones with the lower “agreement” rate among the respondents. Both receive the highest percentage of “disagree” responses (27.3%). Regarding Q07, some respondents probably understood that having a free plan is not enough to infer a lower financial cost (OPEX[22]). Similarly, in Q08, respondents might have not perceived the *tag* attribute as classification information for features – maybe the proposition should have included examples of this attribute to make it easier to understand the inference. Proposition Q08 received the lowest “agree” rate (36.3%), the highest “neutral” rate (36.4%), and the highest “disagree” rate among all propositions. Each third of the respondents had a distinct view about this proposition. The lack of convergence in their answers corroborates the conclusion that it should have been better formulated.

[22] Operating expenses represent the other day-to-day expenses necessary to keep the business running. These are short-term costs and are used up in the same accounting period in which they were purchased.

Finally, it is worth to point out that proposition Q11 received the highest proportion of agreement answers (93.2%). It seems straightforward that developers rely on available documentation to learn about an API. Thus, the vast majority of respondents agreed with the inference relating learnability with the existence of a URL address for a website that could be accessed for retrieving, for instance, sample codes on how to consume the microservice, deal with the returned JSON, etc. Proposition Q11 also received the lowest rate of "neutral" (4.5%) and "disagree" (2.3%) answers among all eleven propositions on this survey.

3.4.3.2 Analyzing the Open Questions

The third part of the questionnaire was comprised of two open questions:

1. *Would you envisage a tool being able to support the architectural decision process regarding choosing a cloud microservice for a solution? If so, what are the features such tool would have? (Please, consider only the decision-making process, not the governance/management that takes place after one has been chosen.)*
2. *Would you kindly consider providing us with additional comments? If so, you may use the space below.*

When questioned whether they envisaged a tool being able to support the architectural decision process regarding choosing a cloud microservice for a solution and the features such tool should have, practitioners provided answers, such as:

- *"Allow comparison between the criteria of 'purchase' of the microservice or its requirements";*
- *"Yes, these tools should consider more than one cloud provider to ensure comparability";*
- *"Evaluate and compare one or more microservices on the characteristics: security, accessibility, availability, cohesion, coupling, performance, quality, documentation and support";*
- *"Yes, a tool (a spreadsheet-like on-line tool or app) that could apply a weighted evaluation criterion to score services according to (a series of) nonfunctional requirements fulfillment capabilities, allowing weights to be*

> *customized to fit the particulars of each NFR being evaluated according to an architectural/project view"*.

No significant additional comments were suggested by the participants in the responses for the second survey's open questions.

3.5 Final Remarks

This chapter aimed to describe the survey conducted to validate propositions that were gathered from mapping the platform's metadata based on the Cloud Foundry standard to the SMI metrics. This survey helped in answering RQ1 – *What quality attributes, from a technical perspective, could be extracted from the metadata available on microservice catalogs, in cloud platforms?*

The survey confirmed the propositions and offered relevant input on how to design a recommendation tool that could support the architectural decision activity as regards to acquiring a software component in the shape of a microservice. From the plurality of the responses, we were able to identify the common points and take those in consideration to design the proposal described in the next chapter.

Chapter 4 – DIRECTOR Framework

This chapter describes DIRECTOR, a selection framework for the cloud era. It includes a tool capable of finding, evaluating, comparing and recommending the fittest microservice, according to three complementary perspectives. It can handle multiple SECO platforms and hundreds of microservices. In this chapter, this Thesis' proposal is positioned as an innovative solution for addressing a known problem in a new context.

4.1 Introduction

In modern SECOs, actors should leverage the resources available in the platforms to be able to compete. Software reuse is pivotal in this context and microservices are a popular unit of reuse. Despite having been addressed before in the SE literature, the COTS-selection problem, which involves finding an adequate software component for reuse, becomes more challenging in modern SECO due to the choice overload originated from the increasing number of platforms (PaaS) and microservices. The number of microservices available from different platforms makes it hard to apply manual comparison, requiring an automated way for finding and comparing them. Additionally, there is no single enterprise catalog, so one should rely on open-standards to retrieve information from each available SECO platform, consolidate it and run queries on it.

From a software architecture standpoint, choosing a microservice for a solution is an important decision, which can have financial, performance, and productivity effects. That is why a SE practitioner may benefit from having a tool capable of supporting this microservice selection process. Such a tool must take into consideration multiple criteria and different priorities when evaluating the candidate microservices, among them the factors that were also identified in the survey's results described in the previous chapter. Furthermore, to reduce bias, complementary perspectives could offer insights that might be missed when the number of candidates exceeds the hundreds.

This research proposes a framework to address these challenges, through the adoption of complementary perspectives (FRANÇA & WERNER, 2019). **DIRECTOR** includes a Web tool that works as a recommendation system for selecting microservices available in modern cloud platforms. DIRECTOR is a conceptual framework comprising the discovery, selection, and ranking of microservices through technical, social and semantic evaluation perspectives as depicted in Figure 4.1.

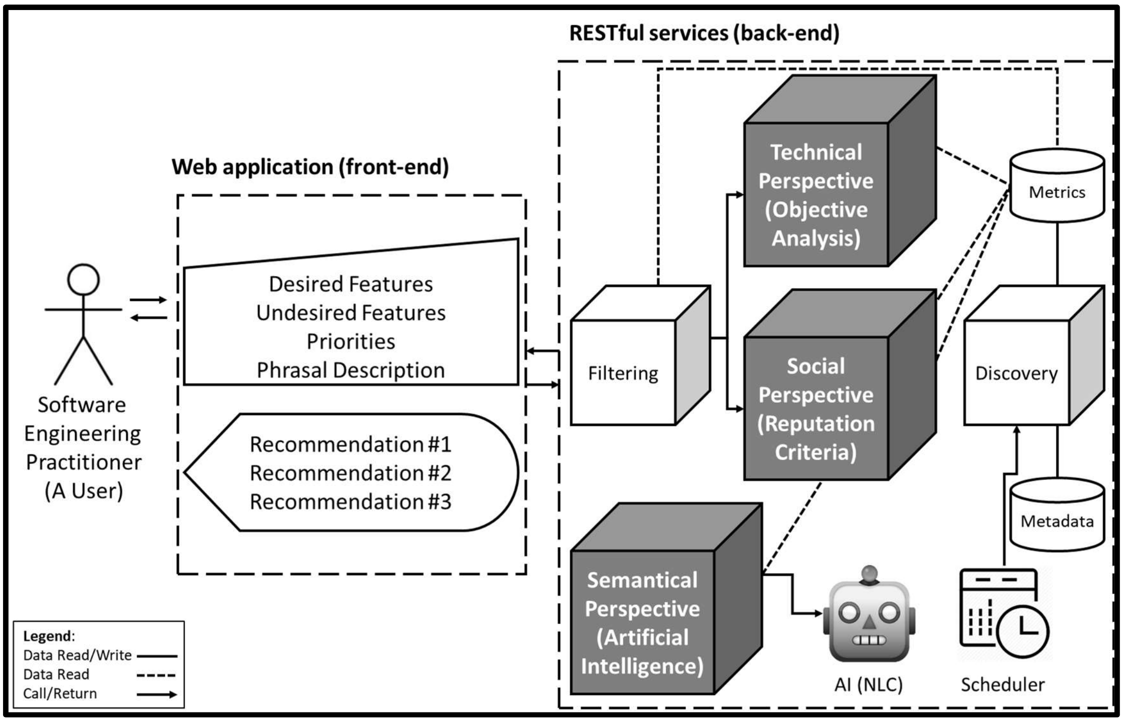

RESTful services (back-end)
Web application (front-end)
Technical Perspective (Objective Analysis)
Metrics
Desired Features
Undesired Features
Priorities
Phrasal Description
Software Engineering Practitioner (A User)
Filtering
Social Perspective (Reputation Criteria)
Discovery
Recommendation #1
Recommendation #2
Recommendation #3
Metadata
Semantical Perspective (Artificial Intelligence)
AI (NLC)
Scheduler
Legend:
Data Read/Write
Data Read
Call/Return

Figure 4.1: DIRECTOR's overview and high-level architecture.

The actor who uses DIRECTOR is likely to be a software engineer or a software architect trying to find a suitable microservice for a cloud solution, such as a software system that will be hosted in a cloud platform instead of on-premises. This is a common scenario in the industry nowadays, due to a series of cloud computing advantages, such as elasticity and charge model.

The selection process starts with the input of requirements, which can be garnered by selecting the desired features from a catalog of available microservices' characteristics, by entering a negative filter as exclusion criteria, or by entering a textual description of the desired microservice. DIRECTOR uses natural language processing to allow non-technically versed professionals (such as business analysts, product owners, and project managers) to participate in microservice selection. This is an important aspect of the framework as it is increasingly common that sponsors and clients participate and influence in technical discussions, particularly in agile teams.

DIRECTOR considers complementary perspectives to avoid a simple answer that could not express the complexities of the selection process. Technical aspects are taken into consideration, but social and semantic perspectives have an important play in the approach. The technical perspective, presented in Chapter 3, depends on metadata available in platform's microservice catalog. The remaining ones will be discussed in the following sections.

As the community's engagement could be pivotal to the survival of technology, such engagement was surveyed to evaluate the strength of a candidate microservice (social aspect). DIRECTOR queries Internet forums, especially popular ones such as StackOverflow, Twitter, Reddit, and Quora, to measure how many discussion threads mention the available microservices and possibly detect the "tone" of the opinions, e.g., if practitioners seem to be complaining about certain microservice's technology.

On its turn, cognitive technologies come into play to support understanding what non-technical users want and determine whether the available microservice could attend to their needs (semantic aspect). This perspective relies on a Natural Language Processing algorithm to allow an end user to express himself/herself in plain English, describe the goal of the desired microservice, and receive a recommendation based on cognitive analysis of the provided text.

In summary, DIRECTOR is capable of discovering and ranking microservices in one or more PaaS. It supports the microservice selection activities performed by SECO actors, e.g., developers or software architects, helping them to decide which microservice to reuse in a cloud context by "directing" them to adequate alternatives. It can be seen as a recommendation system that considers the priorities from the user's point of view, the features he or she needs, and a brief textual description of the main goal the user is trying to achieve. As a result, it provides the suggestions corresponding to the three complementary perspectives: technical, social and semantic.

4.2 Service Discovery

To match a set of requirements with a set of microservices from multiple cloud platforms, the starting point is to gather information about the microservices. Due to the cloud providers' characteristics, the interaction with the platforms may require distinct protocols. To address the challenge of accessing the catalog of active microservices from multiple providers, one could implement an "adapter" for each platform, directly consuming the platform's Web API through the *http/https* protocols. This would be time consuming, so a better approach would be adopting a PaaS standard, which allows the connection with any PaaS following the standard. As discussed in Chapter 3, the Cloud Foundry (CF) standard was adopted for this research. Actually, it is possible to mix both approaches, adopting multiple standards and creating specific adapters for cloud platforms that do not follow such standards. This task is performed by the DIRECTOR's discovery component, as depicted in Figure 4.1.

To deal with more and more microservices available in cloud-based platforms, DIRECTOR's strategy is to discover them by querying the service catalogs of such platforms in a way that should not be specific to a single one, but common to several of them, i.e., by standard means. That is why a widely adopted open-standard for PaaS becomes central in the proposed approach. Therefore, Cloud Foundry was the open-source cloud application platform (PaaS) adopted by this research. It has a marketplace named "The Foundry", which is the gateway to the CF ecosystem, e.g., services and integrations[23]. In turn, services are implementations of the Open Service Broker API[24] designed to work with CF including service integrations, build pack extensions and integrations that work with CF, comprising a total of 194 services (as of May 2019).

The Open Service Broker API project allows independent software vendors, microservice developers to provide backing services to workloads running on cloud native platforms, such as Cloud Foundry and Kubernetes. The specification, which has been adopted by many platforms and thousands of service providers, describes a simple set of API endpoints that can be used to query, provision, gain access to and manage service offerings. The project has contributors from Google, IBM, Pivotal, Red Hat, SAP and many other leading cloud companies.

In order to inspect the "health" of the platform and allow for metrics that depend on a period, a scheduler mechanism should be implemented, so that daily snapshots of the microservice catalog metadata could be retrieved and stored for analysis. As all of the monitored platforms follow the CF standard, a common API can be leveraged to get the microservices' metadata (a sample is provided on Section 4.4). For the metadata, a noSQL database is foreseen as a natural choice, while for the calculated metrics a relational database would speed up the query time.

4.3 Service Filtering

Given a set of microservices available for selection from multiple platforms registered with DIRECTOR (this is still a manual process of informing the endpoint of

[23] https://www.cloudfoundry.org/thefoundry/#services

[24] https://www.openservicebrokerapi.org/

each CF-compatible PaaS), an initial selection needs to be performed to work only with those candidate microservices that possess the characteristics the SE practitioner is looking for – while also accounting for what is not wanted.

Thus, the Filtering component in DIRECTOR is responsible for handling two arguments provided by the user: a list of desirable features and a list of undesirable features (negative filter). It queries the Metrics data store (a relational database loaded with a consolidated view of the microservices' metadata) to find potential candidates, i.e., microservices that have at least one of the desired features (classification *tags*) and none of the undesirable. The daily snapshots taken by the scheduler are saved in a NoSQL database, which in turn is summarized in a relational database to speed up and ease queries. Filtering produces a list of candidate microservices to be ranked in the next step using either a technical (objective) or a social (subjective) perspective. Filtering is not required for the cognitive perspective, which is solely based on the phrasal description by the user.

One should note that the filtering component fosters opportunistic reuse (SEN, 1997) as the practitioner becomes aware of new opportunities for reuse, searches for, and retrieves reusable artifacts. It applies a faceted classification for software reuse (PRIETO-DIAZ, 1991). In a faceted classification scheme, "classes" of software assets are grouped by selecting predefined keywords from faceted lists – in this case, the features or "tags". According to PRIETO-DIAZ (1991), this approach provides higher accuracy and flexibility in classification. Thus, while the Discovery component catalogs all available microservices, the Filtering component supports the search for them.

A subset of the faceted scheme for microservices, constructed based on the available metadata and inspired by SMI, is exemplified by the Table 4.1. There, one can see the "tag" field, which is the main criterion for selecting (and discarding) candidates to be ranked later – a microservice can have one or more classification *tags*, i.e., features, characteristics or functionalities. A list of *tags* is provided when a microservice is discovered, being part of its metadata as collected from its platform.

Table 4.1: Adapting SMI for microservices comparison (sample).

CF attribute's name	Sample values retrieved from CF platform's catalog
Tag (Feature)	analytics, Data Scientist, scheduling, big_data, lite, watson, finance, Security, dev_ops, database, internet_of_things, iPad, Monitoring, Mobile
Commercial Plans (quantity)	1, 2, 3, 4, 5, 7, 10
Cost	Free, Paid

4.4 The Social Perspective

The goal of the social perspective is to rank candidate microservices identified by the Filtering component adopting a reputation criterion based on how the microservice is perceived by the practitioner community. This is a common practice in recommendation systems, since it is based on peer's usage information and opinions (RESNICK & VARIAN, 1997). In that sense, the question answered by DIRECTOR is "*Which of the candidate microservices are other developers using?*" The social perspective does not account for user's priorities, nor is it based on the SMI model. It was inspired by an architect's response on the first survey described in Chapter 3. The architect said he usually asks other architects about which services they are using before settling on a decision.

DIRECTOR makes use of StackOverflow[25], "the largest, most trusted online community for developers", according to their own website. As with CF, by consuming their RESTful APIs, it checks how many questions a certain candidate had in the last year (configurable parameter) and who had posted it. Capturing information about who posted the question allow using information from the community`s user reputation system. DIRECTOR verifies how many questions have been viewed, answered, and how each one has been scored by the community (up/down votes indicating whether it was a good or a silly question). Based on the assumption that the bigger the user base of a technology, the lower the risk of not finding support, the social perspective ranks the candidates accordingly to their community participation – the SECO social dimension discussed in Chapter 2. In short, for each candidate, the five attributes aforementioned (hit count, owner reputation, view hit count, answered hit count, and score) are used as a "proxy"

[25] https://stackoverflow.com/

for the level of community engagement and AHP is used to rank the candidate microservices under evaluation.

Notice that, as multiple platforms are considered, homonyms might exist, i.e., the same technology being published on more than one PaaS – e.g., two providers offering the same database engine. It is reasonable to assume this as non-relevant, since the engagement is measured against the technology behind the microservice, despite the PaaS provider. Actually, there might be cases where the service is offered in the PaaS's marketplace but runs in a separate cloud environment.

Finally, to obtain better results, DIRECTOR applies the following search filters when querying StackOverflow: *cloudfoundry; ibm-bluemix; ibm-cloud; pivotal-cloud-foundry; sap-cloud-platform*, since many technologies have existed before their cloud version, e.g., a popular database technology that only recently started to be offered as a service. This is necessary in order to measure the engagement with the microservice version of the technology instead of the traditional (on-premises) one.

4.5 The Semantic Perspective

The goal of the semantic perspective is to classify all available microservices, not only the ones identified by the Filtering component. The evaluation is performed by applying an AI technique named natural language processing (semantic analysis) on the phrasal description (a "free text") provided by the user to "understand" the meaning of the requirement. The semantic perspective does not take into consideration the user priorities, nor is it based on SMI model, just like the predecessor. The purpose is to be as simple and user-friendly as possible, i.e., easy for an end-user to learn and work with.

The single input briefly describes the user's main intention or need, so DIRECTOR can find the candidate with a description, and functionalities (*tags*) that best matches that input. For that, it consumes an IBM Watson service named Natural Language Classifier[26] (NLC) that implements natural language processing (NLP) and machine learning. NLC returns the best matching microservices for a phrasal description, accounting for synonyms and typographical errors. NLC works in three steps or phases:

[26] https://www.ibm.com/watson/services/natural-language-classifier/

1) On the classify phase (this one can take hours), Watson is taught about classes, through the ingestion of a training data containing, for each class, one or more descriptions. The metadata “description”, “long_description” and “tag” fields are used to generate, at least, three lines for each candidate. For instance, the following information is available for the Redis microservice: “Redis is an open-source, blazingly fast, low maintenance key/value store.” and “281_compose-for-redis”. By the end of this phase, a classifier is trained and ready for being consumed through its RESTful API;
2) On the evaluate phase, Watson understands the intent behind text and returns a corresponding classification (suggestion) with a confidence score;
3) Finally, on the learn phase, the user can give feedback regarding the results, so Watson can improve the statistical confidence of subsequent responses. The more it is used, the more the confidence increases due to the machine-learning algorithm.

This third type of analysis is one of DIRECTOR’s differentiators since no previous work in the SOA/MSA context has considered AI for recommending a microservice for reuse to the best of our knowledge. Watson services are considered examples of Cognitive Computing, which describes technology platforms based on the scientific disciplines of AI and signal processing. These platforms encompass machine learning, reasoning, natural language processing, speech recognition and vision (object recognition), human–computer interaction, dialog and narrative generation, among other technologies (KELLY, 2015).

NLP is an established field of computer science that deals with the interaction between computers and human language (KIM, 2014; PONS et al., 2016). In recent years, the field has undergone considerable change attributable to improved technology, processing power, and increased accessibility of machine-learning (TRIVEDI et al., 2018). In this sense, a powerful tool – IBM’s Watson supercomputer – gained fame as the Jeopardy![27] champion in 2011 and has since branched out into various machine learning tasks, including natural language classification (FERRUCCI et al., 2013).

To develop such an approach, a labeled training dataset was created by classifying each microservice (<id>_<name>) with its characteristics (*tags*) and descriptions, as

[27] https://www.jeopardy.com/

retrieved from the catalog’s metadata. Below, a sample of the .csv file is presented. The full dataset used to train Watson has 1,132 lines in total.

```
"Tone Analyzer uses linguistic analysis to detect three types of tones from
communications: emotion, social, and language.  This insight can then be used to drive
high impact communications.","21_tone_analyzer"
"ibm_created","21_tone_analyzer"
"ibm_dedicated_public","21_tone_analyzer"
"lite","21_tone_analyzer"
"watson","21_tone_analyzer"
```

Thus, deep learning-based natural language classification was conducted using a proprietary natural language classifier from IBM Watson (FERRUCCI et al., 2013; FERRUCCI, 2012). The algorithm uses hypothesis generation, string analysis, and deep learning-based word scoring to generate a prediction for each microservice class. After it is trained, queries can be made using phrases written in plain English.

Finally, Watson also supports machine learning, i.e., for each recommendation a statistical confidence is provided. After each query, the user can provide feedback, which increases the statistical confidence of future responses. The feedback is a binary “yes” or “no” for each suggested microservice stating whether that recommendation is reasonable and acceptable as correct.

This semantic perspective allows a person, despite his/her technical background, to benefit from DIRECTOR. For instance, a business analyst might write a simple sentence in plain English and DIRECTOR will provide a list of candidates that can be later confirmed by an IT architect. In that sense, it works as both a dynamic catalog of microservices offerings in multiple platforms and an assistant capable of suggesting alternatives based on natural language input.

4.6 Implementation

Figure 4.1 shows an overview of the framework and its high-level architecture, which consists of a front-end layer as a Web application (DIRECTOR Web tool) and a back-end layer comprised of RESTful services and two data stores, the AI component and a scheduler mechanism, all running in the cloud at **http://director-services.mybluemix.net**.

It is also possible to interact with DIRECTOR’s own API, receiving JSON documents as responses. For instance, to retrieve all PaaS registered in the framework, one could target *https://director-services.mybluemix.net/api/platform/info*, which will list status (“A”ctive), name, description, query, and authorization endpoints, the dates when different types of metadata were retrieved and stored, the number of microservices found and the total number of days being monitored.

The main purpose of the Web tool is to allow a user to receive recommendations by supplying the input arguments: desired and undesired features (tags), QoS priorities (1 to 5), and the phrasal description (free text). Default values are provided for ease of use, but can be replaced at one’s will. One should bear in mind that the current version of the tool, built for research purposes only, does not yet provide a nice-looking interface for end-users.

DIRECTOR relies on a scheduler mechanism for automatically (daily) retrieving up-to-date information from the registered PaaS (the microservices metadata). It periodically invokes the Discovery component for querying and storing the data related to all available microservices in that platform. In turn, that history of daily snapshots makes it possible to infer the microservice availability, stability and evolution, among other QoS attributes. The metadata encompasses information related not only to the microservices, but to their commercial plans too. It also contains a general description used to feed the AI component – presented in the previous section.

DIRECTOR’s datastores are implemented as independent databases: a NoSQL Cloudant[28] and a relational Aurora MySQL[29]. Cloudant, a managed JSON document database compatible with Apache CouchDB[30], is used for storing the metadata retrieved from the platforms. A NoSQL was chosen for the metadata data store due to its flexibility, which allows it to be quickly adapted to new formats being returned by future versions of the CF’s API – for each PaaS, there is a separate database. Aurora MySQL is periodically updated by the Discovery component to provide a consolidated view, including the available PaaS and microservices, together with the classification scheme (features per microservice) and QoS metrics – an architectural pattern named Command Query Responsibility Segregation (CQRS)[31].

[28] https://console.bluemix.net/catalog/services/cloudant

[29] https://aws.amazon.com/rds/aurora/

[30] http://couchdb.apache.org/

[31] https://cqrs.files.wordpress.com/2010/11/cqrs_documents.pdf

In its current version, DIRECTOR supports CF standard, an open-source cloud platform for the PaaS layer, via a RESTful API (version 2.84.0). The current version of the Discovery component is able to connect to any CF compatible PaaS offering, e.g., IBM Cloud[32] (previously known as Bluemix), Pivotal Cloud Foundry[33] and SAP Cloud Platform[34]. Though Salesforce Heroku[35] was initially considered, since it supports the core concept of *buildpacks* (runtime support), it was later discarded because it runs on a proprietary solution instead of CF. Additional platforms are being considered for this research as future work. Currently, the following CF resources are being consumed:

- GET */v2/info*: to retrieve the basic information required to connect and authenticate with a PaaS;
- GET */v2/services*: to retrieve information about all microservices available on a PaaS (catalog);
- GET */v2/services/:guid/service_plans*: to retrieve information about commercial plans of a particular microservice, using its unique identifier (*guid*).

What follows is an example (response fragment) of Organizations API's operation for listing all services for the organization[36]:

```
{
   "total_results":149,
   "total_pages":3,
   "prev_url":null,
   "next_url":"/v2/organizations/803643ef-a699-41ce-9d62-a6266aa0c03d/services?order-
direction=asc&page=2&results-per-page=50",
   "resources":[
      {
         "metadata":{
            "guid":"9f6d8c63-4b5e-4197-aa3f-d2d7fafc8ffb",
            "url":"/v2/services/9f6d8c63-4b5e-4197-aa3f-d2d7fafc8ffb",
            "created_at":"2013-10-05T16:29:35Z",
            "updated_at":"2016-07-07T23:32:00Z"
         },
         "entity":{
            "label":"mysql",
            "provider":"core",
```

[32] https://www.ibm.com/cloud/

[33] https://pivotal.io/platform

[34] https://cloudplatform.sap.com/index.html

[35] https://developer.salesforce.com/platform/heroku

[36] https://apidocs.cloudfoundry.org/253/organizations/list_all_services_for_the_organization.html

```
            "url":"http://75.126.167.146:54321",
            "description":"MySQL database",
            "long_description":null,
            "version":"5.5",
            "info_url":null,
            "active":true,
            "bindable":true,
            "unique_id":"92a06847-9446-407f-8aaa-a6f20c1d2cdc",
            "extra":"{\"listing\":{\"imageUrl\":null,\"blurb\":null},\"provider\":
{\"name\":\"Core\"}}",
            "tags":[
               "mysql",
               "relational",
               "data_management",
               "ibm_experimental"
            ],
            "requires":[  ],
            "documentation_url":null,
            "service_broker_guid":"49bdc628-46c1-4d7e-a324-0c1aac4f7764",
            "plan_updateable":false,
"service_plans_url":"/v2/services/9f6d8c63-4b5e-4197-aa3f-d2d7fafc8ffb/service_plans"
         }
      }
```

All source-code for DIRECTOR framework, including all the Apache Maven[37] sub-projects (*director-console*, *director-dao*, *director-front-end*, *director-services*, and *director-util*) is publicly available at: **https://github.com/MFranca/DIRECTOR**.

4.7 Proof of Concept

In July 2017, just after the first survey described in the previous chapter, a PoC was conducted, as a preliminary experiment, to evaluate the significance and quality of DIRECTOR's recommendations. Java was chosen as the programming language for the application prototype, since it is popular and could allow for new contributors in the future. The Web interface was constructed based on the JSF specification, together with RESTful webservices using the JAX-RS specification. The scheduler is a microservice named Workload Scheduler[38]. One should bear in mind that, as new microservice are added or decommissioned, the following results may change periodically.

[37] https://maven.apache.org/

[38] https://console.bluemix.net/catalog/services/workload-scheduler

DIRECTOR's Discovery component has been running since June 22th, 2017. As this Thesis was written, it had been more than a year of monitoring, which already presented us with interesting data about IBM Cloud PaaS. For instance, the number of microservices grew from 148 available on June 23th, 2017 to 166 on December 30th of the same year. Then, it grew to 181 in June 27th, 2018, and finally to 163 in April 27th, 2019. This represents an increase of 12% in about six months, followed by another increase of 10% in the following six months, and a decrease of 10% in the last ten months. In software reuse, it seems natural to start with a higher number of assets and, as only some are effectively reused, to see such numbers shrink later.

Besides IBM Cloud, other two PaaS are being monitored: Pivotal Cloud Foundry and SAP Cloud Platform, having, at this moment, 26 and 8 microservices, respectively. Though this may seem like a small number, one should bear in mind that newly created SECOs might take some time before engaging new members and forming a thriving community. As more platforms are connected to DIRECTOR, the number of microservices may increase rapidly.

For this PoC, an IT architect was requested to provide the input values, which are listed on Table 4.3. The fictional scenario could be described as a development team building a new cloud application that should be hosted on a PaaS and need to consume a microservice for unstructured data persistence. The first six parameters are weights for the QoS attributes used in technical perspective – where "1" means lowest importance and "5" means highest. The desired features are the microservice functionalities that will be used to filter out the candidates – a list of available features was supplied to the practitioner[39]. Lastly, the brief description was used to query the AI component.

The Filtering component selected four candidates to be used by the technical and social perspectives, using the desired features informed by the practitioner (the negative filter was not used this time):

1. ***dashDB For Transactions' tags***: big_data, db2, sqldb, purescale, sql, db2 on cloud, db2oncloud, dash, dashdb, oracle, database, transactions, flex, **dbaas**;
2. ***db2oncloud's tags***: data_management, db2oncloud, db2cloud, db2hosted, db2, relational, database, db, hosted, oracle, **dbaas**, purescale, db2 on cloud;

[39] The updated version can be checked at http://director-services.mybluemix.net/listFeatures

3. ***redis' tags***: key-value, **nosql**, web_and_app;
4. ***mongo-db's tags***: data_management, document, mongodb, **nosql**.

Table 4.2: Input arguments for a unstructured data persistence microservice.

User input parameter	Informed value
QoS Agility Priority (1-5)	1
QoS Assurance Priority (1-5)	5
QoS Financial Priority (1-5)	3
QoS Performance Priority (1-5)	4
QoS Security & Privacy Priority (1-5)	5
QoS Usability Priority (1-5)	2
Desired Features (filter)	dbaas, nosql
Brief Description (free text)	I need to store json objects in a nosql database.

Then, the objective analysis takes place and calculates the values for each QoS attribute, as can be seen on Table 4.4. Since only four candidates were competing against each other, the best classified scored four points on that attribute, the second scored three, and so on. After that, weights were applied to each category, and then summed to provide the global rank. As can be seen, "db2oncloud" was classified as the most adequate one from a technical perspective, given the user's priorities.

For instance, take the Agility category, consisting of four attributes. First, the four candidates are compared regarding "Scalability", where a draw (number of inactive days equal to zero) resulted in all candidates initially scoring four points. Then, on "Portability", another draw happened (none of the microservices required anything to run), resulting in all of them scoring additional four points (at this point, all candidates scored eight points in total). However, on "Flexibility", "db2oncloud" had the higher number of service plans, making it score four points (alone), while "dashDB for Transactions" scored three points, and both "redis" and "mongodb" scored only two points – at this point, "db2oncloud" scored 12 points in total. Finally, on "Adaptability", "db2oncloud" was again the best (lower number of days since the last update or release), scoring alone four points again (16 in total, since it scored four points in each of the four attributes, i.e., it was the best in each one). Note that Adaptability values are negative because we maximize all attributes and, the smaller the better for this attribute. Once all attributes are calculated and the candidates are scored (summing up individual scores), it is possible to calculate the local rank (and apply weights) of that specific category – where “db2oncloud” was the best one, scoring four points. A similar procedure was applied for the other categories and their attributes.

Table 4.3: AHP calculation for the technical perspective.

		Candidate Microservice			
QoS Category	**QoS Attribute**	***dashDB***	***db2***	***redis***	***mongodb***
Agility	Scalability	0	0	0	0
	Portability	0	0	0	0
	Flexibility	4	9	1	1
	Adaptability	-9	-7	-51	-51
	TOTAL	**14**	**16**	**12**	**12**
	(*local rank plus 1*)	**3**	**4**	**2**	**2**
Assurance	Availability	0	0	0	0
	Stability	268	780	51	51
	TOTAL	**7**	**8**	**6**	**6**
	(*local rank plus 5*)	**15**	**20**	**10**	**10**
Financial	On-going cost	0	0	1	1
	TOTAL	**3**	**3**	**4**	**4**
	(*local rank plus 3*)	**9**	**9**	**12**	**12**
Performance	Functionality	16	16	5	6
	TOTAL	**4**	**4**	**2**	**3**
	(*local rank plus 4*)	**16**	**16**	**8**	**12**
Security & Privacy	Access control & privacy mgmt.	1	1	1	1
	TOTAL	**4**	**4**	**4**	**4**
	(*local rank plus 5*)	**20**	**20**	**20**	**20**
Usability	Accessibility	1	1	1	1
	Learnability	0	0	0	0
	TOTAL	**8**	**8**	**8**	**8**
	(*local rank plus 2*)	**8**	**8**	**8**	**8**
TOTAL	**(*global rank*)**	**71**	**77**	**60**	**64**

Next, DIRECTOR shifts the perspective used for calculating the ranking for each candidate, taking into consideration the social perspective. For this perspective, “mongodb” was the best classified with 22 questions or hits (10 answered), followed by “redis” with six hits (three answered), while “dashDB For Transactions” and “db2oncloud” scored zero hits. It is worth to point out that “dashDB”, the abbreviated form of one candidate’s label, scored 18 hits (10 answered). This could be further investigated in order to fine tune the query for getting better results, though accounting for not mixing with traditional versions of a technology, as mentioned earlier.

Finally, the semantic perspective takes place and queries the NLC *classifier* using the provided phrase. Again, in this type of analysis, the component considers all available candidates in the catalog so that Watson could recommend a microservice that might have been forgot or unknown to the practitioner. From all available candidates, the NLC scored "redis" with the higher degree of confidence (27%), followed by "dashDB For Transactions" (15%), "mongodb" (9%), "IBM Graph" (8%) and "db2oncloud" with just 6%. The reason behind the suggestion of "redis" could be explained by taking a closer look at its description: *"an in-memory database implementing a distributed key-value store with optional durability"*. Probably, when the user mentioned *"to store json objects"*, Watson understood *"in memory objects"*, thus, suggesting an in-memory database.

It is interesting to note that all the four candidates selected by the Filtering mechanism were also included in the result provided by Watson, as well as additional candidates. On the other hand, the relatively low statistical confidence values can be explained by the fact that the *classifier* had been just recently created, and was not used much before the PoC was conducted. Additionally, Watson always concludes a query by asking the user whether those alternatives were "ok", allowing the machine-learning algorithm to improve future queries while increasing the confidence.

These results, as can be seen on Table 4.5, were presented to the participant and his feedback was requested. He confirmed that all suggestions made sense, although the semantic perspective was discarded because he was looking for a disk storage system. According to him, further analysis would still be required, though only on the suggestions provided by DIRECTOR (instead of the initial two hundred options). He was not aware of all the recommended alternatives, and would have to spend much time searching for them. In general, he was pleased with the results.

For each of the perspectives, a recommendation, followed by a justification, was provided. The justification is a short description of the rationale behind that choice. The intention is to help IT architects to avoid doing a judgmental decision when preparing an Architectural Decisions artifact (FOWLER, 2003) – design decisions that address architecturally significant requirements, perceived as hard to make and/or costly to change. Albeit only four candidates (microservice alternatives) were considered in this

PoC (from the technical and social perspectives), that number might grow exponentially depending on the number of selected features, as well as the number of registered PaaS.

Table 4.4: PoC's output results.

Perspective	Microservice	Justification
Technical	DB2	This candidate was best classified in the following categories: Agility, Assurance, Performance, Security and Usability.
Social	Mongodb	This candidate has the strongest community engagement, being used on a CF and/or Bluemix context.
Semantic	Redis	Watson suggested this candidate, with a confidence of 27%, due to the following functionalities: key-value, nosql, web_and_app

4.8 Final Remarks

This chapter described the DIRECTOR framework, comprised of a discovery, filtering, ranking and selection mechanisms, compatible with the CF standard. The PoC detailed in the previous section was successful in demonstrating that the DIRECTOR approach is feasible in the sense that it is possible to determine a set of QoS information by analyzing microservice metadata and it to measure the community engagement to find out whether other practitioners are adopting or working with a certain technology. Finally, it was also possible to take advantage of Watson so that DIRECTOR can be used by a business analyst or an end user, i.e. a person not so technically versed. The next chapter will present a more robust evaluation of the framework, engaging a larger number of participants and observing the impact of each perspective.

This page was left blank intentionally.

Chapter 5 – Evaluating DIRECTOR

This chapter describes DIRECTOR framework's evaluation. A feasibility study was conducted with 27 subject matter experts, some of them software and solution architects. PaaS' metadata was collected from three distinct cloud providers.

5.1 Introduction

To answer the second and last research question presented in Section 1.4 (***RQ2***. *Can DIRECTOR complementary perspectives help in the architectural decision of choosing a microservice for a PaaS-based solution?*), DIRECTOR was applied and analyzed in industrial scenarios. The goal was to evaluate DIRECTOR framework's correctness and usefulness, using its perspectives in a real context. To contribute to the SECO research and practice, a *feasibility study* was conducted with SE practitioners, in a real-life scenario, to evaluate DIRECTOR and, more specifically, its three complementary perspectives, as presented in the previous chapters. These perspectives were implemented at DIRECTOR-Services – the set of *webservices* and web application available at **http://director-services.mybluemix.net/**.

Feasibility studies are used to determine whether a program, service, policy, or product is appropriate for further testing. In other words, they enable researchers to assess whether or not the ideas and findings can be shaped to be relevant and sustainable (BOWEN et al., 2009). According to BOHEM (1981), it is an alternative for acquiring information to support SE decisions. Feasibility studies try to characterize a technology in order to ensure that it actually does what it claims to do and that it is worth of extra effort to develop it. SHULL et al. (2001) state that such reviews cause the greatest changes in emerging technologies. Therefore, they have to be applied in the beginning of the evaluation process. That is why feasibility studies are often conducted to evaluate a new technology or approach. Thus, a feasibility study seemed the most suitable choice for evaluating the **DIRECTOR** framework.

This chapter presents details on the study that was planned and executed with 27 SMEs in software architecture (two participants for the pilot and 25 for the main part of the study) to answer the aforementioned research question. The next section defines the study and presents its planning. Then, details about the execution are provided and the results analyzed in Sections 5.3 and 5.4, respectively, while the threats to its validity are listed in Section 5.5. The chapter is concluded in Section 5.6 with final remarks.

5.2 Study Planning

Inspired by other studies designed and performed at the Software Reuse Lab (VASCONCELOS, 2007; MAGDALENO, 2013; NUNES, 2014; ALBERT, 2014; SANTOS, 2016) and other SE research groups at COPPE/UFRJ (TRAVASSOS et al., 2002; MAFRA & TRAVASSOS, 2006; BARRETO, 2011; SANTO, 2012), the approach defined by SHULL et al. (2001), which includes a first study to determine the feasibility of using a solution in practice, was followed.

The main purpose of this study is to evaluate the DIRECTOR framework's complementary perspectives regarding the architectural decision of selecting cloud microservices for reuse from a software and/or solution architect point of view. Secondary goals involve evaluation of the infrastructure's (tool) correctness and utility. The goal of the study is defined according to the Goal-Question-Metric (**GQM**) paradigm (BASILI et al., 1999), as described in Figure 5.1 and Table 5.1.

Table 5.1: The goal of the proposed study in GQM format

Analyze	*the DIRECTOR framework*
for the purpose of	*characterizing*
with respect to	*the impact of recommendation systems in the architectural decision activities*
under the point of view of	*software and solution architects*
in the context of	*software reuse of microservices in a cloud-based SECO*

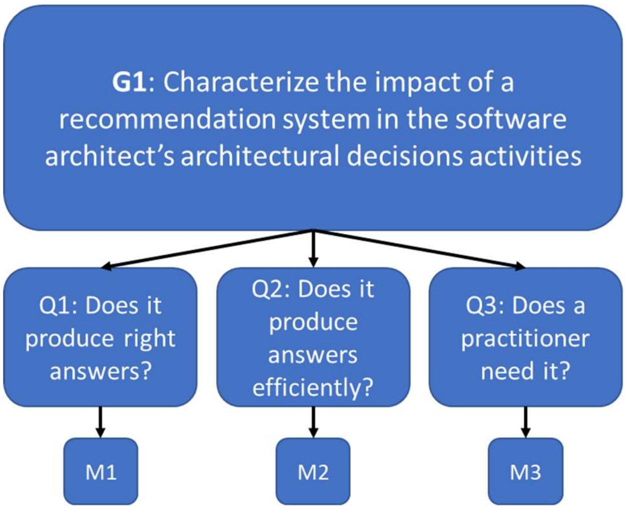

G1: Characterize the impact of a recommendation system in the software architect's architectural decisions activities
Q1: Does it produce right answers?
Q2: Does it produce answers efficiently?
Q3: Does a practitioner need it?
M1
M2
M3

Figure 5.1: GQM model for DIRECTOR evaluation (perspective modules).

5.2.1 Questions and Metrics

This section presents the questions and metrics defined for this feasibility study. Some of them rely on an *oracle* (correct answers) that will be created from the real answers provided by the identified group of senior professionals, as will be described in Section 5.4.3. The questions investigated in this study are as follows:

Q1: Are the participants able to realize the impact of recommendation systems in the architectural decision activities for choosing a cloud microservices in a SECO regarding effectiveness, i.e., does the framework provide right answers?

This perception is measured by the senior participants' answers in comparison with the ones suggested by the framework, i.e., whether the produced recommendations are considered valid or invalid alternatives, e.g., the framework was able to recommend the same microservice that was suggested by a senior SE practitioner to a particular scenario. Therefore, the following metric is defined as follows:

M1: Effectiveness: The effectiveness (correctness) measures the relation between the results obtained with the framework and the objectives with regards to the answers provided by the experienced practitioners from the industry. The calculation is done by the following formula:

$$Effectiveness = \frac{number\ of\ valid\ recommendations}{total\ number\ of\ recommendations}$$

To be considered “valid”, a recommendation must be included in the list of microservices (up to three) provided by the senior participants – assuming there will be convergence in their responses. In this study, the framework will provide up to nine distinct recommendations: three from the technical perspective, three from the social perspective, and three from the semantic perspective – which could result in a minimum of three and a maximum of nine different candidate microservices.

Q2: Are the participants able to realize the impact of recommendation systems in the architectural decision activities for choosing a cloud microservices in a SECO regarding efficiency, i.e., does the framework provide right answers faster than the manual process?

This perception is measured by the average time required by the participants to execute the tasks related to the proposed scenarios, compared to the time required to execute the same tasks using the framework. The following metric is defined:

> **M2: Efficiency:** The efficiency measures the relation between the results and the objectives. In this case, the senior SE practitioners’ answers are our *oracle*, so all of their responses are considered valid recommendations. Then, we compare the average time (in minutes) of seniors and juniors and the researcher using the framework tool. The calculation is done by the following formula:

$$Efficiency = \frac{number\ of\ valid\ recommendations}{time\ taken\ to\ participate}$$

Q3: Are the participants able to realize the impact of recommendation systems in the architectural decision activities for choosing cloud microservices in a SECO regarding utility, i.e., does a practitioner need it?

This perception is measured by the less experienced participants’ answers in comparison with the ones given by the senior participants, i.e., whether the scenarios are so simple that anyone could address them, e.g., a less experienced practitioner without using the framework was unable to provide the same recommendation suggested by a senior practitioner to a particular scenario. The following metric is defined as follows:

M3: Utility: The utility measures the relation between the results without the framework and the objectives, taking into consideration the answers provided by the less experienced practitioners from industry. For instance, if all responses from the less experienced were wrong, i.e., not included in the list provided by the seniors, then we would have a utility rate of 100%. The calculation is done by the following formula:

$$Utility = \frac{number\ of\ invalid\ recommendations}{total\ number\ of\ recommendations}$$

5.2.2 Hypothesis

Empirical research is based on observed and measured phenomena and derives knowledge from actual experience rather than from theory or belief. As such, an empirical study is generally based on one or more hypotheses, instead of a theory. The main hypothesis is known as null hypothesis and states that there is no significant relation between the cause and the effect. The main objective of the study is to reject the null hypothesis in favor of one or some alternative hypotheses. The decision on the rejection of a null hypothesis can be taken based on the results of its evaluation using a dataset analysis (TRAVASSOS et al., 2002). In our study, the following hypotheses were defined:

- **Null Hypothesis (H01)**: There is no difference in **effectiveness** between practitioners choosing cloud microservices in a SECO with or without the framework for cloud microservice selection.

  Alternative Hypothesis (HA1): Practitioners choosing cloud microservices in a SECO with the framework for cloud microservice selection were more effective in their tasks than those without the framework were.

  H01: $\text{Effectiveness}_0 = \text{Effectiveness}_1$, where:

  $Effectiveness_0$ = Effectiveness without the framework

  $Effectiveness_1$ = Effectiveness with the framework

  HA1: $\text{Effectiveness}_1 > \text{Effectiveness}_0$

- **Null Hypothesis (H02)**: There is no difference in **efficiency** between practitioners choosing cloud microservices in a SECO with or without the framework for cloud microservice selection.

- **Alternative Hypothesis (HA2)**: Practitioners choosing cloud microservices in a SECO with the framework for cloud microservice selection were more efficient than those without the framework were.

H02: $\text{Efficiency}_0 = \text{Efficiency}_1$, where:

Efficiency_0 = Efficiency without the framework

Efficiency_1 = Efficiency with the framework

HA2: $\text{Efficiency}_1 > \text{Efficiency}_0$

5.2.3 Participants

Participants were selected by convenience from the researcher's professional network of contacts. They are practitioners, e.g., software and solution architects, developers and trainees, responsible at some degree for design and/or architectural decisions in their organizations. Their organizations range from private and public companies in Brazil and in other countries. Some of these companies are IT consulting firms, while others are startups. The participants work in IT departments, most of them technically leading development teams. They are responsible for solution analysis and software acquisition.

Since an experimental context similar to the software industry is desirable, participants must have an *architectural thinking*[40] mindset, even with different levels of experience in software architecture. Based on information provided through the characterization form, participants were divided into two groups, comprised of the most and less experienced professionals, respectively. Participants' sessions were individually performed. There was no kind of compensation or reward for the participants.

5.2.4 Tasks

A set of eight tasks was designed to explore if practitioners were able to realize the impacts of recommendation systems in the architectural decision activities for choosing a cloud microservices in a SECO. These tasks are classified into three categories according to complexity in execution, based on the work of (OLIVEIRA, 2011):

[40] https://architecturalthinking.net/2015/03/17/what-is-architectural-thinking/

- **Filtering tasks**: this category comprises simple tasks that depend on reading information using the framework's infrastructure or the provided support spreadsheet (also in Annex 1) to answer some questions. If a participant is unable to execute such tasks, he/she should be removed from the analysis because this situation can affect the understanding of the framework, compromising the execution of the remaining tasks. The specific tasks executed in the context of this category in our study were:

  1. What are the currently active registered PaaS?
  2. How many microservices are available in the SAP Cloud Platform PaaS?

- **Basic tasks**: this category comprises the main tasks of this study, which depend on reading information (scenarios are detailed in Annex 1) using the framework's infrastructure or the provided spreadsheet and interpreting the results to answer some questions. The tasks executed in the context of this category in our study were:

  3. What (up to three) microservices would be adequate for scenario #1?
  4. What (up to three) microservices would be adequate for scenario #2?
  5. What (up to three) microservices would be adequate for scenario #3?
  6. What (up to three) microservices would be adequate for scenario #4?
  7. What (up to three) microservices would be adequate for scenario #5?

- **Assimilation tasks**: this category comprises difficult, complex tasks that depend on the participant's background to understand and interpret information related to software and solutions architecture to answer some questions. The specific task proposed in the context of this category in the study was:

  8. With regards to scenario #5, what an architectural decision document would look like (describe it in one to three paragraphs, i.e., the considered alternatives, the chosen one, and the tradeoffs plus the rationale for that)?

Participants should not be informed about such categories to avoid any influence in the proposed tasks. The second group, then, was submitted to the same tasks and their answers were compared with the *oracle* to check the correctness.

5.2.5 Data Collection

This study focuses on analyzing information to observe the impact of recommendation systems in the architectural decision activities for software acquisition of cloud microservices in a SECO. We collected real data from three PaaS[41] (IBM Cloud, Pivotal Cloud Foundry and SAP Cloud Platform) regarding: (1) list of available microservices; (2) list of microservices commercial plans; and (3) microservice metadata, including the features (*tags*) list. Additionally, a set of five scenarios (detailed in Annex 1) were written based on real-life situations, where one needs to reuse or acquire a software asset to compose a solution – descriptions included a main requirement for each scenario, feature list and restrictions. Data were structured in a spreadsheet, consolidating the metadata gathered via PaaS RESTful API services and extracted from the cloud providers' public documents available on the Internet[42]. The researcher, who is also a software architect, was available for questions throughout the execution time, for all participants.

5.2.6 Variables

There are two types of variables: independent and dependent. Independent variables refer to the inputs of the experimental process. Such variables have the cause that affects the result of the experimental process. Their objective is to identify the forces that influence (or can influence) the results of the execution. In turn, dependent variables refer to the outputs of the experimental process (TRAVASSOS et al., 2002). They correspond to those that we are interested in evaluating in the study execution. Such variables are defined according to goals and questions established for the study.

The *independent* variable in this study is the framework used to support choosing cloud microservices in a SECO. This variable has two treatments: (a) the use of DIRECTOR framework for choosing cloud microservices in a SECO; and (b) the use of traditional tools and (a manual) process, besides other resources available on the Internet, such as the provider website, technology blogs and community forums.

[41] http://director-services.mybluemix.net/listPlatforms

[42] https://console.bluemix.net/catalog/, https://pivotal.io/platform/services-marketplace and https://cloudplatform.sap.com/support/service-description.html

The *dependent* variables in this study are: (i) the number of correct answers for the researcher using the framework; (ii) the number of incorrect answers for each participant from Group G2; and (iii) the time spent to execute the eight proposed tasks – the time spent by each participant, by recording the start and end time.

5.2.7 Instruments and Preparation

This section defines the instruments applied during the evaluation and how the evaluation procedure was prepared. We prepared six instruments presented in Annex 1 (applied in Portuguese and English).

1. Informed Consent Form (Section A1.1): informs the study objective and participant's rights and responsibilities. It also informs that collected data should not be used to evaluate participants' performances, and explains confidentiality terms. This form was sent to participants before the study execution. Each participant returned this document;
2. Characterization Form (Section A1.2): allows the researcher to analyze participants' profiles and classify them into groups. This information is also used for the analysis of results;
3. Execution Form (Section A1.3): presents the context of the work and the eight proposed tasks, together with the five real-life scenarios. The participants were asked to play as they currently do in daily IT architectural/development team' activities within their current organizations. This document is also used to collect answers for each task;
4. Microservices Catalog (Section A1.4): a Microsoft Excel workbook with a spreadsheet for each PaaS, containing the list of available microservices on that platform (software catalog), together with the metadata information (name, description, commercial plans and features);
5. Self-Evaluation Form (Section A1.5): consists of a questionnaire in which each participant should evaluate his/her experience after the study execution. Qualitative information on the study execution was collected, as well as suggestions of improvement for the framework and considerations regarding the experience in the study;

6. Background (Section A1.6): before starting the study, participants were submitted to a short training on the SECO dimensions and key concepts, Microservices Architecture (MSA) and an overview of Cloud Computing technologies. They could use this document whenever they needed it.

5.2.8 Planning Validity

As suggested by MAFRA & TRAVASSOS (2006), planning and instruments should be validated with other researchers before executing the study. Such researchers should not be interested in the study's results in order to reduce bias. In our study, the planning was validated by another researcher from our university who has previous experience in studies such as this one. In addition, a pilot study was conducted with two participants simulating both groups, as detailed in Subsection 5.2.10. Such pilot aimed to identify possible difficulties in executing the study, including the understanding of related concepts or even the infrastructure. This pilot also helped to fine-tune the instruments and to have an idea of how long the study execution would take.

5.2.9 Interpretation and Analysis

The results were analyzed in a quantitative way. Quantitative analysis refers to effectiveness, efficiency and utility. The results were analyzed based on participants' answers and the duration of activity. The "worst" scenarios were investigated, i.e., from the initial list of five, which ones caused the framework to have the lowest performance, e.g., the greater number of misses regarding senior answers, or the greater number of correct answers by junior professionals. This could mean the lack of maturity of the framework affected its recommendations and/or the proposed tasks were not complex enough, respectively.

5.2.10 Pilot Study

A pilot study was conducted in early February, 2019 with two participants through individual sessions. The first participant informed a reasonable amount of experience in IT/Software sector and software architecture (12 and 6 years, respectively), whereas the second informed medium to low levels of experience in the same areas (6 and 3, respectively). Though both have experience in SE, the first works in a startup company, while the second is still an undergraduate and a junior software developer in an IT consulting firm. Both do not have experience with recommendation systems neither are experts in software acquisition in SECO contexts, though the first has experience creating and documenting architectural decisions.

After signing the informed consent form and filling the characterization form, they were given the modern SECO background document, which explained the main concepts of this research: (i) modern SECOs; (ii) microservice architecture; and (iii) COTS selection. The researcher responsible for the study also explained those concepts, besides asking for questions or doubts, making sure the participants understood them before proceeding to the execution of the study plan. Both executed the proposed tasks without the framework and, afterwards, were introduced to the framework Web tool and asked to fill in the evaluation form – remaining five questions. Both performed all the tasks and none pointed them as difficult.

The main problem reported was the duration of the study (more than 70 minutes each), the lack of a centralized, automatic infrastructure to help answering the tasks. They reported that most of the information required depends on consulting the provided software catalog in the spreadsheet or going to the Web site of the PaaS provider (SECO keystone). They also struggled with the *search* functionality of Microsoft Excel. Once they mastered it ([Ctrl] + [L] and “Locate All”), it became easier (faster) to execute the subsequent tasks. They also complained about the format of the catalog (JSON), which is not so “human-readable”, and the amount of available microservice they had to choose from.

The pilot studies were important to refine the study’s instruments, such as the supporting spreadsheet. A shorter version of the explanation of the main concepts was envisioned, since the introduction was long (10 to 15 minutes). The last (assimilation) task regarding the construction of an ADR was excluded due to the effort required and time constraints. Answering to this question was not pivotal to the framework evaluation and the participants from the pilot complained about the time required to complete all eight tasks in the second part of the study.

5.3 Study Execution

After some adjustments from the feedback of the two pilot participants, the study was conducted with 25 additional participants, all of them SE practitioners working in several organizations, most of them in the IT market in late February, 2019. Their roles range from interns to IT managers in medium organizations, as previously described.

First, the participants signed an informed consent form and answered the characterization form. This allowed us to distribute them into groups, as previously described. Then, they received a brief explanation of about 5 to 10 minutes on the concepts involved in this research: modern SECOs, microservice architecture, and COTS selection.

Next, both groups received the supporting spreadsheet containing the catalog of available SECO platforms and microservices and the description of each of the five proposed scenarios. An evaluation session with each participant was individually conducted. They used the spreadsheet and around 80% of the participants used an Internet browser to query Google[43] and the three PaaS providers website and get more information about some of the microservices available.

5.4 Analysis of the Results

In this section, the data collected in the study is analyzed. As such, participants' profile and dataset analysis are discussed.

5.4.1 Participants' Profile

Table 5.2 shows the participants' academic degree and experience level (in years). Regarding the academic education, one participant reported having a PhD degree, seven had a Specialization degree (two are currently Master students), 12 have a Bachelor degree, and five are undergraduates – 25 participants in total. The participants have also informed years of experience in SE and Software Architecture. Most of them (80%) informed 5 years or more experience in SE and IT and two-fifths (40%) with Software Architecture. This profile is interesting since it allows us to observe whether the proposed framework could influence architectural decision activities for choosing cloud microservices in a SECO, since the average experience of the participants with SE was 10 years, while with Software Architecture was four years. Regarding experience with

[43] https://www.google.com/

similar tools, where “No” is “I have no familiarity” and “Yes” means “I have some familiarity”, only five participants reported being familiar to recommendation systems.

Table 5.2: Participants’ academic and professional background.

Subject ID	***Academic degree***	***Experience with SE (years)***	***Experience with SA (years)***	***Know similar tools***
P1	PhD	18	8	No
P2	Specialization	24	12	No
P3	Undergraduate	1	0	No
P4	Undergraduate	3	0.5	No
P5	Undergraduate	5	0	No
P6	Specialization	10	5	No
P7	Specialization	18	10	No
P8	Undergraduate	4	0.5	No
P9	Bachelor	10	6	Yes
P10	Bachelor	19	5	No
P11	Bachelor	4	2	Yes
P12	Specialization	20	5	No
P13	Bachelor	5	1	No
P14	Bachelor	6	2	No
P15	Bachelor	10	3	No
P16	Specialization	12	0	No
P17	Bachelor	7	1	No
P18	Bachelor	10	1	No
P19	Undergraduate	12	10	No
P20	Bachelor	12	12	Yes
P21	Specialization	12	6	No
P22	Bachelor	5	3	No
P23	Bachelor	4	1	Yes
P24	Bachelor	8	0	No
P25	Specialization	13	4	No
Max.	*N/A*	*24*	*12*	*N/A*
Min.	*N/A*	*1*	*0*	*N/A*
Avg.	*N/A*	*10.08*	*3.92*	*N/A*

5.4.2 Participant Groups

Participants were characterized before the execution to allow distributing them into two heterogeneous groups. None of the groups used the framework to execute the proposed tasks. Later, the researcher used the framework to perform the same tasks and compared the results with both groups. Participants were ranked and grouped based on their professional experience registered in the characterization form as in Table 5.3.

Table 5.3: Criteria applied for ranking participants.

Question	Observation unit
Experience with Information Technology and/or Software Engineering	Years.
Experience with Software Architecture	Years.

The correlation between Software Engineering experience and experience on software architecture among the participants was calculated as 0.73, indicating high correlation and meaning that participants experienced in Software Engineering are, in general, also experienced in architecture. Thus, we can consider a single experience aspect to separate participants into groups. Software architecture was elected due to its proximity to the research area. Figure 5.2 also confirms the correlation, showing that most participants have a great deal of architecture experience when they also have it in Software Engineering and vice-versa, as exemplified by participants P2 and P7.

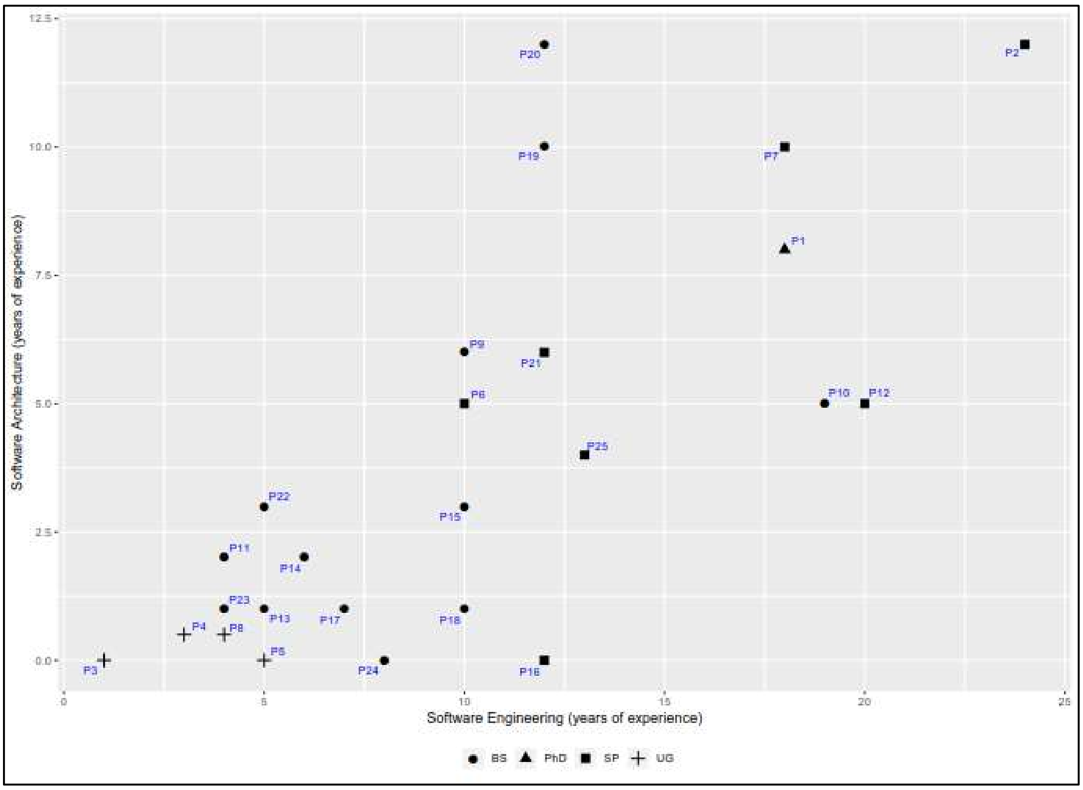


Figure 5.2: Participant's experience in Software Engineering and architecture.

Next, a dendrogram was used to identify subgroups of participants. A dendrogram is a diagram representing a tree, frequently used to represent hierarchical clustering, i.e., the arrangement of the clusters produced by the corresponding analyses (EVERITT, 1998). The dendrogram in Figure 5.3 clearly shows the existence of two groups: the first one comprises the participants in the upper right corner of the scatter plot from the previous figure, representing the most experienced; and the second one will be considered as the group of less experienced practitioners.

Based on the aforementioned results, five senior participants (G1) were selected (P1, P2, P7, P19, P20) from the initial set of 25 participants. The remaining participants (G2) were classified as junior participants. These groups are shown in Table 5.4. This strategy helped us to check whether DIRECTOR's recommendation “confidence” is closer to a senior practitioner or to a junior one. Thus, it is possible to confront the recommendations from DIRECTOR complementary perspectives with those provided by the senior participants, at the same time that its utility is checked on whether junior professionals provide valid recommendations according to the oracle.

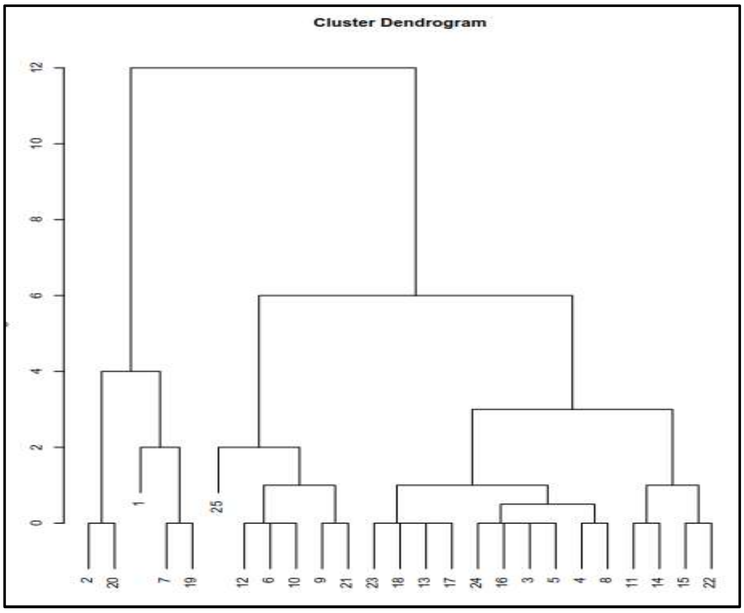


Figure 5.3: Distribution of the study’s participants.

Both groups were submitted to the same eight tasks in the first part of the study execution. In the proposed experimental design, the treatment has not varied (only the researcher used the framework to perform the tasks). The participants used their daily tool support, e.g., their preferred Web browser, the provided spreadsheet and word processor, to perform the architectural decision activities for software acquisition of cloud microservices in a SECO. Participants were instructed to use any means they thought to be adequate, with the exception of another recommendation tool, since the goal was to compare DIRECTOR with their expertise. Additionally, information about the time each participant spent to execute the aforementioned tasks was gathered, along with the rationale for each choice and the confidence level of the responses.

Table 5.4: Experimental design – groups and treatments.

Group	**Execution**	**Participants**
G1	Most experienced professionals	P1, P2, P7, P19, P20
G2	Less experienced professionals	P3, P4, P5, P6, P8, P9, P10, P11, P12, P13, P14, P15, P16, P17, P18, P21, P22, P23, P24, P25

5.4.3 Convergence in the Senior Responses

After identifying the senior practitioners and forming G1, an *oracle* for the proposed scenarios needs to be defined. For that, the responses from G1 were analyzed to confirm whether they converge, *i.e.*, to verify whether there is a consensus among the senior participants. The recommendations from the most experienced professionals, identified by the number of the respective microservice, are listed in Table 5.5.

Table 5.5: Recommendations from the most experienced professionals.

Scenario	***P1***	***P2***	***P7***	***P19***	***P20***
S1	**1511**	**1511**, 2081, 2111	**1511**, 2081	2111, 2155, 1851	91
S2	1061, 1771, 2153	2011, 1781, 931	1061, 1221, 2153	1, 1781	1221
S3	481, **571**	1081	**571**	**571**	**571**
S4	**1961**, 301, 661	**1961**	**1961**	661, **1961**, 2159	2159
S5	**1251**, **1051**, 1771	**1051**, **1251**, 791	1601, **1251**, **1051**	**1251**, **1051**	**1051**

Table 5.6 shows the microservices participating in three to five answers of the senior participants. While we observe some convergence on scenarios S1, S3, S4, and S5, there is none with respect to scenario S2. As the oracle is composed of five professionals,

convergence could be considered when three or more agreed on a recommendation (more than half of the group).

One can see a high convergence on scenarios S3 and S5, where the most voted microservices represent 66% and 75% of all responses, respectively. Scenarios S1 and S4 show medium convergence, as the most voted microservices represent 30% and 44% of all responses, respectively.

Table 5.6: Convergence in the senior responses.

Scenario	***5 answers***	***4 answers***	***3 answers***
S1	-	-	1511
S2	-	-	-
S3	-	571	-
S4	-	1961	-
S5	1051	1251	-

Regarding scenario S2, there is no majority consensus. Therefore, this scenario will be discarded from future analysis. Scenario S2 was summarized as "*I need to store relational data in a cloud database as a service*". It could be argued that a relational database is like a commodity service, a widely available technology that is not markedly dissimilar from one offer to another. As relational databases are available since the 1970s, they can be considered a mature technology – as technologies and markets for a given product mature, it is more likely to be considered a commodity. With plenty of feasible options, it is understandable that each professional decided for the one he or she is more used to, leading to a lack of convergence.

5.4.4 Results

After classifying the participants, results are analyzed based on their answers, duration of activity, and participants' feedback, which includes the confidence level of the answer, provided in the evaluation form, as explained in Section 5.2.9. Thus, an analysis could be done to understand the study from the data collected and evaluated according to statistical resources.

The dataset is presented in Table 5.7, while measures of central tendency and dispersion are presented in Table 5.8. The confidence level is measured as the median of the confidence informed by the participant on the answer given for the four scenarios under analysis. Moreover, boxplots were generated to allow the visual analysis of the data distribution in each group, G3 meaning DIRECTOR (Figures 5.4 and 5.5 for *effectiveness* and *efficiency*, respectively).

As can be seen in Table 5.7, participants of G1 were also ranked according to the final answers from the *oracle*. P7 was the most accurate senior, having five correct recommendations, from a total of seven – a 71% of effectiveness. From the 25 participants in this study, P7 was the fourth most experienced in Software Engineering and the third in software architecture. Conversely, P19 had only 44% of effectiveness, with only four valid answers from nine recommendations.

Table 5.7: The study's dataset.

Participant	*Group*	*Effectiveness*	*Efficiency*	*Utility*	*Confidence*
P1	G1	0,56	0,13	N/A	3,50
P2	G1	0,50	0,04	N/A	4,50
P7	G1	0,71	0,12	N/A	4,00
P19	G1	0,44	0,11	N/A	4,50
P20	G1	0,50	0,04	N/A	4,50
P3	G2	0,40	0,05	0,60	3,00
P4	G2	0,75	0,08	0,25	3,50
P5	G2	0,50	0,02	0,50	2,50
P6	G2	0,43	0,07	0,57	3,50
P8	G2	0,25	0,04	0,75	3,50
P9	G2	0,00	0,00	1,00	4,50
P10	G2	0,50	0,10	0,50	4,00
P11	G2	0,33	0,18	0,67	3,00
P12	G2	0,50	0,04	0,50	5,00
P13	G2	0,43	0,04	0,57	2,50
P14	G2	0,25	0,02	0,75	2,00
P15	G2	0,50	0,04	0,50	3,00
P16	G2	0,40	0,04	0,60	2,00
P17	G2	0,60	0,18	0,40	3,00
P18	G2	0,50	0,07	0,50	3,00
P21	G2	0,75	0,12	0,25	3,00
P22	G2	0,20	0,07	0,80	2,00
P23	G2	0,20	0,06	0,80	1,00
P24	G2	0,25	0,02	0,75	1,00
P25	G2	0,25	0,04	0,75	3,50

Table 5.8: Measures of central tendency and dispersion for the dataset.

		Measures of Central Tendency			*Measures of Dispersion*			
Variable	*Group*	*Mean*	*Median*	*Mode*	*Range*	*St. Dev.*	*Min*	*Max*
Effectiveness	G1	0,54	0,50	0,50	0,27	0,09	0,44	0,71
	G2	0,40	0,41	0,50	0,75	0,18	0,00	0,75
	DIRECTOR	0,28	0,28	0,28	0,0	N/A	0,28	0,28
Efficiency	G1	0,09	0,11	#N/D	0,09	0,04	0,04	0,13
	G2	0,06	0,05	0,18	0,18	0,05	0,00	0,18
	DIRECTOR	0,67	0,67	0,67	0,0	N/A	0,67	0,67
Utility	G2	0,60	0,59	0,50	0,75	0,18	0,25	1,00

In total, participants were asked to execute twelve tasks that covered architectural decision activities for choosing a component, and explored available assets from modern

SECO platforms, in the shape of cloud microservices, as listed in Section 5.2. The first two tasks were not subjected to evaluation, tasks #3 to #7 were the main ones, related to the microservice selection process.

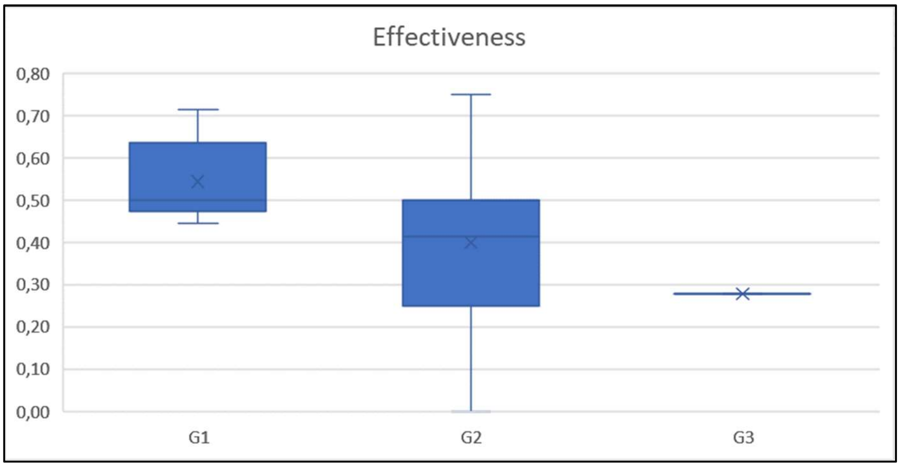


Figure 5.4: Boxplots for effectiveness.

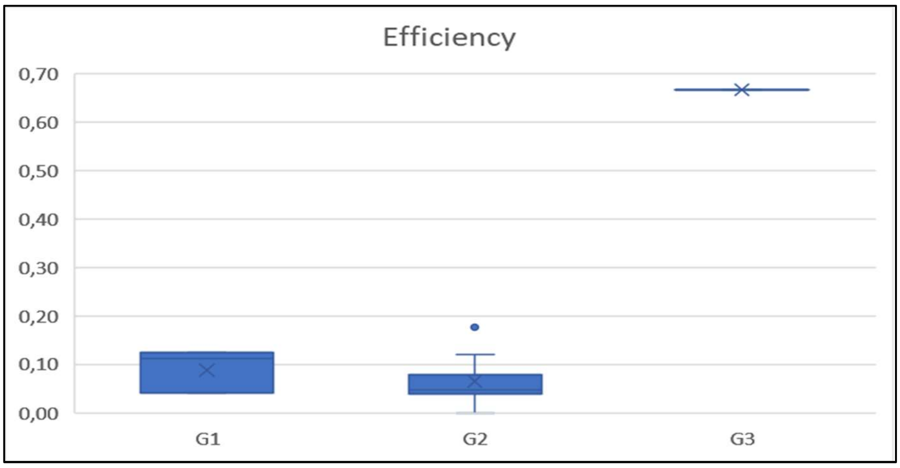


Figure 5.5: Boxplots for efficiency.

All participants completed all tasks. The groups were then analyzed as follows:

- Regarding *effectiveness*, G2 participants had an average of 40%, against 28% obtained by the DIRECTOR framework. A Wilcoxon test shows that the effectiveness of G2 participants is significantly different from a fixed mean of 0.28 (p-value = 0.01573), thus refusing H01. On the other hand, the data show that the framework's recommendations, in a preliminary analysis, seem to be closer to a junior than to a senior SE practitioner;

- Regarding *efficiency*, as expected, it can be seen a great improvement by the use of the framework's automated tool – besides the fact that a manual process is naturally *error-prone*, e.g., one could miss a potential candidate microservice. Participants of G1 spent on average 52 minutes to perform the proposed tasks, while G2 participants spent on average 43 minutes. While G1 and G2 obtained an average of 9% and 6% efficiency, respectively, the researcher using the framework's Web tool hit 67% of efficiency (10 right answers out of 36 in 15 minutes). Again, a Wilcoxon test shows that the efficiency of G2 participants is significantly different from a fixed mean of 0.67 (p-value < 0.00001), while G1 participants (a smaller group) show a less significant difference (p-value = 0.05791). We thus reject H02 on the basis of junior participants only;
- Finally, the senior group (G1) shows greater confidence and lower range probably due to its larger experience, while the confidence of participants from the less experienced group (G2) varies much more. This means that, though G2 participants had more effectiveness recommending microservices, their confidence is limited. Therefore, one could claim that DIRECTOR could be used to improve the confidence of less experienced SE practitioners.

5.4.5 Discussion

In a preliminary analysis, the *effectiveness* of DIRECTOR was reported as lower than G2's. This could be explained by the fact that, by default, DIRECTOR will always recommend three microservices for each scenario, while in three out of four scenarios the *oracle* gave a single recommendation. As the number of total recommendations has a direct impact on effectiveness, this caused apparently lower rates of effectiveness. For example, in scenario S1, which had only one right answer by the *oracle*, DIRECTOR scored 33.33% (three right recommendations out of nine guesses), which is the maximum possible – note that all perspectives were able to provide the right answer. As only S5 had two right answers, it is possible to claim that the maximum achievable effectiveness would be 41.25%.

By looking at Table 5.9, one can see the 36 recommendations by DIRECTOR – three per perspective. It is also possible to check the recommendations by each participant from G1, in comparison with the *oracle*. In the table, a red fill means a wrong recommendation, while a yellow means a partial right answer, *i.e.,* a valid recommendation along with an incorrect one. Green means a correct answer from the participant. Purple represents the oracle answers, while navy the answers produced by the researcher using DIRECTOR framework. Note that in 50% of the scenarios (S1 and S3), DIRECTOR was able to get the right answer in all perspectives (100%). In S3, two perspectives (66%) got it right, while only one (33%) got it right in S5.

Table 5.9: Analytic view of the results (G1 and DIRECTOR).

SME		***DIRECTOR***	***P1***	***P2***	***P7***	***P19***	***P20***
Scenario	***Oracle***	***Complementary Perspectives***	***Sr***	***Sr***	***Sr***	***Sr***	***Sr***
S1	**1511**	➢ Technical: **1511**, 1711, 811 ➢ Social: 1711, **1511**, 811 ➢ Semantic: 1691, 221, **1511**	**1511**	**1511**, 2081, 2111	**1511**, 2081	2111, 2155, 1851	91
S3	**571**	➢ Technical: **571**, 241, 301 ➢ Social: 481, 1421, 491 ➢ Semantic: 641, 2071, **571**	481, **571**	1081	**571**	**571**	**571**
S4	**1961**	➢ Technical: **1961**, 1971, 1871 ➢ Social: **1961**, 1971, 1871 ➢ Semantic: 831, **1961**, 71	**1961**, 301, 661	**1961**	**1961**	661, **1961**, 2159	2159
S5	**1051** and **1251**	➢ Technical: 2011, 2001, 1781 ➢ Social: 311, 1641, 621 ➢ Semantic: 2151, **1051**, **1251**	**1251**, **1051**, 1771	**1051**, **1251**, 791	1601, **1251**, **1051**	**1251**, **1051**	**1051**
#answers		36	9	8	7	9	4
#valid answers		10	5	4	5	4	2
% valid		28	56	50	71	44	50

Table 5.10 presents detailed information about DIRECTOR's perspectives recommendations. For a detailed analysis, three combinations were provided: with a single recommendation per perspective, with two and finally three (the default). First, regarding the scenarios, it is possible to see that S4 was the scenario where the framework performed the best, with 44% of right answers (eight hits in 18 recommendations in the derived scenarios).

On the other hand, the scenario where DIRECTOR performed the worst was the last one (S5), where only three hits were right, representing a 17% rate of effectiveness. Scenario S5 was summarized as "*I need to: store json documents in a nosql database*". By considering only the first two recommendations by DIRECTOR, since the oracle gave only two choices, it could be argued that DIRECTOR recommendations were also correct alternatives (though not as popular), as all of them could be considered noSQL databases, as follows:

- 2011: *elephantsql*, PostgreSQL as a Service (hybrid database), in Pivotal Cloud Foundry;

- 2001: *pubnub*, Build Realtime Apps that Scale (data streams), in Pivotal Cloud Foundry;
- 311: *attm2x*, Time Series IoT Data Service, in IBM Cloud;
- 1641: *pitneybowes-apis*, Add enterprise-class geodata and commerce technology your application, in IBM Cloud;
- 2151: *databases-for-redis*, Redis is a blazingly fast, in-memory data structure store, in IBM Cloud;
- 1051: *compose-for-mongodb*, MongoDB is a JSON document store with a rich query and aggregation framework, in IBM Cloud.

Additionally, by analyzing each perspective in detail, it is possible to recognize that the best effectiveness results were obtained by adopting only the first recommendation of the technical perspective (75%), while the worst result was obtained by adopting only the first recommendation of the semantic perspective (0%).

Table 5.10: Analytic view of the results (DIRECTOR's perspectives).

SME		***DIRECTOR Complementary Perspectives***										
Scenario	***Oracle***	***Technical***			***Social***			***Semantic***			***#valid answers***	***% valid***
		1st	*+ 2nd*	*+ 3rd*	*1st*	*+ 2nd*	*+ 3rd*	*1st*	*+ 2nd*	*+ 3rd*		
S1	**1511**	**1511**	**1511**, 1711	**1511**, 1711, 811	1711	1711, **1511**	1711, **1511**, 811	1691	1691, 221	1691, 221, **1511**	6	33
S3	**571**	**571**	**571**, 241	**571**, 241, 301	481	481, 1421	481, 1421, 491	641	641, 2071	641, 2071, **571**	4	22
S4	**1961**	**1961**	**1961**, 1971	**1961**, 1971, 1871	**1961**	**1961**, 1971	**1961**, 1971, 1871	831	831, **1961**	831, **1961**, 71	8	44
S5	**1051, 1251**	2011	2011, 2001	2011, 2001, 1781	311	311, 1641	311, 1641, 621	2151	2151, **1051**	2151, **1051**, **1251**	3	17
#answers		4	8	12	4	8	12	4	8	12		
#valid answers		3	3	3	1	2	2	0	2	5		
% valid		75	38	25	25	25	17	0	25	42		

The overall results from G2 are listed on Table 5.11 as follows. Considering that the *oracle* provided a single recommendation for S1, among 201 available microservices from three modern SECOs PaaS, the chances of DIRECTOR randomly getting a right answer would be 9 out of 201 (or 4.48%) – considering S5 the chances would be of 8.96%. Such low probability supports the claim that DIRECTOR's complementary perspectives are reasonable and necessary, *i.e.,* the selection does not depend on chance. Besides, with an expected increase in the number of microservices and platforms, that chance will become even lower, making the support of smart recommendation systems more important for SE practitioners. The *utility* metric also ratifies this claim, since G2 in average missed 60% of the answers, as can be seen by the red blocks in the table below. DIRECTOR would be useful for those practitioners, so they could get adequate recommendations.

Table 5.11: Analytic view of the results (G2).

SME		**P3**	**P4**	**P5**	**P6**	**P8**	**P9**	**P10**	**P11**	**P12**	**P13**	**P14**	**P15**	**P16**	**P17**	**P18**	**P21**	**P22**	**P23**	**P24**	**P25**
Scenario	**ORACLE**	Jr	Jr	Jr	Jr	Jr	Jr	Jr	Jr	Jr	Jr	Jr	Jr	Jr	Jr	Jr	Jr	Jr	Jr	Jr	Jr
S1	**1511**	91	**1511**	2081	2081	2081	621	**1511**	**1511,** 2081, 1601	**1511**	**1511,** 2081	91	2081	**1511,** 2081	**1511**	**1511**	**1511**	2158, 211	2081	11, 2081, 211	**1511**
S3	**571**	**571**	**571**	**571**	471, **571,** 1621	1081	1661	481	I don't know	481	1661	**571**	1661	481	481, **571**	**571**	**571**	**571**	1081, 1101	471, 481, **571**	481, 701
S4	**1961**	661, 2159	611	2159	**1961**	I don't know	301	171	661, 2159, **1961**	2159	301	301	**1961**	661	661	661	151	571	661	661, 2159, **1961**	301, 661, 2159
S5	**1051 and 1251**	**1251**	**1251**	**1051**	**1051,** 1791	**1251**	1461	**1051**	**1251,** 1791	**1051**	791, **1051,** **1251**	1791	**1251**	**1051**	**1051**	1791	**1051**	1791	**1251**	2151, 2158, **1251**	**1051,** 1771
	# answers	5	4	4	7	4	4	4	9	4	7	4	4	5	5	4	4	5	5	12	8
	# valid answers	2	3	2	3	1	0	2	3	2	3	1	2	2	3	2	3	1	1	3	2
	%	40%	75%	50%	43%	25%	0%	50%	33%	50%	43%	25%	50%	40%	60%	50%	75%	20%	20%	25%	25%

From the perspectives of G2, task #7 (scenario S5) was the one where even the less experienced SE practitioners were able to recommend, or guess, at least one microservice included in the *oracle's* list. Interesting enough, scenario S5, where DIRECTOR has performed the worst, G2 performed the best, with 17 right answers from 27 responses (63% *effectiveness*). This might be explained by the fact that this is the only scenario where the *oracle* provided two possible answers, instead of just one – doubling the chances of getting a right answer.

Finally, regarding *utility*, DIRECTOR achieved a mark of 60% considering G2 responses as defined in Subsection 5.2.3. Considering the adoption of the framework by a SE practitioner, it would be more useful to a junior than to a senior practitioner. In addition, taken into consideration the answers from participants in G2, this could also improve the confidence levels of their responses. Nevertheless, it is acknowledge that some improvement is needed, e.g., evolving the Semantic Perspective through *machine learning*, although one could recognize its value in more complex scenarios.

5.5 Threats to Validity

Every study has issues that can influence or limit the results' validity. Such issues are known as *threats to validity* and are classified into four categories (WHÖLIN et al., 1999; TRAVASSOS et al., 2002): (a) internal validity: defines if the relation between the treatment and the result is casual and derived from influences of other uncontrolled (or even not measured) factors. Sampling, grouping, treatment application, and social aspects are concerns in this category; (b) external validity: defines the conditions that make it difficult to generalize results to other contexts. We should consider participants' interaction with the treatment, location and occasion in this category; (c) construct validity: considers the relations between theory and observation, i.e., whether the treatment reflects the cause and the result reflects the effect. Undesirable behavior from the participants' or researcher's sides should be analyzed in this category; and (d) conclusion validity: refers to the conditions to make right conclusions on the relations between the treatment and the results. We should discuss statistical methods and sample size choices, as well as measures' confidentiality in this category.

Therefore, the threats to internal validity identified in this study are:

- the exchange of information with other participants who conducted the study – to reduce this risk, the study was executed with persons not known by each other, and we explicitly requested participants to not exchange information;

- the infrastructure can influence the results, if the participants face unexpected difficulties (e.g., slowness, server errors etc.), and the interactions with the tool can influence the way they perform the tasks – to reduce this risk, a pilot was run to capture any confounding factor, and participants had a computer with Internet access and with Microsoft Excel installed;
- the understanding of the execution form is directly influenced by the way the questions were designed, i.e., if the question was poorly worded, the study may be adversely affected – to reduce this risk, a pilot study was previously run to capture any confounding factor;
- the learning effect can manifest itself in the order the study's tasks were executed – to reduce this risk, tasks were arranged in an increasing complexity sequence and without entanglement, not to affect the thinking and the execution. Thus, the participant has the chance to understand the problem by running first with simpler tasks. It is noteworthy that the task sequence was the same in both groups;
- the proficiency in the English language could affect the results, as most of the information available (about microservices) is written in English – to reduce this risk, the research responsible for the study was available all the time during the study to translate terms or to help with any doubt related to translation issues;
- for the data analysis, the participants' characterization information should be used – unfortunately, it is not possible to meticulously verify that such information is correct, although the researcher would recommend the participants to be precise in their answers.

The threats to external validity identified in this study are:

- the study considers a mass of data related to a reality of three particular SECO platforms – a real dataset of three PaaS' catalogs were used, and five scenarios were drawn based on the available microservices in the aforementioned software catalogs;

- it is not possible to represent all the situations of a SECO context, then studies with different platforms and a greater number of SE practitioners should be performed – unfortunately, specialists from industry are rare and have serious time constraints, i.e., they are not so easily available. However, one strength of the study is the fact that a large dataset of 201 microservices was used, from three popular cloud platforms.

The threats to construct validity identified in this study are:

- the selected measures might not be good indicators for the feasibility of the proposed approach – to reduce this risk, measures were chosen based on the information needed to answer the tasks, and a pilot study was previously run to capture any confounding factor;
- since participants were chosen by convenience, their behavior might reflect assumptions on the expected results for this study – to reduce this risk, we executed the study with participants from different organizations, where participants have no academic relationship with the researcher responsible for the study. A random selection was not possible, since the approach requires participants who work as SE practitioners and have some experience in software architecture;
- since some of the participants have previously participated in a survey (described in Chapter 3) for QoS mapping, priorities were not used in this feasibility study, though DIRECTOR framework accounts for that, i.e., which QoS metric was most important for each of the five scenarios could have been indicated, but it was not;
- the tasks were grouped by type in order to aid data analysis and the same weight is assigned to all tasks; however, some tasks might have higher difficulty degree when compared to others and this fact can influence the results –this setting was kept because of the subjectivity in assessing difficulty degrees (which would introduce bias in the analysis).

Finally, the threats to conclusion validity identified in this study are:

- the main threat here is the sample size, with a relatively small number of participants, not being ideal from a statistical angle – to reduce this risk, our analysis included all data collected from the participants. Unfortunately, this is a recurrent challenge for empirical studies in the SE area. This is especially true for approaches that require industrial evaluation, as it is the case. Thus, the study presents a limitation on the results, which are considered as indications (and not evidences).

5.6 Final Remarks

In this chapter, we presented a feasibility study conducted with practitioners in a real scenario to evaluate DIRECTOR framework and contribute to the SECO community research and practice. The details on the study's planning and execution were discussed, as well as how the pilot was conducted with two participants at first. After refinement, the study was performed with additional 25 participants. As a result of RQ2, the effectiveness and efficiency to perform architectural decision activities for choosing a microservice was arguably improved with the framework support in the selected and applied context. Besides, the utility was above medium (60%) with the use of the framework, since less experienced SE practitioners were not always able to get the right answers (as provided by the *oracle*).

After analyzing participants' answers for the proposed tasks and for the study evaluation, there is some indication that the framework is applicable for cloud microservices selection in a SECO context to support architectural decision activities, especially regarding software acquisition. Several opportunities for improvements were identified, including regarding the graphical user interface of the Web tool. Considering usefulness, participants stated that, even when their choices were different from the ones provided by the framework, the latter were valid and adequate, especially by taking into consideration complementary perspectives and priorities in a multi-criteria scenario. It also reinforced the importance of having a good justification behind an architectural decision record.

This page was left blank intentionally.

Chapter 6 – Conclusion

This final chapter presents the Thesis' contributions and results, including some open questions suggested as future work.

6.1 Epilogue

SECO is reality for most organizations trying to keep up with the competition in the Internet era. Companies cannot depend only on internal employees to fulfill their customers' demands. On the other hand, it does not make much sense to pay in advance for hardware and software resources when one could be "on the cloud" and *pay-as-you-go*[44]… This combination resulted in modern SECO platforms, or PaaS, whose marketplaces have hundreds of microservices available to be reused.

By conducting an exploratory study and surveying specialists from industry, as presented in Chapter 3, it was verified that microservice selection in a cloud context is an open and relevant challenge, thus, motivating this Thesis. In turn, SECO and MSA are relative new topics in SE, so there is still plenty of opportunities for research.

To succeed in environments such as cloud SECOs, individuals should take advantage of all available resources published in a PaaS. When it comes to PaaS, a popular format nowadays is the microservice, which can be seen as a SOA evolution. These microservices are usually of the infrastructure kind, i.e., business-domain agnostic, which in turn increase reuse opportunities. Data store (relational, memory cache, NoSQL, ledgers), translation, IoT and device management and Messaging (queues and topics), are examples of crosscutting concerns[45] "as a service". With so many alternatives, it is reasonable to assume that soon it will become impossible to consider them all without some tool support.

[44] Pay-as-you-go is a system in which a person or organization pays for the costs of something when they occur rather than before or afterwards.

[45] The crosscutting concern is a concern that is applicable throughout the application and affects the entire application. For example: logging, security and data transfer are the concerns that are needed in almost every module of an application, hence they are crosscutting concerns.

6.2 Contributions and Results

In Chapter 4, the proposal was detailed and a preliminary experiment was described. It was a PoC, which was successful in demonstrating that DIRECTOR's approaches are feasible, i.e., by analyzing the PaaS' metadata, it was possible to determine a set of useful QoS metrics, which were used to support a SECO participant, in this example, a software architect, from a technical perspective. Furthermore, it was possible to measure the community engagement, i.e., to infer the technology's level of adoption by other SE practitioners. It was also possible to take advantage of IBM Watson cognitive system for dealing with questions posed in natural language. Provided that DIRECTOR can be used even by a person not so technically versed, several opportunities and use scenarios for this research (including the Web recommendation tool) are foreseen, not only in the academy, but in industry as well.

Later, as detailed in the previous chapter, a feasibility study with 27 SE professionals from industry was conducted, aimed at gathering evidence of DIRECTOR's correctness, efficiency and relevancy (utility). The results were promising, though it has been found that the "seniority" of DIRECTOR framework is somewhere between a junior and a senior practitioner – ideally, it should be closer to a senior. Considering the growing number of candidates, the efficiency of such tool is pivotal for the challenge of choosing a microservice in a cloud environment. Besides, results indicated that it could increase the confidence of the choice made by the practitioner, e.g., developer or software architect, once it offers a rationale for each recommendation, avoiding a judgmental decision.

6.2.1 Main Contribution

This PhD Thesis contributes with: (a) the development of a framework to help researchers to better understand modern SECO architectures and key concepts and to analyze cloud-based platforms; (b) the identification of microservice's quality criteria that are critical for SE practitioners regarding choosing a microservice in a SECO context, based on the experts' opinion; (c) the definition of three perspectives (strategies) for discovering and comparing cloud microservices to support IT architectural decision activities, more specifically selecting a software component to be reused in a cloud-based solution; and (d) the evaluation of the perspective modules of the framework (recommendation tool) with SE practitioners (IT architects and developers) performing architectural decisions in real scenarios.

This PhD research provided the SE community with the following detailed contributions in the context of choosing a microservice in cloud SECO platforms:

- two surveys with experts (Chapter 3): the conducted survey with SMEs ratified the assumptions used for the technical evaluation. Responses from 44 SE practitioners around the globe were received, regarding the selection of cloud microservices. Respondents included enterprise architects, cognitive specialists, IT executives, solutions architects, software engineers, among other job roles, from Australia, Brazil, Portugal, Germany, Hungary and United States. The vast majority of them responsible for making architectural and/or design decisions in their organizations;
- a PoC (Section 4.7): a preliminary experiment to validate this Thesis' ideas was performed. The PoC was conducted with a software architect from a multinational company in the IT segment. The prototype applied MCDM-analysis to calculate the technical and social "quality" of the candidates, beside an artificial intelligence technique. Evidence suggested that DIRECTOR can support software acquisition via discovery, evaluation and comparison of microservices in multiple cloud platforms;
- DIRECTOR conceptualization (Chapter 4): the studies performed throughout the research activities allowed the conclusion that SE practitioners need a framework (including a recommendation tool) to be able to choose a microservice, performing architectural decision activities in SECO platforms considering the increasing number of choices. A QoS model based on SMI was defined, and then mapped to the metadata extracted from the PaaS – leveraging the CF open-standard. This mapping allows new cloud platforms to be connected to the tool in the future;
- DIRECTOR Web tool (Chapter 4): an infrastructure to support the proposed framework was developed as a Web application in a cloud platform (available at http://director-services.mybluemix.net). Three platforms are currently connected to the framework, allowing for the quick

evaluation of more than 200 microservices, accordingly to multiple criteria and priorities informed by the user;

- A feasibility study with real data and in real scenarios (Chapter 5): 27 practitioners evaluated the proposed framework and infrastructure (DIRECTOR Web tool) in real scenarios. The efficiency to choose a microservice and perform architectural decisions activities in cloud SECOs was greatly improved with the framework support in the selected and applied context. However, effectiveness was not so high with the use of the framework, compared to junior, i.e., less experienced SE practitioners, considering the reduced number of possible right answers given by the *oracle*.

6.2.2 Secondary Contributions

We have collaborated with/supported a Master's work in the context of this PhD research: "SECO-AM: an Approach for Maintenance of IT Architecture In Software Ecosystems", and an in-progress study named "Quasi Systematic Review / Literature Narrative – Model for Suggestion of Criticity in Microservice Oriented Architecture", both at COPPE – Federal University of Rio de Janeiro, Rio de Janeiro, Brazil.

6.2.3 Publications

Research activities performed in this PhD produced the following publications:

- "A Roadmap for Cloud SECO: EcoData and the New Actors in the IoT Era", Jun 12th, 2015, IEEE International Conference on Distributed Computing in Sensor Systems, pp. 218-223, Fortaleza, Brazil;
- "EcoData: Architecting Cross-Platform Software Ecosystem Applications", May 29th, 2017, Doctoral Symposium - The 16th International Conference on Software Reuse (ICSR 2017), pp. 195-200, Salvador, Brazil;
- "Perspectives for Selecting Cloud Microservices", May 2nd, 2018, IEEE International Conference On Software Architecture (ICSA 2018), p. 56-59, Seattle, USA;

- “Evaluating Cloud Microservices with DIRECTOR”, Apr 7th, 2019, 2019 IEEE International Conference on Service-Oriented System Engineering (SOSE), pp. 47-56, California, USA;
- “Software architecture for Software Ecosystems: A Systematic Literature Mapping and Research Agenda”, May, 2019, Technical Report at COPPE/UFRJ and submitted to Technological Forecasting and Social Change, an International Journal by ELSEVIER in 05-11-2019.

6.3 Limitations

As a limitation of both the preliminary experiment (PoC) and the feasibility study, only three platforms are being monitored, i.e., DIRECTOR has only three connected PaaS: IBM Cloud, Pivotal PCF and SAP Cloud Platform. Therefore, the prototype and the Web tool only evaluated μSaaS published on such marketplaces.

A current research limitation is the fact that the technical perspective is restricted to the data made available via the CF’s API and published by the μSaaS provider. Therefore, it is important to revisit the framework data model whenever there is a change in the APIs, e.g., when a new version is released. That way, it would be possible to expand the framework encompassing new QoS attributes and categories. Note that DIRECTOR is **not** limited to a single PaaS, nor by IBM Cloud, i.e., it is possible to compare μSaaS from different PaaS.

Another limitation is the fact that we conducted the feasibility study with a group of few people, i.e., only 27 practitioners, which does not provide us with a high statistical confidence – though this is an expressive number of participants from industry. This is a known challenge when it comes to specialists from industry – they are rare to find and have a busy agenda. We have also explored little of the multi-criteria features, i.e., in the experiments we have not taken into consideration priorities.

Regarding IBM Watson, we have not extensively trained it before running the experiment, which caused low rates of statistical confidence as well. We have not implemented the machine learning feature as well, i.e., in the current implementation of the tool, the practitioner cannot give feedback to the machine, thus it cannot learn from its previous experiences (interactions with a human).

Lastly, it is wise to state that it is not expected that DIRECTOR would replace a software engineer or an IT architect when it comes to making such important architectural decisions. As the name implies, the intention is to "direct" SE practitioners to the alternatives that best fit their necessities and priorities, though sometimes they might not be so easily spotted. The ultimate goal is for the framework to be useful for such professionals, especially in a vast and diverse environment such as a SECO.

6.4 Open Questions and Future Work

As for the continuous evaluation of DIRECTOR framework, a new experiment is being planned with another group of practitioners (developers, software engineers or IT architects) from both industry and academy. We want to evaluate more complex scenarios, including prioritization of multiple criteria, as well as improve the statistical confidence.

Currently, DIRECTOR only accounts for the prioritization of QoS categories. A possible enhancement is to increase the granularity, allowing for the prioritization of attributes and sub-attributes (following SMI's ontology). Additionally, the "extra" field of a service plan (entity) can be further analyzed to figure out additional characteristics (e.g., capacity) and price, which would become new QoS attributes. Plus, to evolve the current QoS model, SMI will be compared with other related standards, such as ISO/IEC 25000 (ISO, 2005) and NIST Cloud Computing Service Metrics Description (NIST, 2015).

For the social perspective, the goal is to reach additional forums, i.e., to retrieve posts from other community sources, such as Twitter[46], Reddit[47] and Quora[48]. Then, by leveraging another Watson service named Tone Analyzer[49], and applying an AI technique known as sentiment analysis, DIRECTOR will try to discover what community members "are saying" about a certain microservice – evaluating whether it is positive, negative or neutral. The motivation for this approach is the fact that it could be difficult, and take long, to produce a data set of reviews big enough to allow for such comparisons, while there is plenty of information available on the WWW already.

[46] https://twitter.com/

[47] https://www.reddit.com/

[48] https://quora.com

[49] https://www.ibm.com/watson/services/tone-analyzer/

The comparison between PaaS offerings and the analysis of the platform's "health" (is it growing or diminishing?) are also being considered as potential research extensions in the near future – the number of µSaaS has varied (increased and decreased) over time. Yet regarding PaaS (on top of IaaS), two very popular offerings are being considered to be registered in the framework: Google Cloud Platform[50] and Amazon AWS[51] – both have recently announced a partnership with Pivotal Cloud Foundry. This needs to be further investigated to determine the degree of compatibility with the CF standard.

Another possible path is related to dynamic reconfiguration of systems. On-demand service provisioning is not part of DIRECTOR's goals. However, it should be possible, via PaaS' APIs, not only to retrieve information about microservices, but also to provision new instances of them. The CF standard would allow for this approach to be applied in a multi-cloud strategy, which is a common pattern in industry to avoid vendor lock-in. If a microservice instance fails, it should be possible to instantiate a new one, on a different provider, and reconfigure the system to consume the new one. This is especially interesting when it comes to IoT, a very popular trend nowadays.

Finally, an issue yet to be addressed is related to the stability attribute and how to include the "Interface Stability" sub-attribute. Being a very technical QoS attribute, this could be of special interest to developers. The intention is to use the "http options" method plus the OpenAPI[52] (former Swagger) specification to find out changes in the operation signatures of RESTful service' operations – something equivalent to the traditional SOA "wsdl" files. A request for enhancement[53] was submitted in 2017 for this to be considered in a future release of CF API – unfortunately, the current version does not provide a metadata field for the API specification (documentation) endpoint.

[50] https://pivotal.io/partners/google

[51] https://content.pivotal.io/blog/meet-the-new-aws-service-broker-for-pcf-now-an-open-beta-heres-why-its-the-easiest-way-to-add-18-aws-services-to-your-cloud-native-apps

[52] https://www.openapis.org/

[53] https://www.pivotaltracker.com/n/projects/966314/stories/152069882

This page was left blank intentionally.

References

ALBERT, B.E., 2014, "SECOGov: A Software Ecosystem Governance Model to Support IT Architecture Activities". Master Thesis in Computer Science and System Engineering. COPPE – Federal University of Rio de Janeiro, Rio de Janeiro, Brazil, 171p. (In Portuguese)

ALVES, C. F., 2005, "Managing Mismatches in COTS-Based Development". PhD Thesis in Computer Science. University College London (UCL), London, UK, pp. 14-18, 32-44.

ALVES, C., CASTRO, J., 2001, "CRE: A systematic method for cots selection." In *Proceedings of the 15th Brazilian Symposium on Software Engineering (SBES)*. Rio de Janeiro, Brazil.

ALVES, C., FINKELSTEIN, A., 2003, "Investigating conflicts in COTS decision-making." International Journal of Software Engineering and Knowledge Engineering, v. 13, n. 05, p. 473-493.

AMORIM, S. S., ALMEIDA, E. S., McGREGOR, J. D., 2013, "Extensibility in ecosystem architectures: an initial study." In: *Proceedings of the 2013 International Workshop on Ecosystem Architectures*. ACM, 2013. p. 11-15.

ANVAARI, M., JANSEN, S., 2010, "Evaluating architectural openness in mobile software platforms". In *Proceedings of the Fourth European Conference on Software Architecture: Companion Volume* (pp. 85-92). ACM.

ARAUJO, R. M., MAGALHÃES, A., 2015, "Ecossistemas Digitais para o Apoio a Sistemas de Governo Abertos e Colaborativos." In *Proceedings of the XI Brazilian Symposium on Information Systems*, Goiânia, Brazil, pp. 647-650. In Portuguese.

ASHTON, K., 2009, "That 'Internet of Things' Thing.", available at http://www.rfidjournal.com/articles/pdf?4986, retrieved in 12-05-2015, *RFiD Journal*, 22(7), 97-114.

AXELSSON, J., SKOGLUND, M., 2016, "Quality assurance in software ecosystems: A systematic literature mapping and research agenda." Journal of Systems and Software, 114, pp. 69-81.

BARBOSA, O., DOS SANTOS, R. P., ALVES, C., WERNER, C., JANSEN, S., 2013, "A systematic mapping study on software ecosystems from a three-dimensional perspective". In: *Jansen, S., Brinkkemper, S., Cusumano, M. (Eds.), Software Ecosystems*

– *Analyzing and Managing Business Networks in the Software Industry*. Edward Elgar Publishing, pp. 59–81.

BARRETO, A.O.S., 2011, “Definition and Management of Strategically Aligned Software Improvement Goals”. PhD Thesis in Computer Science and System Engineering. COPPE – Federal University of Rio de Janeiro, Rio de Janeiro, Brazil, 240p. (In Portuguese)

BARUCH, Y., 1999, “Response rate in academic studies-A comparative analysis.”, Human relations, v. 52, n. 4, p. 421-438.

BASILI, V.R., SHULL, F., LANUBILE, F., 1999, “Building Knowledge through Families of Experiments”. *IEEE Transactions on Software Engineering* 25(4):456-473.

BEUS-DUKIC, L., 2000, “Non-functional requirements for COTS software components.” In *Proceedings of ICSE workshop on COTS Software* (pp. 4-5) June.

BOEHM, B. W., 1981, “Software engineering economics”, (Vol. 197). Englewood Cliffs (NJ): Prentice-hall.

BOEHM, B., 2006, “A View of 20th and 21st Century Software Engineering”, In: *Proceedings of the 28th International Conference on Software Engineering,* pp. 12-29, Shanghai, China, May.

BOOCH, G., 2006, "On design" cited in: *Frank Buschmann, Kevlin Henney, Douglas C. Schmidt (2007) Pattern-Oriented Software Architecture*, On Patterns and Pattern Languages. p. 214.

BOSCH, J., 2009, “From Software Product Lines to Software Ecosystem”, *Proceedings of 13th International Software Product Line Conference*, pp. 1-10, San Francisco, CA, USA, August.

BOSCH, J., 2012, “Software Ecosystems: Implications for Strategy, Business Model and Architecture”. In: *Proceedings of the 34th International Conference on Software Engineering* (ICSE), Tutorial, Zurich, Switzerland, June.

BOSCH, J., BOSCH-SIJTSEMA, P. M., 2010a, “Software Product Lines, Global Development and Ecosystems: Collaboration in Software Engineering”, in *Collaborative Software Engineering*, I. Mistrík, J. Grundy, A. Hoek, and J. Whitehead, Eds., pp. 77–92, Springer, Berlin, Germany.

BOSCH, J., BOSCH-SIJTSEMA, P., 2010b, “From integration to composition: On the impact of software product lines, global development and ecosystems.”, Journal of Systems and Software, v. 83, n. 1, p. 67-76.

BOWEN, D. J., KREUTER, M., SPRING, B., COFTA-WOERPEL, L., LINNAN, L., WEINER, D., FERNANDEZ, M., 2009, "How we design feasibility studies." In: *American journal of preventive medicine*, 36(5), 452-457.

CAMPBELL, P. R., AHMED, F., 2010, "A three-dimensional view of software ecosystems". In: *Proceedings of the Fourth European Conference on Software Architecture*: Companion Volume (pp. 81-84). ACM. Copenhagen, Denmark. August.

CARNEY, D., 1998, "COTS evaluation in the real world." SEI Interactive, Carnegie Mellon University.

CARVALLO, J. P., FRANCH, X., QUER, C., 2003, "Defining a quality model for mail servers." In *International Conference on COTS-Based Software Systems* (pp. 51-61). Springer, Berlin, Heidelberg, February.

CHESBROUGH, H. W., 2003, "Open Innovation: The new imperative for creating and profiting from technology." Harvard Business School Press, pp.72.

CHUNG, L., COOPER, K., 2002, "A knowledge-based COTS-aware requirements engineering approach." In: *Proceedings of the 14th international conference on Software engineering and knowledge engineering.* ACM, 2002. p. 175-182.

CLARK, B., TORCHIANO, M., 2004, "COTS terminology and categories: Can we reach a consensus?" Lecture Notes in Computer Science: 4-5.

CUKIERMAN, H.L., TEIXEIRA, C., PRIKLADNICKI, R., 2007, "Um Olhar Sociotécnico sobre a Engenharia de Software". *Revista de Informática Teórica e Aplicada* 14(2):207-227. (In Portuguese)

DAVIS, F.D., 1993, "User Acceptance of Information Technology: System Characteristics, User Perceptions and Behavioral Impacts". *International Journal of Man-Machine Studies* 38(3):475-487.

DELOITTE, 2016, "2016 Technology Industry Outlook", Deloitte.com, available at https://www2.deloitte.com/us/en/pages/technology-media-and-telecommunications /articles/2016-telecommunications-industry-outlook.html, retrieved in 07-25-2016.

DI FRANCESCO, P., MALAVOLTA, I., LAGO, P., 2018, "Migrating towards microservice architectures: an industrial survey.", In: *2018 IEEE International Conference on Software Architecture*, ICSA 2018, Seattle, USA, April 30, - May 4, 2018, pp. 29–38.

DI FRANCESCO, P., LAGO, P., MALAVOLTA, I., 2019, "Architecting with microservices: A systematic mapping study." Journal of Systems and Software, v. 150, p. 77-97.

DYBÁ, T., DINGSOYR, T., HANSSEN, G., 2007, "Applying systematic reviews to diverse study types: an experience report". In: *First International Symposium on Empirical Software Engineering and Measurement*, ESEM 2007, pp. 225-234. Madrid, Spain. September.

DUEÑAS, J. C., CAPILLA, R., 2005, "The Decision View of Software Architecture", In: *Morrison R., Oquendo F. (eds) Software Architecture*. EWSA. Lecture Notes in Computer Science, vol 3527. Springer, Berlin, Heidelberg.

EISA, M., YOUNAS, M., BASU, K., ZHU, H., 2016, "Trends and directions in cloud service selection". In: *Service-Oriented System Engineering* (SOSE), 2016 IEEE Symposium on (pp. 423-432). IEEE. March.

EKLUND, U., BOSCH, J., 2014, "Architecture for embedded open software ecosystems". Journal of Systems and Software, 92, 128-142.

ENGSTROM, E., RUNESON, P., SKOGLUND, M., 2010, "A systematic review on regression test selection techniques". Inf. Software Technol. 52 (1), 14-30.

ERL, T., 2005, "Service-Oriented Architecture – Concepts, Technology, and Design", Prentice Hall: Boston, p.557.

EVERITT, B. S., 1998, "The cambridge dictionary of statistics cambridge university press". Cambridge, UK Google Scholar.

FERRUCCI, D. A., 2012, "Introduction to 'this is watson'." IBM Journal of Research and Development, v. 56, n. 3.4, p. 1: 1-1: 15.

FERRUCCI, D., LEVAS, A., BAGCHI, S., GONDEK, D., MUELLER, E. T., 2013, "Watson: beyond jeopardy!". Artificial Intelligence, 199, 93-105.

FIELDING, R. T., 2000, "Chapter 5: Representational State Transfer (REST)", In: *Architectural Styles and the Design of Network-based Software Architectures* (Ph.D. Thesis). University of California.

FINKELSTEIN, A., 2014, "Rethinking Software: Business Change and the Consequences for Software Engineering". In: *Proceedings of the 22nd IEEE International Requirements Engineering Conference* (RE), Keynote, Karlskrona, Sweeden, August.

FINKELSTEIIN, A., SPANOUDAKIS, G., RYAN, M., 1996, “Software package requirements and procurement.” In *Proceedings of the 8th international workshop on software specification and design* (p. 141). IEEE Computer Society, March.

FORBES, 2012, “Is Data the new Oil?”, available at http://www.forbes.com/sites/perryrotella/2012/04/02/is-data-the-new-oil/, retrieved in 12-05-2015.

FORRESTER, 2013, “The Total Economic Impact of IBM’s Worklight Platform”, April, A Forrester Total Economic Impact™ Study Prepared For IBM, available at http://mobileenterprise.edgl.com/Libraries/Images/PDFs/mees14_ibm.pdf, retrieved in 12-12-2016.

FOWLER, M., 2003, “Design – Who needs an architect?”. IEEE Software. 20 (5): pp. 11-44. Doi:10.1109/MS.2003.123114.

FOWLER, M., LEWIS, J., 2014. Microservices a definition of this new architectural term. http://martinfowler.com/articles/microservices.html.

FRAIR, L., 1995, “Student peer evaluations using the analytic hierarchy process method.” In *Proceedings Frontiers in Education 1995 25th Annual Conference.* Engineering Education for the 21st Century (Vol. 2, pp. 4c3-1). IEEE, November.

FRANÇA, M., 2017, “EcoData: Architecting Cross-Platform Software Ecosystem Applications”, In: *ICSR 2017 - The 16th International Conference on Software Reuse*, Doctoral Symposium, Salvador, Bahia, Brazil. May, 2017.

FRANÇA, M., WERNER, C., 2018, “Perspectives for Selecting Cloud Microservices”. In *2018 IEEE International Conference on Software Architecture Companion Proceedings ICSA-C*, pp. 63-66, April 30 – May 4, Seattle, USA.

FRANÇA, M., WERNER, C., 2019, “Evaluating Cloud Microservices with DIRECTOR”, In: *Proceedings of the 13th IEEE International Conference on Service-Oriented System Engineering* (SOSE 2019), pp. 47-56, California, USA, April.

FRANÇA, M., SANTOS, R., WERNER, C., 2015, “A Roadmap for Cloud SECO: EcoData and the New Actors in IoT Era”. In: *2015 International Conference on Distributed Computing in Sensor Systems* (pp. 218-223). IEEE.

GARG, S. K., VERSTEEG, S., BUYYA, R., 2011, “Smicloud: A framework for comparing and ranking cloud services.” In: *2011 Fourth IEEE International Conference on Utility and Cloud Computing*. IEEE, p. 210-218.

GARLAN, D. and SHAW, M., 1993, “An Introduction to software architecture.” Advances in software engineering and knowledge engineering Vol 1. P. 1-40.

GARTNER, 2012, “The Nexus of Forces: Social, Mobile, Cloud and Information”, available at https://www.gartner.com/doc/2049315, accessed 12-10-2016.

GARTNER, 2013, “Predicts 2014: Infrastructure Protection,” Gartner #G002147953, Ray Wagner, Kelly M. Kavanagh, Mark Nicolett, Anton Chuvakin, Andrew Walls, Joseph Feiman, Lawrence Orans, Ian Keene, November 25, 2013.

GARTNER, 2014a, “The Nexus of Forces: Social, Mobile, Cloud and Information”, available at https://www.gartner.com/technology/research/nexus-of-forces/, accessed 07-25-2016.

GARTNER, 2014b, “Comparing Open-Source PaaS Frameworks: Cloud Foundry and OpenShift”, Analyst(s): Eric Knipp, Published: 04 September 2014, accessible at https://www.gartner.com/doc/2838118/comparing-opensource-paas-frameworks-cloud

GODSE, M., MULIK, S., 2009, “An approach for selecting software-as-a-service (SaaS) product.” In: *2009 IEEE International Conference on Cloud Computing*. IEEE, p. 155-158.

GRACIANO NETO, V. V., CAVALCANTE, EL HACHEM, E., SANTOS, D. S., 2017a, “On the Interplay of Business Process Modeling and Missions in Systems-of-Information Systems.” In *Proceedings of the 2017 IEEE/ACM Joint 5th International Workshop on Software Engineering for Systems-of-Systems and 11th Workshop on Distributed Software Development, Software Ecosystems and Systems-of-Systems (JSOS)*, Buenos Aires, Argentina, pp. 72-73.

GRACIANO NETO, V. V., SANTOS, R. P., ARAUJO, R., 2017b, “Sistemas de Sistemas de Informação e Ecossistemas de Software: Conceitos e Aplicações.” In *B. Zarpelão et al. (Org.) Tópicos em Anais do VIII Workshop sobre Aspectos da Interação Humano-Computador na Web Social Sistemas de Informação: Minicursos* SBSI 2017. 1ed. Lavras: UFLA, v. 1, pp. 22-41. In Portuguese.

GROVES, R. M., FOWLER Jr, F. J., COUPER, M. P., LEPKOWSKI, J. M., SINGER, E., TOURANGEAU, R., 2011, “Survey methodology”, (Vol. 561). John Wiley & Sons.

GUINARD, D., TRIFA, V., KARNOUSKOS, S., SPIESS, P., SAVIO, D., 2010, “Interacting with the soa-based internet of things: Discovery, query, selection, and on-demand provisioning of web services.”. In *IEEE transactions on Services Computing*, pp. 223-235.

HAENNI, N., LUNGU, M., SCHWARZ, N., NIERSTRASZ, O., 2014, “A quantitative analysis of developer information needs in software ecosystems.”, *8th*

European Conference on Software Architecture Workshops, ECSAW 2014; Vienna; Austria; 25 August 2014 through 29 August 2014.

HANSSEN, G., DYBÅ, T., 2012, "Theoretical Foundations of Software Ecosystems". In: *Proceedings of the 4th International Workshop on Software Ecosystems* (IWSECO), in conjunction with the 3rd International Conference on Software Business (ICSOB), pp. 6-17, Cambridge, USA, June.

HERNANDES, E., ZAMBONI, A., DI THOMMAZO, A., FABBRI, S., 2010, "Avaliação da ferramenta StArt utilizando o modelo TAM e o paradigma GQM". In: *Proceedings of the VII Experimental Software Engineering Latin American Workshop* (ESELAW), pp. 30-39, Goiania, Brazil, November. (In Portuguese)

HEVNER, A.R., MARCH, S.T., PARK, J., 2004, "Design Science in Information System Research". *MIS Quarterly* 28(1):75-105.

HILWA, A., 2015, "The emergence of microservices as a new architectural approach to building new software systems", IDC Study, June.

HU, Y., SUN, X., ZHANG, J., ZHANG, X., LUO, F., HUANG, L., 2009, "A University Student Behavioral Intention Model of Online Shopping". In: *Proceedings of the International Conference on Information Management, Innovation Management and Industrial Engineering* (ICIII), pp. 625-628, Xi'an, China, December.

IANSITI, M., LEVIEN, R., 2004, "Strategy as Ecology." Harvard Business Review, 82 (3), 68-78.

IBM, 2014, "The data center impact of cloud, analytics, mobile, social and security. A mandate for infrastructure agility, dynamic optimization and software defined environments", IBM Global Technology Services - Thought leadership white paper, available at http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?infotype=SA&subtype=WH&htmlfid=RLW03031USEN, retrieved in 07-26-2016.

IBM, 2015, "CAMSS Ready", In IBM Systems Magazine, available at http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=PO103036USEN&appname=skmwww, retrieved in 07-25-2016.

IDC, 2014, "EMC Digital Universe Study, with Data and Analysis by IDC". April, technical report.

IDC, 2016, "Explore 3rd Platform Transformation", available at http://www.idc.com/prodserv/3rd-platform/, accessed 07-25-2016.

IEEE Computer Society, 2000, “IEEE Recommended Practice for Architectural Description of Software-Intensive Systems”, IEEE Std 1472000.

ISO, 2005, IEC, ISO. “ISO/IEC 25000 software engineering software product quality requirements and evaluation (SquaRE) guide to SquaRE.” Systems Engineering, v. 41.

ISO, 2008, ISO/IEC 12207: “Systems and software engineering – Software life cycle processes”. International Organization for Standardization: Geneva.

IYENGAR, S. S., LEPPER, M. R., 2000, “When choice is demotivating: Can one desire too much of a good thing?”. Journal of personality and social psychology, v. 79, n. 6, p. 995.

JANSEN, S., 2013, “How quality attributes of software platform architectures influence software ecosystems”. In *Proceedings of the 2013 International Workshop on Ecosystem Architectures* (pp. 6-10). ACM. Saint Petersburg, Russian Federation. August.

JANSEN, S., CUSUMANO, M., 2012, “Defining Software Ecosystems: A Survey of Software Ecosystems and Ecosystem”. In: *Proceedings of the 4th International Workshop on Software Ecosystems* (IWSECO), in conjunction with the 3rd International Conference on Software Business (ICSOB), pp. 41-58, Cambridge, USA, June.

JANSEN, S., FINKELSTEIN, A., BRINKKEMPER, S., 2009, “A Sense of Community: A Research Agenda for Software Ecosystems.” In: *Proceedings Of the 31st ICSE, New and Emerging Research Track*, pp. 187-190. Vancouver, BC, Canada, May.

JAMSHIDI, M., 2005, “System-of-systems engineering-a definition”. IEEE SMC, 32.

KARIM, R., DING, C., MIRI, A., 2013, “An end-to-end QoS mapping approach for cloud service selection.” In: *2013 IEEE Ninth World Congress on Services.* IEEE, p. 341-348.

KELLY, J. III, 2015, “Computing, cognition and the future of knowing”. IBM Research: Cognitive Computing. IBM Corporation. Available at http://www.research.ibm.com/software/IBMResearch/multimedia/Computing_Cognition_WhitePaper.pdf, retrieved in 07-19-2017.

KIM, Y., 2014, “Convolutional Neural Networks for Sentence Classification.” In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing* (EMNLP). https://doi.org/10.3115/v1/d14-1181

KNODEL, J., NAAB, M., ROST, D., 2014, “Supporting architects in mastering the complexity of open software ecosystems”. In *Proceedings of the 2014 European Conference on Software Architecture Workshops* (p. 13). ACM. Vienna, Austria. August.

KONTIO, J., 1995, “OTSO: a systematic process for reusable software component selection.” Technical report, University of Maryland.

KONTIO, J., 1996, “A case study in applying a systematic method for COTS selection.” In *Proceedings of the 18th international conference on Software engineering* (pp. 201-209). IEEE Computer Society, May.

KRUCHTEN, P., CAPILLA, R., DUEÑAS, J. C., 2009, “The decision view's role in software architecture practice.”, IEEE software, v. 26, n. 2, p. 36-42.

KUNDRA, V., 2010, “State of Public Sector Cloud Computing”. May. Available at https://www.cio.gov/2010/05/20/state-of-public-sector-cloud-computing/, retrieved in 07-19-2017.

LEWIS, J., 2012, “Micro Services – Java the Unix way”, 33rd Degree Conference, 19-21 March 2012, Poland, available at http://2012.33degree.org/pdf/JamesLewisMicroServices.pdf, retrieved in 12-05-2015.

LEWIS, J., FOWLER, M., 2014, “Microservices”, available at http://www.martinfowler.com/articles/microservices.html, accessed in 12-05-2015.

LI, S., 2018, “Understanding quality attributes in microservice architecture” In: *Proceedings 2017 24th Asia-Pacific Software Engineering Conference Workshops*, APSECW 2017, January, pp. 9-10

LIMA, T., SANTOS, R. P., OLIVEIRA, J., & WERNER, C., 2016, “The importance of socio-technical resources for software ecosystems management”. *Journal of Innovation in Digital Ecosystems*, 3(2), 98-113.

LUNDBERG, L., BOSCH, J., HÄGGANDER, D., BENGTSSON, P., 1999, “Quality Attributes in Software Architecture Design”, In: *Proceedings of 3rd IASTED – International Conference Software Engineering and Applications*. October 6-8, Arizona, USA.

MAFRA, S., TRAVASSOS, G.H., 2006, “Primary and Secondary Studies to Support Evidence in Software Engineering”. Technical Report PESC/COPPE – ES-687/06, Federal University of Rio de Janeiro.

MAGDALENO, A.M., 2013, “COMPOOTIM: Towards Planning, Monitoring and Optimization of Collaboration in Software Process Definition”. PhD Thesis in

Computer Science and System Engineering. COPPE – Federal University of Rio de Janeiro, Rio de Janeiro, Brazil, 253p. (In Portuguese)

MAIDEN, N. A., NCUBE, C., 1998, "Acquiring COTS software selection requirements". IEEE software, v. 15, n. 2, p. 46-56.

MAIER, M. W., 1996, Architecting principles for systems-of-systems. Systems Engineering 1(4):267-284.

MAJD, S., MARIE-HÉLÈNE, A., ALOK, M., 2015, "An Architectural Model for System of Information Systems." In Ciuciu I. et al. (eds.). On the Move to Meaningful Internet Systems: OTM 2015 Workshops. Lecture Notes in Computer Science, vol. 9416. Springer, Cham.

MANIKAS, K., 2016, "Revisiting Software Ecosystems Research: A Longitudinal Literature Study". Journal of Systems and Software. Volume 117, July, pp. 84–103.

MANIKAS, K., HANSEN, K. M., 2013, "Software ecosystems: A systematic literature review." *Journal of System Software 86*, 1294-1306.

MARR, B., 2015, Big Data: Using SMART Big Data, analytics and metrics to make better decisions and improve performance, Wiley, March.

MARTIN, R., 2015, "The first micro-service architecture", available at https://8thlight.com/blog/uncle-bob/2015/05/28/TheFirstMicroserviceArchitecture.html, accessed in 12-05-2015.

McGOLDRICK, P. J., COLLINS, N., 2007, "Multichannel retailing: Profiling the multichannel shopper". The International Review of Retail, Distribution and Consumer Research.

McILROY, M. D., 1968, "Mass Produced Software Components", In: *Proceedings of NATO Conference on Software Engineering*, Garmisch, Germany, pp. 88-98, October.

MENDES, E., TURHAN, B., RODRIGUEZ, P., FREITAS, V., 2015, "Estimating the Value of Decisions Relating to Managing and Developing Software-intensive Products and Projects". In: *Proceedings of the 11th International Conference on Predictive Models and Data Analytics in Software Engineering* (PROMISE), Article No. 7, Beijing, China, October.

MENZEL, M., SCHÖNHERR, M., TAI, S., 2013, "(MC2)2: criteria, requirements and a software prototype for Cloud infrastructure decisions." Software: Practice and experience, 43(11), pp. 1283-1297.

MESSERSCHMITT, D. G., SZYPERSKI, C., 2003, *Software Ecosystem: Understanding an Indispensable Technology and Industry*. 1. ed., The MIT Press.

MEYERS, B., OBERNDORF, P., 2001, Managing Software Acquisition: Open Systems and COTS Products. Addison Wesley.

MOHAMED, A., RUHE, G., EBERLEIN, A., 2007, "COTS selection: past, present, and future". In.: *Engineering of Computer-Based Systems*, 2007. ECBS'07. 14th Annual IEEE International Conference and Workshops on the (pp. 103-114). IEEE. March.

MOLLES, M. C., 1999, "Ecology: Concepts and Applications." Boston: WCB/McGraw-Hill.

MOORE, J. F., 1996, The Death of Competition: Leadership and Strategy in the Age of Business Ecosystems. Harper Business.

MOORE, G., 2011, "Systems of Engagement and The Future of Enterprise IT. A Sea Change in Enterprise IT", White Paper, available at http://info.aiim.org/systems-of-engagement-and-the-future-of-enterprise-it, retrieved in 12-05-2015.

NCUBE, C., MAIDEN, N. A., 1999, "PORE: Procurement-oriented requirements engineering method for the component-based systems engineering development paradigm." In *International workshop on component-based software engineering* (pp. 130-140), May.

NETO, V. V. G., ARAUJO, R., & DOS SANTOS, R. P., 2017, "New challenges in the social web: Towards systems-of-information systems ecosystems." In *Anais do VIII Workshop sobre Aspectos da Interação Humano-Computador para a Web Social* (pp. 1-12). SBC. December. In Portuguese.

NETO, V. V. G., MANZANO, W., KASSAB, M., NAKAGAWA, E. Y., 2018, "Model-based engineering & simulation of software-intensive systems-of-systems: experience report and lessons learned.", In: *Proceedings of the 12th European Conference on Software Architecture: Companion Proceedings* (p. 27). ACM, September.

NEWMAN, S., 2015, Building Microservices. "O'Reilly Media, Inc.".

NIE, G., SHE, Q., CHEN, D., 2011, "Evaluation index system of cloud service and the purchase decision-making process based on AHP." In *Proceedings of the 2011 International Conference on Informatics, Cybernetics, and Computer Engineering* (ICCE2011) November 19-20, 2011, Melbourne, Australia, pp. 345-352. Springer, Berlin, Heidelberg.

NIST, 2015, “Cloud Computing Service Metrics Description”, available at https://www.nist.gov/sites/default/files/documents/itl/cloud/RATAX-CloudServiceMetricsDescription-DRAFT-20141111.pdf, retrieved in 09-01-2017.

NIZAMANI, S. A., 2012, “A quality-aware cloud selection service for computational modellers”. University of Leeds.

NUNES, V.T., 2014, “Dynamic Process Adaptation: Planning in a Context-Aware Approach”. PhD Thesis in Computer Science and System Engineering. COPPE – Federal University of Rio de Janeiro, Rio de Janeiro, Brazil, 133p.

NYGARD, M., 2011, “Documenting Architecture Decisions”, available at http://thinkrelevance.com/blog/2011/11/15/documenting-architecture-decisions accessed in 05-25-2019.

O’GRADY, S, 2017, “The Difference Between SOA and Microservices Isn’t Size”, available at http://redmonk.com/sogrady/2017/07/20/soa-microservices/, retrieved in 09-05-2017, July.

OLIVEIRA, M.S., 2011, “Previa: An Approach for Visualizing the Evolution of Software Models”. Master Thesis in Computer Science and System Engineering. COPPE – Federal University of Rio de Janeiro, Rio de Janeiro, Brazil, 185p. (In Portuguese)

OPEN GROUP, 2011, “SOA Reference Architecture”, Technical Standard, available at http://www.opengroup.org/soa/source-book/soa_refarch/, accessed in 12-12-2016.

ORACLE, 2011, “The top 10 technical considerations for evaluating e-commerce platforms”, White paper. March 2011.

ORACLE, 2014a, “Cloud, Analytics, Mobile, And Social: Convergence Will Bring Even More Disruption”, Forbes.com, available at http://www.forbes.com/sites/oracle/2014/05/06/cloud-analytics-mobile-and-social-convergence-will-bring-even-more-disruption/, retrieved in 07-25-2016.

ORACLE, 2014b, “Information Management and Big Data A Reference Architecture”, Oracle White Paper, available at https://www.oracle.com/technetwork/database/bigdata-appliance/overview/bigdatarefarchitecture-2297765.pdf, accessed in 05-25-2019.

ORACLE, 2016, “Where Is SOA Going?”, Oracle Magazine, January/February 2016, COMMUNITY: Architect, Bob Rhubart, available at http://www.oracle.com/technetwork/issue-archive/2016/16-jan/o16architect-2882207.html, retrieved in 07/18/2015.

PAHL, C., JAMSHIDI, P., 2016, “Microservices: a systematic mapping study.”, In: *Proceedings of the 6th International Conference on Cloud Computing and Services Science*, Volume 1, Rome, Italy, April 23–25, pp. 137–146.

PAN, Y., 2016, “Heading toward artificial intelligence 2.0”, Engineering, ELSEVIER, v. 2, n. 4, p. 409-413.

PETERSEN, K., FELDT, R., MUJTABA, S., MATTSSON, M., 2008, “Systematic mapping studies in software engineering”. In: *Proceedings of the 12th international conference on evaluation and assessment in software engineering* (Vol. 17, No. 1), Swinton, UK, pp. 68-77.

PLAMONDON, J., 2012, “Introducing the API Economy: A Dialogue”. Cutter Consortium, Executive Report, available at: https://www.cutter.com/article/introducing-api-economy-dialogue-425066, retrieved in 08-13-2015.

POLANČIČ, G., HERIČKO, M., ROZMAN, I., 2010, “An Empirical Examination of Application Frameworks Success Based on Technology Acceptance Model”. *The Journal of Systems and Software* 83(4):574-584.

PONS, E., BRAUN, L. M., HUNINK, M. M., KORS, J. A., 2016, “Natural language processing in radiology: a systematic review”. Radiology, 279(2), 329-343.

PRIETO-DIAZ, R., 1991, “Implementing faceted classification for software reuse”. Communications of the ACM, 34(5), pp. 88-97.

ROLLAND, C., 1999, “A goal driven approach to modelling COTS acquisition impacts.” In Entreprise Management and Resource Planning Systems (p. 1).

RESNICK, P., VARIAN, H. R., 1997, “Recommender systems”. Communications of the ACM, 40(3), pp. 56-58.

SAATY, T., 1980, The Analytic Hierarchy Process, Planning, Priority Setting, Resource Allocation. McGraw-Hill, New York.

SAATY, T. L., 1990, “How to make a decision: the analytic hierarchy process.”, European journal of operational research, v. 48, n. 1, p. 9-26.

SAATY, T. L., 2000, “Fundamentals of decision making and priority theory with the analytic hierarchy process”, (Vol. 6). RWS publications.

SALEH, M., ABEL, M. H., 2015, Information Systems: Towards a System of Information Systems. In *Proceedings of the KMIS 2015 7th International Conference on Knowledge Management and Information Sharing*, Lisbonne, Portugal, vol. 3, pp. 193-200.

SALMINEN, A., MIKKONEN, T., 2012, "Mashups – Software ecosystems for the web era", In *IWSECO@ ICSOB International Workshop on Software Ecosystems*, pp. 18-32. Boston, USA. June.

SANTO, R.E., 2012, "Services to Support Planning of Systematic Literature Reviews". Master Thesis in Computer Science and System Engineering. COPPE – Federal University of Rio de Janeiro, Rio de Janeiro, Brazil, 155p. (In Portuguese)

SANTOS, R. P., 2016, "Managing and monitoring software ecosystem to support demand and solution analysis" (Doctoral dissertation, Ph. D. Dissertation. Universidade Federal do Rio de Janeiro).

SANTOS, R. P., WERNER, C. M. L., 2010, "Analyzing the Concept of Components in the Brechó-VCM Approach through a Sociotechnical and Software Reuse Management Perspective". In: *Proceedings of the Fourth Brazilian Symposium on Software Components, Architectures and Reuse*, pp. 21-30, Salvador, BA, Brazil. September.

SANTOS, R.P., WERNER, C.M.L., 2011, "Treating Business Dimension in Software Ecosystems". In: Proceedings of the 3rd ACM/IFIP International Conference on Management of Emergent Digital EcoSystems (MEDES), pp. 197-201, San Francisco, USA, November.

SANTOS, R.P., WERNER, C.M.L., BARBOSA, O.A.L.P., ALVES, C.F., 2012, "Software Ecosystems: Trends and Impacts on Software Engineering". In: *Proceedings of the XXVI Brazilian Symposium on Software Engineering* (SBES), Special Track "Grand Challenges in Software/System Engineering", pp. 206-210, Natal, Brazil, September.

SANTOS, R., GONÇALVES, M., NAKAGAWA, E. Y., WERNER, C., 2014, "On the relations between systems-of-systems and software ecosystems". In: *Proceedings of the V Brazilian Congress on Software: Theory and Practice (CBSoft 2014) WDES*, pp. 58-62. Maceió, Alagoas, Brazil. September-October.

SCHWARTZ, B., 2004, "The paradox of choice: Why more is less.", New York: Ecco, January.

SEI, 2019, "Software Product Lines Collection", Software Engineering Institute, available at https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=513819, retrieved in May 20th, 2019.

SEN, A., 1997, "The role of opportunism in the software design reuse process". IEEE Transactions on Software Engineering, 23(7), 418-436.

SEREBRENIK, A., MENS, T., 2015, "Challenges in Software Ecosystems Research", In: *Proceedings of the 2015 European Conference on Software Architecture Workshops (ECSA 2015)*, p. 40. Dubrovnik /Cavtat, Croatia. September

SIEGEL, J., PERDUE, J., 2012, "Cloud services measures for global use: the Service Measurement Index (SMI)". In *SRII Global Conference (SRII)*, 2012 Annual (pp. 411-415). IEEE. July.

SHAW, M., 2003, "Writing good software engineering research papers: mini-tutorial". In: *Proceedings of the 25th International Conference on Software Engineering*, ICSE '03. IEEE Computer Society, Washington, DC, USA, pp. 726-736.

SHAW, M., GARLAN, D., 1996, "Software Architecture: Perspective on an Emerging Discipline". Prentice Hall.

SHULL, F., CARVER, J., TRAVASSOS, G.H., 2001, "An Empirical Methodology for Introducing Software Processes". *SIGSOFT Software Engineering Notes* 26(5):288-296.

SINCERO, S. M., 2012, "Types of Survey", available at https://explorable.com/types-of-survey, retrieved in 07-01-2019.

STOLTERMAN, E., FORS, A. C., 2004, "Information Technology and the Good Life", in: *Information Syst. Research: relevant theory and informed practice*, ISBN 1-4020-8094-8, p. 689.

SUBRAMANIAN, N., CHUNG, L., 2001, "Software architecture adaptability: an NFR approach". In: *Proceedings of the 4th International Workshop on Principles of Software Evolution*, pp. 52-61. ACM. September.

SUN, L., DONG, H., HUSSAIN, F. K., HUSSAIN, O. K., CHANG, E., 2014, "Cloud service selection: State-of-the-art and future research directions". Journal of Network and Computer Applications, 45, 134-150.

TAYLOR, R. N., MEDVIDOVIC, N., DASHOFY, E. M., 2010, "Software Architecture – Foundations, Theory, and Practice". Wiley.

TOMICIC-PUPEK, K., DOBROVIC, Z., TOMICIC FURJAN, M., 2012, "Strategies for Information Systems Integration." In *Proceedings of the ITI 2012 34th International Conference on Information Technology Interfaces*, Cavtat/Dubrovnik, Croatia, pp. 311-316.

TRAVASSOS, G. H., GUROV, D., AMARAL, E. A. G., 2002, "Introduction to Experimental Software Engineering". Technical Report PESC/COPPE – ES-590/02, Federal University of Rio de Janeiro.

TRIANTAPHYLLOU, E., 2000, “Multi-criteria decision making methods. In Multi-criteria decision making methods: A comparative study”, pp. 5-21. Springer US.

TRIVEDI, H., MESTERHAZY, J., LAGUNA, B., VU, T., SOHN, J. H., 2018, “Automatic determination of the need for intravenous contrast in musculoskeletal MRI examinations using IBM Watson’s natural language processing algorithm.” Journal of digital imaging, 31(2), 245-251.

VASCONCELOS, A.P.V., 2007, “An Approach to Support the Creation of Domain Reference Architectures Based on Legacy Systems Analysis”. PhD Thesis in Computer Science and System Engineering. COPPE – Federal University of Rio de Janeiro, Rio de Janeiro, Brazil, 267p. (In Portuguese)

VILLAMIZAR, M., GARCÉS, O., CASTRO, H., VERANO, M., SALAMANCA, L., CASALLAS, R., GIL, S., 2015. “Evaluating the monolithic and the microservice architecture pattern to deploy web applications in the cloud.” In: *Computing Colombian Conference* (10CCC), 2015 10th. IEEE, pp. 583–590.

WALLNAU, K., HISSAM, S., SEACORD, R., 2002, Building Systems from Commerical Components. Addison-Wesley Longman Publishing.

WEBER, C., ARAÚJO, E. E. R. de, SCALET, D., ANDRADE, E. L. P. de, ROCHA, A. R. C. da, MONTONI, M. A., 2007, “MPS model-based software acquisition process improvement in Brazil”. In: *Proceedings of the 6th International Conference on the Quality of Information and Communications Technology* (QUATIC 2007), pp. 110–119, Lisbon, Portugal, September.

WERNER, C.M.L., 1992, *Reutilização de Software no Desenvolvimento de Software Científico.* Tese de D.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil. (In Portuguese)

WHITMORE, P., 2001, “The Psychology of e-Commerce: Choice Overload and other Paradoxes”, Paul Whitmore of E*trade Group – Multi-University/Research Laboratory Seminar Series, available at http://www.informationdesign.org/the_psychology_of_e-commerce_c/, accessed in 05-25-2019.

WÖHLIN, C., RUNESON, P., HÖST, M., 1999, *Experimentation in Software Engineering: An Introduction.* 1. Ed. Berlin: Springer.

YAHIA, E. B. H., RÉVEILLÈRE, L., BROMBERG, Y.-D., CHEVALIER, R., CADOT, A., 2016, “Medley: an event-driven lightweight platform for service composition.”, In: *International Conference on Web Engineering.* Springer, pp. 3–20.

Annex A1 – Evaluation Instruments

A1.1 Informed Consent Form (In Portuguese)

Investigação sobre Seleção de Microsserviços
Termo de Consentimento Livre Esclarecido

OBJETIVO DO ESTUDO
Este estudo visa realizar uma investigação sobre seleção de microsserviços.

IDADE
Eu declaro ter mais de 18 (dezoito) anos de idade e concordar em participar de um estudo conduzido por Marcelo de França Costa da COPPE/UFRJ, sob a orientação da Profa. Cláudia Maria Lima Werner.

CONFIDENCIALIDADE
Eu estou ciente de que meu nome não será divulgado em hipótese alguma. Também estou ciente de que os dados obtidos por meio deste estudo serão mantidos sob confidencialidade, e os resultados serão posteriormente apresentados de forma agregada, de modo que um participante não seja associado a um dado específico.
Da mesma forma, me comprometo a não comunicar meus resultados enquanto o estudo não for concluído, bem como manter sigilo das técnicas e documentos apresentados e que fazem parte do experimento.

BENEFÍCIOS E LIBERDADE DE DESISTÊNCIA
Eu entendo que, uma vez o experimento tenha terminado, os trabalhos que desenvolvi serão estudados visando entender a eficiência dos procedimentos e as técnicas que me foram ensinadas.
Os benefícios que receberei deste estudo são limitados ao aprendizado do material que é distribuído e ensinado. Também entendo que sou livre para realizar perguntas a qualquer momento, solicitar que qualquer informação relacionada à minha pessoa não seja incluída no estudo ou comunicar minha desistência de participação, sem qualquer penalidade. Por fim, declaro que participo de livre e espontânea vontade com o único intuito de contribuir para o avanço e desenvolvimento de técnicas e processos para a Engenharia de Software.

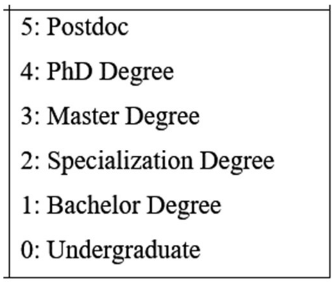
PESQUISADOR RESPONSÁVEL
Marcelo de França Costa (mafranca@cos.ufrj.br)
Programa de Engenharia de Sistemas e Computação - COPPE/UFRJ

PROFESSORA RESPONSÁVEL
Profa. Cláudia Maria Lima Werner (werner@cos.ufrj.br)
Programa de Engenharia de Sistemas e Computação - COPPE/UFRJ

Data, nome do participante e rubrica

A1.2 Characterization Form

Name:	
e-mail:	
Academic Background:	(1-5) See table:
Years of Experience with IT/Software sector:	Years
Years of Experience with IT/Software architecture:	Years
Do you have experience with similar tools (recommendation systems)?	(Yes/No)


5: Postdoc
4: PhD Degree
3: Master Degree
2: Specialization Degree
1: Bachelor Degree
0: Undergraduate

A1.3 Execution Form

DIRECTOR - Evaluation Tasks

Hint: [Ctrl + L]

Obs.: You can use anything, except another recommendation system.

++++++++++
+ (1st part) +
++++++++++

------- --------- --------

==> PLEASE, take note of your start time:

Task #	Description	Answer	Justification/Rationale	Confidence (1-5)
1	*What are the currently active registered PaaS?*			
2	*How many microservices are available in the "SAP Cloud Platform" PaaS?*			

3	*What (up to three) microservices would be adequate for scenario #1?*			
4	*What (up to three) microservices would be adequate for scenario #2?*			
5	*What (up to three) microservices would be adequate for scenario #3?*			
6	*What (up to three) microservices would be adequate for scenario #4?*			
7	*What (up to three) microservices would be adequate for scenario #5?*			

8	*With regards to scenario #5, what an architectural decision document would look like (describe it in one to three paragraphs, i.e., the considered alternatives, the chosen one, and the tradeoffs plus the rationale for that)?*			

==> PLEASE, take note of your end time:

++++++++++
+ (2nd part) +
++++++++++

Additional TAM Questions (Feedback)

-------- ---------- ---

Please, pick one of the aforementioned scenarios and go to:

(In Portuguese) http://director-services.mybluemix.net/

a) Utilize a seção "1) (Discovery) Are you searching for a cloud microservice?" para informar a lista de features desejadas e tags a serem excluídas (atenção para separar termos com vírgula). Clique no botão "Show Me Some Candidates". Reflita sobre a resposta;

b) Copie todos os "serviceId" dos microsserviços candidatos encontrados (confirme o total) - atenção para compor uma lista separada por vírgula. Utilize a seção "2) Technical Perspective" para informar a lista de candidatos. Clique no botão "Rank These Candidates". Reflita sobre a resposta;

c) Repita o processo com a seção "3) Social Perspective". Reflita sobre a resposta;

d) Digite a frase (resumo) do cenário escolhido na seção "4) Semantical Perspective", no campo "Brief description (free-text):". Clique no botão "Rank These Candidates". Reflita sobre a resposta;

We need a microservice (Scenarios):

....

#1: ***I need to store and view logs from my application.***

with one of the following features: kibana, splunk, elk, log analytics, logging, logs, Operations Support

without the tags: ibm_deprecated

....

#2 ***I need to store relational data in a cloud database as a service.***

with one of the following features: sqldb, db2 on cloud, db2cloud, mysql, oracle, db2 cloud, managed-relational, relational, db2hosted, db2, db2oncloud, sql, postgresql

without the tags: ibm_deprecated

....

#3 ***I need to set up and manage my connected devices in an IoT solution.***

with one of the following features: internet_of_things, Internet of Things, (S) Internet of Things, (P) Internet of Things

without the tags: ibm_deprecated

....

#4 ***I need to provision a rabbit mq cluster to allow asynchronous messaging.***

with one of the following features: Messaging and Queuing, amqp, rabbitmq, message-queue

without the tags: ibm_deprecated

....

#5 ***I need to store json documents in a nosql database.***

with one of the following features: nosql, mongodb, Data Stores, Data Store

without the tags: ibm_deprecated, caching

Registered PaaS

id	Name	Description	Endpoint	Status	Days Being Monitored
11	IBM Cloud	Former IBM Bluemix, it is a cloud platform as a service developed by IBM. It supports several programming languages and services as well as integrated DevOps to build, run, deploy and manage applications on the cloud.	https://api.ng.bluemix.net	A	366
21	Pivotal Cloud Foundry	Pivotal Cloud Foundry is engineered to deliver a single software platform to run an entire enterprise?capable of scaling to support hundreds of IT teams and thousands of applications.	https://api.run.pivotal.io	A	145
31	Salesforce Heroku	Heroku is a platform as a service (PaaS) that enables developers to build, run, and operate applications entirely in the cloud.	https://api.heroku.com	I	1
51	SAP Cloud Platform	SAP Cloud Platform is the enterprise platform-as-a-service with comprehensive application development services and capabilities.	https://api.cf.us10.hana.ondemand.com	A	143

A1.4 Microservices Catalog

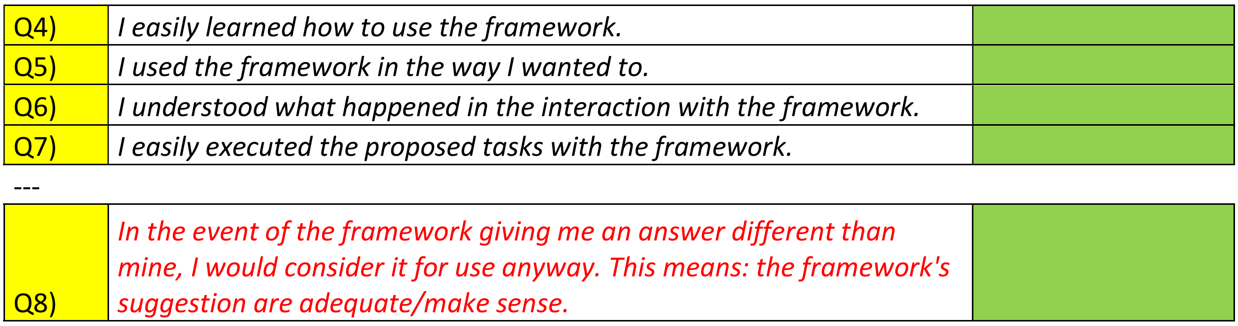
```
{
  "platforms": [
    {
      "id": 11,
      "name": "IBM Cloud",
      "platformsSyncDate": "Jan 26, 2019 12:00:00 AM",
      "servicesSyncDate": "Jan 26, 2019 12:00:00 AM",
      "servicesQuantity": 163,
      "services": [
        {
          "serviceId": 1,
          "serviceGuid": "604e3b5d-f1d1-42a8-978c-d4c5b49c85cd",
          "serviceURL": "/v2/services/604e3b5d-f1d1-42a8-978c-d4c5b49c85cd",
          "serviceName": "elephantsql",
          "serviceDescription": "PostgreSQL as a Service",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Apr 15, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 11,
          "serviceGuid": "9a0d03b3-23f3-4ac7-b931-b50b1c8f2c2a",
          "serviceURL": "/v2/services/9a0d03b3-23f3-4ac7-b931-b50b1c8f2c2a",
          "serviceName": "pubnub-sandbox",
          "serviceDescription": "Data Streaming and Realtime Communication",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "May 7, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 21,
          "serviceGuid": "9dd2e109-7daa-4300-86f0-13df2075b11f",
          "serviceURL": "/v2/services/9dd2e109-7daa-4300-86f0-13df2075b11f",
          "serviceName": "tone_analyzer",
          "serviceDescription": "Tone Analyzer uses linguistic analysis to detect three types of tones from communications: emotion, social, and language. This insight can then be used to drive high impact communications.",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 6, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Jan 8, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 31,
          "serviceGuid": "a344f2e9-6c76-4039-a5b3-f8281b6427fc",
          "serviceURL": "/v2/services/a344f2e9-6c76-4039-a5b3-f8281b6427fc",
          "serviceName": "language_translator",
          "serviceDescription": "Translate text from one language to another for specific
domains.",
          "servicePlansQuantity": 4,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 20, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jan 8, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 41,
          "serviceGuid": "8d34e584-fee0-4021-a8b6-b269389e09b7",
          "serviceURL": "/v2/services/8d34e584-fee0-4021-a8b6-b269389e09b7",
          "serviceName": "Difitek",
          "serviceDescription": "API-first platform for fintech applications",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Apr 10, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 51,
          "serviceGuid": "fc6c8b5e-593a-4c44-a926-e81e71fe9822",
          "serviceURL": "/v2/services/fc6c8b5e-593a-4c44-a926-e81e71fe9822",
          "serviceName": "natural_language_classifier",
          "serviceDescription": "Natural Language Classifier performs natural language
classification on question texts. A user would be able to train their data and the predict
the appropriate class for a input question.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 12, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Jan 7, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 61,
          "serviceGuid": "7f5cf926-b7cb-4985-bc42-7a9e77e0aa01",
          "serviceURL": "/v2/services/7f5cf926-b7cb-4985-bc42-7a9e77e0aa01",
          "serviceName": "kinetise",
          "serviceDescription": "Rapid development of mobile apps. With Native Source
Code.",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Apr 10, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 71,
          "serviceGuid": "2b71dd7a-4af3-450a-8e04-88db1fc04a5b",
          "serviceURL": "/v2/services/2b71dd7a-4af3-450a-8e04-88db1fc04a5b",
          "serviceName": "box",
          "serviceDescription": "Powering Content and data for your application. Whether
you are building a line of business app, content management software or need to display
content beautifully on web and mobile, the Box API can ",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Apr 4, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 81,
          "serviceGuid": "9218abab-c6ba-4e40-ab42-cb971d6d1835",
          "serviceURL": "/v2/services/9218abab-c6ba-4e40-ab42-cb971d6d1835",
          "serviceName": "sql-query",
          "serviceDescription": "Analyze data in Object Storage with ANSI SQL.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jan 17, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Aug 15, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 91,
          "serviceGuid": "ba720d65-b425-4773-bedc-74aa770ac866",
          "serviceURL": "/v2/services/ba720d65-b425-4773-bedc-74aa770ac866",
          "serviceName": "datacatalog",
          "serviceDescription": "Discover, catalog, and securely share enterprise data.",
          "servicePlansQuantity": 4,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 8, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jan 25, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 101,
          "serviceGuid": "e4d429bb-58fd-4384-ac77-726c0ccc3d5c",
          "serviceURL": "/v2/services/e4d429bb-58fd-4384-ac77-726c0ccc3d5c",
          "serviceName": "Phunware Mobile Marketing Automation",
          "serviceDescription": "Phunware Mobile Marketing Automation",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 1, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 111,
          "serviceGuid": "f1ee10bb-e290-412a-9297-9a4507d523ab",
          "serviceURL": "/v2/services/f1ee10bb-e290-412a-9297-9a4507d523ab",
          "serviceName": "ecs-checker",
          "serviceDescription": "Automate accessibility verification of HTML and EPUB
documents.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 16, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Apr 20, 2016 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 121,
          "serviceGuid": "289dab26-3200-4140-b405-725ba4ed2e5f",
          "serviceURL": "/v2/services/289dab26-3200-4140-b405-725ba4ed2e5f",
          "serviceName": "rediscloud",
          "serviceDescription": "Enterprise-Class Redis for Developers",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Apr 23, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 131,
          "serviceGuid": "2b0b3df2-ad4e-4fb3-b1fd-d0f3f7ddc821",
          "serviceURL": "/v2/services/2b0b3df2-ad4e-4fb3-b1fd-d0f3f7ddc821",
          "serviceName": "cloudcerts",
          "serviceDescription": "Use the Certificate Manager service to manage SSL
certificates.",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Nov 16, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Nov 30, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Dec 1, 2018 12:00:00 AM"
        },
        {
          "serviceId": 141,
          "serviceGuid": "065595fd-c7bb-4cbb-8a19-58a4e319098f",
          "serviceURL": "/v2/services/065595fd-c7bb-4cbb-8a19-58a4e319098f",
          "serviceName": "VoiceAgent",
          "serviceDescription": "Create a cognitive voice agent that uses Watson services
to speak directly with customers using natural language over the telephone",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Aug 23, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Dec 7, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 151,
          "serviceGuid": "fe959ac5-aa47-43a6-9c58-6fc265ee9b0e",
          "serviceURL": "/v2/services/fe959ac5-aa47-43a6-9c58-6fc265ee9b0e",
          "serviceName": "messagehub",
          "serviceDescription": "IBM Event Streams is a high-throughput message bus built
with Apache Kafka. It is optimized for event ingestion into IBM Cloud and event stream
distribution between your services and applications.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 27, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Sep 19, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 161,
          "serviceGuid": "3bf645a1-4cb5-4b79-983e-ecb3fe7c2c42",
          "serviceURL": "/v2/services/3bf645a1-4cb5-4b79-983e-ecb3fe7c2c42",
          "serviceName": "postgresql-replaced",
          "serviceDescription": "This service is no longer available. Please search for
Compose services in the main catalog instead.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jun 6, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jun 6, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 171,
          "serviceGuid": "5296dc86-e2a8-47d0-b4b3-a7901254bcff",
          "serviceURL": "/v2/services/5296dc86-e2a8-47d0-b4b3-a7901254bcff",
          "serviceName": "mqcloud",
          "serviceDescription": "Enterprise-grade messaging hosted in the cloud",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Nov 28, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jun 5, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 181,
          "serviceGuid": "d343399c-9016-4598-994f-77667d20c5c3",
          "serviceURL": "/v2/services/d343399c-9016-4598-994f-77667d20c5c3",
          "serviceName": "Accrete.AI: Rational Exuberance",
          "serviceDescription": "Tackle information overload in confusing Fed speak",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 8, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 191,
          "serviceGuid": "5aaa2533-8d33-4cde-be0f-1ddd4c69b7e2",
          "serviceURL": "/v2/services/5aaa2533-8d33-4cde-be0f-1ddd4c69b7e2",
          "serviceName": "namara-catalog",
          "serviceDescription": "Open Data. Clean and simple.",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Mar 27, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 201,
          "serviceGuid": "aa206aa9-1c49-499b-86ad-add09f73fabd",
          "serviceURL": "/v2/services/aa206aa9-1c49-499b-86ad-add09f73fabd",
          "serviceName": "imfpush",
          "serviceDescription": "Scalable and reliable Push Notifications service for
mobile and web applications",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 12, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Dec 10, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 211,
          "serviceGuid": "4a502300-6a0a-4e62-aadf-d05d3ce36833",
          "serviceURL": "/v2/services/4a502300-6a0a-4e62-aadf-d05d3ce36833",
          "serviceName": "InfluxCloud",
          "serviceDescription": "A modern time series data platform for metrics & events",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Aug 9, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 221,
          "serviceGuid": "83e32d49-5203-40f8-a0fd-6f6eff0beaa7",
          "serviceURL": "/v2/services/83e32d49-5203-40f8-a0fd-6f6eff0beaa7",
          "serviceName": "Monitoring",
          "serviceDescription": "Collect, store, and analyze metrics from your dynamic
cloud environments and micro-service applications.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 14, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Nov 17, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 231,
          "serviceGuid": "7730ed47-4e36-46a0-8313-698869dfc354",
          "serviceURL": "/v2/services/7730ed47-4e36-46a0-8313-698869dfc354",
          "serviceName": "Envestnet | Yodlee",
          "serviceDescription": "APIs for Financial Data Aggregation",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Apr 13, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 241,
          "serviceGuid": "8a0c35ba-a8e2-4b84-9a13-3706e46bd2f9",
          "serviceURL": "/v2/services/8a0c35ba-a8e2-4b84-9a13-3706e46bd2f9",
          "serviceName": "apersona-amfa",
          "serviceDescription": "Frictionless Adaptive Multi-Factor Authentication",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Feb 17, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 251,
          "serviceGuid": "a56821ea-3f5e-41cb-ab4f-6ea4e18d4c99",
          "serviceURL": "/v2/services/a56821ea-3f5e-41cb-ab4f-6ea4e18d4c99",
          "serviceName": "hiptest",
          "serviceDescription": "The most simple and powerful test management platform",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Feb 26, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 261,
          "serviceGuid": "18cfd285-46c2-457b-bfba-7e469e1bddf9",
          "serviceURL": "/v2/services/18cfd285-46c2-457b-bfba-7e469e1bddf9",
          "serviceName": "AvailabilityMonitoring",
          "serviceDescription": "Around the world, around the clock availability and
performance monitoring.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 9, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Aug 21, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 271,
          "serviceGuid": "79e12717-d0a0-46ee-aa43-26303555a79c",
          "serviceURL": "/v2/services/79e12717-d0a0-46ee-aa43-26303555a79c",
          "serviceName": "conversation",
          "serviceDescription": "Add a natural language interface to your application to
automate interactions with your end users. Common applications include virtual agents and
chat bots that can integrate and communicate on any ch",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 18, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jan 4, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 281,
          "serviceGuid": "0e314aea-d2fb-41b9-9eac-bccb2e0f9595",
          "serviceURL": "/v2/services/0e314aea-d2fb-41b9-9eac-bccb2e0f9595",
          "serviceName": "compose-for-redis",
          "serviceDescription": "Redis is an open-source, blazingly fast, low maintenance
key/value store.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 6, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Dec 8, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 291,
          "serviceGuid": "ee8b3275-190a-438b-a768-4eca4136b8ac",
          "serviceURL": "/v2/services/ee8b3275-190a-438b-a768-4eca4136b8ac",
          "serviceName": "Mapbox Maps",
          "serviceDescription": "Add powerful custom maps to your app",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Oct 12, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 301,
          "serviceGuid": "7732ee9c-0570-43cc-a61f-75d303b57c05",
          "serviceURL": "/v2/services/7732ee9c-0570-43cc-a61f-75d303b57c05",
          "serviceName": "cloudamqp",
          "serviceDescription": "Managed HA RabbitMQ servers in the cloud",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Apr 15, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 311,
          "serviceGuid": "6707f079-383e-43a9-b97e-402fe9e6b4a6",
          "serviceURL": "/v2/services/6707f079-383e-43a9-b97e-402fe9e6b4a6",
          "serviceName": "attm2x",
          "serviceDescription": "Time Series IoT Data Service",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Aug 16, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 321,
          "serviceGuid": "b3ac3c8e-b30d-4497-bf2e-27cee33fa077",
          "serviceURL": "/v2/services/b3ac3c8e-b30d-4497-bf2e-27cee33fa077",
          "serviceName": "statica",
          "serviceDescription": "Enterprise Static IP Addresses",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Feb 4, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 331,
          "serviceGuid": "16a6054a-83f2-4251-a221-ab8853e1f18c",
          "serviceURL": "/v2/services/16a6054a-83f2-4251-a221-ab8853e1f18c",
          "serviceName": "compose-enterprise",
          "serviceDescription": "IBM Compose Enterprise is a service which provides a
private isolated cluster for IBM Cloud users to optionally provision their Compose
databases into.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 26, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Dec 8, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 341,
          "serviceGuid": "bed0b74d-6d3e-47b4-ade5-b1407a5b1795",
          "serviceURL": "/v2/services/bed0b74d-6d3e-47b4-ade5-b1407a5b1795",
          "serviceName": "blazemeter",
          "serviceDescription": "Performance Testing Platform",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Apr 15, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 351,
          "serviceGuid": "53299077-26d6-4ac2-b07e-eba9417fe275",
          "serviceURL": "/v2/services/53299077-26d6-4ac2-b07e-eba9417fe275",
          "serviceName": "Twilio Programmable Voice",
          "serviceDescription": "Build calling experiences with Twilio?s Voice API.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 25, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 361,
          "serviceGuid": "b45d95a2-92ac-480a-a11b-23c5242f3d12",
          "serviceURL": "/v2/services/b45d95a2-92ac-480a-a11b-23c5242f3d12",
          "serviceName": "mobile-analytics_Prod",
          "serviceDescription": "Mobile app developers and business stakeholders: Use IBM
Mobile Analytics to gain insight into how your app is performing and how it is being
used.",
          "servicePlansQuantity": 3,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Apr 29, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Aug 2, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Aug 25, 2018 12:00:00 AM"
        },
        {
          "serviceId": 371,
          "serviceGuid": "a004fb83-d76f-4fef-a33a-f9af86e148a4",
          "serviceURL": "/v2/services/a004fb83-d76f-4fef-a33a-f9af86e148a4",
          "serviceName": "IBMAnalyticsEngine",
          "serviceDescription": "Flexible framework to deploy Hadoop and Spark analytics
applications.",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 17, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Sep 14, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 381,
          "serviceGuid": "9e0fc976-9adc-4aea-ae0f-b6e37e6dccdc",
          "serviceURL": "/v2/services/9e0fc976-9adc-4aea-ae0f-b6e37e6dccdc",
          "serviceName": "compose-for-scylladb",
          "serviceDescription": "ScyllaDB is a highly performant, in-place replacement for
the Cassandra wide-column distributed database.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 15, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Dec 8, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 391,
          "serviceGuid": "8ce13841-6e8e-45bf-9e09-deac89c08d83",
          "serviceURL": "/v2/services/8ce13841-6e8e-45bf-9e09-deac89c08d83",
          "serviceName": "Accrete.AI: Topic Deltas",
          "serviceDescription": "Analyzes earnings to find delta in topic sentiment",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 8, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 401,
          "serviceGuid": "d1446769-bbea-4716-927a-b8b3c6ec749d",
          "serviceURL": "/v2/services/d1446769-bbea-4716-927a-b8b3c6ec749d",
          "serviceName": "fss-financial-optimization-service",
          "serviceDescription": "Construct or rebalance investment portfolios based on
investor goals, mandates, and preferences.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 27, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Aug 14, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 411,
          "serviceGuid": "ed46bd78-96ee-40cc-9242-86abfd678b48",
          "serviceURL": "/v2/services/ed46bd78-96ee-40cc-9242-86abfd678b48",
          "serviceName": "push-reappt",
          "serviceDescription": "Real Time Data Distribution Service",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Jun 29, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 421,
          "serviceGuid": "cd9be175-a5a4-403a-aab8-e4a0ff5e48db",
          "serviceURL": "/v2/services/cd9be175-a5a4-403a-aab8-e4a0ff5e48db",
          "serviceName": "Zuznow",
          "serviceDescription": "Automatically develop mobile apps",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 25, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 431,
          "serviceGuid": "b24049b1-a518-4b02-9f0d-98a4b7adb7ba",
          "serviceURL": "/v2/services/b24049b1-a518-4b02-9f0d-98a4b7adb7ba",
          "serviceName": "simplicite",
          "serviceDescription": "Versatile Cloud Platform for Enterprise Applications",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Dec 17, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 441,
          "serviceGuid": "c9b97ed0-2b3a-45b3-b3a6-2765158ba284",
          "serviceURL": "/v2/services/c9b97ed0-2b3a-45b3-b3a6-2765158ba284",
          "serviceName": "docplexcloud",
          "serviceDescription": "Develop optimization applications, such as planning or
scheduling, using our APIs to connect to the CPLEX optimization engines.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Oct 26, 2016 12:00:00 AM",
          "serviceUpdatedAt": "May 23, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 451,
          "serviceGuid": "c997d729-21cb-4669-9410-f77cef3ee3f4",
          "serviceURL": "/v2/services/c997d729-21cb-4669-9410-f77cef3ee3f4",
          "serviceName": "fss-predictive-scenario-analytics-service",
          "serviceDescription": "Create conditional scenarios to model how, given a change
to a subset of factors the broader set of market factors are expected to change.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 17, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Aug 14, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 461,
          "serviceGuid": "805f39f7-d815-4059-b9a6-9ea587b0bb87",
          "serviceURL": "/v2/services/805f39f7-d815-4059-b9a6-9ea587b0bb87",
          "serviceName": "xpertrule-node-red",
          "serviceDescription": "Decision Author for node-RED",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Jul 21, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 471,
          "serviceGuid": "dd7c0c93-a8a0-4625-ba88-3ced8a8c09f2",
          "serviceURL": "/v2/services/dd7c0c93-a8a0-4625-ba88-3ced8a8c09f2",
          "serviceName": "iotforautomotive",
          "serviceDescription": "IoT for Automotive provides automotive domain specialized
application development enablers for data acquisition, storage, real-time processing, and
business rules support.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jun 20, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Feb 2, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 481,
          "serviceGuid": "ad6e056f-c4a8-409f-86da-607003ee57d2",
          "serviceURL": "/v2/services/ad6e056f-c4a8-409f-86da-607003ee57d2",
          "serviceName": "AT&T Flow Designer",
          "serviceDescription": "Design, Build and Deploy IoT Solutions in Minutes",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jan 3, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 491,
          "serviceGuid": "aefbfa7f-93e0-46ad-ac29-ed23981b8c8d",
          "serviceURL": "/v2/services/aefbfa7f-93e0-46ad-ac29-ed23981b8c8d",
          "serviceName": "newrelic",
          "serviceDescription": "Manage and monitor your apps",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Apr 18, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 501,
          "serviceGuid": "76ce47a9-32f9-481c-a2e6-c7d54b83313f",
          "serviceURL": "/v2/services/76ce47a9-32f9-481c-a2e6-c7d54b83313f",
          "serviceName": "alertnotification",
          "serviceDescription": "Never miss critical alerts. Notify the right people
immediately. Speed up response with automated escalation policies.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Nov 25, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Jul 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 511,
          "serviceGuid": "824eb9fc-8033-443c-b8b7-ff698e68fad2",
          "serviceURL": "/v2/services/824eb9fc-8033-443c-b8b7-ff698e68fad2",
          "serviceName": "Twilio Verify",
          "serviceDescription": "Check phone numbers, reduce fraud, increase trust.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 25, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 521,
          "serviceGuid": "031d2a70-6361-49cc-8441-259aac83d797",
          "serviceURL": "/v2/services/031d2a70-6361-49cc-8441-259aac83d797",
          "serviceName": "AppID",
          "serviceDescription": "User Authentication and User Profiles for your apps.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 12, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Jan 20, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 531,
          "serviceGuid": "96b12c35-f9f0-4e5c-b763-b5a4efe740f0",
          "serviceURL": "/v2/services/96b12c35-f9f0-4e5c-b763-b5a4efe740f0",
          "serviceName": "informix_on_cloud",
          "serviceDescription": "IBM Informix on Cloud helps businesses gain a trusted view of data in a hybrid computing environment.",
          "servicePlansQuantity": 8,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 7, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Oct 24, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 541,
          "serviceGuid": "4aa839aa-f5e0-45ce-9d14-c1f34b70a434",
          "serviceURL": "/v2/services/4aa839aa-f5e0-45ce-9d14-c1f34b70a434",
          "serviceName": "compose-for-janusgraph",
          "serviceDescription": "JanusGraph is a scalable graph database optimized for storing and querying highly-interconnected data",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 10, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Dec 8, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 551,
          "serviceGuid": "c44e8afb-5821-4880-b137-52d11f62f06b",
          "serviceURL": "/v2/services/c44e8afb-5821-4880-b137-52d11f62f06b",
          "serviceName": "PagerDuty",
          "serviceDescription": "Incident Management and Resolution Platform",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Apr 21, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 561,
          "serviceGuid": "8d0d0719-09a0-4e27-b34a-94732de68706",
          "serviceURL": "/v2/services/8d0d0719-09a0-4e27-b34a-94732de68706",
          "serviceName": "cloud-object-storage",
          "serviceDescription": "Provides flexible, cost-effective, and scalable cloud
storage for unstructured data.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 18, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Dec 19, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 571,
          "serviceGuid": "8e3a9040-7ce8-4022-a36b-47f836d2b83e",
          "serviceURL": "/v2/services/8e3a9040-7ce8-4022-a36b-47f836d2b83e",
          "serviceName": "iotf-service",
          "serviceDescription": "This service is the hub of all things IBM IoT, it is
where you can set up and manage your connected devices so that your apps can access their
live and historical data.",
          "servicePlansQuantity": 4,
          "serviceStatus": "active",
          "serviceCreatedAt": "Oct 21, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Nov 28, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 581,
          "serviceGuid": "fa452c7f-8e57-453e-92f9-ab293107ba09",
          "serviceURL": "/v2/services/fa452c7f-8e57-453e-92f9-ab293107ba09",
          "serviceName": "Rocket Mainframe Data",
          "serviceDescription": "Rocket Mainframe Data Service on IBM Cloud provides an
easy way to leverage your mainframe data for new cloud services and mobile apps. Built on
our proven data virtualization technology, this new mai",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Apr 4, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 591,
          "serviceGuid": "54e655b9-4791-4025-8a09-4fa643e91312",
          "serviceURL": "/v2/services/54e655b9-4791-4025-8a09-4fa643e91312",
          "serviceName": "driverinsights",
          "serviceDescription": "IBM Watson IoT Driver Behavior Service lets you analyze
drivers' behavior from vehicle probe data and contextual data.",
          "servicePlansQuantity": 2,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Mar 7, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Feb 2, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Nov 1, 2018 12:00:00 AM"
        },
        {
          "serviceId": 601,
          "serviceGuid": "7ae5567f-1d0b-4b7b-822d-29376d20b6a8",
          "serviceURL": "/v2/services/7ae5567f-1d0b-4b7b-822d-29376d20b6a8",
          "serviceName": "Payeezy",
          "serviceDescription": "Simple, powerful payments",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 24, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 611,
          "serviceGuid": "0dae1106-cd91-4f2b-a5c2-c96da7167d16",
          "serviceURL": "/v2/services/0dae1106-cd91-4f2b-a5c2-c96da7167d16",
          "serviceName": "rabbitmq-replaced",
          "serviceDescription": "This service is no longer available. Please search for
Compose services in the main catalog instead.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jun 6, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jun 6, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 621,
          "serviceGuid": "db7b3818-73ab-44da-b786-67792d152291",
          "serviceURL": "/v2/services/db7b3818-73ab-44da-b786-67792d152291",
          "serviceName": "searchly",
          "serviceDescription": "Search Made Simple. Powered-by Elasticsearch",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Sep 2, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 631,
          "serviceGuid": "3d5b1ca2-df4d-489d-88e5-c2e681844930",
          "serviceURL": "/v2/services/3d5b1ca2-df4d-489d-88e5-c2e681844930",
          "serviceName": "dsoncloud",
          "serviceDescription": "IBM® Information Server on Cloud allows you to rapidly
expand data integration and governance capabilities into the cloud for new or ad hoc
development and testing environments.",
          "servicePlansQuantity": 17,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 6, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Nov 30, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 641,
          "serviceGuid": "29e39fa8-644f-4d7c-a230-43612f84f84e",
          "serviceURL": "/v2/services/29e39fa8-644f-4d7c-a230-43612f84f84e",
          "serviceName": "WorkloadScheduler",
          "serviceDescription": "Automate your tasks to run one time or on recurring
schedules. Far beyond Cron, exploit job scheduling within and outside Bluemix.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 24, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Mar 17, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 651,
          "serviceGuid": "a132a5a9-4eb3-4b8d-bb43-3248b4badf5e",
          "serviceURL": "/v2/services/a132a5a9-4eb3-4b8d-bb43-3248b4badf5e",
          "serviceName": "SecureGateway",
          "serviceDescription": "IBM Secure Gateway for Bluemix enables users to integrate
cloud services with enterprise systems on premises.",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 16, 2015 12:00:00 AM",
          "serviceUpdatedAt": "May 25, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 661,
          "serviceGuid": "669635c2-9163-4f06-80cc-aa909fe6bbb0",
          "serviceURL": "/v2/services/669635c2-9163-4f06-80cc-aa909fe6bbb0",
          "serviceName": "compose-for-rabbitmq",
          "serviceDescription": "RabbitMQ asynchronously handles the messages between your
applications and databases",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 6, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Dec 8, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 671,
          "serviceGuid": "8b9a616e-83f9-4ca4-b917-2010c2bc0a4d",
          "serviceURL": "/v2/services/8b9a616e-83f9-4ca4-b917-2010c2bc0a4d",
          "serviceName": "fss-portfolio-service",
          "serviceDescription": "Maintain a record of your investment portfolios through
time.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Mar 14, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Aug 14, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 681,
          "serviceGuid": "8b1cb70f-7c81-433d-814b-a935561f5801",
          "serviceURL": "/v2/services/8b1cb70f-7c81-433d-814b-a935561f5801",
          "serviceName": "fss-historical-scenario-analytics-service",
          "serviceDescription": "Leverage sophisticated IBM Algorithmics financial models
to price and compute analytics on financial securities for a historical date, under a
scenario.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Mar 14, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Aug 14, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 691,
          "serviceGuid": "f9c4eb2f-c62a-4bbe-b2a5-8f40b09ba01a",
          "serviceURL": "/v2/services/f9c4eb2f-c62a-4bbe-b2a5-8f40b09ba01a",
          "serviceName": "Intelligent Travel API",
          "serviceDescription": "ZUMATA's Artificial Intelligence for personalized hotel
search experience.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 13, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 701,
          "serviceGuid": "3b402ae1-9b33-45d6-b635-1218be88e949",
          "serviceURL": "/v2/services/3b402ae1-9b33-45d6-b635-1218be88e949",
          "serviceName": "moni-ai",
          "serviceDescription": "Virtual Assistant for the IoT",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Dec 3, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 711,
          "serviceGuid": "3ce533f0-5c29-4bfe-bfec-6c44661609aa",
          "serviceURL": "/v2/services/3ce533f0-5c29-4bfe-bfec-6c44661609aa",
          "serviceName": "data-science-experience",
          "serviceDescription": "Embed AI and machine learning into your business. Create custom models using your own data.",
          "servicePlansQuantity": 4,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 24, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Aug 1, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 721,
          "serviceGuid": "5c69a958-74a8-47a4-b9e9-3efd078c8958",
          "serviceURL": "/v2/services/5c69a958-74a8-47a4-b9e9-3efd078c8958",
          "serviceName": "Ylabs",
          "serviceDescription": "Full banking stack with enhanced KYC and real time risk monitoring.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 19, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 731,
          "serviceGuid": "03428381-79c8-4cb0-8211-4ca7b3e6229d",
          "serviceURL": "/v2/services/03428381-79c8-4cb0-8211-4ca7b3e6229d",
          "serviceName": "Accern-API",
          "serviceDescription": "Get the most advanced breaking news technology for your investment strategies.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Mar 10, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 741,
          "serviceGuid": "f4a97023-eb84-47e7-ab96-5d00de51d000",
          "serviceURL": "/v2/services/f4a97023-eb84-47e7-ab96-5d00de51d000",
          "serviceName": "mongodb-replaced",
          "serviceDescription": "This service is no longer available. Please search for Compose services in the main catalog instead.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jun 6, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jun 6, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 751,
          "serviceGuid": "ca925d63-b971-4c42-bbf3-3ae0de867a3d",
          "serviceURL": "/v2/services/ca925d63-b971-4c42-bbf3-3ae0de867a3d",
          "serviceName": "compose-for-etcd",
          "serviceDescription": "etcd is a key/value store for distributed server
configuration management.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 6, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Dec 8, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 761,
          "serviceGuid": "508bd176-fcbb-41ac-adfa-a909b5694552",
          "serviceURL": "/v2/services/508bd176-fcbb-41ac-adfa-a909b5694552",
          "serviceName": "Esri ArcGIS for Developers",
          "serviceDescription": "Bring the power of location to your apps with ArcGIS.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 13, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 771,
          "serviceGuid": "0a609717-5c4f-4044-ba36-0157e769b2b3",
          "serviceURL": "/v2/services/0a609717-5c4f-4044-ba36-0157e769b2b3",
          "serviceName": "TradeIt",
          "serviceDescription": "TradeIt enables developers to link to brokers.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 7, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 781,
          "serviceGuid": "cbea77c4-9297-4c36-8522-6baba6d815d0",
          "serviceURL": "/v2/services/cbea77c4-9297-4c36-8522-6baba6d815d0",
          "serviceName": "spark",
          "serviceDescription": "IBM Analytics for Apache Spark for IBM Cloud.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 6, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Sep 18, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 791,
          "serviceGuid": "d6ad5fe0-d736-400e-a4e6-8b80b1bd397f",
          "serviceURL": "/v2/services/d6ad5fe0-d736-400e-a4e6-8b80b1bd397f",
          "serviceName": "compose-for-rethinkdb",
          "serviceDescription": "RethinkDB is a JSON document based, distributed database with an integrated administration and exploration console.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 6, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Dec 8, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 801,
          "serviceGuid": "ecba8956-c5e2-4496-97ef-756460920f67",
          "serviceURL": "/v2/services/ecba8956-c5e2-4496-97ef-756460920f67",
          "serviceName": "lift-cli",
          "serviceDescription": "Migrate data quickly, easily and securely from your on-premises data source to the IBM Cloud.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 26, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Dec 7, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 811,
          "serviceGuid": "dce48839-4924-48bc-b473-2c7905466dc4",
          "serviceURL": "/v2/services/dce48839-4924-48bc-b473-2c7905466dc4",
          "serviceName": "loadimpact",
          "serviceDescription": "Performance and load testing for DevOps",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Apr 21, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 821,
          "serviceGuid": "0f0f26e5-0562-46c9-afa2-353503ce1ddc",
          "serviceURL": "/v2/services/0f0f26e5-0562-46c9-afa2-353503ce1ddc",
          "serviceName": "Auto-Scaling",
          "serviceDescription": "Automatically increase or decrease the number of application instances based on a policy you define.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 30, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Sep 28, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 831,
          "serviceGuid": "176b952b-ddb0-4add-9538-12012e7fade3",
          "serviceURL": "/v2/services/176b952b-ddb0-4add-9538-12012e7fade3",
          "serviceName": "WebSphereAppSvr",
          "serviceDescription": "Allows you to quickly get up and running on a pre-
configured WebSphere Application Server installation in a hosted cloud environment in IBM
Cloud.",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 20, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Dec 19, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 841,
          "serviceGuid": "9738fe08-f35e-4833-b066-a295a197911f",
          "serviceURL": "/v2/services/9738fe08-f35e-4833-b066-a295a197911f",
          "serviceName": "Twilio Programmable SMS",
          "serviceDescription": "Not just an API to exchange SMS text messages.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 25, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 851,
          "serviceGuid": "455b511c-29e6-4020-a364-9734706357a3",
          "serviceURL": "/v2/services/455b511c-29e6-4020-a364-9734706357a3",
          "serviceName": "fss-instrument-analytics-service",
          "serviceDescription": "Leverage sophisticated IBM Algorithmics financial models
to price and compute analytics on financial securities.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Mar 14, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Aug 14, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 861,
          "serviceGuid": "3d00b690-a39a-40c7-8eba-aed1a7b1739d",
          "serviceURL": "/v2/services/3d00b690-a39a-40c7-8eba-aed1a7b1739d",
          "serviceName": "Precision Location",
          "serviceDescription": "Skyhook Precision Location",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jun 13, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 871,
          "serviceGuid": "c7dc4533-16ac-4377-a24b-b8a9cfbcdbc8",
          "serviceURL": "/v2/services/c7dc4533-16ac-4377-a24b-b8a9cfbcdbc8",
          "serviceName": "runbookautomation",
          "serviceDescription": "Support Operators by providing a structured way of
executing Runbooks.",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Mar 8, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Oct 3, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 881,
          "serviceGuid": "8d4a21c3-7ef2-4f24-a5f6-751ecc8c6675",
          "serviceURL": "/v2/services/8d4a21c3-7ef2-4f24-a5f6-751ecc8c6675",
          "serviceName": "hs-crypto",
          "serviceDescription": "IBM Cloud Hyper Protect Crypto Services provides
cryptographic functions from a highly secure, FIPS-140-2 level 4 certified HSM on IBM Z",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 21, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Mar 14, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 891,
          "serviceGuid": "34f9d20d-b407-42bd-9325-acc51a041504",
          "serviceURL": "/v2/services/34f9d20d-b407-42bd-9325-acc51a041504",
          "serviceName": "XPagesData",
          "serviceDescription": "Create an IBM Notes .NSF database to store your XPages
Domino data.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jun 24, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Oct 27, 2015 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 901,
          "serviceGuid": "b8d348ea-c9a0-4a1c-bd54-efba97a9cbb3",
          "serviceURL": "/v2/services/b8d348ea-c9a0-4a1c-bd54-efba97a9cbb3",
          "serviceName": "ecs-dashboard",
          "serviceDescription": "Integrate automated accessibility auditing and reporting
capabilities into your deployment DevOps processes.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 16, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Apr 19, 2016 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 911,
          "serviceGuid": "6e395e18-5954-4fb1-ac75-b8ae504c4005",
          "serviceURL": "/v2/services/6e395e18-5954-4fb1-ac75-b8ae504c4005",
          "serviceName": "Xignite Market Data APIs",
          "serviceDescription": "Real-time and reference market data",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Mar 17, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 921,
          "serviceGuid": "11a84001-3ab3-4eaa-af39-23b8240edfdd",
          "serviceURL": "/v2/services/11a84001-3ab3-4eaa-af39-23b8240edfdd",
          "serviceName": "Plaid",
          "serviceDescription": "Innovate in financial services.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Mar 10, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 931,
          "serviceGuid": "3a85b6bc-952c-4c4a-9e9b-12722bf2cbbd",
          "serviceURL": "/v2/services/3a85b6bc-952c-4c4a-9e9b-12722bf2cbbd",
          "serviceName": "db2oncloud",
          "serviceDescription": "Db2 Hosted: Offers customers the rich features of an on-premise Db2 deployment without the cost, complexity, and risk of managing their own infrastructure.",
          "servicePlansQuantity": 11,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jun 8, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Dec 3, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 941,
          "serviceGuid": "1577ff61-d898-4244-b973-e6cd09ea8c3f",
          "serviceURL": "/v2/services/1577ff61-d898-4244-b973-e6cd09ea8c3f",
          "serviceName": "mdmoncloud",
          "serviceDescription": "IBM® Master Data Management (MDM) on Cloud helps businesses gain a trusted view of data in a hybrid computing environment.",
          "servicePlansQuantity": 9,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 6, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Dec 3, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 951,
          "serviceGuid": "36a2ec3e-a565-45f4-a244-48eb8865d1a0",
          "serviceURL": "/v2/services/36a2ec3e-a565-45f4-a244-48eb8865d1a0",
          "serviceName": "Morningstar",
          "serviceDescription": "Managed investments portfolio statistics",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Mar 20, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 961,
          "serviceGuid": "8012c0a1-25d5-4e25-8020-b5d7dbffb4eb",
          "serviceURL": "/v2/services/8012c0a1-25d5-4e25-8020-b5d7dbffb4eb",
          "serviceName": "Continuous Release",
          "serviceDescription": "Manage software deployments with this enterprise-scale
release management solution.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 20, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Mar 2, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 971,
          "serviceGuid": "d111fbed-ec1f-4759-b4f8-f998a207e6ec",
          "serviceURL": "/v2/services/d111fbed-ec1f-4759-b4f8-f998a207e6ec",
          "serviceName": "personality_insights",
          "serviceDescription": "The Watson Personality Insights derives insights from
transactional and social media data to identify psychological traits",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 12, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Jan 7, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 981,
          "serviceGuid": "f29d484a-21e1-4c88-9c8a-a6c07b520b83",
          "serviceURL": "/v2/services/f29d484a-21e1-4c88-9c8a-a6c07b520b83",
          "serviceName": "Passport",
          "serviceDescription": "Modern Identity and User Management",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Feb 24, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Sep 23, 2018 12:00:00 AM"
        },
        {
          "serviceId": 991,
          "serviceGuid": "cf5d12b8-8a0e-4aa1-83ce-b04929c6b651",
          "serviceURL": "/v2/services/cf5d12b8-8a0e-4aa1-83ce-b04929c6b651",
          "serviceName": "xpertrule-nodejs",
          "serviceDescription": "Non-coders and developers can automate and execute
business decisions",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Dec 2, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1001,
          "serviceGuid": "3e622e8c-54e0-4b8f-993a-f635ca9bcd35",
          "serviceURL": "/v2/services/3e622e8c-54e0-4b8f-993a-f635ca9bcd35",
          "serviceName": "knowledge-studio",
          "serviceDescription": "Build custom models to teach Watson the language of your
domain.",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 20, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jan 7, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1011,
          "serviceGuid": "1af6af74-8b1a-429e-aeef-51119651c09e",
          "serviceURL": "/v2/services/1af6af74-8b1a-429e-aeef-51119651c09e",
          "serviceName": "AppLaunch",
          "serviceDescription": "Accelerate the delivery of innovations to mobile apps by
avoiding release cycle complexities.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 29, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jan 25, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1021,
          "serviceGuid": "76ac7746-7ddb-4e4b-9b8b-382954c5cf57",
          "serviceURL": "/v2/services/76ac7746-7ddb-4e4b-9b8b-382954c5cf57",
          "serviceName": "ibm-blockchain-5-prod",
          "serviceDescription": "IBM Blockchain Platform is a flexible software-as-a-
service offering that simplifies the blockchain journey of developing, governing, and
operating a blockchain network.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Apr 21, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Dec 7, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1031,
          "serviceGuid": "e0643793-e03e-49ec-8db5-6bc244f3ec8c",
          "serviceURL": "/v2/services/e0643793-e03e-49ec-8db5-6bc244f3ec8c",
          "serviceName": "IBM_Cloud_Brokerage_CAM",
          "serviceDescription": "Hybrid Cloud Cost and Asset management service broker",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jun 23, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jun 23, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1041,
          "serviceGuid": "5145dc97-d149-416f-97e1-efbba6c56b7f",
          "serviceURL": "/v2/services/5145dc97-d149-416f-97e1-efbba6c56b7f",
          "serviceName": "Mobile Foundation",
          "serviceDescription": "Build secure, cognitive, engaging and personalized mobile
apps faster at scale",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Mar 14, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Dec 26, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1051,
          "serviceGuid": "4d546349-a9d5-42e9-b0bb-dd069efbb456",
          "serviceURL": "/v2/services/4d546349-a9d5-42e9-b0bb-dd069efbb456",
          "serviceName": "compose-for-mongodb",
          "serviceDescription": "MongoDB is a JSON document store with a rich query and
aggregation framework",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 6, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Dec 8, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1061,
          "serviceGuid": "4e29b98a-2175-4d72-951d-bb52b1733ef2",
          "serviceURL": "/v2/services/4e29b98a-2175-4d72-951d-bb52b1733ef2",
          "serviceName": "compose-for-postgresql",
          "serviceDescription": "Postgres is a powerful, open source object-relational
database that is highly customizable.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 6, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Dec 8, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1071,
          "serviceGuid": "a90c6af0-2ed0-4b43-8cb6-9f6c908aef3c",
          "serviceURL": "/v2/services/a90c6af0-2ed0-4b43-8cb6-9f6c908aef3c",
          "serviceName": "SingleSignOn",
          "serviceDescription": "Implement user authentication for your web and mobile
apps quickly, using simple policy-based configurations.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 23, 2014 12:00:00 AM",
          "serviceUpdatedAt": "May 31, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1081,
          "serviceGuid": "11394918-7c61-4e85-a84d-1030c4372807",
          "serviceURL": "/v2/services/11394918-7c61-4e85-a84d-1030c4372807",
          "serviceName": "ibm-iot-for-electronics",
          "serviceDescription": "The IoT for Electronics service supports user and device
registration and notifications. As part of the IoT for Electronics Starter, it is
preconfigured with other services to help you connect your de",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Apr 7, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Apr 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1091,
          "serviceGuid": "0299549f-8e09-44fc-9d7f-94d771d13122",
          "serviceURL": "/v2/services/0299549f-8e09-44fc-9d7f-94d771d13122",
          "serviceName": "Twilio Authy",
          "serviceDescription": "Secure your users with 2FA for mobile and web.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 25, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1101,
          "serviceGuid": "2e5dfb52-5340-4a7c-b603-06d124da7159",
          "serviceURL": "/v2/services/2e5dfb52-5340-4a7c-b603-06d124da7159",
          "serviceName": "Bosch IoT Rollouts",
          "serviceDescription": "Rollout software and firmware updates to devices",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Nov 7, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1111,
          "serviceGuid": "288381bc-20d8-46eb-ad91-3137c8e249e4",
          "serviceURL": "/v2/services/288381bc-20d8-46eb-ad91-3137c8e249e4",
          "serviceName": "Twilio Programmable Video",
          "serviceDescription": "Embed WebRTC video calling into web & mobile apps.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 25, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1121,
          "serviceGuid": "8688d5ee-a7c2-4f43-afea-346533e58903",
          "serviceURL": "/v2/services/8688d5ee-a7c2-4f43-afea-346533e58903",
          "serviceName": "WealthEngine API",
          "serviceDescription": "Look up anyone's net worth in real-time.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 8, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1131,
          "serviceGuid": "ce6a0e45-8b6a-4c95-af72-c5bdb835ae60",
          "serviceURL": "/v2/services/ce6a0e45-8b6a-4c95-af72-c5bdb835ae60",
          "serviceName": "Bondevalue-API",
          "serviceDescription": "Real time bonds data to manage one?s bond investments.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jun 12, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1141,
          "serviceGuid": "c281bb9f-3958-4213-903c-95b6097a53bf",
          "serviceURL": "/v2/services/c281bb9f-3958-4213-903c-95b6097a53bf",
          "serviceName": "weatherinsights",
          "serviceDescription": "Use the Weather Company Data for IBM Bluemix service to
incorporate weather data into your Bluemix applications.",
          "servicePlansQuantity": 4,
          "serviceStatus": "active",
          "serviceCreatedAt": "Oct 20, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Jan 15, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1151,
          "serviceGuid": "240ccbf2-2f67-419b-8726-5019f90e0530",
          "serviceURL": "/v2/services/240ccbf2-2f67-419b-8726-5019f90e0530",
          "serviceName": "discovery",
          "serviceDescription": "Add a cognitive search and content analytics engine to applications.",
          "servicePlansQuantity": 4,
          "serviceStatus": "active",
          "serviceCreatedAt": "Apr 15, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jan 4, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1161,
          "serviceGuid": "02f38717-9869-4736-846a-0e416e2fc7a0",
          "serviceURL": "/v2/services/02f38717-9869-4736-846a-0e416e2fc7a0",
          "serviceName": "tinyqueries",
          "serviceDescription": "Create complex queries easily",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Mar 9, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1171,
          "serviceGuid": "9d57abb6-8bdc-40bd-9d9a-a71aae498567",
          "serviceURL": "/v2/services/9d57abb6-8bdc-40bd-9d9a-a71aae498567",
          "serviceName": "internet-svcs",
          "serviceDescription": "Provides network security, reliability and performance for applications.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jan 19, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Sep 18, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1181,
          "serviceGuid": "8f89831e-5cc8-4fa8-9a2d-0e7989296f25",
          "serviceURL": "/v2/services/8f89831e-5cc8-4fa8-9a2d-0e7989296f25",
          "serviceName": "watson_vision_combined",
          "serviceDescription": "Find meaning in visual content! Analyze images for scenes, objects, faces, and other content. Choose a default model off the shelf, or create your own custom classifier. Develop smart applications tha",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 17, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jan 3, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1191,
  "serviceGuid": "0aa8ede4-979b-4f83-846a-bc151c275175",
  "serviceURL": "/v2/services/0aa8ede4-979b-4f83-846a-bc151c275175",
  "serviceName": "continuous-delivery",
  "serviceDescription": "Build, test and deliver using DevOps best practices.",
  "servicePlansQuantity": 2,
  "serviceStatus": "active",
  "serviceCreatedAt": "Sep 21, 2016 12:00:00 AM",
  "serviceUpdatedAt": "Nov 28, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 1201,
  "serviceGuid": "3289a5f1-6126-451a-9a70-d3789222470c",
  "serviceURL": "/v2/services/3289a5f1-6126-451a-9a70-d3789222470c",
  "serviceName": "GEO Web Services",
  "serviceDescription": "Adding geo-intelligence to your business.",
  "servicePlansQuantity": 1,
  "serviceStatus": "active",
  "serviceCreatedAt": "Jun 26, 2017 12:00:00 AM",
  "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 1211,
  "serviceGuid": "634c4aa8-640b-4d12-a2ac-e129fdaa54d9",
  "serviceURL": "/v2/services/634c4aa8-640b-4d12-a2ac-e129fdaa54d9",
  "serviceName": "vantrix-transcoder",
  "serviceDescription": "Video Transcoding",
  "servicePlansQuantity": 1,
  "serviceStatus": "inactive",
  "serviceCreatedAt": "Feb 16, 2016 12:00:00 AM",
  "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
},
{
  "serviceId": 1221,
  "serviceGuid": "f84204c1-a082-440f-941e-3ef10fbe222a",
  "serviceURL": "/v2/services/f84204c1-a082-440f-941e-3ef10fbe222a",
  "serviceName": "compose-for-mysql",
  "serviceDescription": "MySQL is a fast, easy-to-use, and flexible RDBMS.",
  "servicePlansQuantity": 2,
  "serviceStatus": "active",
  "serviceCreatedAt": "Dec 15, 2016 12:00:00 AM",
  "serviceUpdatedAt": "Dec 8, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 1231,
  "serviceGuid": "92394a9c-4509-4398-9740-8b01ed16ac3a",
  "serviceURL": "/v2/services/92394a9c-4509-4398-9740-8b01ed16ac3a",
  "serviceName": "kms",
          "serviceDescription": "An app-independent service for protecting, managing, and generating keys.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 7, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Dec 13, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1241,
          "serviceGuid": "c438906e-563a-4879-8f0e-2b29d161caec",
          "serviceURL": "/v2/services/c438906e-563a-4879-8f0e-2b29d161caec",
          "serviceName": "dashDB For Transactions",
          "serviceDescription": "A next generation SQL database. Formerly dashDB For Transactions.",
          "servicePlansQuantity": 6,
          "serviceStatus": "active",
          "serviceCreatedAt": "Nov 1, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jan 22, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1251,
          "serviceGuid": "14c83ad2-6fd4-439a-8c3a-d1a20f8a2381",
          "serviceURL": "/v2/services/14c83ad2-6fd4-439a-8c3a-d1a20f8a2381",
          "serviceName": "cloudantNoSQLDB",
          "serviceDescription": "IBM Cloudant is a fully managed JSON document database. Cloudant is compatible with Apache CouchDB and accessible through a simple to use HTTPS API for web, mobile, and IoT applications. See https://i",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jun 3, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Jul 9, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1261,
          "serviceGuid": "05447c76-4e42-4fcb-b1ed-0ddb29eacd04",
          "serviceURL": "/v2/services/05447c76-4e42-4fcb-b1ed-0ddb29eacd04",
          "serviceName": "ustream",
          "serviceDescription": "Video streaming, storage and publishing.",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Sep 30, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1271,
          "serviceGuid": "99222288-7beb-4d0d-a3e4-729f7b2caf15",
          "serviceURL": "/v2/services/99222288-7beb-4d0d-a3e4-729f7b2caf15",
          "serviceName": "Document Generation",
          "serviceDescription": "Generate documents from any standard data source with the
Document Generation for Bluemix service.",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Jun 3, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Oct 13, 2015 12:00:00 AM",
          "serviceSynchronizedAt": "Sep 23, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1281,
          "serviceGuid": "aff58576-c0fc-4d9a-a57d-c6dd492bede1",
          "serviceURL": "/v2/services/aff58576-c0fc-4d9a-a57d-c6dd492bede1",
          "serviceName": "BigInsightsonCloud",
          "serviceDescription": "Provision managed bare metal Apache Hadoop clusters for
production use or POCs at scale.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 18, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Mar 17, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1291,
          "serviceGuid": "5d647166-56cc-41a6-a660-53219a05f3d7",
          "serviceURL": "/v2/services/5d647166-56cc-41a6-a660-53219a05f3d7",
          "serviceName": "APIConnect",
          "serviceDescription": "Create, manage, enforce, and run APIs.",
          "servicePlansQuantity": 10,
          "serviceStatus": "active",
          "serviceCreatedAt": "Mar 21, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Sep 1, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1301,
          "serviceGuid": "0c98401f-7e66-498e-8f0c-e1c9b667237f",
          "serviceURL": "/v2/services/0c98401f-7e66-498e-8f0c-e1c9b667237f",
          "serviceName": "Accrete.AI: Rumor Hound",
          "serviceDescription": "Scours digital sources to find M&A rumors",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jan 30, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1311,
          "serviceGuid": "8d774aae-3f11-4258-935a-c023a9ddf817",
          "serviceURL": "/v2/services/8d774aae-3f11-4258-935a-c023a9ddf817",
          "serviceName": "mysql-replaced",
          "serviceDescription": "This service is no longer available. Please search for Compose services in the main catalog instead.",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jun 6, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jun 6, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1321,
          "serviceGuid": "e685162a-2a48-454a-8167-6bf2f27da58b",
          "serviceURL": "/v2/services/e685162a-2a48-454a-8167-6bf2f27da58b",
          "serviceName": "fss-scenario-analytics-service",
          "serviceDescription": "Leverage sophisticated IBM Algorithmics financial models to price and compute analytics on financial securities under a given scenario.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Mar 14, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Aug 14, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1331,
          "serviceGuid": "cb77c376-ecf5-466d-99da-8b1a2fc7de3f",
          "serviceURL": "/v2/services/cb77c376-ecf5-466d-99da-8b1a2fc7de3f",
          "serviceName": "Car Diagnostic API",
          "serviceDescription": "Translation service for OBD error codes.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 21, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1341,
          "serviceGuid": "d14f3880-6a1d-4c41-806d-6f7c0769e0e8",
          "serviceURL": "/v2/services/d14f3880-6a1d-4c41-806d-6f7c0769e0e8",
          "serviceName": "cleardb",
          "serviceDescription": "Highly available MySQL for Apps.",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Apr 15, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1351,
          "serviceGuid": "92755da1-fc09-4217-a594-db780becf6f6",
          "serviceURL": "/v2/services/92755da1-fc09-4217-a594-db780becf6f6",
          "serviceName": "pm-20",
          "serviceDescription": "IBM Watson Machine Learning - make smarter decisions,
solve tough problems, and improve user outcomes.",
          "servicePlansQuantity": 4,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 9, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Jun 18, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1361,
          "serviceGuid": "331a0869-047e-4e24-a9a6-16c4aaf29f1f",
          "serviceURL": "/v2/services/331a0869-047e-4e24-a9a6-16c4aaf29f1f",
          "serviceName": "AppConnect",
          "serviceDescription": "Connect your applications, automate tasks, and improve
productivity",
          "servicePlansQuantity": 5,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 13, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jan 15, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1371,
          "serviceGuid": "f5c45150-8023-4d3e-a3d4-5a3a8ca8e407",
          "serviceURL": "/v2/services/f5c45150-8023-4d3e-a3d4-5a3a8ca8e407",
          "serviceName": "Geospatial Analytics",
          "serviceDescription": "Expand the boundaries of your application. Leverage real-
time geospatial analytics to track when devices enter, leave or hang out in defined
regions. Powered by IBM Streaming Analytics on IBM Cloud.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 18, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Apr 19, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1381,
          "serviceGuid": "4237b2fc-c536-4e6e-a75a-74b9a65b4e69",
          "serviceURL": "/v2/services/4237b2fc-c536-4e6e-a75a-74b9a65b4e69",
          "serviceName": "streaming-analytics",
          "serviceDescription": "Leverage IBM Streams to ingest, analyze, monitor, and
correlate data as it arrives from real-time data sources. View information and events as
they unfold.",
          "servicePlansQuantity": 15,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 14, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Jul 10, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1391,
          "serviceGuid": "08a471f9-5be2-4b4f-aeb9-fe90f1197307",
          "serviceURL": "/v2/services/08a471f9-5be2-4b4f-aeb9-fe90f1197307",
          "serviceName": "DevOpsInsights",
          "serviceDescription": "Improve agility, reliability, and security by using
machine learning and analytics",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "May 12, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jul 27, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Sep 23, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1401,
          "serviceGuid": "9f901350-8603-4d51-9830-a768f757e10b",
          "serviceURL": "/v2/services/9f901350-8603-4d51-9830-a768f757e10b",
          "serviceName": "memcachedcloud",
          "serviceDescription": "Enterprise-Class Memcached for Developers",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Apr 23, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1411,
          "serviceGuid": "95f00ec8-07e9-4f02-b767-5eaf9caf22e3",
          "serviceURL": "/v2/services/95f00ec8-07e9-4f02-b767-5eaf9caf22e3",
          "serviceName": "Natural Language Generation APIs",
          "serviceDescription": "Generate expertly written narratives in seconds",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Oct 24, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1421,
          "serviceGuid": "df3ebc3a-4268-4e0e-8ab3-164fe95a4118",
          "serviceURL": "/v2/services/df3ebc3a-4268-4e0e-8ab3-164fe95a4118",
          "serviceName": "AT&T IoT Data Plans",
          "serviceDescription": "Launch your IoT product fast with IoT data plans",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Apr 3, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1431,
          "serviceGuid": "7291730c-bce7-4c8d-99c0-4d968daf0c7f",
          "serviceURL": "/v2/services/7291730c-bce7-4c8d-99c0-4d968daf0c7f",
          "serviceName": "g11n-pipeline",
          "serviceDescription": "Manage the translation of your cloud and mobile
applications using IBM Globalization Pipeline.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 8, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Dec 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1441,
          "serviceGuid": "1ab5e541-c007-412b-a65e-598b0ef6337f",
          "serviceURL": "/v2/services/1ab5e541-c007-412b-a65e-598b0ef6337f",
          "serviceName": "Alloy",
          "serviceDescription": "API for identity (KYC, AML & fraud)",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 7, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1451,
          "serviceGuid": "7c87c148-e1a4-4cb8-81f8-c5e74be7684b",
          "serviceURL": "/v2/services/7c87c148-e1a4-4cb8-81f8-c5e74be7684b",
          "serviceName": "dashDB",
          "serviceDescription": "Db2 Warehouse on Cloud is a flexible and powerful data
warehouse for enterprise-level analytics.",
          "servicePlansQuantity": 15,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 17, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Jan 22, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1461,
          "serviceGuid": "6382b237-f88e-4ac6-a01c-ee6903c06d95",
          "serviceURL": "/v2/services/6382b237-f88e-4ac6-a01c-ee6903c06d95",
          "serviceName": "Object-Storage",
          "serviceDescription": "Provides a cost-effective, scalable, unstructured cloud
data store to build and deliver cloud apps and services.",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Oct 13, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Sep 15, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1471,
          "serviceGuid": "2524f345-ff25-4819-8f46-f98bb95020d9",
          "serviceURL": "/v2/services/2524f345-ff25-4819-8f46-f98bb95020d9",
          "serviceName": "Conversation.one",
          "serviceDescription": "Build Voicebots In Minutes",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Nov 15, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1481,
          "serviceGuid": "adfd3a72-b33e-4648-97ca-031a2843324c",
          "serviceURL": "/v2/services/adfd3a72-b33e-4648-97ca-031a2843324c",
          "serviceName": "RiskSpan RS Edge Loan Analytics",
          "serviceDescription": "A loan analytics and predictive modeling platform",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Mar 16, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1491,
          "serviceGuid": "1122a3f2-5787-4c43-8fa7-099ae661b8d3",
          "serviceURL": "/v2/services/1122a3f2-5787-4c43-8fa7-099ae661b8d3",
          "serviceName": "Coalesce for Risk and Compliance Monitoring",
          "serviceDescription": "Automates Risk and Compliance Monitoring",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Nov 27, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1501,
          "serviceGuid": "96dcf868-b670-4156-9db1-ab3605cd184d",
          "serviceURL": "/v2/services/96dcf868-b670-4156-9db1-ab3605cd184d",
          "serviceName": "cloudeventmanagement",
          "serviceDescription": "Consolidated operational event and incident management.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 27, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Nov 28, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1511,
          "serviceGuid": "bac507c9-a0a2-4fd9-bc52-b7ffa57113e8",
          "serviceURL": "/v2/services/bac507c9-a0a2-4fd9-bc52-b7ffa57113e8",
          "serviceName": "ibmLogAnalysis",
          "serviceDescription": "Collect, store, and analyze your application's log
data.",
          "servicePlansQuantity": 5,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 19, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Aug 8, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1521,
          "serviceGuid": "6ee1daf9-2977-4730-a4e7-71c1a7fb23e7",
          "serviceURL": "/v2/services/6ee1daf9-2977-4730-a4e7-71c1a7fb23e7",
          "serviceName": "mapinsights",
          "serviceDescription": "IBM Watson IoT Context Mapping Service brings the power
to your application to analyze moving object trajectories by leveraging road network-based
geospatial services.",
          "servicePlansQuantity": 2,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Mar 7, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Feb 2, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Nov 1, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1531,
          "serviceGuid": "bba57e76-4dde-45d1-a9cb-be879d07e6e4",
          "serviceURL": "/v2/services/bba57e76-4dde-45d1-a9cb-be879d07e6e4",
          "serviceName": "real-time-payments-service",
          "serviceDescription": "Manage participants, tokens and recipients, and initiate
and receive real time payments.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Nov 13, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Dec 1, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1541,
          "serviceGuid": "37b045ec-879c-42f6-8d80-9650d0b94906",
          "serviceURL": "/v2/services/37b045ec-879c-42f6-8d80-9650d0b94906",
          "serviceName": "SPLICE Pre-CAT Insurance Notifications",
          "serviceDescription": "Pre-CAT Notifications for Insurance Companies",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 28, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1551,
          "serviceGuid": "b0a3e1e7-c7d6-418b-8612-3f2cbb8c812d",
          "serviceURL": "/v2/services/b0a3e1e7-c7d6-418b-8612-3f2cbb8c812d",
          "serviceName": "ibmcloud-link",
          "serviceDescription": "An IBM provided service that enables aliasing to service
instances in the IBM Cloud.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Apr 11, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Apr 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1561,
          "serviceGuid": "cb583050-0723-4da7-bf41-23f5acf0aa8d",
          "serviceURL": "/v2/services/cb583050-0723-4da7-bf41-23f5acf0aa8d",
          "serviceName": "speech_to_text",
          "serviceDescription": "Low-latency, streaming transcription",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jan 30, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Jan 7, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1571,
          "serviceGuid": "fe6c8f90-6e69-4571-954f-377a893357c5",
          "serviceURL": "/v2/services/fe6c8f90-6e69-4571-954f-377a893357c5",
          "serviceName": "apprenda",
          "serviceDescription": "Bluemix .NET Powered by Apprenda",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Mar 17, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1581,
          "serviceGuid": "9cf06f4f-8f4e-4084-aefe-bb4d821920b8",
          "serviceURL": "/v2/services/9cf06f4f-8f4e-4084-aefe-bb4d821920b8",
          "serviceName": "Quovo",
          "serviceDescription": "Connecting You to Your Users' Financial Accounts",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Apr 3, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1591,
          "serviceGuid": "038b44f7-5fe3-43b8-90cc-759422db9b65",
          "serviceURL": "/v2/services/038b44f7-5fe3-43b8-90cc-759422db9b65",
          "serviceName": "natural-language-understanding",
          "serviceDescription": "Analyze text to extract meta-data from content such as
concepts, entities, emotion, relations, sentiment and more.",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Feb 24, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jan 7, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1601,
          "serviceGuid": "1a3cc31a-c58e-4c05-a0c2-c5087a6ee8bd",
          "serviceURL": "/v2/services/1a3cc31a-c58e-4c05-a0c2-c5087a6ee8bd",
          "serviceName": "compose-for-elasticsearch",
          "serviceDescription": "Elasticsearch combines the power of a full text search
engine with the indexing strengths of a JSON document database",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 6, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Dec 8, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1611,
          "serviceGuid": "f1ed3a0a-4d7c-421b-b8fd-a01ba3ac4838",
          "serviceURL": "/v2/services/f1ed3a0a-4d7c-421b-b8fd-a01ba3ac4838",
          "serviceName": "testdroid",
          "serviceDescription": "Mobile testing cloud service",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Mar 21, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1621,
          "serviceGuid": "3130604f-8c14-47d6-b78f-9d5381187c5c",
          "serviceURL": "/v2/services/3130604f-8c14-47d6-b78f-9d5381187c5c",
          "serviceName": "UnificationEngine",
          "serviceDescription": "Intelligent IoT messaging for all H2M communications.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 30, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1631,
          "serviceGuid": "ca7e360c-821b-43e8-a011-e23e8be64db9",
          "serviceURL": "/v2/services/ca7e360c-821b-43e8-a011-e23e8be64db9",
          "serviceName": "Contrast Security",
          "serviceDescription": "Detect vulnerabilities and block attacks",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 24, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1641,
          "serviceGuid": "46b2a9ec-1813-4f82-88e3-a6c9bf776ae2",
          "serviceURL": "/v2/services/46b2a9ec-1813-4f82-88e3-a6c9bf776ae2",
          "serviceName": "pitneybowes-apis",
          "serviceDescription": "Add enterprise-class geodata and commerce technology your
application",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Sep 15, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1651,
          "serviceGuid": "91650649-da8f-4865-9eb0-2525906e218e",
          "serviceURL": "/v2/services/91650649-da8f-4865-9eb0-2525906e218e",
          "serviceName": "text_to_speech",
          "serviceDescription": "Synthesizes natural-sounding speech from text.",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jan 30, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Jan 7, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1661,
          "serviceGuid": "67ec0b70-b198-4cd8-8d3b-0071f1a6987c",
          "serviceURL": "/v2/services/67ec0b70-b198-4cd8-8d3b-0071f1a6987c",
          "serviceName": "Internet of Things Workbench",
          "serviceDescription": "An intuitive development environment for rapid design,
simulation, & construction of complete Internet of Things solutions and services",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jun 8, 2015 12:00:00 AM",
          "serviceUpdatedAt": "May 4, 2016 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1671,
          "serviceGuid": "7e0715bc-b608-4bfd-838d-044bfc6f3d7e",
          "serviceURL": "/v2/services/7e0715bc-b608-4bfd-838d-044bfc6f3d7e",
          "serviceName": "fss-historical-instrument-analytics-service",
          "serviceDescription": "Leverage sophisticated IBM Algorithmics financial models
to price and evaluate financial securities for historical dates.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Mar 14, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Aug 14, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1681,
          "serviceGuid": "7e30d08c-c25d-4069-b4e0-ddfcc335f732",
          "serviceURL": "/v2/services/7e30d08c-c25d-4069-b4e0-ddfcc335f732",
          "serviceName": "Phunware Location Based Services",
          "serviceDescription": "Phunware Location Based Services",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 1, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1691,
          "serviceGuid": "ec7b2a1e-b390-45a0-bc85-7d3cc3632708",
          "serviceURL": "/v2/services/ec7b2a1e-b390-45a0-bc85-7d3cc3632708",
          "serviceName": "dynamic-dashboard-embedded",
          "serviceDescription": "Bring data to life directly from your application with
this powerful and easy-to-use visualization service.",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 5, 2017 12:00:00 AM",
          "serviceUpdatedAt": "May 16, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1701,
          "serviceGuid": "86e8a7c0-f399-41d3-ae28-92de91ccca86",
          "serviceURL": "/v2/services/86e8a7c0-f399-41d3-ae28-92de91ccca86",
          "serviceName": "Nexmo",
          "serviceDescription": "Build great communication experiences.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Apr 10, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jul 11, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1711,
          "serviceGuid": "d12e1a82-ecc0-4bd4-95e9-93b29b1d968d",
          "serviceURL": "/v2/services/d12e1a82-ecc0-4bd4-95e9-93b29b1d968d",
          "serviceName": "sendgrid",
          "serviceDescription": "Delivering your email through one reliable platform.",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Apr 18, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1721,
          "serviceGuid": "cdcd770b-e9d0-4ac1-b2cf-984f19f0156a",
          "serviceURL": "/v2/services/cdcd770b-e9d0-4ac1-b2cf-984f19f0156a",
          "serviceName": "cpy-insights",
          "serviceDescription": "Business Activity Insights for Bluemix© apps",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Mar 17, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1731,
          "serviceGuid": "a1da956b-cc22-42ac-af6b-38f91844f26f",
          "serviceURL": "/v2/services/a1da956b-cc22-42ac-af6b-38f91844f26f",
          "serviceName": "jkoolapi",
          "serviceDescription": "jKool provides real-time and historical visualization and analytics",
          "servicePlansQuantity": 1,
          "serviceStatus": "inactive",
          "serviceCreatedAt": "Aug 3, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Feb 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jun 3, 2018 12:00:00 AM"
        },
        {
          "serviceId": 1741,
          "serviceGuid": "035875d1-4e23-4eb5-96ff-db75803c9064",
          "serviceURL": "/v2/services/035875d1-4e23-4eb5-96ff-db75803c9064",
          "serviceName": "redis-replaced",
          "serviceDescription": "This service is no longer available. Please search for Compose services in the main catalog instead.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jun 6, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jun 6, 2017 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1751,
          "serviceGuid": "b13b9cf8-fff8-4e3a-b794-acf3558b91f9",
          "serviceURL": "/v2/services/b13b9cf8-fff8-4e3a-b794-acf3558b91f9",
          "serviceName": "businessrules",
          "serviceDescription": "Automate and manage business logic in applications using business rules.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 11, 2014 12:00:00 AM",
          "serviceUpdatedAt": "Feb 24, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1761,
          "serviceGuid": "4fb727f3-6f6a-44a8-9102-d8a6c3140b04",
          "serviceURL": "/v2/services/4fb727f3-6f6a-44a8-9102-d8a6c3140b04",
          "serviceName": "accessTrail",
          "serviceDescription": "Capture, store, and visualize your IBM Cloud activities",
          "servicePlansQuantity": 2,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 11, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Feb 9, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1771,
          "serviceGuid": "dd9436db-ce71-48ff-9095-c9f537feadfc",
          "serviceURL": "/v2/services/dd9436db-ce71-48ff-9095-c9f537feadfc",
          "serviceName": "hypersecuredbaas",
          "serviceDescription": "Hyper Protect DBaaS is a highly secured enterprise
service. It provides capabilities to manage different database types like MongoDB or
PostgreSQL through standardized interfaces.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Mar 8, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Nov 15, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2142,
          "serviceGuid": "7120c147-b3f4-4802-a1c4-cba61a32107d",
          "serviceURL": "/v2/services/7120c147-b3f4-4802-a1c4-cba61a32107d",
          "serviceName": "SizeUp Small Business Intelligence",
          "serviceDescription": "SizeUp Provides Big Data for Small Businesses",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 9, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jul 16, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2143,
          "serviceGuid": "15288659-1de0-4a06-ad9e-b515c93eb448",
          "serviceURL": "/v2/services/15288659-1de0-4a06-ad9e-b515c93eb448",
          "serviceName": "HazardHub Property Risk Data API",
          "serviceDescription": "Property Level hazard risk data",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 29, 2018 12:00:00 AM",
          "serviceUpdatedAt": "May 30, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
```

```
},
{
  "serviceId": 2144,
  "serviceGuid": "2d0709a9-3989-42af-91c7-b5058be3d169",
  "serviceURL": "/v2/services/2d0709a9-3989-42af-91c7-b5058be3d169",
  "serviceName": "RelSci",
  "serviceDescription": "Integrate people and relationship intelligence.",
  "servicePlansQuantity": 1,
  "serviceStatus": "active",
  "serviceCreatedAt": "Jun 29, 2018 12:00:00 AM",
  "serviceUpdatedAt": "Jun 29, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 2145,
  "serviceGuid": "19124649-3759-4cfd-904c-c041fc4c8261",
  "serviceURL": "/v2/services/19124649-3759-4cfd-904c-c041fc4c8261",
  "serviceName": "Strands Business Financial Management",
  "serviceDescription": "Business Financial Management",
  "servicePlansQuantity": 1,
  "serviceStatus": "active",
  "serviceCreatedAt": "Jun 29, 2018 12:00:00 AM",
  "serviceUpdatedAt": "Jun 29, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 2146,
  "serviceGuid": "85bb4b26-0ef8-48d6-981f-3d48f3a24737",
  "serviceURL": "/v2/services/85bb4b26-0ef8-48d6-981f-3d48f3a24737",
  "serviceName": "TrueRisk Labs - Equity Predictions Using Advanced AI",
  "serviceDescription": "Russell 5000 Price and Volatility Predictions",
  "servicePlansQuantity": 1,
  "serviceStatus": "active",
  "serviceCreatedAt": "Aug 9, 2018 12:00:00 AM",
  "serviceUpdatedAt": "Aug 15, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 2148,
  "serviceGuid": "c4784776-a075-4f3c-8e52-4ee2b470341a",
  "serviceURL": "/v2/services/c4784776-a075-4f3c-8e52-4ee2b470341a",
  "serviceName": "Telstra Messaging API",
  "serviceDescription": "Send and receive SMS/MMS messages globally.",
  "servicePlansQuantity": 1,
  "serviceStatus": "active",
  "serviceCreatedAt": "Aug 27, 2018 12:00:00 AM",
  "serviceUpdatedAt": "Oct 25, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 2149,
          "serviceGuid": "c722aaa2-c3f1-4f28-91d2-180c58de838a",
          "serviceURL": "/v2/services/c722aaa2-c3f1-4f28-91d2-180c58de838a",
          "serviceName": "Risk Engine",
          "serviceDescription": "Calculate health risks with dacadoo Risk Engine",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 19, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Sep 7, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2150,
          "serviceGuid": "f3358077-418a-47ca-91a7-b4cc15c029a0",
          "serviceURL": "/v2/services/f3358077-418a-47ca-91a7-b4cc15c029a0",
          "serviceName": "FusionAuth",
          "serviceDescription": "Modern Identity and User Management",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Oct 2, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Oct 5, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2151,
          "serviceGuid": "99115a8e-8726-4155-9453-04cccb477ccc",
          "serviceURL": "/v2/services/99115a8e-8726-4155-9453-04cccb477ccc",
          "serviceName": "databases-for-redis",
          "serviceDescription": "Redis is a blazingly fast, in-memory data structure
store.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Aug 21, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jan 24, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2152,
          "serviceGuid": "afc511fa-b17d-4eb3-8517-dc6a62e6e791",
          "serviceURL": "/v2/services/afc511fa-b17d-4eb3-8517-dc6a62e6e791",
          "serviceName": "Health Score",
          "serviceDescription": "The dacadoo Health Score measures health",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 19, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Sep 7, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2153,
          "serviceGuid": "f0b39106-f3b0-4a64-8035-36119191e74f",
          "serviceURL": "/v2/services/f0b39106-f3b0-4a64-8035-36119191e74f",
          "serviceName": "databases-for-postgresql",
          "serviceDescription": "PostgreSQL is a powerful, open source object-relational
database that is highly customizable.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 2, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jan 24, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2154,
          "serviceGuid": "bd7684d1-9b06-4ab4-b6db-9851cddee096",
          "serviceURL": "/v2/services/bd7684d1-9b06-4ab4-b6db-9851cddee096",
          "serviceName": "mendix-platform",
          "serviceDescription": "License your Mendix app on the IBM Cloud Portal.",
          "servicePlansQuantity": 5,
          "serviceStatus": "active",
          "serviceCreatedAt": "Oct 1, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jan 9, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2156,
          "serviceGuid": "02f5ebba-28f8-4bc7-b3a9-a084348f231c",
          "serviceURL": "/v2/services/02f5ebba-28f8-4bc7-b3a9-a084348f231c",
          "serviceName": "Powerlytics Consumer Income API",
          "serviceDescription": "The income profile of consumers at the ZIP+4 level",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 6, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Dec 10, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2157,
          "serviceGuid": "8a7531b5-44a0-4d36-b255-1d2c403a00ed",
          "serviceURL": "/v2/services/8a7531b5-44a0-4d36-b255-1d2c403a00ed",
          "serviceName": "FundingShield - Wire Account Verification Service (WAVS)",
          "serviceDescription": "Wire fraud prevention and compliance confirmation",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Nov 2, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Dec 6, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2158,
          "serviceGuid": "448e05b2-4e59-4d93-8d6a-32606450f6bc",
          "serviceURL": "/v2/services/448e05b2-4e59-4d93-8d6a-32606450f6bc",
          "serviceName": "databases-for-elasticsearch",
          "serviceDescription": "Elasticsearch combines the power of a full text search
engine with the indexing strengths of a JSON document database.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Oct 23, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jan 24, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2159,
          "serviceGuid": "60181d81-7295-47d5-8ee5-55025b5eeb14",
          "serviceURL": "/v2/services/60181d81-7295-47d5-8ee5-55025b5eeb14",
          "serviceName": "messages-for-rabbitmq",
          "serviceDescription": "RabbitMQ is an open source multi-protocol messaging
broker.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Nov 6, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jan 24, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2160,
          "serviceGuid": "b86123ff-94f7-4dd6-86e3-2d825e116141",
          "serviceURL": "/v2/services/b86123ff-94f7-4dd6-86e3-2d825e116141",
          "serviceName": "Powerlytics Investable Assets & Wealth API",
          "serviceDescription": "Investable assets & wealth income at Zip+4 level.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 6, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Dec 10, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2161,
          "serviceGuid": "0457f00c-1186-4d4f-8d64-afbc8c2d478a",
          "serviceURL": "/v2/services/0457f00c-1186-4d4f-8d64-afbc8c2d478a",
          "serviceName": "Powerlytics Behavior/Propensity Model API",
          "serviceDescription": "Improve customer behavior/propensity models",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 6, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Dec 10, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2162,
          "serviceGuid": "f51086d0-498d-4924-8c74-09353f83d408",
          "serviceURL": "/v2/services/f51086d0-498d-4924-8c74-09353f83d408",
          "serviceName": "sysdig-monitor",
          "serviceDescription": "Offers visibility into the performance and health of your
infrastructure and apps, in-depth troubleshooting, and alerting.",
          "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 19, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jan 9, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2163,
          "serviceGuid": "0ff44943-e912-4e5b-bf4a-b4f8da8c4984",
          "serviceURL": "/v2/services/0ff44943-e912-4e5b-bf4a-b4f8da8c4984",
          "serviceName": "Dwolla",
          "serviceDescription": "Dwolla is a powerful payments platform.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 20, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Dec 5, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2164,
          "serviceGuid": "e06d5546-cfd6-4027-a92f-74979ecf6bfc",
          "serviceURL": "/v2/services/e06d5546-cfd6-4027-a92f-74979ecf6bfc",
          "serviceName": "AccountScore",
          "serviceDescription": "AccountScore Open Banking & transaction analytics",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Nov 30, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Nov 30, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2165,
          "serviceGuid": "b0bce49d-e64f-42a7-a762-7cf8e1d55fff",
          "serviceURL": "/v2/services/b0bce49d-e64f-42a7-a762-7cf8e1d55fff",
          "serviceName": "databases-for-etcd",
          "serviceDescription": "Etcd is a distributed key value store.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 2, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jan 24, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2166,
          "serviceGuid": "6e2ec651-9f6f-4217-9bf9-2b7d9b3cc737",
          "serviceURL": "/v2/services/6e2ec651-9f6f-4217-9bf9-2b7d9b3cc737",
          "serviceName": "Totum Risk",
          "serviceDescription": "Risk tolerance tool for financial advisors",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 13, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Dec 13, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2167,
          "serviceGuid": "0f3c6d5b-1f69-447b-a906-8940462280fc",
          "serviceURL": "/v2/services/0f3c6d5b-1f69-447b-a906-8940462280fc",
          "serviceName": "Hydrogen",
          "serviceDescription": "Build fintech apps with ease with Hydrogen's APIs.",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 12, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Dec 12, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        }
      ]
    },
    {
      "id": 21,
      "name": "Pivotal Cloud Foundry",
      "platformsSyncDate": "Jan 26, 2019 12:00:00 AM",
      "servicesSyncDate": "Jan 26, 2019 12:00:00 AM",
      "servicesQuantity": 28,
      "services": [
        {
          "serviceId": 1781,
          "serviceGuid": "5e30ff7e-d857-4aa7-9eda-7db9a0d7b19b",
          "serviceURL": "/v2/services/5e30ff7e-d857-4aa7-9eda-7db9a0d7b19b",
          "serviceName": "cleardb",
          "serviceDescription": "Highly available MySQL for your Apps.",
          "servicePlansQuantity": 4,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 1791,
          "serviceGuid": "c72eba27-c3f6-4ccb-b2fb-0a74e8ce0a25",
          "serviceURL": "/v2/services/c72eba27-c3f6-4ccb-b2fb-0a74e8ce0a25",
          "serviceName": "mlab",
          "serviceDescription": "Fully managed MongoDB-as-a-Service",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
```

```
      "serviceId": 1801,
      "serviceGuid": "092b0ecd-b4ac-4dc1-bd6e-6e147b8f94ec",
      "serviceURL": "/v2/services/092b0ecd-b4ac-4dc1-bd6e-6e147b8f94ec",
      "serviceName": "searchify",
      "serviceDescription": "Custom search you control",
      "servicePlansQuantity": 3,
      "serviceStatus": "active",
      "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
      "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
      "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
    },
    {
      "serviceId": 1811,
      "serviceGuid": "f460748d-1983-4fba-9b7b-f97cd5127854",
      "serviceURL": "/v2/services/f460748d-1983-4fba-9b7b-f97cd5127854",
      "serviceName": "quotaguard",
      "serviceDescription": "High Availability Enterprise-Ready Static IPs",
      "servicePlansQuantity": 11,
      "serviceStatus": "active",
      "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
      "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
      "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
    },
    {
      "serviceId": 1821,
      "serviceGuid": "bc4b05aa-9b47-414c-a234-e2d182f7be86",
      "serviceURL": "/v2/services/bc4b05aa-9b47-414c-a234-e2d182f7be86",
      "serviceName": "streamdata",
      "serviceDescription": "Future-proof your APIs !",
      "servicePlansQuantity": 3,
      "serviceStatus": "active",
      "serviceCreatedAt": "Feb 10, 2017 12:00:00 AM",
      "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
      "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
    },
    {
      "serviceId": 1831,
      "serviceGuid": "e15ed959-70a9-4ee7-8c84-056bba5b63d7",
      "serviceURL": "/v2/services/e15ed959-70a9-4ee7-8c84-056bba5b63d7",
      "serviceName": "sendgrid",
      "serviceDescription": "Email Delivery. Simplified.",
      "servicePlansQuantity": 3,
      "serviceStatus": "active",
      "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
      "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
      "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
    },
    {
      "serviceId": 1841,
      "serviceGuid": "515ce213-3ba5-417e-8db9-71d42b093246",
      "serviceURL": "/v2/services/515ce213-3ba5-417e-8db9-71d42b093246",
      "serviceName": "memcachedcloud",
      "serviceDescription": "Enterprise-Class Memcached for Developers",
      "servicePlansQuantity": 7,
      "serviceStatus": "active",
      "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
      "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
      "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
    },
    {
      "serviceId": 1851,
      "serviceGuid": "fe54e2dc-b603-48b0-888e-cd32003713a5",
      "serviceURL": "/v2/services/fe54e2dc-b603-48b0-888e-cd32003713a5",
      "serviceName": "newrelic",
      "serviceDescription": "Manage and monitor your apps",
      "servicePlansQuantity": 1,
      "serviceStatus": "active",
      "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
      "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
      "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
    },
    {
      "serviceId": 1861,
      "serviceGuid": "8b0404f7-ac4f-4003-913e-f586d8340f17",
      "serviceURL": "/v2/services/8b0404f7-ac4f-4003-913e-f586d8340f17",
      "serviceName": "loadimpact",
      "serviceDescription": "Performance testing for DevOps",
      "servicePlansQuantity": 4,
      "serviceStatus": "active",
      "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
      "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
      "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
    },
    {
      "serviceId": 1871,
      "serviceGuid": "50b38b63-deea-4dd9-bfcb-2c1f4d0aadbd",
      "serviceURL": "/v2/services/50b38b63-deea-4dd9-bfcb-2c1f4d0aadbd",
      "serviceName": "ironworker",
      "serviceDescription": "Job Scheduling and Processing",
      "servicePlansQuantity": 3,
      "serviceStatus": "inactive",
      "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
      "serviceUpdatedAt": "Aug 2, 2018 12:00:00 AM",
      "serviceSynchronizedAt": "Sep 12, 2018 12:00:00 AM"
    },
    {
      "serviceId": 1881,
      "serviceGuid": "5a1ca545-2a3a-423f-bafb-21d72061bd63",
      "serviceURL": "/v2/services/5a1ca545-2a3a-423f-bafb-21d72061bd63",
      "serviceName": "cedexisopenmix",
      "serviceDescription": "Openmix Global Cloud and Data Center Load Balancer",
      "servicePlansQuantity": 2,
      "serviceStatus": "active",
      "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
      "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
      "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
    },
    {
      "serviceId": 1891,
      "serviceGuid": "f40ff591-6f1e-4bbf-a4ae-ff88d91c36ff",
      "serviceURL": "/v2/services/f40ff591-6f1e-4bbf-a4ae-ff88d91c36ff",
      "serviceName": "stream",
      "serviceDescription": "Timelines, Build Scalable Newsfeeds & Activity Streams",
      "servicePlansQuantity": 4,
      "serviceStatus": "active",
      "serviceCreatedAt": "Mar 26, 2018 12:00:00 AM",
      "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
      "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
    },
    {
      "serviceId": 1901,
      "serviceGuid": "f4532de3-af6d-4fc9-8dbe-f86fa06711ec",
      "serviceURL": "/v2/services/f4532de3-af6d-4fc9-8dbe-f86fa06711ec",
      "serviceName": "metrics-forwarder",
      "serviceDescription": "Custom metrics service",
      "servicePlansQuantity": 3,
      "serviceStatus": "active",
      "serviceCreatedAt": "Jul 19, 2017 12:00:00 AM",
      "serviceUpdatedAt": "Jan 25, 2019 12:00:00 AM",
      "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
    },
    {
      "serviceId": 1911,
      "serviceGuid": "b5df1be2-f32a-45e3-b916-b479bff9c23d",
      "serviceURL": "/v2/services/b5df1be2-f32a-45e3-b916-b479bff9c23d",
      "serviceName": "gluon",
      "serviceDescription": "Mobile Synchronization and Cloud Integration",
      "servicePlansQuantity": 4,
      "serviceStatus": "active",
      "serviceCreatedAt": "Oct 21, 2016 12:00:00 AM",
      "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
      "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
    },
    {
      "serviceId": 1921,
      "serviceGuid": "eab026f9-e031-400c-990b-b144a0428bca",
      "serviceURL": "/v2/services/eab026f9-e031-400c-990b-b144a0428bca",
      "serviceName": "cloudforge",
      "serviceDescription": "Development Tools In The Cloud",
      "servicePlansQuantity": 3,
      "serviceStatus": "active",
      "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
      "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
    "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
  },
  {
    "serviceId": 1931,
    "serviceGuid": "07ce7df4-66c4-4b55-87cf-a183fb371b65",
    "serviceURL": "/v2/services/07ce7df4-66c4-4b55-87cf-a183fb371b65",
    "serviceName": "p-config-server",
    "serviceDescription": "Config Server for Spring Cloud Applications",
    "servicePlansQuantity": 2,
    "serviceStatus": "active",
    "serviceCreatedAt": "Apr 13, 2016 12:00:00 AM",
    "serviceUpdatedAt": "Sep 4, 2018 12:00:00 AM",
    "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
  },
  {
    "serviceId": 1941,
    "serviceGuid": "2fa8809e-64fc-4b63-975b-555d2bd5b1a1",
    "serviceURL": "/v2/services/2fa8809e-64fc-4b63-975b-555d2bd5b1a1",
    "serviceName": "memcachier",
    "serviceDescription": "The easiest, most advanced memcache.",
    "servicePlansQuantity": 12,
    "serviceStatus": "active",
    "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
    "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
    "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
  },
  {
    "serviceId": 1951,
    "serviceGuid": "7d633c57-abab-48af-be01-c992b9b77ecb",
    "serviceURL": "/v2/services/7d633c57-abab-48af-be01-c992b9b77ecb",
    "serviceName": "Greenplum",
    "serviceDescription": "Greenplum for Pivotal Cloud Foundry",
    "servicePlansQuantity": 1,
    "serviceStatus": "active",
    "serviceCreatedAt": "Jan 31, 2018 12:00:00 AM",
    "serviceUpdatedAt": "Jan 31, 2018 12:00:00 AM",
    "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
  },
  {
    "serviceId": 1961,
    "serviceGuid": "3ba9445c-c709-4153-a343-e4ff5807316a",
    "serviceURL": "/v2/services/3ba9445c-c709-4153-a343-e4ff5807316a",
    "serviceName": "cloudamqp",
    "serviceDescription": "Managed HA RabbitMQ servers in the cloud",
    "servicePlansQuantity": 5,
    "serviceStatus": "active",
    "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
    "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
    "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
  },
  {
```

```
  "serviceId": 1971,
  "serviceGuid": "99b6e48d-f319-4a72-b854-a6e43eea9c3c",
  "serviceURL": "/v2/services/99b6e48d-f319-4a72-b854-a6e43eea9c3c",
  "serviceName": "rediscloud",
  "serviceDescription": "Enterprise-Class Redis for Developers",
  "servicePlansQuantity": 9,
  "serviceStatus": "active",
  "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
  "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 1981,
  "serviceGuid": "120c9e32-463e-4132-89b1-2af5c7a93f64",
  "serviceURL": "/v2/services/120c9e32-463e-4132-89b1-2af5c7a93f64",
  "serviceName": "p-circuit-breaker-dashboard",
  "serviceDescription": "Circuit Breaker Dashboard for Spring Cloud Applications",
  "servicePlansQuantity": 2,
  "serviceStatus": "active",
  "serviceCreatedAt": "Apr 13, 2016 12:00:00 AM",
  "serviceUpdatedAt": "Sep 4, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 1991,
  "serviceGuid": "19b09b99-0bdc-495b-80e4-89ae013a04eb",
  "serviceURL": "/v2/services/19b09b99-0bdc-495b-80e4-89ae013a04eb",
  "serviceName": "app-autoscaler",
  "serviceDescription": "Scales bound applications in response to load",
  "servicePlansQuantity": 1,
  "serviceStatus": "active",
  "serviceCreatedAt": "Aug 20, 2014 12:00:00 AM",
  "serviceUpdatedAt": "Jan 25, 2019 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 2001,
  "serviceGuid": "bc5f4870-0d21-41ee-b968-e14286196b95",
  "serviceURL": "/v2/services/bc5f4870-0d21-41ee-b968-e14286196b95",
  "serviceName": "pubnub",
  "serviceDescription": "Build Realtime Apps that Scale",
  "servicePlansQuantity": 1,
  "serviceStatus": "active",
  "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
  "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 2011,
  "serviceGuid": "34dbc753-34ed-4cf1-9a87-a224dfca569b",
  "serviceURL": "/v2/services/34dbc753-34ed-4cf1-9a87-a224dfca569b",
  "serviceName": "elephantsql",
  "serviceDescription": "PostgreSQL as a Service",
  "servicePlansQuantity": 4,
  "serviceStatus": "active",
  "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
  "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 2021,
  "serviceGuid": "dfb4bee2-c56a-4257-93c4-0499e35637b3",
  "serviceURL": "/v2/services/dfb4bee2-c56a-4257-93c4-0499e35637b3",
  "serviceName": "p-service-registry",
  "serviceDescription": "Service Registry for Spring Cloud Applications",
  "servicePlansQuantity": 2,
  "serviceStatus": "active",
  "serviceCreatedAt": "Apr 13, 2016 12:00:00 AM",
  "serviceUpdatedAt": "Sep 4, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 2031,
  "serviceGuid": "c7ca0fdf-7f24-41e9-840e-08b435de5481",
  "serviceURL": "/v2/services/c7ca0fdf-7f24-41e9-840e-08b435de5481",
  "serviceName": "searchly",
  "serviceDescription": "Search Made Simple. Powered-by Elasticsearch",
  "servicePlansQuantity": 7,
  "serviceStatus": "active",
  "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
  "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 2041,
  "serviceGuid": "0b24bc9b-c267-4607-9d80-6f4c9f32ccdf",
  "serviceURL": "/v2/services/0b24bc9b-c267-4607-9d80-6f4c9f32ccdf",
  "serviceName": "scheduler-for-pcf",
  "serviceDescription": "Scheduler service",
  "servicePlansQuantity": 1,
  "serviceStatus": "active",
  "serviceCreatedAt": "Oct 26, 2017 12:00:00 AM",
  "serviceUpdatedAt": "Sep 25, 2018 12:00:00 AM",
  "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
},
{
  "serviceId": 2051,
  "serviceGuid": "8ba01a99-9590-4069-8972-c96c3e7e8698",
  "serviceURL": "/v2/services/8ba01a99-9590-4069-8972-c96c3e7e8698",
  "serviceName": "blazemeter",
  "serviceDescription": "Performance Testing Platform",
  "servicePlansQuantity": 3,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jan 19, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Nov 26, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2061,
          "serviceGuid": "a98173ff-7c74-4473-ae88-8607977ad29d",
          "serviceURL": "/v2/services/a98173ff-7c74-4473-ae88-8607977ad29d",
          "serviceName": "ssl",
          "serviceDescription": "Upload your SSL certificate for your app(s) on your custom domain",
          "servicePlansQuantity": 1,
          "serviceStatus": "active",
          "serviceCreatedAt": "Jul 7, 2015 12:00:00 AM",
          "serviceUpdatedAt": "Aug 24, 2016 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        }
      ]
    },
    {
      "id": 51,
      "name": "SAP Cloud Platform",
      "platformsSyncDate": "Jan 26, 2019 12:00:00 AM",
      "servicesSyncDate": "Jan 26, 2019 12:00:00 AM",
      "servicesQuantity": 10,
      "services": [
        {
          "serviceId": 2071,
          "serviceGuid": "cfffaca0-92d0-423f-87df-3a0c57a48036",
          "serviceURL": "/v2/services/cfffaca0-92d0-423f-87df-3a0c57a48036",
          "serviceName": "connectivity",
          "serviceDescription": "Establishes a secure and reliable connectivity between cloud applications and on-premise systems.",
          "servicePlansQuantity": 0,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 4, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jan 24, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2081,
          "serviceGuid": "dcd98b5f-d4c3-4f44-b4a6-0f62435ceae3",
          "serviceURL": "/v2/services/dcd98b5f-d4c3-4f44-b4a6-0f62435ceae3",
          "serviceName": "application-logs",
          "serviceDescription": "Create, store, access, and analyze application logs.",
          "servicePlansQuantity": 0,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 4, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Dec 20, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2091,
          "serviceGuid": "2f47932d-3208-4767-b8fb-c10467dc551b",
          "serviceURL": "/v2/services/2f47932d-3208-4767-b8fb-c10467dc551b",
          "serviceName": "xsuaa",
          "serviceDescription": "Manage application authorizations and trust to identity
providers.",
          "servicePlansQuantity": 0,
          "serviceStatus": "active",
          "serviceCreatedAt": "Aug 5, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jan 24, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2101,
          "serviceGuid": "610d1a84-c678-46a4-a07e-95f3b1603fbf",
          "serviceURL": "/v2/services/610d1a84-c678-46a4-a07e-95f3b1603fbf",
          "serviceName": "hana",
          "serviceDescription": "Manage schemas and HDI containers on an existing SAP HANA
database.",
          "servicePlansQuantity": 0,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 16, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Dec 6, 2018 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2111,
          "serviceGuid": "75311192-394d-4a4c-9b9d-94dd816b3ebc",
          "serviceURL": "/v2/services/75311192-394d-4a4c-9b9d-94dd816b3ebc",
          "serviceName": "auditlog-api",
          "serviceDescription": "Auditlog API",
          "servicePlansQuantity": 0,
          "serviceStatus": "active",
          "serviceCreatedAt": "May 25, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jan 24, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2121,
          "serviceGuid": "52e988db-7b8b-47fd-993b-81224bacbf61",
          "serviceURL": "/v2/services/52e988db-7b8b-47fd-993b-81224bacbf61",
          "serviceName": "destination",
          "serviceDescription": "Provides a secure and reliable access to destination
configurations",
          "servicePlansQuantity": 0,
          "serviceStatus": "active",
          "serviceCreatedAt": "Aug 31, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jan 24, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2131,
          "serviceGuid": "f62a7fd0-4d87-4aa8-ad9e-05e6edae3d51",
          "serviceURL": "/v2/services/f62a7fd0-4d87-4aa8-ad9e-05e6edae3d51",
          "serviceName": "html5-apps-repo",
          "serviceDescription": "Enables storage of HTML5 applications and provides
runtime environment for HTML5 applications.",
          "servicePlansQuantity": 0,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 21, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jan 24, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2141,
          "serviceGuid": "afdd9c14-0d8b-4364-9ec6-ebd6975a4c79",
          "serviceURL": "/v2/services/afdd9c14-0d8b-4364-9ec6-ebd6975a4c79",
          "serviceName": "managed-hana",
          "serviceDescription": "Creates service instances at runtime of: Manage schemas
and HDI containers on an existing SAP HANA database.",
          "servicePlansQuantity": 0,
          "serviceStatus": "active",
          "serviceCreatedAt": "Dec 16, 2016 12:00:00 AM",
          "serviceUpdatedAt": "Jan 24, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2147,
          "serviceGuid": "9af2f2d7-822a-4a43-9fdb-fdf9d2382b56",
          "serviceURL": "/v2/services/9af2f2d7-822a-4a43-9fdb-fdf9d2382b56",
          "serviceName": "saas-registry",
          "serviceDescription": "Service for application providers to register multitenant
applications and services",
          "servicePlansQuantity": 0,
          "serviceStatus": "active",
          "serviceCreatedAt": "Sep 28, 2017 12:00:00 AM",
          "serviceUpdatedAt": "Jan 24, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        },
        {
          "serviceId": 2155,
          "serviceGuid": "8859d0f2-55f0-4640-ab55-ab7ad7963ac3",
          "serviceURL": "/v2/services/8859d0f2-55f0-4640-ab55-ab7ad7963ac3",
          "serviceName": "auditlog-management",
          "serviceDescription": "Auditlog Management - Retrieve logs and change
retention",
          "servicePlansQuantity": 0,
          "serviceStatus": "active",
          "serviceCreatedAt": "Nov 22, 2018 12:00:00 AM",
          "serviceUpdatedAt": "Jan 24, 2019 12:00:00 AM",
          "serviceSynchronizedAt": "Jan 26, 2019 12:00:00 AM"
        }
      ]
    }
  ]
}
```

A1.5 Self-Evaluation Form



Q4)	*I easily learned how to use the framework.*	
Q5)	*I used the framework in the way I wanted to.*	
Q6)	*I understood what happened in the interaction with the framework.*	
Q7)	*I easily executed the proposed tasks with the framework.*	

Q8)	*In the event of the framework giving me an answer different than mine, I would consider it for use anyway. This means: the framework's suggestion are adequate/make sense.*	

Scale	Value
totally agree	5
agree	4
neutral/don’t know	3
disagree	2
strongly disagree	1

A1.6 Background

(In Portuguese)

Explicação sobre os seguintes conceitos:

1) Ecossistemas de Software

Exemplo: Eclipse, Windows, Android, Bluemix, Azure

2) Arquitetura Orientada a Microsserviços

Como evolução SOA, Exemplo: SQL Server as a service.

3) COTS (component out of the shelf) Selection

Descoberta, Comparação, Avaliação, Seleção de componentes reutilizáveis