IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 13, 2020, accepted June 29, 2020, date of publication July 14, 2020, date of current version July 28, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3009079

A Taxonomy of Quality Metrics for Cloud Services

XIMENA GUERRON[1,2], SILVIA ABRAHÃO[1], (Member, IEEE), EMILIO INSFRAN[1], (Member, IEEE), MARTA FERNÁNDEZ-DIEGO[3], AND FERNANDO GONZÁLEZ-LADRÓN-DE-GUEVARA[3]

[1]Instituto Universitario Mixto de Tecnología Informática, Universitat Politècnica de València (UPV), 46022 Valencia, Spain
[2]Facultad de Ingeniería, Ciencias Físicas y Matemática, Universidad Central del Ecuador (UCE), Quito 170129, Ecuador
[3]Departamento de Organización de Empresas, Universitat Politècnica de València (UPV), 46022 Valencia, Spain

Corresponding author: Silvia Abrahão (sabrahao@dsic.upv.es)

This work was supported by the Spanish Ministry of Science, Innovation and Universities through the Adapt@Cloud Project under Grant TIN2017-84550-R. The work of Ximena Guerron was supported in part by the Universidad Central del Ecuador (UCE), and in part by the Banco Central del Ecuador.

ABSTRACT A large number of metrics with which to assess the quality of cloud services have been proposed over the last years. However, this knowledge is still dispersed, and stakeholders have little or no guidance when choosing metrics that will be suitable to evaluate their cloud services. The objective of this paper is, therefore, to systematically identify, taxonomically classify, and compare existing quality of service (QoS) metrics in the cloud computing domain. We conducted a systematic literature review of 84 studies selected from a set of 4333 studies that were published from 2006 to November 2018. We specifically identified 470 metric operationalizations that were then classified using a taxonomy, which is also introduced in this paper. The data extracted from the metrics were subsequently analyzed using thematic analysis. The findings indicated that most metrics evaluate quality attributes related to performance efficiency (64%) and that there is a need for metrics that evaluate other characteristics, such as security and compatibility. The majority of the metrics are used during the Operation phase of the cloud services and are applied to the running service. Our results also revealed that metrics for cloud services are still in the early stages of maturity – only 10% of the metrics had been empirically validated. The proposed taxonomy can be used by practitioners as a guideline when specifying service level objectives or deciding which metric is best suited to the evaluation of their cloud services, and by researchers as a comprehensive quality framework in which to evaluate their approaches.

INDEX TERMS Software quality, metrics, cloud services, systematic literature review.

I. INTRODUCTION

Information and communication technology (ICT) companies have widely exploited cloud computing as a strategic opportunity to meet business objectives and remain competitive in the market. The National Institute of Standards and Technology (NIST) defined cloud computing as a model that allows ubiquitous, convenient and on-demand access to a shared set of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be quickly provisioned and released with minimal management effort or interactions with the service provider [1]. According to NIST, a cloud service has five characteristics: i) on demand self-service, where a consumer can unilaterally provision computing capabilities as needed automatically without interacting with providers; ii) broad network access, where capabilities are available over the network; iii) resource pooling, where provider's resources are pooled to serve multiple consumers using a multi-tenant model with resources dynamically assigned and reassigned on demand; iv) rapid elasticity, where capabilities can be elastically provisioned and released; and v) measured service, where resource usage are monitored, controlled, and reported.

Cloud service providers (CSP) are continuously competing for customers. This competition was, in its beginnings, based primarily on the cost of the resources provided, but quantifying and comparing the actual capabilities is now becoming more critical. Quality of Service (QoS), therefore, plays a critical role in monitoring, controlling, reporting, and billing [1]. As an example, if cloud service performance levels become unpredictable or do not meet expectations, customers will refuse the service or avoid its adoption. However,

The associate editor coordinating the review of this manuscript and approving it for publication was Porfirio Tramontana.

if expectations are met or exceeded, the cloud provider's reputation will increase and its services will, therefore, be better recognized and used [2]. The service providers must consequently consider conscious investments and efforts in order to continue in business, because any improvement made to the quality of service will be perceived and valued by their customers. Lastly, modern service development approaches based on agility and DevOps techniques require the continuous monitoring of cloud services in order to allow the dynamic adaption and evolution of the service behavior in short cycles.

The increasing interest in addressing the challenges associated with the quality of cloud services has, in recent years, led to the proposal of numerous metrics with which to assess the quality of cloud services [3], [4]. Metrics provide useful data that can be analyzed and used in technical, operational, and business decisions throughout the organization. However, the current knowledge of metrics for cloud services is still dispersed. No study has, to the best of our knowledge, systematically identified, analyzed, and consolidated the knowledge regarding the existing metrics proposed for the evaluation of the internal and external quality of cloud services. This means that cloud stakeholders (e.g., customers, providers, brokers, cloud architects) have little or no guidance when choosing suitable metrics with which to evaluate their cloud services in different cloud service models (e.g., SaaS, PaaS, IaaS). One key difficulty is the selection of metrics that can be applied to specific cloud artifacts (e.g., SLA specification, cloud architecture or the actual cloud service) in different service lifecycle phases.

The objective of this paper is, therefore, to systematically identify, taxonomically classify, and compare existing QoS metrics in the cloud computing domain according to the ISO/IEC 25010 quality model [5]. Since Systematic Literature Reviews (SLRs) are useful as regards objectively finding and aggregating all the existing evidence concerning an area of study [6], we used this methodology to identify and analyze what metrics have been used to evaluate the internal and external quality of cloud services and how they were measured and used.

We selected 84 primary studies from a set of 4333 papers that were published from 2006 to November 2018. A total of 470 metric operationalizations were identified and classified. Overall, our study makes the following contributions:

- A catalogue of quality metrics for cloud services retrieved from the relevant literature.
- A taxonomy of metrics for cloud services hierarchically organized according to the quality model from the ISO/IEC 25010 [5] and the concepts defined by the NIST SP 800-145 [1] and NIST SP 800-146 [7]. The purpose of this taxonomy is to systematically classify and compare the metrics according to different criteria.
- A metamodel that supports the proposed taxonomy by representing the different concepts and relationships among them.

Our taxonomy supports different cloud stakeholders' viewpoints by allowing them to make better-informed decisions through the provision of an approach with which to understand why, where, and how metrics can be applied to their cloud artifacts. The results of this study could specifically be useful as a guideline for practitioners when defining service level agreement objectives or deciding which metrics are best suited to the evaluation of their cloud artifacts. Our findings are also useful for researchers, as we identify future research efforts that should be made in order to advance the state of the art of the assessment of cloud services.

The remainder of the paper is organized as follows: In Section II, we review the related works, outlining the differences and gaps identified. In Section III, we explain the research method used to build the taxonomy and identify, classify, and analyze the existing metrics for cloud services. In Section IV, we present the process followed to create and refine the proposed taxonomy, while in Section V, we aggregate the results. In Section VI, we comment on the threats to validity. Finally, in Section VII, we present our conclusions and directions for future work.

II. RELATED WORK

In the last years, a large body of research has focused on developing frameworks, tools, and technologies with which to assess or monitor QoS in cloud environments. As this study focuses on metrics, we first discuss existing taxonomies and surveys whose purpose is to classify metrics for cloud services. We then discuss existing secondary studies related to QoS evaluation in cloud computing.

A. EXISTING TAXONOMIES AND SURVEYS

In order to verify that a similar taxonomy of metrics for cloud services had not already been reported, we searched IEEE Xplore, ACM Digital Library, SpringerLink, and Science Direct, using the following search string: (metric or measure) AND cloud AND (taxonomy or ontology or classification). The metadata used to carry out the search were title, abstract and keywords.

None of the studies retrieved was related to our research questions detailed in Section III.A. Nevertheless, we found some related studies that focused on some specific approaches or quality characteristics. For instance, Li *et al.* [4] proposed a taxonomy of the performance evaluation of commercial cloud services. This taxonomy was constructed in two dimensions: performance feature and experiment. The performance feature was further decomposed into 4 physical property elements (e.g., communication, storage) and 7 capacity elements (e.g., availability, reliability), while the experiment feature was further decomposed into 5 environmental scenes (e.g., experimental resources, such as single cloud provider vs multiple cloud providers) and 15 operational scenes (e.g., processes with human interference, such as repeating an experiment for a period of time). A scene is considered to be an atomic unit in which to construct a complete experiment for the evaluation

of a commercial cloud service. Although this taxonomy is useful to analyze existing evaluation practices and design new experiments, it provides only seven metrics, one per each capacity part (i.e., speed, uptime ratio, latency, failure rate, actual throughput, scalability, variability) and is limited to performance evaluation.

The properties of trust modeled by Habib *et al.* [8] concerned the cloud providers' capabilities. This was done using a Consensus Assessment Initiative Questionnaire (CAIQ) designed by the Cloud Security Alliance (CSA). The aim of the framework was to verify the properties modeled from CAIQ controls and provide a solution as regards assessing cloud providers' claims. To this end, the authors introduced a taxonomy with which to map and classify CAIQ controls into trust properties, including the type control and the validation authorities. Finally, a decision model was proposed that takes both the verification of trust properties (from the taxonomy) and the consumers' requirements into account in order to determine cloud providers' trustworthiness. The focus of this work is, therefore, not on evaluating cloud services, but rather on evaluating the trustworthiness of cloud providers.

Herbst *et al.* [9] proposed a taxonomy for cloud metrics focused on four system properties: the elasticity of the cloud service, performance isolation between the tenants and the resulting performance variability, the availability of cloud services, and the operational risk of running a production system in a cloud environment. The taxonomy proposed four levels of abstraction for measurement and assessment metrics (i.e., traditional performance metrics, cloud infrastructure metrics, policy metrics and metrics for managerial decisions). Its goal was to enable a comparison between cloud offerings and technology and to provide a common understanding to providers, customers, and end-users. Although the authors proposed a hierarchical taxonomy for cloud-relevant metrics and their corresponding measurement approaches, these metrics covered only performance properties.

Elasticity is a critical factor for cloud services, and several studies have focused on how to measure it. In [10], the authors identified the requirements and challenges as regards managing elastic resources for a PaaS provider, along with possible solutions. Coutinho *et al.* [11] proposed definitions, metrics, and tools with which to measure elasticity. Both studies analyzed elasticity as an isolated property. This is not, however, sufficient when considering the impact that this property has on other quality attributes of cloud service quality, such as scalability and efficiency [12].

Other surveys whose objective was to analyze and classify metrics for cloud services have also been published. Jelassi *et al.* [13] took performance as a reference and presented several QoS parameters and methods that could be used to measure this characteristic. The QoS parameters were based on nine properties and their corresponding measurement approaches (e.g., throughput expressed in requests per minute), while the methods represented the available methods (i.e., admission control, resource management, waiting queue management), techniques and mechanisms (i.e., scheduling and monitoring) with which to ensure and guarantee quality. Although the survey discussed some current approaches for the measurement of performance, this is an informal survey, and a more systematic study analyzing the existing metrics employed to measure both performance and other QoS characteristics is required.

Bardsiri & Hashemi [14] categorized metrics into four groups (performance, economics, security and general). A set of metrics with which to measure specific features was then suggested for each group. The survey covered the main service types (SaaS. PaaS and IaaS) from the perspective of service providers. The most important limitation of this study is that only the names of the metrics were provided (e.g., flexibility, readability, and service modularity) and it is unclear how these metrics can be measured.

Although some studies have proposed several metrics with which to assess the quality of cloud services, we are not aware of a study that has consolidated this knowledge and classified it according to internal and external QoS characteristics, and has aligned it to quality standards (e.g., ISO/IEC 25010) and cloud computing concepts (e.g., NIST SP 800-145 [1], NIST SP 800-146 [7]).

B. EXISTING SECONDARY STUDIES

The two forms of secondary studies most frequently used in Software Engineering are systematic literature reviews (SLRs) and mapping studies. A systematic mapping study provides an overview of a research area by classifying papers and results on the basis of relevant categories and counting the frequency of papers in each of those categories. Systematic mappings are exploratory in nature, whereas the purpose of SLRs is to provide synthesized summaries in order to answer well-defined research questions [15]. Several systematic reviews and mapping studies related to the quality of cloud services have been proposed in the last few years.

Abdelmaboud *et al.* [16] presented a systematic mapping in order to survey the existing approaches employed to assess the quality of cloud services. The results showed that the type of services addressed was focused principally on Infrastructure as a Service – IaaS (48%) and Software as a Service – SaaS (36%). The contribution types were mainly methods (48%) and models (32%), and the research types focused on validation studies (64%). The stakeholders' viewpoints were limited to the providers and consumers of cloud services.

Other studies, such as that of Li *et al.* [17] conducted a systematic literature review focusing on a specific set of quality attributes and metrics. These authors obtained a catalog of 97 metrics focused on the evaluation of cloud service performance, economics, and security. This subsequently resulted in the definition of a framework with which to support the selection of commercial cloud services that covered IaaS and PaaS but not SaaS. Later, Li *et al.* [18] used their previous systematic literature review as a baseline and

extended it in order to investigate the cloud service evaluation procedures, properties, metrics, benchmarks, and experimental environments involved in the evaluation of commercial cloud services. The results showed that the existing works have employed many metrics to measure various performance features, in addition to the cost of commercial cloud services. Finally, Lehrig *et al.* [19] conducted a systematic literature review in order to examine the definitions and metrics related to the scalability, elasticity, and efficiency of cloud services. Their source was limited to Google Scholar. Their results showed a common concept and recommended metrics with which to evaluate these attributes.

Scheuner & Leitner [20] presented a multi-vocal review on Function as a Service (FaaS) performance evaluation. FaaS provides an entirely new cloud service model which allows to achieve Serveless architectures (microservices). Some of its advantages are dynamic resource provisioning and auto scaling. This work was based on academic and grey literature, and examined current trends, platform configurations, and performance characteristics. However, the performance characteristics were limited to four attributes (i.e., platform overload, workload concurrency, instance duration, and infrastructure inspection) and no metrics were collected.

Kanashi *et al.* [21] presented a systematic literature review in order to identify and classify the existing knowledge of QoS in fog computing. Fog computing decentralizes services and resources outside the cloud and near the end devices. This work addressed three management categories (service/ resource, communication, and application) with eleven QoS factors (i.e., throughput, deadline, response time, resource utilization, cost, execution time, energy consumption, reliability, availability, scalability, and security). Finally, the authors ranked the relevance of the QoS factors usage. Response time, cost, and resource utilization were found to be the most used whereas availability and scalability were the least used. Again, the study does not provided a full coverage of quality characteristics, and no metrics were collected.

Table 1 presents a comparison of the aforementioned secondary studies. These studies were, overall, limited to a specific quality characteristic (e.g., performance) or stakeholder viewpoint (e.g., the cloud service provider).

None of these studies collected and analyzed all the existing evidence regarding metrics for cloud services or introduced a reference model or taxonomy for a process-centric classification and a comparison of the metrics collected.

This means that the knowledge of quality metrics for cloud services is dispersed, and stakeholders have little or no guidance when choosing suitable metrics with which to evaluate their own or acquired cloud services. Considering the importance of cloud computing and the relative maturity of this field, a consolidation of existing evidence on quality metrics for cloud services is, therefore, timely.

III. METHOD

Our objective is to systematically identify and taxonomically classify available evidence on quality metrics for cloud services and to provide a holistic comparison so as to analyze the potential limitations of existing research.

The purpose of the proposed taxonomy is, therefore, to increase the body of knowledge regarding the evaluation of cloud services by: (i) providing a set of quality metrics for cloud services; (ii) identifying the quality attributes that measure the selected metrics and align them with the quality characteristics proposed by the ISO/IEC 25010 and the cloud computing concepts defined by the NIST SP 800-145 [1] and NIST SP 800-146 [7], i.e., service models and stakeholders' viewpoint, and (iii) identifying limitations in previous work in order to suggest an agenda for further research.

The taxonomy will provide a common terminology for the concepts involved in the evaluation of cloud services. It will be supported by a metamodel that represents the different concepts and relationships among the concepts, thus facilitating communication and allowing its subsequent reuse by practitioners and researchers.

We identified the existing quality metrics for cloud services and created the proposed taxonomy by conducting a systematic literature review according to the guidelines proposed by Kitchenham *et al.* [6]. This research method involves three main phases: planning a review, conducting a review, and reporting a review. Figure 1 describes the process followed to create the proposed taxonomy. It includes the activities and artifacts, along with the inputs and outputs of the activities.

The activities concerning the planning, conducting, and reporting stages are detailed in this section, while the creation of the taxonomy and its refinement is reported in Section IV.

A. PLANNING

The definition of a review protocol provides a framework in which to document the required study design decisions with the aim of minimizing bias. The set of activities that we performed to define a review protocol were the following: definition of a research question, definition of the search strategy and definition of the inclusion and exclusion criteria. These activities are described in the following sub-sections.

1) RESEARCH QUESTION

The PICOC strategy, which was suggested by Petticrew *et al.* [22] and is used to frame the research question elements in order to develop the review protocol, has been employed herein. The PICOC elements utilized in this study are:

- *Intervention (I)*: Characterization, Extracting data, Synthesis.
- *Comparison (C)*: A comparison, carried out by mapping the primary studies onto a taxonomy (characterization framework)
- *Outcome (O)*: A taxonomy of metrics for cloud services aligned to the ISO/IEC 25010 [5], the NIST SP 800-145 [1] and the NIST SP 800-146 [7] standards.
- *Context (C)*: A systematic investigation in order to consolidate peer-reviewed research.

TABLE 1. Comparing the search strategies of existing SIrs.

Options	Li *et al.* 2012	Li *et al.* 2013	Abdelmaboud *et al.* 2015	Lehrig *et al.* 2015	Scheuner & Leitner 2020
Time frame	2006-2011	2006-2011	01/2008-12/2012	2005-04/2014	2016-2019
Research Questions	What metrics have been used for the evaluation of commercial cloud services?	What metrics have been used for the evaluation of commercial cloud services?	Which topics related to QoS in cloud computing have been investigated and to what extent?	What metrics can be used to compare the scalability, efficiency and elasticity of different cloud computing services, and how are they measured and used?	Which general performance characteristics (e.g., platform overhead / cold starts) are commonly evaluated?
Search strategy	Database search	Database search	Database search	Database search	Database search, web search & complementary search, snowballing
Search engines	a set of popular digital publication databases (Not specified)	ACM Digital Library Google Scholar IEEE Xplore ScienceDirect SpringerLink	IEEE Xplore ACM Digital Library Springer Link ScienceDirect Scopus Google Scholar	Google Scholar	ACM Digital Library IEEE Explore ISI Web of Science Science Direct SpringerLink Wiley InterScience Scopus, Google Scholar Google Search, Twitter Search Hacker News Algolia Search, Reddit Search, Medium Search
Search String	N/A	(“cloud computing” OR “cloud platform” OR “cloud provider” OR “cloud service” OR “cloud offering”) AND (evaluation OR evaluating OR evaluate OR evaluated OR experiment OR benchmark OR metric OR simulation) AND (<Cloud provider’s name> OR. . .)	(QoS OR “quality of service” OR SLA OR “service level agreement” OR “quality”) AND (“cloud computing” OR “cloud services”)	Phrase Def.: (scalability OR elasticity OR efficiency) AND “cloud computing” Phrase Met.: metric AND (scalability OR elasticity OR efficiency) AND “cloud computing”	(serverless OR FaaS) AND (performance OR benchmark) AND experiment AND lambda. (serverless OR FaaS) AND (performance OR benchmark). serverless. serverless benchmark.
# retrieved papers	N/A	4017	515	418	956 academic literature 663 grey literature
# primary studies	46	82	67 (only 6 related to metrics)	20	112 (51 academic and 61 grey literature)
# metrics	97	80	N/A	N/A	N/A

NA= Not available.

We followed the guidelines provided by Easterbrook *et al.* [15] and Kitchenham & Charters [6] in order to define an exploratory and descriptive research question. The main research question addressed in this study is: *what metrics have been used to evaluate the internal and external quality of cloud services and how are they measured and used?*

Internal quality attributes are those attributes of a software artifact that can be measured on the basis of knowledge of the artifact alone [23]. Examples of internal quality attributes of cloud artifacts are: i) capacity, which can be measured in terms of resource capacity such as storage with size in gigabytes, or network capacity, which can be measured by studying the number of available connections, and ii) density, which can be measured by focusing on the number of applications or number of virtual machines. Other examples are: iii) complexity, which can be measured in terms of the cloud service’s source code structure, or service interface definition complexity (e.g., Web Services Description Language (WSDL) interfaces), which can be measured as minimal refactoring effort, and iv) legibility, which can be measured by means of a readability metric for web service descriptions (WSDL specifications).

External quality attributes are those attributes that cannot be measured using only the knowledge of the software artifact but can be measured by taking into account the artifact, its environment and the interactions between the artifact and the environment [23]. Examples of external quality attributes are reliability, performance, usability and maintainability.

This research question will allow us to understand and summarize the current evidence regarding the existing metrics and identify areas for further research. Since this main question is very general, we refined it into finer-grained questions. In particular, we wish to make explicit which characteristic and quality attribute is being measured, how the metric has been used, to what type of service the metrics have been applied, for which type of stakeholders the metrics are useful, and which method was applied to validate the metrics. The resulting questions are, therefore, the following:

- **RQ1**: What quality characteristics and attributes were evaluated?

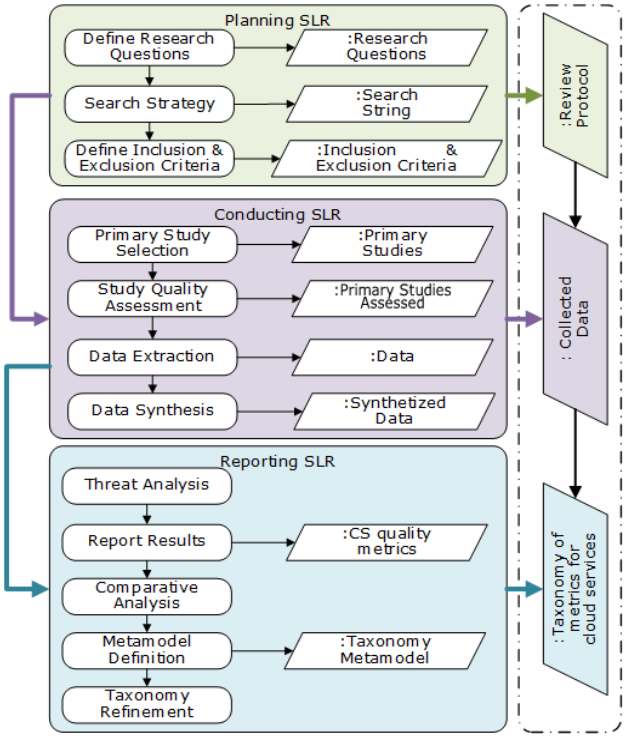

FIGURE 1. Phases of research method.

- **RQ2**: What type of metrics were they?
  - RQ2.1: If the metric is base, what measurement method and unit were used to calculate it?
  - RQ2.2: If the metric is derived, what measurement function and unit were used to calculate it?
- **RQ3**: Are there tools with which to support the measurement process, and, if so, what are they?
- **RQ4**: What type of measurement results from the metrics provided?
- **RQ5**: During which phases of the cloud service lifecycle were these metrics used?
- **RQ6**: For which type of stakeholders (cloud roles) are these metrics relevant?
- **RQ7**: To what type of cloud service (i.e., SaaS, PaaS, IaaS) were these metrics applied?
- **RQ8**: What cloud artifacts or resources were measured?
- **RQ9**: Which validation method was used to provide evidence about the metrics' validity and usefulness?

2) SEARCH STRATEGY

We defined an unbiased and effective search by establishing an accurate publication time interval, defining a search *string,* and employing well-known automatic searchers in digital libraries, and we complemented these searchers with a manual search in order to avoid a possible lack of relevant works. The period reviewed included studies that were published from 2006 until November 2018. We selected this date because Amazon Web Services (a pioneer cloud service) was launched in that year.

We applied a search process that combined both automated searches of selected digital libraries and additional manual searches of the most relevant conferences and journals if they were missing. The digital libraries selected were IEEE Xplore, ACM Digital Library, SpringerLink and Science Direct. These are the most commonly used sources in software engineering [24]. It is worth noting that Google Scholar was not selected as a data source because of the low precision of search results and the generation of many irrelevant results [25]. The following steps were applied in order to build the search string:

1. Derivation of major terms (keywords) from the research question;
2. Identification of alternative spellings and synonyms for major terms;
3. The usage of the Boolean OR in order to incorporate alternative spellings and synonyms;
4. The usage of the Boolean AND in order to link the major terms.

We then modified and combined these search terms so as to build a set of candidate search strings. Several pilot searches were carried out, and the search string was refined as many times as necessary in order to improve the completeness of the results and find the most suitable search string. The completeness of the results was assessed using a set of ten known studies. Table 2 presents the resulting search string.

TABLE 2. Keywords and Related Terms.

Keywords	Alternative Terms & Synonyms
Metric Quality	((metric* OR measur*) AND (QoS OR "quality of service" OR "quality model" OR "evaluation model" OR "assessment model" OR "quality in cloud" OR "quality of cloud") AND
Cloud	(cloud*))

The search was conducted by applying the search string to the same metadata (i.e., title, abstract, and keywords) of each data source, signifying that the syntax of the search string was adapted to be applied in each digital library. The final search string used in each source is available on the website that accompanies this paper (*https://bit.ly/taxonomyqoscs*).

In order to ensure the search quality, we verified that top-ranked journals and conferences (see Table 3) relevant to the cloud computing and software quality domains were included in the digital libraries. The list of journals was obtained from the JCR impact factor. The list of conferences was similarly obtained from the top-ranked conferences based on the CORE conference ranking (http://www.core.edu.au/conference-portal), and we selected those that had a CORE A* and A classification. In particular, we verified that all the editions of each conference proceedings and journal from 2006 to 2018 were indexed in at least one of the digital libraries. We then performed a manual search to attain those editions that were missing.

TABLE 3. List of Journals and Conferences.

Type	Name
Journals	IEEE Transactions on Cloud Computing (TCC)
	Journal of Cloud Computing
	ACM Transactions on the Web (TWEB)
	IEEE Internet Computing
	World Wide Web Journal
	ACM Transactions on Internet Technology (TOIT)
	ACM Transactions on Software Engineering and Methodology (TOSEM)
	Information and Software Technology (IST)
	Journal of Cloud Computing: Advances, Systems and Applications (JoCCASA)
	Journal of Systems and Software (JSS)
	IEEE Software
	IEEE Transactions on Software Engineering (TSE)
	Software Quality Journal (SQJ)
	Empirical Software Engineering (EMSE)
Conferences	International Conference on Software Engineering (ICSE)
	International Conference on Service Oriented Computing (ICSOC)
	International Conference on Web Services (ICWS)
	International Conference on Services Computing (SCC)
	International World Wide Web Conference (WWW)
	Web Information Systems Engineering (WISE)
	International Symposium on Empirical Software Engineering and Measurement (ESEM)
	International Conference on Evaluation and Assessment in Software Engineering (EASE)

3) INCLUSION AND EXCLUSION CRITERIA

We used the following inclusion and exclusion criteria to select candidate papers for our study:

- Inclusion Criteria: (1) Studies in the form of a scientific peer-reviewed paper; (2) Studies that propose or use metrics to assess the internal or external quality of cloud services; (3) Studies that introduce frameworks or methods to evaluate the QoS of cloud services; and (4) Studies that report empirical studies whose objective is to validate the usefulness of metrics for cloud services.
- Exclusion Criteria: (1) Studies that propose metrics that are not related to the internal or external quality of the cloud services; (2) Studies that propose metrics, but do not explain how to measure them; (3) Studies that present quality attributes, but do not propose metrics with which to measure them; (4) Editorials, abstracts or short papers (shorter than five pages); (5) Duplicate papers of the same study found in different sources; (6) Studies not written in English.

4) PROTOCOL EVALUATION

As suggested by Brereton *et al.* [24], we externally evaluated the protocol before its execution. We asked an external expert, who had experience in conducting SLRs, for feedback. The main contribution was the refinement of the definition of the quality items used to assess the primary studies. In particular the inclusion of question 7 to assess the usefulness of the metrics in practice, and to improve the value items of questions 8 and 10 which are related to the contribution and limitations of the studies. Later, in the data extraction criteria, the expert suggested to include indicator as an additional option for the metric type criterion. We also performed a pilot study with the expert in order to test and improve the search strategy and the inclusion and exclusion criteria.

B. CONDUCTING THE STUDY

As part of the review protocol, we also specified a set of processes that would be conducted when performing the SLR: primary study selection process, study quality assessment process, data extraction process and data synthesis process.

1) SELECTION OF PRIMARY STUDIES

In order to identify the primary studies using the predefined search string, we customized the automated search to each digital library. We then performed three main activities: a quick scanning of all the studies retrieved after carrying out the automated and manual searches, a full reading of the selected studies, and team meetings in order to reach agreements among the reviewers in the case of any discrepancies.

1. *Quick scanning*: relevant primary studies were initially selected by scanning the title, keywords and abstract of the paper. In some cases, we also reviewed the introduction and conclusions.
2. *Full reading*: when the decision regarding inclusion or exclusion was not clear, a further review was required, and the text was, therefore, read in full.
3. *Team meetings*: if doubts still remain, team meetings were arranged to discuss them and reach an agreement.

In order to assess the accuracy of the primary study selection phase, three different authors executed a second iteration of a random sample of papers containing ten of the studies included in the first interaction and ten that had been excluded. The level of agreement among the researchers was assessed using the Fleiss' Kappa index [26]. The value of the index was between 0 (not coincident) and 1 (fully coincident).

2) QUALITY ASSESSMENT

There is common agreement that the quality of the chosen primary studies is critical if trustworthy results are to be obtained. We, therefore, defined a checklist according to the criteria proposed by Kitchenham and Charters [6] in order to assess the quality of the selected primary studies. The quality of the primary studies was scored on the basis of how well they satisfied the ten quality items. Each criterion was assessed using a predefined scale (Y, N, P) to indicate whether the study fully complied (Y = 1), partially complied (P = 0.5), or did not comply (N = 0) with those items.

The overall quality of a paper was calculated by summing up all the scores attained for the quality items. The highest score a paper could attain as regards quality was, therefore, 10 points, signifying that the study satisfied all the quality criteria. The quality threshold established was that of employing the mean as the cutoff point, which was equivalent to

the second quartile (5 points), in order to minimize bias and maximize internal and external validity. This signifies that any studies that scored less than the minimum score were no longer considered to be primary studies and were discarded from the review. Table 4 shows the quality items and their criteria.

TABLE 4. Quality Items Used to Assess Primary Studies.

Scale	Quality Question
Q1. (Motivation) Is the research problem clearly specified?	
1	Explicit problem description
0.5	General problem description
0	No problem description
Q2. (Aim) Are the research aim(s)/objective(s) clearly established?	
1	Explicit aims/objectives
0.5	General aims/objectives
0	No aims/objectives
Q3. (Context) Is the context of the study clearly specified?	
1	Explicit problem context supported by references
0.5	General problem context supported by references
0	No problem context description
Q4. (Data) Are the metrics used to assess the quality of cloud services clearly defined?	
1	Explicit metric description
0.5	General metric description
0	No metric description
Q5. (Data) Are the measurement methods or functions used to calculate the metrics clearly defined?	
1	Explicit description of measurement methods/functions
0.5	General description of measurement methods/functions
0	No description of measurement methods/ functions
Q6. (Usefulness) Are the metric(s) empirically validated?	
1	Explicit empirical validation of metrics
0.5	No validation but a proof of concept
0	No validation
Q7. (Usefulness) Is there sufficient evidence to show how the metrics can be used in practice?	
1	Explicit use of metrics supported by practice
0.5	General use of metrics supported by practice
0	Not supported by practice
Q8. (Contributions) Are the contributions/results of the paper discussed?	
1	Explicit list of study contributions/results
0.5	General discussion about study contributions/results
0	No description of the study contributions/results
Q9. (Insights) Are the insights/lessons learned of the study reported?	
1	Explicit list of insights and/or lessons learned
0.5	General discussion of insights or lessons learned
0	No description of insights or lessons learned
Q10. (Limitations) Are the limitations of the study discussed?	
1	Explicit list of study limitations
0.5	General description of study limitations
0	No description of limitations

We also assessed the quality of the venue at which each primary study was published. This may indicate the potential impact and influence of the selected studies. To do this, we assessed two criteria: (1) the relevance of the journal or conference where the paper was published; (2) the number of paper citations.

With regard to the first criterion, we assessed the impact of the publication using in the CORE-ERA ranking for conference papers, and the impact factor in Journal Citations Reports (JCR) for journal papers. And regarding the second criterion, the number of citations for each primary study was assessed according to Google Scholar. Since recent articles tend to have fewer citations and should not be penalized for this, we followed a strategy similar to that of [27] and established the year 2017 in order to differentiate early publications. Table 5 shows the criteria and scale that have been used to assess the potential impact and influence of the selected primary studies.

TABLE 5. Potential Impact and Influence of Primary Studies.

Publication / Citation	Description	Points
PUBLICATION IMPACT		
Very relevant	Papers published in conferences ranked as Core A* and A and journals indexed in JCR	10
Relevant	Papers published in conferences ranked as CORE B	5
Not relevant	The remaining publications	0
CITATION		
Publication prior to 2017		
High	Studies with more than 50 citations	10
Medium	Studies that had between 10 and 49 citations	5
Low	Studies with less than 10 citations	2
None	Studies with no citations	0
Early Publication		
Cited	Studies that had any citations	10
Not cited	Studies with no citations	5

3) DATA EXTRACTION STRATEGY

The data extraction strategy provides a framework in which to characterize individual metrics and helps us taxonomically classify and compare all the metrics collected. We first collected some basic information for each publication: date of publication, publication type (journal, conference, workshop), publication source (journal, conference, or workshop name), number of citations and authors' names and country. We then collected the study data required to address our research questions (features and some textual information) by defining a data extraction form and checklist. Table 6 summarizes the data extraction criteria used, which are further explained in the following subsections.

The objective of this strategy was to ensure a consistent classification of all the primary studies and an understanding of the current state of the art of QoS metrics for cloud services.

a: QOS CHARACTERISTIC AND QUALITY ATTRIBUTE

The purpose of this criterion was to extract the quality characteristic being measured by the metric and its quality attribute. We classified each metric according to the following internal and external quality characteristics proposed by the ISO/IEC 25010 [5]:

- *Performance Efficiency:* whether the metric measures an attribute related to the amount of resources used by the cloud services under certain conditions;

TABLE 6. Data Extraction Form and Criteria.

Criterion	Possible Options/ Information Gathered [a]
QoS characteristic	Performance Efficiency, Reliability, Portability, Security, Maintainability, Functional Suitability, Usability, Compatibility.
Quality attribute	Attribute name
Metric type	Base (measurement method), Derived (measurement function), Indicator (analysis model).
Unit of measurement	Metric unit
Measurement function	Calculation formula and an explanation on how the metric is calculated
Tool support	Manual, Automated (tool name).
Measurement result	Qualitative, Quantitative, Hybrid.
Cloud lifecycle phase	Requirements, Acquisition, Development, Integration, Operation, Retirement
Measured cloud artifact	Cloud service specification, Cloud architecture, Cloud service
Service type	SaaS, PaaS, IaaS
Stakeholder's viewpoint	Provider, Consumer, Broker, Developer, End-User
Validation procedure	Theoretical validation, Empirical validation, No validation

- *Reliability:* whether the metric measures an attribute related to the ability of a cloud service to perform the specified functions when used under certain conditions and in a certain interval of time;
- *Portability:* whether the metric measures an attribute related to the ability of the cloud service to be transferred effectively and efficiently from a hardware, software, operational, or usage environment to another. This feature also includes aspects related to the scalability and elasticity of the cloud service;
- *Security:* whether the metric measures an attribute related to the ability of the cloud service to protect information and data such that unauthorized persons or systems cannot read or modify them;
- *Maintainability*: whether the metric measures an attribute related to the ability of the cloud service to be modified effectively and efficiently owing to evolutionary, corrective or perfective needs;
- *Functional Suitability*: whether the metric measures an attribute related to the capacity of the cloud service to provide functions that meet the explicit and implicit needs of users under specific conditions;
- *Usability:* whether the metric measures an attribute related to the ability of the cloud service to be understood, learned, operated and attractive to the user in the specific context of use;
- *Compatibility:* whether the metric measures an attribute related to the ability of two or more cloud services to exchange information and/or perform their required functions when they share the same hardware or software environment.

b: METRIC TYPE

There are three types of metrics: base, derived and indicator. A *base* metric (also known as a direct metric) is defined in terms of an attribute and can be calculated directly. A *derived* metric (also known as an indirect metric) is, meanwhile, defined as a function of two or more base or derived metrics and should, therefore, contain a measurement function (formula) that explains how to calculate the metric. An *indicator* is defined from other measures using an analysis model as a measurement approach.

Base metrics are, therefore, independent, while a derived metric can only be calculated using other measures [28]. We extracted the metric type for each metric, and in the case of the base metrics, we also extracted their *measurement method* and *unit of measurement*. A measurement method is a logical sequence of operations that are described generically and are used to quantify an attribute [28]. With regard to the derived metrics, we extracted their *measurement function* and *unit of measurement*. A measurement function is an algorithm or calculation performed to combine two or more base or derived measures [29], while a unit of measurement is used to express one or more measures of interval or ratio types [29]. Finally, in the case of indicators, we extracted the measurement function that describes an algorithm or calculation that combines one or more measures with associated decision criteria [29]. A decision criterion uses thresholds, targets, or patterns to determine the need for action or further investigation or to describe the level of confidence in a given result [28].

c: TOOL SUPPORT

The objective of this criterion was to assess whether the metric was supported by a tool or algorithm that facilitated the calculation of that metric. If the metric had a tool that supported its automated or semi-automated calculation, then it was classified as *Automated*, and the tool name and reference were registered. Otherwise, it was classified as *Manual*.

d: MEASUREMENT RESULT

A measurement result is a set of numbers and references together with any other available relevant information which are attributed to a magnitude, i.e., the property of a phenomenon, body, or substance [30]. In this study, the result obtained by performing a measurement is a number or category that is assigned to a quality attribute of a cloud service (definition adapted from [31]). For example, the data transmission rate or bandwidth is commonly expressed in Mbs/sec, or the disk usage in Gbs [32].

The purpose of this criterion was, therefore, to understand the type of measurement result obtained when the metric was applied. In particular, a metric can measure a quality attribute in a *qualitative*, *quantitative,* or *hybrid* manner.

Quantitative evaluations are concerned with evaluating the attributes quantitatively, using continuous values (e.g., an attribute such as function commonality that measures the average of commonality of each functional feature discovered from software requirements specification (SRS) in the same domain and defined in a target SaaS [33]. It can be measured with continuous values (between 0 and 1). *Qualitative*

evaluations are those that indicate qualities or qualitative categories (e.g., an attribute such as the flexibility of a cloud service that rates the ability to add or remove predefined features from service in order to customize it [34]. It can be measured as High, Medium, or Low). *Hybrid* evaluations are those that use both qualitative and quantitative evaluations [28]. For example, the data center distance measures the distance between the location of the data center and the expected service location and then rates the provider using the sum of distances [35].

e: CLOUD LIFECYCLE PHASES

The objective of this criterion was to understand the phases of the cloud service lifecycle to which the metric can be applied, which could be one or more phases. We classified each metric according to the lifecycle phases proposed by Schneider *et al.* [36]:

- *Requirements*: those metrics that are applied to documents or specifications that describe the customer's decision concerning whether and to what extent a cloud service is used.
- *Acquisition*: those metrics that support the evaluation of cloud providers and services. Examples of these are metrics that are used to determine the estimation of demand or to assess opportunities and related risks.
- *Development*: those metrics that are applied to cloud artifacts of all the activities related to the service requirements specification, architecture design, programming, hardware configuration, testing, deployment, orchestration, release management, and integration.
- *Operation*: those metrics that are used to evaluate the service at runtime for management purposes (e.g., monitoring the QoS at runtime). These include metrics with which to supervise the consumers' and providers' fulfillment of their contract (e.g., detecting SLA violations). It also includes metrics used by service providers when performing maintenance, evolution, support and billing, and metrics used by consumers to monitor and evaluate service usage.
- *Retirement*: those metrics that are used to ensure a safe and organized discontinuity of the service or when the customer switches to another provider. The provider's metrics can include metrics to ensure compliance with data protection regulations regarding keeping or deleting customer data.

f: MEASURED CLOUD ARTIFACT

An artifact is a cloud service representation that is used to apply the metric and to perform a quality evaluation. The state of artifacts can be early, intermediate, or ended. Each metric is classified according to the artifact that it measures. We consider the following artifacts:

- *Cloud service specification*: a document or model that represents the functional interface and lists the operations and attributes that customers can access. Examples of cloud service specification artifacts are the requirement documents which define specific features to meet or service-level agreements (SLAs) which basically are a commitment between a service provider and a customer usually defined in terms of QoS.
- *Cloud architecture*: a model encompassing all the elements in a cloud environment. It represents how all the components and capabilities required to build a cloud service are connected in order to deliver a platform on which applications can run. Software diagrams which encapsulate application layers of abstraction (e.g., middleware) and network diagrams which encapsulate network layers connection (e.g., routers) can be considered as instances of cloud architecture artifacts.
- *Cloud service*: the actual service being used in a cloud environment. This includes actual service representations. Source code successfully deployed as a versioned service, and network or virtual machine configurations as settings of hardware and software to enable networking and virtualization services can be considered as examples of cloud service artifacts.

A metric may be applied to more than one artifact. If this is the case, we extract different operationalizations (i.e., different measurement functions that show how the metrics are calculated in each artifact).

g: SERVICE TYPE

This objective of this criterion is to assess the type of service that the metric evaluates. A service type corresponds to a group of services that share a common set of quality features and properties. A metric may be used to evaluate more than one service type. In this study, we consider the three service types (or models or capabilities) proposed by the NIST SP800-145 [1] and ISO/IEC 17788 [37]:

- *Software as a Service (SaaS)*: consumers using a running provider's applications deployed on a cloud infrastructure. Consumers do not manage or control the cloud service, with the exception of the application configuration settings.
- *Platform as a Service (PaaS)*: consumers can use a programming language, libraries, services, and execution environment, and tools supported by the provider to deploy, in the cloud infrastructure, the applications created or acquired. Consumers do not manage or control the cloud infrastructure but control deployed applications and the configuration settings for the hosting environment.
- *Infrastructure as a Service (IaaS)*: consumers can provide and use resources for processing, storage, and access to networks (networking) in the cloud, where they deploy and run additional software (e.g., operating systems and applications). Consumers do not manage or control the core cloud infrastructure but control operating systems, storage, applications, and specific network settings.

h: STAKEHOLDER'S VIEWPOINT

A stakeholder's viewpoint represents those who can measure and use the metric. Each metric has been classified according to the roles proposed by the NIST SP 800-146 [7]:

- *Provider*: person, organization, or entity responsible for making a service available to consumers. A provider builds the requested software/platform/infrastructure services, manages the technical infrastructure required to provide the services, provisions the services at agreed-upon service levels, and ensures the quality of services.
- *Consumer*: person or organization that maintains a business relationship with and uses services made available by cloud providers. A consumer browses the cloud provider's service catalog, requests the appropriate service, sets up service contracts with the cloud provider, and uses the service.
- *Broker*: a cloud consumer may request cloud services from a cloud broker rather than contacting a cloud provider directly. A cloud broker manages the use, performance, and delivery of services, and negotiates relationships between cloud providers and cloud consumers.

We also added two other roles focused on the cloud service *Developer*, who acts as a service partner and can be a developer, integrator, tester, etc., and *End-User*, which represents the individuals or organizations who are the customers of the cloud service.

i: VALIDATION PROCEDURE

The goal of this criterion is to obtain the procedure used to validate the metric. A metric must be both theoretically and empirically validated. The first type of validation ensures that the metric measures the attribute that it is supposed to measure, while the second provides evidence on the usefulness of the metrics in practice. The theoretical validation also makes it possible to confirm that the measurement does not violate any necessary properties of the measurement elements.

Each metric has been classified according to the research method/strategy used. If the metric was theoretically validated, it was classified according to the type of approach that was used: *Property-based approach*, *Measurement theory-based approach*, or other. If the metric was empirically validated, it was classified according to the method that was used: *Controlled Experiment*, *Case Study*, or *Survey*. Otherwise, the metric was classified as *Not Validated*.

With regard the theoretical strategies employed to validate metrics, property-based approaches [38] can be used to prove that a metric satisfies the properties that characterize a concept (e.g., size, complexity, coupling), while measurement-theory-based approaches [39], [40] are more rigorous than property-based approaches since they prescribe theories and conditions for the modeling and definition of metrics. The theory provides an empirical interpretation of the numbers (of software metrics) by means of the hypothetical empirical relational system.

With regard to the empirical strategies used to validate metrics, a Case Study is an observational study, and data are collected for a specific purpose throughout the study. A Survey is a piece of research that is performed in retrospect when the metric has been in use for a certain period of time. A Controlled Experiment is a formal, rigorous, and controlled study. Experiments provide a high level of control and are useful when validating software metrics. They can, for example, be used to validate the effectiveness of a set of design metrics as regards predicting the usability of cloud services.

We facilitated the data extraction task carried out by the researchers by designing a template (spreadsheet) together with a guideline containing the details of each criterion. One sheet of the template was used to gather information about the selected primary studies, while another gathered the data about the cloud service metrics (e.g., metric name, metric description, attribute measured, characteristic). We also collected common information such as i) authors names, ii) paper title, iii) publication details, iv) digital library name v) publication type, vi) citations.

We performed a pilot study using a sample of papers in order to test the understandability and correctness of the data extraction criteria and spreadsheet.

In order to assess the reliability of the data extraction process, three different authors carried out a second iteration of the classification on a random sample of 30 papers. The level of agreement among the researchers was assessed using Fleiss' Kappa index.

4) SYNTHESIS METHOD

We used two qualitative synthesis methods to synthesize the data extracted from the primary studies and to answer the research questions: narrative synthesis and thematic analysis.

Narrative synthesis reports the results of a systematic review in terms of text and words. We used the Mendeley tool [41] to store all the papers and to annotate the pieces of evidence employed to classify the metrics according to each criterion described in Section IV.B. 3. We also analyzed the number of papers found in each bibliographic source per year and the frequencies of the studies classified in each criterion.

Thematic analysis [42] involves identifying and coding the major or recurrent themes in the primary studies and summarizing the results under these thematic headings. We used this method in combination with narrative synthesis in order to answer the research questions. We specifically followed the steps shown below:

1. *Reading the papers and identifying specific segments of text*: we read all the text related to the primary studies and identified specific segments of text that were relevant to the research questions (e.g., QoS characteristics, quality attributes, metrics, cloud lifecycle phases, cloud artifacts measured, service type, stakeholder's viewpoint and validation procedure) in order to form an initial idea for analysis.

2. *Generating initial codes*: we defined the initial codes for the QoS characteristics, quality attributes, metrics (and their associated information), cloud lifecycle phases, cloud artifacts measured, service type, stakeholder's viewpoint, and validation procedure. We also labeled and coded those segments in the text that were related to these concepts. It should be noted that, in some cases, we had to recheck the papers.
3. *Searching for themes*: for each data item, we attempted to combine different initial codes generated from the second step into potential themes.
4. *Analyzing the codes to reduce overlaps and define themes:* It was possible to define some themes in advance as a result of the research questions (e.g., the quality characteristics that conform to the ISO/IEC 25010 or the stakeholder's viewpoint and service type that conforms to the NIST SP 800-145 [1] and NIST SP 800-146 [7], and the cloud lifecycle phases and validation procedure that are based on classifications taken from existing works), while others appeared as a result of reading the primary studies (e.g., quality attributes, metrics and their associated information, measured cloud artifacts, and tool support).
5. *Reviewing and refining themes*: the quality attributes, metrics (including their measurement methods or measurement functions, measurement result, unity of measurement), and measured cloud artifacts identified from the fourth step were checked against each other in order to understand what themes had to be merged with others or dropped. For example, availability and serviceability were merged because they had the same purpose, while throughput was dropped because it was a repeated metric (it was the same as the number of service requests served over total service time with the same measurement function [43]–[45]). We initially categorized and organized the metrics by considering the QoS characteristics, the quality attributes, and the name of the metrics. We then grouped the metrics in search of synonyms and homonyms, by analyzing the name of the metric and its definition, measurement method, measurement function(s) and unit of measurement (e.g., milliseconds, request/min).
   We considered those metrics that had a different name but the same definition (purpose) to be synonymous. For example, availability [46] and serviceability [47] use the same base metrics (i.e., uptime and downtime) to measure service availability. Similarly, AVAL-CQ [48] and Uptime [49] both measure the percentage of service availability by using the uptime metric, which is calculated as the total time of an operational period. Note that a metric can have several operationalizations (different measurement functions that can be used to measure the same quality attribute).
   We considered those metrics that had the same name but a different definition (purpose) to be homonyms. For example, in [43], reliability is measured using the failure rate, while in [50] it is measured using the success rate.
   This step allowed us to define clear and concise themes for quality attributes, metrics (along with their name, definition, measurement method, measurement functions, measurement result and unity of measurement), service types, cloud lifecycle phases, and the cloud artifacts measured.

We additionally created bubble plots [51] in order to report the frequencies of the combination of different criteria. A bubble plot comprises two x–y scatter plots with bubbles in the category intersections, in which the size of the bubble is proportional to the frequency. This synthesis method is effective as regards providing a map and a quick overview of a research topic.

C. RESULTS

The results of the primary study selection phase are presented on the basis of the metadata analysis and the quality assessment. On November 8, 2018, we ran the search string on the four digital libraries and retrieved 4333 papers. The list of publications from each database was then combined. We subsequently eliminated a total of 147 duplicate publications. The duplicates included studies that were obtained from more than one source and studies that had been published in both conference proceedings and journals. In the former case, we selected the publication only once by adopting the following order of priority, which otherwise does not have implications on the results: (1) IEEE Xplore, (2) ACM, (3) ScienceDirect, and (4) SpringerLink. We started with the specialized digital libraries that have the narrowest focus on the computer science/software engineering domain, i.e., IEEE and ACM [52], and later on, Science Direct and SpringerLink, which are multidisciplinary libraries, were considered. In the latter case, we selected only the most complete version of the study.

We then checked that all the editions of each conference proceedings and journal shown in Table 3 were indexed in at least one of the digital libraries. As the Journal of Cloud Computing: Advances, Systems and Applications (JoCCASA) was missing, we performed a manual search on this journal. This resulted in 44 additional candidate papers.

The title, abstract, and keywords of each publication were then reviewed by two reviewers against a set of inclusion and exclusion criteria; if necessary, the introduction and conclusions were also checked. This step led to the removal of a total of 3947 publications. The main reasons for exclusion were: studies that proposed metrics not related to internal or external quality (e.g., [53]); studies describing quality attributes without metrics to measure them (e.g. [54]); and studies that proposed or used metrics but did not provide an explanation of how to measure them (e.g., [55]).

After this early screening, the two reviewers read the remaining papers in full in order to apply the inclusion and exclusion criteria. This resulted in a set of 195 papers

(194 from the digital libraries and one paper from the manual search) that were stored in the Mendeley tool [38]. We then performed the quality assessment of these publications using the quality items reported in Table 4. All discrepancies concerning the quality assessment results were discussed by the authors with the aim of reaching a consensus. The reliability of the findings of this assessment was accomplished by considering only those relevant studies with an acceptable quality rate, i.e., those that had attained a quality score of more than 5 points (50% of the percentage score). Table 7 shows the results obtained for each quality item.

TABLE 7. Quality Assessment Checklist.

Quality Item	Score	Number of Papers			Percentage
		Y	P	N	
Q1	69,0	52	34	2	97,73%
Q2	49,0	32	33	23	75,00%
Q3	81,5	75	11	2	98,86%
Q4	82,5	77	11	0	100,00%
Q5	82,0	76	12	0	100,00%
Q6	28,5	6	45	37	57,95%
Q7	28,5	19	19	50	43,18%
Q8	75,0	64	22	2	97,73%
Q9	24,5	14	21	53	39,77%
Q10	10,5	7	7	74	15,91%
Total				88	100,00%

The assessment of each paper are available on the website accompanying this paper (*https://bit.ly/taxonomyqoscs*). The papers that did not fulfill the minimum threshold established were removed. As a result, four papers were excluded, signifying that a total of 84 relevant studies was eventually selected (see Figure 2). When analyzing the quality items, it is necessary to indicate some of the limitations of the selected studies. First, there is little discussion on the limitations of the studies (Q10; 15.91%). Second, there is a lack of documentation regarding insights and lessons learned (Q9; 39.77%). Third, there is a shortage of evidence concerning the application of metrics in practice (Q7; 47.18%), and fourth, the metrics reported are rarely validated (Q6; 57.95%).

The transfer of knowledge related to problems or negative outcomes of work is unusual. The retrospective assessment of insights and lessons learned from the work is, therefore, a less widespread practice because reporting focuses on the contributions and results that meet the objectives of the research. Furthermore, few empirical validations of metrics indicate that more evidence about the usefulness of these metrics is required. Finally, we consider that gaps in validation have influenced their reduced application in practice.

Table 8 shows the quality of the venues at which the selected papers were published. The results show that over 55% of the papers were published at very relevant or relevant venues. With regard to the number of citations, the results show that most of the papers are high (>50 citations) and medium-cited (between 10 and 49 citations) papers, with 32.95% of the publications each; 30 papers have no citations.

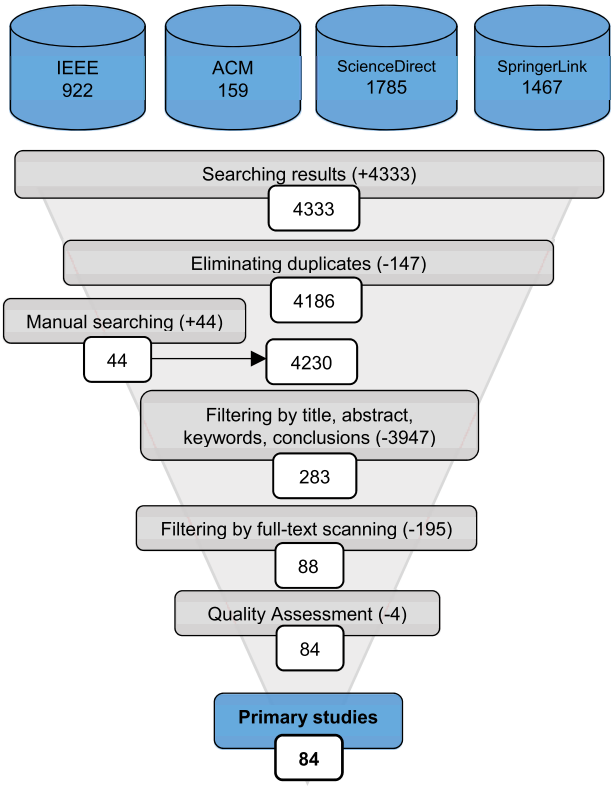

FIGURE 2. Primary studies results.

TABLE 8. Primary Studies Relevance.

Type	Score	Number of papers	Percentage
Publication impact	10	24	27.27%
	5	24	27.27%
	0	40	45.46%
	TOTAL	88	100.00%
Citation	10	29	32.95%
	5	29	32.95%
	2	28	31.80%
	0	2	2.30%
	TOTAL	88	100.00%

This can be considered as an indicator of how this topic has gained importance in recent years. There are no conclusions with regard to which the best bibliographic sources are, since those papers that appeared in several sources were considered only once. However, most of the relevant studies concerning quality metrics for cloud services were found in IEEE Xplore (55 papers, representing 65% of the selected primary studies). With regard to the type of study, 54 papers (60%) were published at conferences, 6 papers (7%) were published in workshops, and 28 papers (33%) were published in journals (see Figure 3). The list of selected studies is shown in Appendix A.

IV. CREATING AND REFINING THE TAXONOMY

This section describes how the taxonomy of metrics for cloud services has been created and refined. We started by defining

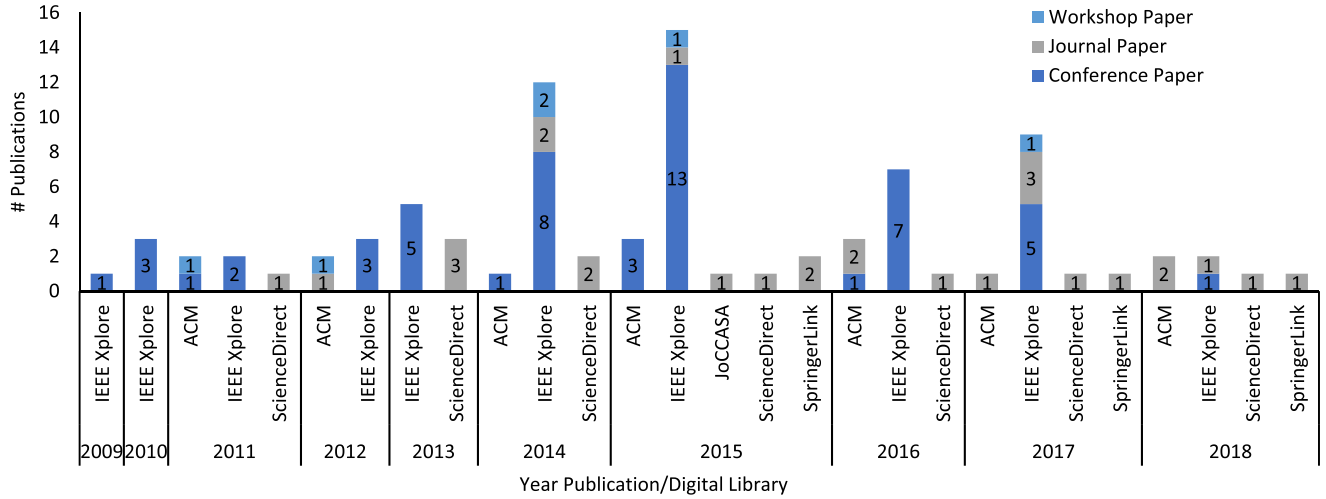

FIGURE 3. Number of primary studies by year and source.

a metamodel in which to structure the different concepts of the taxonomy as a baseline. These concepts are related to the data extraction criteria defined in Section IV.B. We then used qualitative synthesis methods (narrative synthesis and thematic analysis) in order to extract the data from the selected primary studies and create the taxonomy.

A. TAXONOMY METAMODEL

We organized the data and answered the research questions by defining a metamodel with which to guide the construction of the taxonomy of metrics for cloud services. The usefulness of the metamodel lies in its ability to decompose and hierarchically organize the elements of the taxonomy on the basis of the principles and notations of models in a technology-independent manner. It, therefore, allows the representation of concepts concerning metrics for cloud services, thus facilitating common understanding and communication.

Figure 4 introduces the metamodel employed to structure the taxonomy in terms of concepts (including their attributes) and their relationships. The most important concepts are characteristics, attributes, metrics, and operationalizations. The relationships indicate how these concepts are related to each other. For instance, an attribute can be measured by means of one or several metrics, and a metric can have one or more operationalizations, which represent different ways in which to calculate the value of the metric.

In the following, we introduce the purpose of the main concepts represented in the metamodel, which are described as metaclasses in Figure 4.

- *CloudServiceQualityModel*: groups the quality characteristics that are relevant to the cloud service domain and establishes the relationships among them. The quality of a cloud service is the degree to which the service satisfies the stated and implied needs of the cloud stakeholders, and thus provides value to them (adapted from ISO/IEC 25010) [5].
- *Characteristic*: according to the ISO/IEC 25010 [5], this is a high-level quality property of a cloud service that is refined into a set of sub-characteristics, which can be refined into quality attributes specific to the cloud domain. For example, performance efficiency, which represents the performance of a cloud service relative to the usage of resources under particular conditions.
- *Attribute*: a measurable physical or abstract property of an entity of a cloud service (e.g., network, virtual machine, container) [29] that can be measured using a quality metric. For example, memory capacity, which is a property of a physical or a virtual machine.
- *Metric*: a measurement scale (i.e., nominal, ordinal, interval, ratio, or absolute) combined with a measurement approach (i.e., measurement method or measurement function) describing how measurement is to be conducted [31]. For example, the metric memory size measures the RAM size, and the metric response time measures the execution time of a request. Each metric may also have operationalizations. There are three types of metrics:
  - *Base Metric*: a metric that does not depend on any other metric and uses a measurement method as a measurement approach [29], e.g., memory size, request time, response time.
  - *Derived Metric*: a metric that is derived from other base or derived metrics, using a measurement function as a measurement approach [29], e.g., response time, which is measured as the round trip time of a request, using the time of the request and the time of the response in the measurement function.
  - *Indicator*: a high-level quantitative metric that is derived from other metrics and uses an analysis model as a measurement approach [29]. In this work, we have collected the measurement function

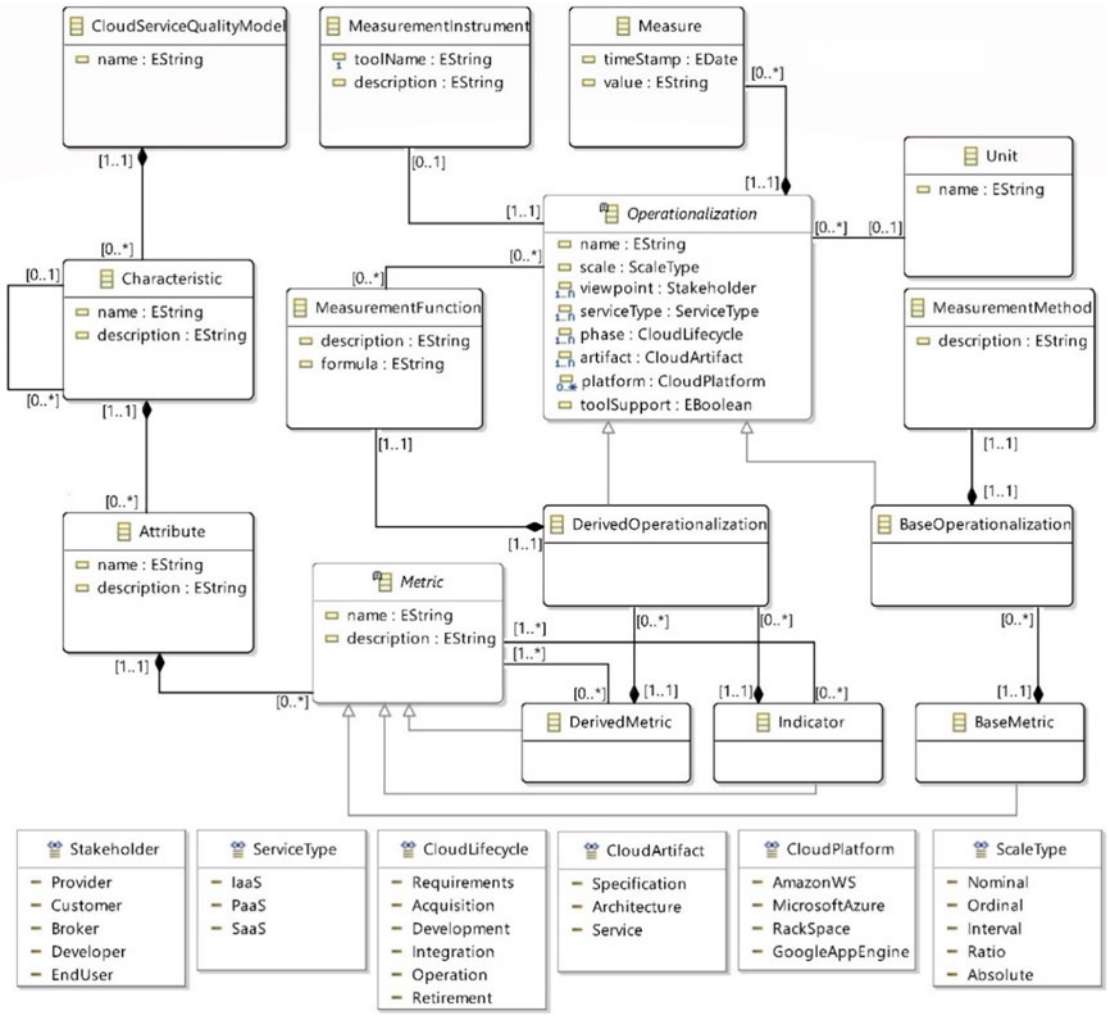

FIGURE 4. Taxonomy of cloud service metrics metamodel.

as being a reference to the analysis model because this concept was not properly described by the primary studies. An example of an indicator is service capacity, which is calculated using a weighted sum of several resources, such as CPU, memory, storage, and networking.

- *Operationalization*: represents different ways in which to calculate a given metric. It establishes a mapping between the generic definition of the metric and the cloud artifact, platform, or environment where it is actually measured. This means that a given metric can have one or more measurement approaches (i.e., measurement methods or measurement functions) that show how the metric can be calculated for a specific cloud platform, artifact, or environment. The operationalization of a metric can be base or derived, and is defined using the following metaclasses:
  - *Derived Operationalization*: this is the operationalization of a derived metric or indicator and has a Measurement Function that describes how a specific derived operationalization is calculated.
  - *Base Operationalization*: this is the operationalization of a base metric and has a Measurement Method that describes how a specific base operationalization is calculated.
- *Measurement Method*: a logical sequence of operations that are used to quantify a quality attribute by means of a base metric [28]. e.g., the sequence of steps that describes how the storage size or memory size in a virtual machine specification should be measured.
- *Measurement Function*: an algorithm or calculation performed to combine two or more base or derived metrics in order to quantify a quality attribute (adapted from ISO/IEC 15939) [28], e.g., response time can be calculated as the difference between the time at which the user sends a request to the cloud and the time at which that user receives a response from the cloud.
- *Measurement Instrument*: an instrument that assists cloud stakeholders or is useful for a measurement method. A measurement instrument can assist one or more measurement methods. For example, a cloud monitoring tool (e.g., Amazon CloudWatch) can be used to

calculate base or derived metric operationalizations for cloud services on the AWS platform. Another example is the use of the JCatascopia [32] tool, by using its parameter rx_bytes to measure the number of bytes received by a service.

- *Measure*: the value assigned as a measurement result for a given quality attribute [28]. For example, the number of Mbs for memory size or the number of milliseconds for latency.
- *Unit*: the unit of measurement [28] for those metric operationalizations with interval and ratio scale types. A unit can be defined for more than one metric operationalization (e.g., MBs for memory size, milliseconds for response time).

We also represented relevant information about the domain that is required in order to properly characterize what entity is being measured, how it is measured, and to whom the measurement results are relevant. Some specific cloud concepts were represented as enumerated types (i.e., Stakeholder, ServiceType, CloudLifecycle, and CloudArtifact), according to the definitions provided in Section III.B.3. In addition, CloudPlatform represents the specific cloud platform on which a given operationalization is applied (e.g., Microsoft Azure, AWS, Google App Engine), and ScaleType defines the nature of the relationship between values on the scale [28]. Five scale types can be found in software measurement literature: Absolute, Nominal, Ordinal, Interval, and Ratio. A scale type determines the type of arithmetic operations that can be carried out with a given metric and, hence, the type of statistical analysis.

As the core of the taxonomy is the concept of metric and its possible operationalizations, the operationalization metaclass contains several attributes that describe its properties:

- *name:* indicates the name of the operationalization.
- *scale:* represents the scale type of the operationalization.
- *viewpoint:* indicates the target audience for the metric (e.g., provider, customer, broker, cloud architect). An operationalization can be of interest to more than one stakeholder type.
- *serviceType:* indicates the type of service (i.e., SaaS, PaaS, IaaS) to which the operationalization is applied. An operationalization can be applied to more than one service type.
- *phase*: indicates the phase of the cloud service lifecycle in which the operationalization can be applied. An operationalization can be applied to more than one phase.
- *artifact*: indicates the cloud artifact that is being measured by the operationalization.
- *platform*: indicates the cloud platform on which the operationalization is applied.
- *tool support*: indicates the tool (if any) that can assist the stakeholders in calculating the metric operationalization.

Overall, the proposed taxonomy integrates relevant concepts that are required in order to understand the quality of cloud services. Specifically, it integrates the high-level quality characteristics proposed by the ISO/IEC 25010 with measurable properties and metrics from the cloud domain.

As discussed in Section II, a quality assessment of cloud services is usually performed for specific quality characteristics. This leads to isolated solutions for QoS assessment. A holistic approach that integrates the different quality characteristics, attributes, and metrics for the cloud domain is, therefore, necessary. Furthermore, current standards for product quality are too abstract. A clear transition to measurements in specific domains is, therefore, required.

We believe that the proposed taxonomy provides a first step in this direction. The taxonomy will be used to guide the data extraction from the primary studies, thus allowing the existing knowledge regarding metrics for cloud services to be gathered and classified. This will allow us to attain a unified view of quality metrics for cloud services.

B. DATA EXTRACTION

The data extraction started on March 2019 and finished on September 2019. The results derived from the data extraction allowed us to answer the research questions and obtain the taxonomy of quality metrics for cloud services.

We categorized and organized the metrics extracted from the primary studies using the data extraction criteria described in Section IV. B.3. This allowed us to obtain an initial version of the taxonomy that has been represented on a spreadsheet. In particular, we gathered a total of 579 metrics retrieved from 84 primary studies. Owing to the wide range of metrics, we decided to analyze them using the thematic analysis method. The flexibility of this method allowed us to refine the taxonomy according to the five steps detailed in Section III.C.4.

First, only one of the researchers (the first author) read all the papers (Step 1). The same researcher identified initial codes and quotes (e.g., segments of text from each study) related to the research questions, e.g., quality attributes being measured and detailed information about the metrics, such as their name, definition, measurement method, measurement function(s), etc. (Step 2). Two researchers then reviewed each quote independently and identified a list of higher-level categories (themes) that described a set of QoS characteristics, quality attributes, metrics, cloud lifecycle phases, cloud artifacts measured, service type, stakeholder's viewpoint and validation procedure (Step 3). The themes comprised a short name and a description. The results were discussed at a meeting and disagreements on the categories were solved by consensus. We also analyzed the codes so as to reduce overlaps and define the themes (Step 4). This allowed us to obtain an initial version of the taxonomy of metrics.

We then reviewed and refined the themes in order to refine and consolidate the taxonomy of metrics (Step 5). This process involved searching for synonyms and homonyms of metrics in order to aggregate common metrics (Step 4). It was, in some cases, necessary to rechecking the papers during this

step. In particular, some of the initial themes (i.e., quality attributes and metrics) were grouped into higher-order themes in order to provide a consolidated view of how the different quality attributes and metrics are related to each other. In this step, we, therefore, eliminated duplicate quality attributes and metrics and combined them as a result of the analysis of synonyms and homonyms. A subsequent analysis was based on analyzing the frequency with which each theme appeared in the primary studies.

After applying the five steps mentioned below, we were able to obtain the final distribution of the quality attributes, metrics, and operationalizations for each of the eight quality characteristics from the ISO/IEC 25010. Figure 5 shows the structure of the refined taxonomy of metrics for cloud services.

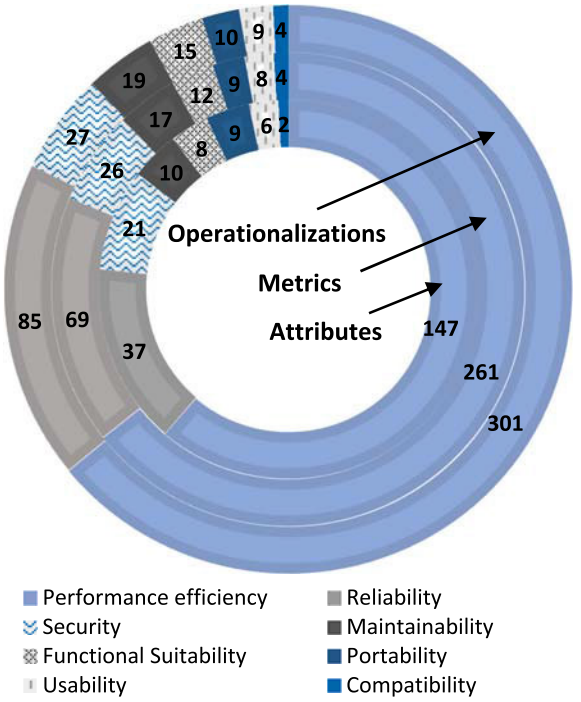

FIGURE 5. Refined taxonomy of metrics for cloud services.

The inner ring of the figure contains 235 different quality attributes distributed among the quality characteristics; the intermediate ring contains 406 unique metrics (from the initial set of 579 metrics that were originally retrieved), while the external ring contains the 470 operationalizations (470 metric operationalizations), which represent the different measurement methods and functions that can be used to calculate the metrics. Of these metrics, 156 measured internal quality attributes, while the other 314 measured external quality attributes of cloud services.

The results show that the quality characteristic with the largest number of quality attributes, metrics, and operationalizations is Performance Efficiency, with more than 50% of the total number of metrics. This characteristic also contained the greatest number of metrics that were eliminated and aggregated as a result of refining the taxonomy (e.g., various synonymous metrics related to delay, latency, throughput, and response time). A detailed discussion of this is presented in the following subsections while answering the research questions.

Table 9 shows the contribution of each primary study to the quality characteristics of the ISO/IEC 25010 standard. It also shows the number of papers that contribute to each characteristic and the percentage with respect to the total number of papers.

TABLE 9. Paper contribution to quality characteristics.

Quality Characteristic	Primary Studies	# Papers	% Papers
Performance Efficiency	S01, S02, S03, S04, S05, S06, S07, S08, S09, S10, S13, S14, S16, S17, S18, S19, S20, S21, S23, S24, S25, S26, S27, S28, S30, S31, S32, S33, S35, S36, S37, S38, S40, S41, S42, S43, S44, S45, S47, S48, S49, S50, S51, S53, S54, S55, S56, S57, S58, S61, S63, S64, S65, S66, S67, S68, S69, S70, S71, S73, S74, S75, S76, S77, S78, S79, S80, S81, S82, S83.	69	82 %
Reliability	S03, S07, S08, S09, S10, S12, S13, S14, S15, S16, S21, S22, S23, S24, S26, S27, S28, S30, S34, S36, S44, S45, S48, S49, S50, S51, S52, S53, S58, S61, S62, S65, S67, S69, S72, S75, S78, S79, S81, S82, S83, S84.	42	50 %
Security	S10, S21, S27, S29, S36, S39, S48, S50, S58, S60, S65, S75, S79, S82, S83.	15	18 %
Functional Suitability	S23, S24, S30, S44, S49, S53, S65, S71.	8	10 %
Maintainability	S07, S21, S24, S44, S46, S53, S61, S65.	7	8%
Portability	S10, S24, S26, S36, S53, S59, S65.	7	8%
Usability	S07, S21, S24, S65, S82, S83.	6	7%
Compatibility	S24, S49, S53.	3	4%
	Total of papers	84	

The highest contribution is related to Performance Efficiency, with 69 different papers, while the lowest is related to Compatibility, with only three papers (note that some studies contribute to more than one quality characteristic). This is consistent with the total number of quality attributes, metrics, and operationalizations retrieved as the result of data collection.

Furthermore, Figure 6 shows how many quality characteristics are addressed in each selected primary studies. Note that 47 primary studies have focused on a single quality characteristic, none of the studies addressed all the quality characteristics holistically, and only 7 primary studies addressed more than half (i.e., 4) of the quality characteristics proposed by the ISO/IEC 25010.

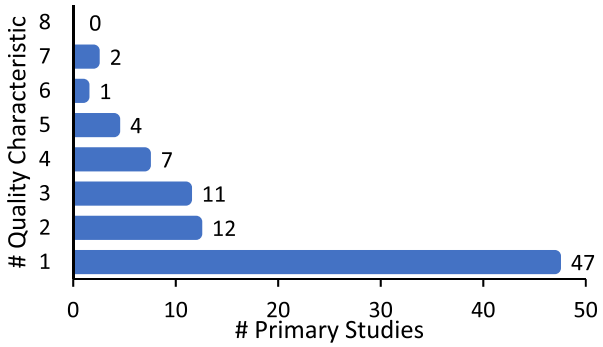

FIGURE 6. Distribution of primary studies to quality characteristics.

Of the studies that addressed a single characteristic, the top characteristic is Performance Efficiency, because issues such as the service performance, capacity, and resource utilization affect the initial perception of the service quality from the stakeholders' point of view. Moreover, performance efficiency is commonly used when defining SLAs between providers and customers, and to monitor the service, both of which are essential activities when adopting cloud services.

The analysis also revealed that the combination of quality characteristics that appears most frequently in the studies is that of Performance Efficiency and Reliability, with 32 papers (e.g., [3], [56], [57]).

Of the studies whose objective was to provide a holistic approach with which to evaluate cloud services, we identified only two that addressed seven out of the eight quality characteristics from the ISO/IEC 25010 (i.e., Singh & Chana [47] and Garg *et al.* [58]). Singh & Chana [47] proposed a QoS metric-based resource provisioning technique. This technique uses 35 metrics that cover all the quality characteristics (with the exception of Compatibility) to support the efficient provisioning of resources. Garg *et al.* [58], meanwhile, proposed the SMICloud framework in order to rank cloud services. This framework uses 23 metrics that address all the ISO/IEC 25010 quality characteristics (with the exception of Security) used to measure the quality level of both cloud services and cloud providers.

Figure 7 presents the taxonomy of QoS metrics for cloud services. In the figure, the quality characteristics from the ISO/IEC 25010 are broken down into quality attributes. These quality attributes have associated the different QoS metric operationalizations collected from the literature.

The results of this classification are available on the website accompanying this paper (*https://bit.ly/taxonomyqoscs*), where the reader can study the different layers of this figure in order to consider further details of the distribution of quality attributes, metrics, and operationalizations for any of the quality characteristics.

Table 10 provides a summary of the results obtained for each criterion in order to answer the stated research questions, which are further discussed in the following subsections.

1) QOS CHARACTERISTIC

Each metric was classified according to the ISO/IEC 25010 quality characteristics. We collected the name and description of the metric, the name of the quality attribute that was measured, and some additional information that helped us understand the context in which the metric was used.

a: PERFORMANCE EFFICIENCY

The results show a high concentration of metrics related to Performance Efficiency, with 301 metric operationalizations. This can be explained by the fact that one of the essential characteristics of cloud services (i.e., measured service) states that cloud systems should automatically control and optimize the resources used by leveraging a metering capability appropriate to the type of service (e.g., storage, processing, bandwidth) [1].

This means that resource usage should be monitored, controlled, and reported for both the provider and the consumer of the service being utilized. It is, therefore, very important to control the performance of cloud services and ensure an appropriate response time for the customer.

An analysis of the metrics collected shows that most of them (i.e., 135 metric operationalizations) measure *time behavior*. This is owing to the fact that, in cloud computing, performance is expressed by the speed with which a cloud service request is completed. That speed is usually represented in terms of the amount of time required, from sending a request until receiving the response (i.e., response time), and the number of successful requests within a certain time interval (i.e., throughput). Response time can be further split into execution time, which represents the time required to process a request on the server-side, and latency, which represents the time required for the one-way delivery of a message.

Examples of the metrics employed to measure time behavior that we collected include response time in terms of the time taken to execute a service request ([57]–[60]) or in terms of the execution time of a virtual machine belonging to a cloud service [60], and latency in terms of the time that elapses between a request and the corresponding response ([45], [47], [60]).

There are also a great number of metrics that measure *resource utilization* (77 metric operationalizations), but most of them are low-level metrics, which are, in most cases, delivered by the monitoring tools provided by cloud platforms. These metrics make it possible to measure the level of resource usage (CPU, memory, disk) or the percentage of currently occupied resources ([32], [61]). Of these metrics, 27 were proposed for the purpose of measuring sustainability in terms of *energy consumption*. For example, in Dou et at. [62], the total energy consumption of a cloud infrastructure is measured in terms of the energy consumption of servers and the energy consumption required for communication between those servers.

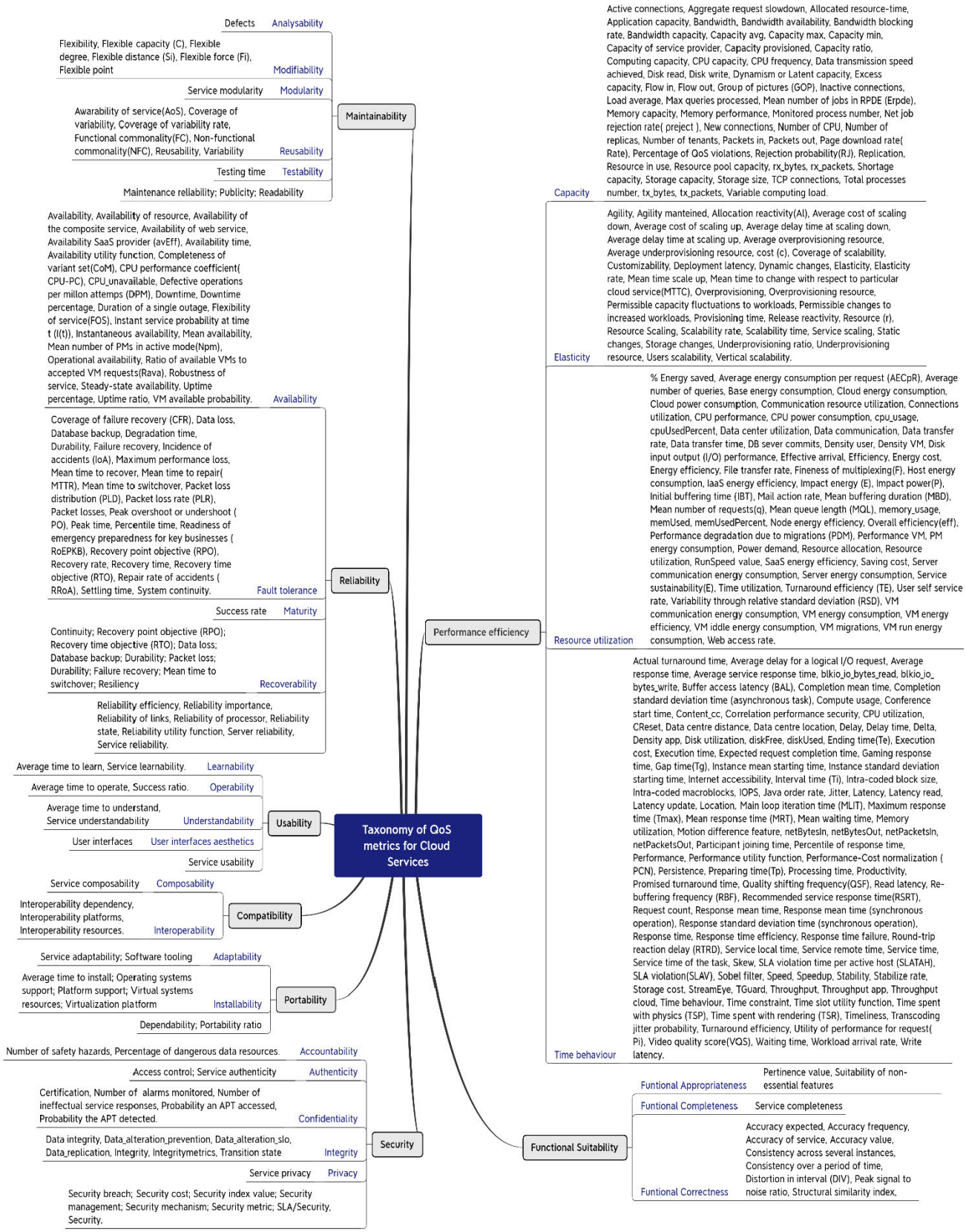

FIGURE 7. Taxonomy of QoS metrics for cloud services.

TABLE 10. Results according to data extraction criteria.

Research Subquestions	Possible Options	Results	
		#Oper.	Percentage
RQ 1. QoS characteristic			
	Performance Efficiency	301	64%
	Reliability	85	18%
	Security	27	6%
	Maintainability	19	4%
	Functional Suitability	15	3%
	Portability	10	2%
	Usability	9	2%
	Compatibility	4	1%
RQ 2. Metric type			
	Base	108	23%
	Derived	347	74%
	Indicator	15	3%
RQ 3. Tool Support			
	Manual	402	85%
	Automated	70	15%
RQ 4. Measurement result			
	Qualitative	12	2,6%
	Quantitative	455	96,8%
	Hybrid	3	0,6%
RQ 5. Cloud lifecycle phase			
	Requirements	53	8%
	Acquisition	174	26%
	Development	58	9%
	Integration	8	1%
	Operation	373	56%
	Retirement	8	1%
RQ 6. Measured cloud artifact			
	Cloud service specification	12	2%
	Cloud architecture	58	11%
	Cloud service	464	87%
RQ 7. Service type			
	SaaS	210	35%
	PaaS	82	14%
	IaaS	304	51%
RQ 8. Stakeholder's viewpoint			
	Provider	374	45%
	Consumer	283	34%
	Broker	46	6%
	Developer	57	7%
	End-user	69	8%
RQ 9. Validation procedure			
Theoretical validation	Axiomatic approach	0	0%
	Measurement Theory - based approach	0	0%
Empirical validation	Controlled Experiment	39	8%
	Case Study	8	2%
	Survey	0	0%
No validation	No Validation	188	39%
	Proof of concept	254	52%

OPER = Operationalizations

We also found 56 metrics that measure *capacity*. Capacity is the degree to which the maximum limits of a resource of a cloud service (e.g., storage, networking) satisfy the expected requirements. For example, Souza *et al.* [59] propose several metrics with which to measure server capacity, storage device capacity and network capacity.

This characteristic also includes several quality attributes and metrics related to *elasticity* (29 metric operationalizations) and *scalability* (11 metric operationalizations), which are essential factors that affect the quality of cloud services. Scalability and efficiency are associated with elasticity, but their meaning is different from elasticity, while they are, in some cases, used interchangeably.

NIST SP 800-145 defines rapid elasticity as capabilities that are rapidly and elastically provisioned and released, in some cases automatically [1]. Herb *et al.* [63] define it as the ability to adapt the change in workload by automatically adding or removing resources. Finally, in this work, we adopt the definition proposed by Al-Dhuraibi [12], which defines elasticity on top of scalability. The authors consider elasticity to be an automation of the concept of scalability (auto-scaling); however, it has the objective of optimizing the resources as best and as quickly as possible at a given time. In this study, scalability is defined as the ability of the cloud service to sustain increasing workloads by making use of additional resources and is time-independent, whereas efficiency describes how the cloud resources can be used efficiently as they increase or decrease in scale [12].

Scalability additionally sustains the essential characteristic resource pooling of cloud computing, during which providers' computer resources are pooled in order to serve multiple consumers using the multi-tenant model [1]. Its measurement is based on three basic methods (i.e., replication, resizing, and migration [61]), which are supported by the metrics collected. Replication employs horizontal scalability to add or remove instances of resources (e.g., virtual machines, Kubernetes). Resizing uses vertical scalability to increase or decrease computing resources (e.g., RAM size, CPU cores). For example, Souza *et al.* [59] proposed that the metric number of virtual servers in the resource pool should be used to measure horizontal server scalability and that the metrics RAM size and number of CPUs should be used to measure the vertical scalability of servers. Migrations transfer resources (e.g., containers, virtual machines) from one server to another.

Efficiency, meanwhile, reflects how well cloud resources are utilized as they scale up or down. It could be measured in terms of time scaling resources and the amount of resources in relation to the cost of optimization. For example, Hu *et al.* [61] used time scaling to measure scaling up and down by using average delay time at scaling up and down. A consumer can, therefore, measure the delay it takes to provision and de-provision a given resource. These authors also measured the cost of scaling as the average cost of scaling up or down using the renting price per use of the virtual machine. This signifies that the higher the elasticity results, the greater the efficiency.

We also found several metrics that are used to control the three provisioning states: over-provisioning, under-provisioning, and just-in-need. The over-provisioning state leads to extra and unnecessary costs when renting cloud resources. Underprovisioning takes place when the resources provided are smaller than the resources required, which may lead to performance degradation and a violation of SLA clauses. For instance, Hu *et al.* [61] defined over-provisioning as the average number of over-provisioning resources and under-provisioning as the average number of

under-provisioning resources. In both cases, the SLA and QoS are not optimal. Finally, just-in-need denotes a balanced resource close to the real demand, signifying that the workload is handled and QoS is assured.

b: RELIABILITY

This was the second QoS characteristic with the highest number of metrics (85 metric operationalizations), which accounted for 18% of the metrics). Reliability has always been a major concern in distributed systems. Providing highly available and reliable services in cloud computing is essential as regards maintaining customer trust and satisfaction and preventing revenue losses. However, assuring reliability in cloud environments is challenging. A cloud service may consist of hundreds of microservices, each running in its own distributed cluster and containing its own multiple dependencies, thus increasing the number of individual components that can fail. In addition, various types of failures are interleaved in the cloud computing environment, such as overflow failure, timeout failure, resource missing failure, network failure, hardware failure, software failure, and database failure. This may explain the large number of metrics with which to evaluate reliability that are proposed in the literature.

Of all the quality attributes considered, the most important as regards measuring reliability are availability, recoverability, fault tolerance, and service reliability. All these attributes are critical to ensure the continuity of the cloud service.

An analysis of the metrics collected shows that most of them (i.e., 34 metric operationalizations) measure *availability*. Availability is the ability of a cloud service to be operational and accessible when it is required for use. Consumers value a highly available service, and it must be part of the SLA negotiation with the cloud providers. For example, Zheng *et al.* [48] propose employing the AVAL metric to measure availability as the uptime percentage of a cloud service during a time interval, where higher uptime represents fewer interruptions. In Garg *et al.* [58], availability is measured in terms of service accessibility using the ratio between unavailable time over the total service time. In Rizvi *et al.* [57], availability is measured in terms of the functional state of the service in an interval of time using uptime (service accessibility) and downtime (service under repair).

There are also a good number of metrics that measure different attributes related to *recoverability* (25 metric operationalizations). Recoverability is the degree to which, in the event of an interruption or a failure, a cloud service can recover the data directly affected and re-establish the desired state of the system. This is an important factor for cloud consumers, as any loss of data could be devastating for the business. Examples of metrics include the coverage of failure recovery, which is measured as the ratio of failures remedied over the total of failures [3], durability, which is measured as the probability of data loss [57], and robustness, which is measured as the probability of service being affected by the failure of a cloud component [60].

In particular, the robustness of an IaaS service can be measured by the number of VMs affected by a host failure, i.e., maximizing robustness means minimizing the number of VMs affected [60]. Similarly, the robustness of multi-cloud or multi-tenant services can be measured by the number of cloud services or the number of tenants affected by a service failure. However, none of the collected metrics address robustness in these scenarios.

We also found 12 metrics that measure *fault tolerance*. This is the capability of the cloud service to remain reachable and working when anomalies occur [60]. Anomalies can occur as a result of errors in the physical machine, network, or software. Examples of metrics include traditional metrics used to measure fault tolerance in software systems, such as mean time between failures (MTBF), which is measured as the time between consecutive service failures, mean time to failure (MTTF), mean time to repair (MTTR), and coverage of fault tolerance (CFT).

There is one metric that measures *maturity*. Maturity is the degree to which a cloud service meets required needs for reliability under normal operation. A service is reliable when it performs specific functions under specified conditions for a specified period of time [5]. These metric operationalizations include the success rate, which measures the successful completion of the accepted job by the cloud service [50].

Finally, several primary studies proposed service reliability as an indicator that combines other quality attributes related to reliability (e.g., availability, fault tolerance, functional correctness, and recoverability). The purpose of these 13 metric operationalizations is to ensure the continuous operation of the cloud service without failures. All the indicators employed the typical Weighted-Sum method. For example, Lee *et al.* [3] weighted the fault tolerance, failure recovery, and functional correctness in order to obtain a measure of reliability, while Zheng *et al.* [48] weighted the storage cloud free state of failures (hardware failures, software faults, and network outages) to provide a measure of service reliability. The decision criteria of both approaches establish that the higher the result, the greater the reliability. This combination of measures allows a different weighting of stakeholder concerns, which may also differ as regards priority setting.

c: SECURITY

This characteristic accounted for 27 metric operationalizations, which represents only 6% of the total number of metrics. Aspects of cloud security include application-level security, tenant-on-tenant security where different tenants share a common infrastructure, provider-on-tenant security, information security and security requirements conformance (i.e., authentication, authorization, confidentiality, identity management, integrity, audit, security monitoring, incident response, and security policy management) [64]. This is, therefore, one of the main concerns when adopting cloud services and is mainly related to the security conformance and trustworthiness of the cloud infrastructure when running the customers' applications and storing their data in the cloud.

We are consequently of the opinion that the number of metrics with which to address these concerns that we have retrieved from literature is low.

Some of the quality attributes that were considered most important by the authors of the selected primary studies are integrity, confidentiality, accountability, authenticity, and privacy. All these attributes are critical as regards ensuring the security of cloud services and the privacy of data stored in the cloud. Some authors also used service security as an indicator that integrates other security base or derived metrics.

Upon analyzing the metrics collected, it will be noted that most of them (i.e., 8 metric operationalizations) measure *integrity*. Integrity is the ability to prevent the unauthorized access to, or modification of, applications or data. This is an important quality attribute when adopting cloud solutions (i.e., moving business data to the cloud) and should be checked at the data level and the computation level. For example, Singh & Chana [47] measured the integrity of a system by using the probability of a threat attack and the probability of repelling an attack in a given time. In contrast, Manuel [50] measured data integrity as a guarantee of data preservation and by checking whether resources store data correctly.

We found 6 metrics that measure attributes related to *confidentiality*. Confidentiality is the ability to ensure that data are accessible only to those who are authorized to do so. This is important because many virtual machines in cloud environments can co-exist on the same physical machine and may adopt different security protection mechanisms. Examples of metrics include the number of fake alarms monitored and the number of ineffectual service responses to the issues identified by the security as control weaknesses of the cloud service [47].

We also found 2 metrics that measure *accountability*, which is the ability to trace the actions of the cloud service. In cloud environments, the controls employed to audit and monitor security are essential owing to their intrinsic sharing principle. Examples of these metrics include the safety hazards proactively identified and dangerous data resources residing on solutions [47]. Furthermore, the rapid evolution of the cloud and the integration of new technologies (e.g., Internet of Things), make the existence of new metrics that allow improvements to be made to the transparency of the service operation even more important.

We also found 2 metrics that measure attributes related to *authenticity* (i.e., the ability to identify a subject or resource and prove that it is what it claims to be). This is critical in cloud environments in order to address access control to resources (i.e., infrastructure, applications, data), because they are usually shared and distributed. Examples of these metrics include authenticity in order to determine whether users have the privilege to employ that cloud service and access control to indicate the users' access state [65].

Some primary studies proposed service security as an indicator that aggregates other metrics related to security (e.g., integrity, privacy, availability, authenticity). The purpose of these nine metric operationalizations is to ensure the satisfaction of security requirements. For example, Zheng *et al.* [48] used SECY as a cumulative distribution function until the first security breach occurs. This is the guarantee that cloud services are free from viruses, intrusions, spyware, attacks and other security vulnerabilities that could put them at risk.

The implications of properly managing data privacy directly impact on the reputation and credibility of cloud providers that comply with laws and regulations (e.g., data protection). This establishes security as one of the most relevant challenges when adopting cloud services. More metrics are, therefore, required to address the whole information security lifecycle and control the usage of sensitive data. Security should also be assessed throughout the cloud lifecycle in order to ensure security requirements from the initial design and architecture of cloud services.

There are yet other challenges related to data security and privacy in cloud environments, in which broad network access [1] is one of the essential features. This promotes access to the cloud from different locations and with different devices (e.g., mobile phones, workstations), which increases the issues that it is necessary to consider when ensuring security. Protecting and controlling access to data, therefore, becomes a real technical challenge that requires more metrics and tools to support it.

Another challenge is related to ensuring security in multi-cloud environments (i.e., a cloud approach composed of more than one cloud service, provided by more than one public or private provider), in which the overall security level of the service will be an aggregation of the security properties of the linked services. There is a particular need for metrics that measure accountability in multi-cloud environments so as to allow a full verification of both physical and virtual resources.

d: MAINTAINABILITY

This characteristic accounted for 19 metric operationalizations, which represent only 4% of the total metrics. This does not appear to be enough when considering that the majority of the cost of software (including the cost of cloud services) is derived not from its initial development, but rather from its continuous maintenance. Several different people will have to make changes to the cloud service over time, both to maintain its current behavior or to adapt/evolve the service to cope with new requirements, and they should be able to make these changes effectively and efficiently, and have a mechanism with which to check this.

Some of the quality attributes considered most relevant (by the authors of the primary studies) as regards measuring maintainability are modifiability, reusability, modularity.

An analysis of the metrics collected shows that most of them (i.e., 8 metric operationalizations) measure *reusability*. Reusability is the degree to which an asset can be used in more than one cloud service or to build other assets. In cloud environments, an asset can be a component or an artifact of the cloud service. Some examples of assets are cloud technologies such as virtual machines at the IaaS level or

containers and microservices at the SaaS level. Cloud services are also developed using methodologies such as Agile and DevOps, in which reuse is a key factor.

Examples of the reusability metrics for SaaS include: i) coverage of variability, which measures how many of the variation points included in the domain are actually realized in the cloud service, and ii) functional commonality, which measures an average amount of the commonality of each functional feature defined in a target service [3], [33]. We also found other indicators with which to measure reusability that combine different metrics (i.e., functional commonality (FC), non-functional commonality (NFC) and coverage of variability (CV) [3]). All these indicators use the Weighted-Sum-based method as a measurement function.

There are also metrics that measure attributes related to *modifiability* (i.e., 7 metric operationalizations). Modifiability is the ability to modify a cloud service effectively and efficiently without introducing defects or degrading the service quality. These metrics can be applied to cloud artifacts obtained in different lifecycle phases (e.g., design, document, test cases) that implement a particular change. Examples of these metrics include flexible force, which measures the ease or difficulty with which a service can be changed as a response to a customer request [47], and rating the ability to add or remove predefined features from a service in order to accommodate users' preferences [34].

We also found one metric that measures *analyzability*, i.e., the number of defects per cloud service [47]. Analyzability is the ability to assess the impact that changes in one or more of its resources have on a cloud service or to diagnose deficiencies or causes of failure.

We found one metric with which to measure *modularity*, i.e., the ratio of the number of elements without external dependencies [33]. Modularity is the degree to which a cloud service can be composed of components, in such a way that changes in one component have a minimal impact on other components.

Finally, there is one metric related to maintainability itself, i.e., maintenance cost, which calculates the amount of capacity that each edge/node needs in order to be restored and cannot exceed the budget [66].

The high degree of granularity (several components) of cloud services and their increasing rate of delivery of short-time releases (daily or several per day) signify that the metrics found in literature are insufficient to help establish an adequate control over the maintainability of cloud services.

e: FUNCTIONAL SUITABILITY

This characteristic accounted for 15 metric operationalizations, which represents only 3% of the total number of metrics. Functional suitability is the degree to which the functions do or do not carry out the basic functions. It focuses particularly on three types of functional suitability [5]: i) functional completeness, which is the degree to which the set of functions covers all the specified tasks and user objectives; ii) functional correctness, which is the degree to which a cloud service provides the correct results with the degree of precision required, and iii) functional appropriateness, which is the degree to which the service functions facilitate the accomplishment of specified tasks and objectives.

The common issues related to functional completeness and correctness are mainly incorrect and ineffective data retrieval or ineffective data edits that may originate inadequate results. The issues associated with functional appropriateness are related to services that are not sufficiently flexible to meet business requirements or service level objectives. The relevance is owing to stakeholders' needs, which are specified in SLAs and must be fulfilled by the cloud service.

Our results indicated that the quality attributes that are considered most important as regards measuring functional suitability are: correctness, suitability, appropriateness, and completeness.

Upon analyzing the metrics collected, it will be noted that most of them (i.e., 12 metric operationalizations) measure *functional correctness*. Examples of metrics used to measure correctness include the accuracy of service at the IaaS level, which is measured by Garg *et al.* [58] as the frequency of failure to fulfill the promised SLA in terms of computing units, network, and storage. In contrast, the accuracy of service at the SaaS level is measured by Nadanam & Rajmohan [33] as the degree to which a response to a user's request is correct, and by Singh and Chana [47] as the ratio between the cloud service that is expected and that which is observed.

Finally, we found one metric for each of the following attributes: suitability, functional appropriateness, and functional completeness. These metrics: are the suitability of non-essential features, which measures whether the degree of a customer's requirements are met by the cloud provider [58]; pertinence value, which measures the service unit value expected by a user and the service unit value sent back to that user [60], and completeness, which measures the total existing cloud services over the total requested cloud services [47].

f: PORTABILITY

This characteristic accounted for 10 metric operationalizations, which represent only 2% of the total number of metrics. In the context of cloud computing, portability concerns the customers' ability to move and suitably adapt their applications and data between their own systems and cloud services, and between the cloud services of different cloud service providers and potentially different cloud deployment models [67]. It is one of the most questioned characteristics owing to the high degree of dependence on the provider, and the challenge here is, therefore, to solve problems regarding the movement of data or services between cloud providers. Portability is significant in cloud computing since customers are interested in avoiding lock-in when they choose to use cloud services.

We found 6 metrics related to *installability*. Installability has a different meaning in cloud computing. As a pay-per-use model it expresses the effort required to get a cloud service deployed or ready for use. It can thus be defined

as the degree to which a service can be migrated (ported) from a source system to a target system, or an application from one cloud provider to another one, or between a cloud service consumer's system and a cloud service. As examples, Baranwal & Vidyarthi [35] proposed application-dependent metrics with which to measure the degree to which the cloud service is portable to other platforms and did this by employing platform support, a virtualization measure, and operating system support. This is useful as regards identifying application installability and may differ between applications deployed in the cloud.

We also found 4 metrics related to *adaptability*. Adaptability is the degree to which a cloud service can effectively and efficiently be adapted to different or evolving hardware, software, or other operational or usage environments. Note that adaptability is different from installability (porting) as the former represents the ability to adjust or change the service based on customer's request or technology changes. As an example, Nadanam & Rajmohan [33] measured adaptability in terms of coverage of variability and completeness of variant set in order to determine the effectiveness of adapting services to the use of each service-based application. Garg *et al.* [58] measured it as the time taken to adapt the cloud service to changes or upgrading it to a higher level.

The essential requirements for consumers are customized solutions and independence from the provider. However, the cost of customization is often tied to proprietary solutions in public cloud environments, which limits access to services among multiple providers. Moving or migrating data or services from one provider to another is, therefore, a challenge.

The lack of portability among cloud providers is a relevant aspect that highlights the dependence on providers, thus making cloud consumers vulnerable to price limitations and the quality of service provided by the specific cloud platform.

g: USABILITY

This characteristic accounted for 9 metric operationalizations, which represents only 2% of the total number of metrics. This is an important factor, which is employed to measure the simplicity of using a cloud service and its rapid adoption. According to Stanton *et al.* [68], end users may customize several attributes of a cloud solution for the whole organization. For example, cloud services should be accessible to customers with a variety of needs (accessibility), should allow consumers to change their user interface to suit their needs (customization) and should ensure ease of use by implementing multiple identity access, such that consumers are not aware of the number of authentication/authorization steps they have to go through to access their applications in the cloud (identify management). Customers should, meanwhile, have a sense of control over the functionality of the cloud service (control) and should have ownership over the data they store in the cloud services they use (data ownership). This characteristic additionally comprises other attributes such as learnability, which represents the effort required to learn how to use a cloud service, and user error protection, which represents the degree to which a cloud service protects users against making errors.

Despite the relevance of usability in cloud environments, our results show that very few metrics have been defined and used to measure certain usability attributes (i.e., operability, learnability, and understandability). We found two metric operationalizations for each of these attributes. For example, Garg *et al.* [58] measured the average time taken by previous users to operate, learn, and understand a cloud service. Singh & Chana [47], meanwhile, measured the ratio of successful operations and the time taken to learn the cloud service. Finally, Nadanam & Rajmohan [33] measured the understandability of service as the ratio of the amount of fields, which has unacceptable readability to the total number of fields.

We found one metric that measures *user interface aesthetics*, i.e., USAB-CQ, which measures how easy, efficient and enjoyable it is to use the interface to a cloud service, or assesses the ease of invocation when the cloud service functionality is shown in the form of APIs [48]. User interface aesthetics is the degree to which a cloud service interface provides a pleasing and satisfying interaction to the user.

However, our main concern is that usability was generally measured subjectively by employing qualitative measures. For example, Ezenwoke *et al.* [34] measured it as the ease with which a cloud service can be used, learned, operated, installed, and understood by the user. Moreover, the current measurement efforts are not focused on users; the metrics are, in most cases, subjective and based on estimates of past user experiences. More research is, therefore, needed in order to engage users and consider them as a critical factor in the success of a cloud service.

h: COMPATIBILITY

Compatibility accounted for only 4 metric operationalizations, which represent 1% of the total number of metrics. In cloud computing, there is an intrinsic need for cloud services to exchange and interact with other services, independently of the provider. The challenge here concerns heterogeneous cloud-based infrastructure services (multi-cloud) and application integration from multiproviders and domains. These issues become more challenging to manage as systems grow more complex and interconnected.

An analysis of the metrics collected shows that 3 of them measure *interoperability*. The ISO/IEC 17788 [37] defines interoperability as the ability of two or more systems or applications to exchange information and mutually use the information that has been exchanged. In cloud computing, there are basically two scenarios of interoperability: i) the ability of an application running in a consumer system to exchange information with a cloud service and use the information from it; and ii) the ability of a cloud service to work with other cloud services. The interoperability between the two cloud services is currently becoming more important. For example, Garg *et al.* [58] measure it as the ratio between platforms provided by the provider and platforms required

by the users, while Nadanam & Rajmohan [33], measure interoperability as the level of efficient interactions between the cloud service and its dependent services.

However, we observed a lack of metrics with which to measure *co-existence*, which is the ability of a cloud service to perform its required functions while sharing a common environment and resources with other cloud services. As an example, Nadanam & Rajmohan [33] proposed composability as an indicator that employed a weighted sum of service modularity and service interoperability. Its purpose is to ensure that the service is adaptable to any cloud environment by incorporating other services that can be more easily and efficaciously customized to service users' specific needs.

There is also a lack of metrics that can be used to support the processes of cloud migration, the composition of new services from multiple services, or multi-cloud management in which mechanisms that check and ensure the compatibility of applications and cloud services are required. The benefits of interoperability include lower costs of integration and increasing the value of cloud services by offering new functionality, which is provided by composing cloud services [69]. There is, therefore, a need for further research into a cloud service compatibility evaluation.

2) METRIC TYPE

The results show that there are 108 Base metric operationalizations (accounting for 23% of the total number of metrics). When performing the thematic analysis, we found 43 duplicated base metrics that were merged. We also collected the method employed to measure each base metric. Most of these metric operationalizations were defined in order to measure the capacity of resources (physical or virtualized) and networking. Examples include metrics with which to measure the capacity of CPU (e.g., CPU frequency [59]), memory (e.g., RAM size [56]), network (e.g., jitter [59], bandwidth [61]), virtual machines (e.g., number of CPU cores assigned [59]) and storage (e.g., disk used [45]). These metrics are commonly used to measure elasticity and scalability attributes.

Derived metric operationalizations accounted for 74% of the total number of metrics. These are high-level measures that contain a measurement function that shows how base or derived metrics can be combined or aggregated. An example of metric aggregation is cloud service capacity, which is measured by Baranwal & Vidyarthi as an aggregation of the capacity of the different service components, i.e., CPU, memory, and storage [56].

Indicators accounted for 3% of the total number of metrics. These are also high-level measures that use an analysis model and could estimate or predict another measure. For example, Lee *et al.* [3], measured service efficiency by using a weighted sum of time behavior and resource utilization, whereas Lim & Thiran [70] measured efficiency using a weighted sum of availability, reliability, and response time. The use of weighted ponderations allows stakeholders to express their main concerns when evaluating cloud services.

3) TOOL SUPPORT

Our results show that only 70 metric operationalizations (15%) are supported by a tool that assists stakeholders to perform the measurement process. These tools are mostly commercial monitoring tools that have principally been used to calculate the performance measures of physical and virtualized resources. As an example, Baranwal & Vidyarthi used the CLOUDSLEUTH' tool as a near real-time visualization tool with which to calculate the availability and response time of different service providers [56].

We also observed a common use of benchmarking tools to test the workload capabilities of cloud platforms. Several tools have been developed by industry (e.g., YCSB, TPC-W), academia (e.g., BenchCouds, CloudSuite), or other areas for specific purposes, such as HiBench for Hadoop applications, which analyzes big data. For example, Hwang *et al.* [45] use five benchmarking tools (i.e., YCSB, CloudSuite, HiBench, BenchClouds, and TPC-W) to carry out controlled experiments in order to evaluate the performance on a hybrid cloud.

The outcomes of benchmarking tools can be used as evidence to perform cloud service modifications or adaptations according to the parameters reported in each tool (e.g., the operations tested, the configuration of instances or VM, the size of the instances). The results can, therefore, be connected to the tools or cloud platform (e.g., Amazon EC2, Rackspace) selected and, consequently, affect the reproduction of metric calculations or the consistent comparisons between outcomes. Other uses of benchmarking tools in primary studies are frameworks for cloud service provider selection, cloud service provider ranking, and price ranking.

Our results show that most of the metric operationalizations (i.e., 402, which accounts for 85% of the metrics) are not supported by tools. This means that the measurement process is performed manually. However, this result should be viewed with caution because most of the primary studies did not explicitly mention the instrument or framework used to obtain the value of the metric, so they were classified as manual. In addition, it is well known that most cloud service providers offer functions, APIs or applications with which to monitor cloud resources (e.g., CloudWatch provided by Amazon or AzureWatch provided by Microsoft Azure). Nevertheless, our study reflects how the metrics have been calculated in the selected primary studies.

The low automation level of the collected metrics could be also explained by the fact that many of those metrics are theoretical, use complex algorithms or require external or accumulative data that cannot be easily/effectively implemented. By theoretical metrics, we refer to those metrics that use abstract concepts in their measurement functions, so these concepts should be redefined in terms of specific cloud artifacts and platforms.

4) MEASUREMENT RESULT

With regard to the measurement results, almost all the metrics provide quantitative results (455 metrics, which accounts for 96.8% of the metrics). This is coherent, considering that

Performance Efficiency and Reliability were found to be the quality characteristics with the highest number of metric operationalizations. In cloud environments, these are critical factors that need quantitative measures that will allow cloud service providers and customers to monitor and control them. Examples are response time [56], [59] and response time efficiency [70], whose measurement results are quantitative and provided in units of time (e.g., seconds, milliseconds). Furthermore, the measurement results of metrics such as the number of monitored processes and the number of TCP connections [61] are also quantitative, but their results are provided in absolute values.

Only 12 metric operationalizations (2.6% of the metrics) present qualitative results. Qualitative metrics are mainly used to measure accountability, agility, and cost when comparing different cloud services. Examples of attributes that use qualitative results are continuity, which uses an ordinal scale to qualify the mechanism for emergency preparedness [49].

Finally, only 3 metric operationalizations (0.6%) provided a hybrid measurement result (i.e., SLA/Security, suitability, and data center location). The objective of SLA/Security is to assess the compliance with SLAs by measuring security, privacy, or copyright regulation [45] attributes. Suitability is the degree to which the cloud service provider meets customer requirements and can be used to quantify essential and non-essential features. The essential features are quantified by rating essential requirement satisfaction as one if all features are satisfied, and as zero otherwise. In contrast, non-essential features are quantified as a ratio between the non-essential features provided and the non-essential features required [58]. Finally, the data center location uses the number of data centers and the distance between them, signifying that the provider with the minimum distance is ranked first, and so on [35].

Overall, we believe that the high number of quantitative metrics is very positive as this facilitates the evaluation of a number of quality attributes in an objective manner and the use of these measurement results to support the continuous adaptation and evolution of cloud services in order to satisfy the stakeholders' needs. However, we observed a lack of threshold-based mechanisms to assist the stakeholders in the interpretation of the measurement results. Establishing suitable thresholds for cloud service metrics (i.e., to measure elasticity) is not an easy task. For instance, the workload or application behavior changes, which makes the accuracy of the metric results subjective and prone to uncertainty. There is, therefore, a need for new threshold-based mechanisms that will take the specific characteristics of cloud services into account.

5) CLOUD LIFECYCLE PHASES

Our results show that most metrics are applied in the Operation phase (373 metric operationalizations, which accounts for 56% of the metrics). This indicates that these metrics are mostly being measured during the actual use of the service (e.g., during the cloud service monitoring at runtime). Monitoring is a key component of continuous service improvement from the provider's perspective, and the measurement results are normally exported to the cloud portal to allow the customers to see how their services are performing. Some examples are metrics such as read and write speed on disk, active and inactive connections, CPU, and memory use, whose objective is to inform stakeholders about the provisioning status of their resources [61].

The lifecycle phase with the second highest number of metrics was Acquisition, with 174 metric operationalizations, which corresponds to 26% of the total. This phase is crucial as regards establishing an SLA between the cloud customer and the cloud service provider. In this phase, a prospective customer can use service offerings published by the cloud service provider to check whether the service meets her/his requirements in terms of, for example, security, personal data protection, performance, etc., and to see how one offers comparing with another on the market. For example, Rizvi *et al.* [71] proposed the security index in order to describe the level of security accomplished by cloud providers. This means that consumers do or do not decide to adopt cloud services and require metrics to support their decision-making process.

Most of the metrics in this phase were, therefore, used to evaluate offerings or check the performance of cloud service providers. For example, Abdeladim *et al.* [72] made use of under-provisioning, over-provisioning, and scalability coverage metrics to estimate the demand for resources (e.g., CPU, memory, space and hard disk performance) in the Acquisition phase.

Although we found a great number of metric operationalizations, few of them can be applied to the Requirements (53 operationalizations, corresponding to 8%), Development (58 operationalizations, corresponding to 9%), and Integration phases (8 operationalizations, corresponding to 1%). The quality assessment in these phases of the cloud service lifecycle is equally important, as it is widely accepted that a good design improves the service that will be delivered and decreases defects. However, our results indicate that most of the evaluation effort is focused on the later phases of the service lifecycle.

Finally, despite the relevance of the Retirement phase, which deals with service contract termination or replacement of service issues, we found a limited number of metrics that can be applied in this phase (8 metrics, which accounts for 1% of the metrics). These metrics are mainly related to support (e.g., platforms, operating systems, virtualization, and software tooling) and interoperability (e.g., platforms and resources). The limited number of metrics may be explained by the fact that this phase deals with legislation compliance (data protection) and these issues can be difficult to control by means of metrics. This fact denotes that the duties of providers in the custody of their customer information does not end at the time of contractual closure or when the provider is replaced. Therefore, the provider remains accountable for compliance with local legislation. However, we believe that

some issues related to service termination and data protection could be controlled by means of metrics. With regard to service termination, there is a need for metrics to evaluate issues related to vendor lock-in or the portability and compatibility of application deployment when transferring the service to other platforms. With regard to data protection, there is a need for metrics to control issues related to the transfer, custody, and secure disposal of data in order to avoid unwanted copies of data, and metrics to control the frequency of the data backups.

It should be noted that some metrics can be applied to more than one phase of the service lifecycle. This occurs, for example, in multi-tenant contexts that are widely used by SaaS applications, in which many tenants share a single software instance, and the metric number of tenants can be applied during the Requirements, Acquisition, and Operation phases. In the requirements phase, it is used to establish the expected quota, while in the acquisition phase, it is used to guarantee the quota; finally, in the operation phase, it is used to evaluate the compliance with the quota. Moreover, as the number of tenants has an impact on the scalability and elasticity of the service, it should also be part of the SLA [44].

6) CLOUD ARTIFACT MEASURED

The results show that most metrics (464 metric operationalizations, which accounts for 87% of the total number of metrics) evaluate the actual cloud service. For example, Lee *et al.* [3] used the Coverage of Failure Recovery (CFR) and Coverage of Fault Tolerance (CFT) metrics to assess the reliability of the running cloud service. This concentration of metrics to evaluate the cloud service is consistent because the studies have focused on the operational phase of the service. 11% of the proposed metrics were applied to the Cloud Service Architecture, such as the metric number of replicas for IaaS proposed by Souza *et al.* [59]. In general, we observed a reduced number of metrics that can be used to evaluate cloud architectures and help architects build cloud solutions. Barely 2% was applied to cloud service specification (e.g., the business emergency plan [49]). The findings suggest that there is a need for metrics to be applied in the early stages of the cloud service lifecycle (service specification and architecture) and that future work is required in order to address these lacks.

7) SERVICE TYPE

The results suggest that most metric operationalizations were applied to the IaaS and SaaS service models (51% and 35%, respectively). In SaaS, applications are provided to consumers, and providers are responsible for service deployment, configuration, and maintenance. The functional commonality, non-functional commonality, coverage of variability and reusability metrics were proposed by Lee *et al.* [3] and make it possible to discover the degree of reusability of the application. With regard to IaaS, some representative examples might be metrics such as the number of TCP connections, the bandwidth of the in and out flow, memory and disk usage, which provide a reading status of the capacity of the servers and also apply to virtual machines [61].

Finally, the metrics for PaaS accounted for 14% of the total. In the case of providing application development environments, the concern is the programming capacity or the effective use of software development kits (SDKs). Some examples of these metrics are testing time, user self-service rate and the computing capacity of the resource [47]. It is important to note that some metrics can be applied to more than one type of service. Examples of this are: the scalability expressed as the dynamic interval of auto-scaling resources with workload variation; the performance efficiency expressed as the speedup by the speed gain using multiple processing nodes, and the elasticity expressed as the minimum time to change from an under-provisioned state to a provisioned one in which the available resources match as closely as possible to existing demand [45].

8) STAKEHOLDER'S VIEWPOINT

As expected, the results show that most of the metrics were used to assist cloud service Providers and Consumers, with 374 metric operationalizations (45%) and 283 metric operationalizations (34%), respectively. Note that a metric may assist more than one stakeholder.

A cloud provider undertakes different tasks for the provision of cloud services at different levels (SaaS, PaaS, and IaaS). For example, at the IaaS level, the provider is responsible for providing and managing the physical processing, storage, networking, the hosting environment, and the cloud infrastructure for IaaS consumers [73]. Metrics such as the data transmission speed achieved to represent transmission rate and the delay in transmission proposed by Saiz *et al.* [74] can be used to assist providers monitor the quality of services and identify possible improvements.

A cloud customer browses and selects a service from a cloud provider, sets up the contract with the provider, and uses the service. The activities and usage scenarios may be different among customers depending on the type of service requested (SaaS, PaaS, and IaaS). Examples of metrics that can be used by customers for any type of service include uptime percentage and the repair rate of accidents proposed by Zhou *et al.* [49].

The results also indicated that End-Users, Brokers, and Developers were the roles least involved when measuring the quality of cloud services, with 69 metric operationalizations (8%), 46 metric operationalizations (6%), and 57 metric operationalizations (7%), respectively.

Some of the metrics employed to assist end-users are the gaming time response and the game mean opinion score, both proposed by Wang &Dey [75]. One of the metrics used to assist cloud service brokers when managing the performance of cloud services and negotiating the service clauses between providers and consumers is availability. This quality attribute can be measured using online benchmarking tools such as CLOUDSLEUTH or low-level metrics (e.g., processor time, current connections, uptime) that are available

and extractable from different types of servers (e.g., virtual machine hosts, virtualization servers, client access servers). Brokers record the service log and review it in order to maintain the history of a particular cloud service, and also collect information from users about their experience with the service. This information assists in the selection of cloud service providers according to the user's needs.

Developers used metrics with different purposes, such as evaluating the service performance, improving the user experience, determining the appropriate cloud environment, or specifying the upper threshold of permitted current users. For example, Wen & Hsiao [76] analyzed the relationship between QoS and quality of experience (QoE) in the domain of cloud gaming services, and considered latency as a relevant attribute, as these types of services handle end-user interactions. They measured latency using the round-trip reaction delay metric, which is composed of transmission delay, server-side processing delay and client-side processing delay. These measurement results can, therefore, allow gaming developers to infer a gaming experience index through the use of service quality metrics and assess the impact of their design.

Overall, our findings indicate that the metrics collected were most frequently used to assist service providers control the provision of cloud services in order to guarantee their behavior, and to assist consumers assess the quality of services and ensure their compliance with service level agreements.

9) VALIDATION PROCEDURE

This criterion assessed the extent to which the metrics collected were theoretically and/or empirically validated (i.e., whether the metrics measured what they were intended to measure and whether the results were as expected).

The results show that 443 metric operationalizations (90%) lack any type of validation. This means that only 47 metric operationalizations (10%) were validated (all of them were empirically validated, and there was an absence of theoretical validations).

Of these metric operationalizations, 39 were validated by means of experiments, although most of these experiments did not involve humans – they were experiments that compared the results of the metrics against benchmarks. As an example, Hwang *et al.* [45] conducted benchmarking experiments to validate the effectiveness of metrics as regards measuring elasticity when considering the efficiency and the performance of cloud services (e.g., the resilience represented by a rate or capacity to recover from a failure).

We also found 8 metrics that were validated by means of case studies. For example, in Zheng *et al.* [48], the authors presented a case study that was carried out to evaluate the QoS offered by storage clouds (Amazon S3, Azure Blob, and Aliyun OSS) thought the use of four operations (i.e., create, upload, download and delete). The technical standard used for assessing the metrics was IEEE Std 1061 [77], while the validity criteria employed were correlation, consistency and discriminative power. Correlation assesses whether a sufficiently strong linear association exists between a quality dimension (i.e., attribute) and a metric, consistency assesses whether a metric can accurately rank a set of services by a quality dimension, and discriminative power assesses whether a metric can separate a set of high-quality services from a set of low-quality ones for a quality dimension. One example of a validated metric is RESP-Evaluation, which measures responsiveness as the promptness of a cloud service to perform a request. The maximum acceptable time used is defined by employing the user's viewpoint rather than the perspective of the system.

We observed that, in general, the authors of the primary studies stated that they were carrying out a case study when they were in fact only presenting a proof of concept on how the metrics could be used. To make the use of a metric feasible, it must be well specified, thus enabling it to be evaluated in a reproducible and repeatable manner.

In this regard, we classified a metric as Not Validated when there was no validation at all or when the validation was carried out using an incorrect method. Of these non-validated metrics, 52% presented a proof of concept regarding how the metrics were used. We also observed that most of the studies did not explain the validation process and its results in detail. This limitation made it challenging to know whether or not a proper design existed or whether it was merely a proof of concept, and hence the high number of non-validated metrics.

Finally, there is a need for further validation of existing metrics for cloud services. There is a particular need to provide evidence of both the usefulness of these metrics as internal quality measures and their ability to predict external quality attributes such as performance, security, and maintainability. However, the evaluation of any metric has an associated cost as regards gathering, processing, and storing the data used to produce the value. In cloud environments, the cost of storing data should be considered. The data (and inputs from cloud experts) must be available, and the evaluation of the metric from those inputs must be made at a cost that is acceptable to the stakeholders intended to use the metric and in a timeframe consistent with the decisions the metric is intended to support.

V. AGGREGATING THE RESULTS

In this section, we further discuss the results of this study by analyzing the frequencies obtained when different criteria are combined. Figure 8 shows the results of a four-dimension bubble chart, which combines data regarding the following criteria: QoS characteristic and cloud lifecycle phase in the x-axis, and stakeholder's viewpoint and type of service in the y-axis. These results may indicate that:

- Performance efficiency, reliability, security, and portability are the characteristics with metrics that can be used to assist all stakeholder viewpoints, although with

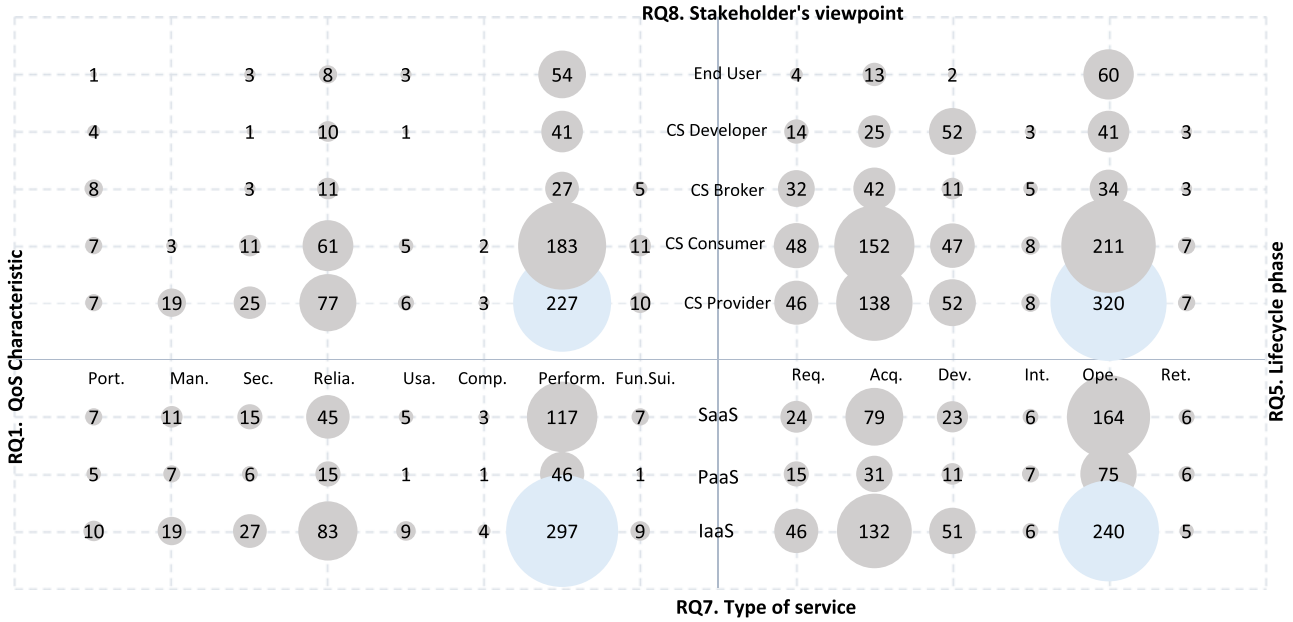

FIGURE 8. Results obtained after combining QoS characteristics and cloud lifecycle phases with stakeholder's viewpoint and type of service.

a different degree of coverage. In this regard, most of these metrics were used to support both providers and consumers. This suggests that most existing metrics are intended to measure external quality attributes, which are mainly concerned with the behavior of the cloud service when it is in use (i.e., performance and reliability), and are of interest to the two major cloud actors (i.e., consumers and providers).

- We found few metrics with which to measure quality attributes related to compatibility, usability, and maintainability, none of them were intended to assist brokers, and very few metrics were found to be useful for developers and end-users.
- The most frequently evaluated phase was operation, which had the highest number of metrics with which to assist providers and end-users. This was followed by the acquisition phase, which had the second highest number of metrics that could be used to assist consumers and providers.
- The majority of the metrics were used to evaluate performance efficiency at the IaaS and SaaS levels, followed by metrics to measure reliability at the IaaS and SaaS levels. This may indicate that most of these metrics were intended to support the monitoring of the performance and reliability of cloud infrastructures and applications. We also observed that the quality characteristics least covered by metrics were compatibility and usability.
- There is a shortage of metrics with which to support developers when evaluating cloud services at the PaaS level, regardless of the quality characteristic.
- The types of cloud services evaluated most frequently are IaaS and SaaS, both in the operation and acquisition phases. This may indicate that these phases represent the main concerns of providers, consumers, and end-users.
- The aforementioned tendency also applies to PaaS services. The phase in which the cloud service is continuously monitored to check whether it meets the committed service level objectives is that of Operation, while the Acquisition phase is crucial as regards establishing an SLA between the cloud customer and the cloud service provider. The focus on IaaS is comprehensive because, from the technical point of view, IaaS gives stakeholders the most control and requires extensive expertise to manage the computing infrastructure. SaaS simultaneously allows the use of cloud-based applications without having to manage the underlying infrastructure. However, very few metrics were oriented toward helping PaaS vendors or developers use runtime environments when developing, testing, and managing their applications.

Figure 9 shows the results of a bubble chart that combines data obtained from metric type, tool support, measurement result and measured cloud artifact on the x-axis, and the QoS characteristic on the y-axis. These results may indicate that:

- The majority of base metrics were used to measure quality attributes related to performance efficiency, while a more significant number of derived metrics gives a greater amount of coverage to performance efficiency and reliability and a lesser amount to all the other quality characteristics from the ISO/IEC 25010 standard. Derived metrics were probably those most frequently used because this type of metric might be more valuable to the stakeholders owing to the fact that they combine

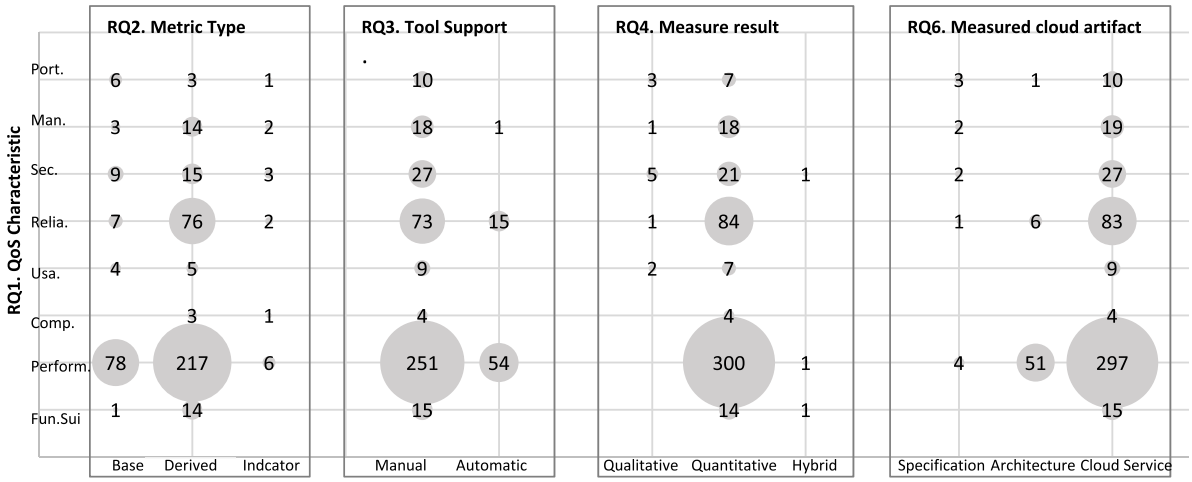

FIGURE 9. Results obtained after combining QoS characteristics with metric type, tool support, measurement result and cloud artifact measured.

base and possibly other derived metrics in order to provide more meaningful information.

- Most of the metrics were measured manually, regardless of the QoS characteristic. The measurement of quality attributes related to performance efficiency [32] and reliability [45] is likely to be more automated than that of the quality attributes related to the other QoS characteristics. This is probably explained by the fact that most of the tools used to calculate these metrics were those provided by cloud platforms in order to monitor the state of cloud services.
- Almost all the metrics were used to evaluate performance efficiency and reliability, and they were measured quantitatively by employing objective measures. This is useful for cloud stakeholders since measurements are easily obtainable, and quantitative values computed from measurements support a detailed analysis of cloud service behavior. Moreover, this is consistent with the fact that most metrics were used during the operation phase, during which automated measures make it possible to continuously monitor the quality of services at runtime.
- The majority of metrics were used to evaluate the performance efficiency and reliability of actual cloud services. Their limited context signifies that this type of metrics can be used to make short-term (action) decisions. There is a need for metrics that cover multiple phases or even the whole cloud service lifecycle. This type of metric is more long-term (vision) oriented.
- There is also a shortage of metrics that can be applied at early stages of the cloud service lifecycle (e.g., specification, cloud architecture and design stages), as an early detection of any deviation allows preventive measures to be taken and less expensive solutions to be employed.

VI. THREATS TO VALIDITY

Despite the fact that a rigorous and systematic process was carried out, it is possible that this work was affected by some threats to its validity. In this section, we shall, therefore, review the actions taken to avoid bias throughout this work.

A. THREATS TO THE IDENTIFICATION OF STUDIES

When employing our search strategies, the key idea was to retrieve as much of the available literature as possible in order to avoid any bias. The scope of the study consequently included research works from different communities, including software engineering, information systems and cloud computing. These communities use different terminologies for the same concepts. We, therefore, searched for common terms and combined them in a search string in an attempt to cover all of them and avoid bias. Moreover, the taxonomy allowed us to properly integrate all the relevant concepts in order to understand how the quality of cloud services was assessed.

Assessing the quality of the search string and the quality of the selected primary studies are key factors when attempting to avoid the possibility of missing or excluding relevant studies.

In order to mitigate the threat related to the search string, we did the following: i) we checked whether the digital libraries included all the relevant journals and conference proceedings from the cloud computing and software quality fields, and performed a manual search in the case of missing sources; ii) we defined the search string on the basis of terms that appeared in relevant papers whose existence was already known (e.g., [3], [48], [49]); iii) we refined the search string by applying different combinations to find that which obtained the best results;. iv) we applied the search string to the same metadata in each paper, and v) we adapted the search string to each digital library. We also avoided full-text searches because this usually leads to a significant number of irrelevant results [24].

In order to identify relevant studies and ensure that the selection process was unbiased, a review protocol was developed. With regard to the primary studies, the main limitation was that the sources selected were academic publications

(i.e., journals and conference proceedings), and gray literature or unpublished reports were, therefore, out of our scope. We focused on academic publications in digital libraries because they are peer-reviewed. Nevertheless, we plan to further validate the relevance of these metrics in industrial contexts.

B. THREATS TO SELECTION AND DATA EXTRACTION CONSISTENCY

In order to validate the selection of primary studies and reduce the research team's subjective judgment, we did the following when selecting the studies:

- The motivations for including or excluding the papers were registered. The first and the second authors had ongoing discussions about which paper should be included, and any discrepancies regarding their inclusion or exclusion were solved by consensus. We also ensured that ten relevant papers from different digital libraries were included (e.g. [3], [48], [49], [58], [60], 75]).
- The other three authors then performed a second iteration on a random sample of 20 papers in order to verify the inclusion and exclusion criteria (10 included and 10 excluded). The discrepancies were solved by consensus, and the team members' level of agreement was assessed using the Kappa Fleiss index. The overall score was 0.87, indicating that the raters had a good level of agreement.
- We performed a quality assessment for all the selected studies. As a result, four studies were discarded. This is an important step as if the quality of the primary studies is low, the conclusions based on those studies are unlikely to be strong and reliable.

With regard to the validation of the data extraction strategy, the following actions were performed:

- The extraction criteria were based on the research questions, and we created a taxonomy and a form to assist us collect the data in a consistent manner.
- The data extraction was performed by the first author and reviewed by the second author. As in the previous phase, these authors had ongoing discussions regarding how the papers could be classified. The discrepancies were solved by consensus.
- We piloted the data extraction strategy externally. An independent researcher assessed the form and the data extraction criteria by classifying two randomly selected papers. Minor changes were made to both the form and the extraction criteria in order to improve clarity.
- The other three authors then performed a second data extraction iteration on a random sample of 30 previously included studies. The results were discussed during a team meeting, and the discrepancies were solved by consensus.

We assessed the reliability of the data extraction using the Fleiss' Kappa statistic index. The results showed a good level of agreement and low variability. The scores obtained were the following: 0.95 for criterion 1, 0.86 for criterion 2, 0.84 for criterion 3, 0.89 for criterion 4, 0.85 for criterion 5, 0.87 for criterion 6, 0.95 for criterion 7, 0.86 for criterion 8 and 0.84 for criterion 9.

C. THREATS TO DATA SYNTHESIS AND RESULTS

With regard to the data synthesis, as described in Section III.C.4, we applied qualitative methods to analyze and synthesize the data (i.e., narrative synthesis and thematic analysis). These research methods ensured a certain amount of consistency in the data analysis. However, it should be noted that in some cases, we had difficulties in extracting and interpreting the data owing to the fact that the information available in the papers was not sufficiently clear or complete for us to be able to answer some research questions. The interpretation bias was, therefore, mitigated as far as possible by involving multiple researchers, having a unified scheme with which to gather the data and piloting the data extraction process with an external researcher.

VII. CONCLUSION AND FUTURE WORK

We have systematically identified, taxonomically classified, and compared existing internal and external quality metrics for cloud services. The metrics were identified by means of a systematic literature review of 84 studies. We specifically identified 470 metrics that were classified and compared on the basis of a taxonomy of quality metrics for cloud services.

The taxonomy allowed us to structure the concepts related to the metrics in a comprehensive manner. We then used narrative synthesis and thematic analysis in order to extract the data from the primary studies and create a catalog of metrics according to the taxonomy.

The taxonomy also allowed the metrics to be aligned with quality attributes and the characteristics from the ISO/IEC 25010, along with the concepts defined by cloud computing standards, thus providing quantitative mechanisms with which to evaluate the quality of cloud services.

The results obtained are useful as regards understanding the state of the art of metrics for cloud services, identifying challenges to be addressed and directing research efforts in the area. In the following subsections, we discuss the implications of our work for practitioners and researchers, and further work.

A. IMPLICATIONS FOR PRACTITIONERS

We believe that our results are relevant to industry, and particularly cloud development companies that are interested in having a mechanism that will allow them to ensure the quality of the service they develop.

Cloud stakeholders (e.g., customers, providers, brokers, cloud architects, infrastructure managers) can select appropriate metrics that can be applied to a specific context according to the type of artifact (e.g., SLA specification, cloud

architecture, actual cloud service), service type (i.e., SaaS, PaaS, IaaS) or cloud lifecycle phase (e.g., Acquisition, Development, Integration, Operation). Specifically, a set of metrics from our catalog can be selected and tailored for inclusion in a larger metrics program.

Cloud customers and providers may use the catalog of metrics as a guideline when specifying service level objectives by identifying the QoS characteristics and attributes that are relevant to their needs and choosing the metrics that should be included in a service-level agreement. Furthermore, since a metric can be calculated using several measurement functions (operationalizations), our catalog of metrics may help stakeholders choose that which best satisfies the organizations' objectives and needs.

The catalog of metrics can also be used for other purposes. For example, a prospective cloud customer could use the metrics to assess a cloud service provider's service offerings in order to verify whether it meets her/his requirements (e.g., security, data protection, performance) and also to see how one offering compares with another one on the market. A customer could also use the metrics to assess the quality of the service acquired.

A provider might use the catalog of metrics for several purposes: to detect defects and remove them before service delivery, to improve the quality characteristics of their services, to guarantee that their customers receive services with the expected quality, or to position their services in the market.

A developer could use the catalog of metrics to evaluate and monitor the performance of the service being developed, thus ensuring that it provides the expected results. This is especially relevant in continuous integration and deployment (CI/CD) or DevOps settings, in which the metrics collected could provide continuous information on the state of the service from different stakeholders' points of view, thus facilitating decision-making and corrective actions.

Overall, the catalog of metrics is a step towards transparency and credibility among the parties involved in the acquisition, development, and operation of cloud services.

B. IMPLICATIONS FOR RESEARCHERS

The findings of our study have implications for researchers who are planning new studies related to the quality of cloud services. We believe that the cloud service quality measurement has not been studied holistically by the authors of the selected primary studies. We have gathered and integrated all the existing knowledge concerning quality metrics for cloud services into a taxonomy. Our results revealed that not all the quality properties or phases of the cloud service lifecycle that are relevant to cloud stakeholders were appropriately covered. For instance, despite the relevance of the retirement phase for organizations owing to its impact on infrastructure and information security, we have found no metrics that help stakeholders manage this phase of the lifecycle.

A large number of metrics are low-level metrics related to performance efficiency (e.g., metrics with which to measure time behavior or resource utilization, such as response time and CPU usage). Moreover, most metrics measure different properties related to cloud infrastructures. More high-level metrics and indicators (e.g., those that support organizations in SLA assessment) are, therefore, required, regardless of the QoS characteristic and type of service. Furthermore, there is a need for metrics that measure the quality of PaaS and SaaS services.

Our findings also show that not all the quality characteristics from the ISO/IEC 25010 are sufficiently covered. portability, usability, and compatibility together accounted for less than 5% of the total number of metrics (22 out of 470 metrics). Usability is a fundamental factor in the success or failure of the adoption of cloud services and is also a differentiating factor in the selection of cloud services. Compatibility is a relevant property in the interoperability of cloud services as it facilitates the integration of services independently of the provider. Moreover, despite the fact that security is considered a key factor when adopting cloud computing, we observed that there were few metrics with which to address the security and privacy concerns of cloud services. More metrics measuring these characteristics are, therefore, needed.

Our results also show that the majority of papers reported evaluations during the operation phase or in a single phase of the cloud service lifecycle. Quality assessments in each phase of the lifecycle are critical to ensure that the service will actually behave as expected. We, therefore, consider that there is an important shortage of metrics that can be applied in the early stages of cloud service development, and not only when the service is being used. The main problem appears to be that most quality assessment practices do not take advantage of the intermediate cloud artifacts that are produced during the early stages of the lifecycle (e.g., requirements specifications, cloud architectures). Moreover, as the quality of services needs to be addressed throughout the entire cloud service lifecycle, it is necessary to combine appropriate metrics identified in this study with other technical solutions in order to provide an overall view of the quality of a cloud service. New research should be oriented toward integrating quality evaluations, whose intermediate artifacts can be effectively evaluated, into the cloud development lifecycle.

A further finding was that the metrics collected were oriented toward the evaluation of services in single cloud environments. We observed a need for further research as regards ensuring the quality of services in multi-cloud environments. The multi-cloud includes hybrid, federated, public, and private cloud infrastructures, and has become fundamental in providing flexible services to organizations. However, this makes it much harder to monitor and ensure the quality of cloud services, since the quality of a service depends on the quality of the related services that may be deployed across several physical or virtual infrastructures.

Metrics are also required to help control and manage new practices for the development, integration, and continuous delivery of cloud services.

Finally, most of the metrics identified in the primary studies have not yet been established in industrial practices, indicating that these metrics still need to be empirically validated on industrial projects of various scales and in different domains. Empirical studies (e.g., controlled experiments or industrial case studies) are, therefore, necessary in order to provide evidence on the usefulness of these metrics.

C. FURTHER WORK

There is a need to evaluate the relevance of the existing quality attributes and metrics for cloud services, considering the possible impacts (trade-offs) among the quality attributes. This will be done by conducting a survey with practitioners in specific domains.

There is also a need for more in-depth analyses of the level of integration of the metrics collected into the different phases of the cloud service lifecycle. We particularly wish to analyze the types of decisions a metric is intended to support, the measurement domain, the context in which the metric is meaningful, and the properties of the cloud service being measured. In this respect, ongoing research is based on using the results of this study to define an operationalized product quality model for cloud services. This quality model will bridge the gap between concrete measurements and abstract quality characteristics. The quality model will be technology-independent but may be tailored to specific cloud domains and platforms.

We identify the need to measure the quality of experience (QoE) of users when interacting with cloud services. Therefore, we also plan to conduct a similar study in order to collect and analyze the existing metrics that have been used to evaluate QoE. This course of action leads to several open issues and research directions on QoS metrics vs. QoE metrics in cloud computing.

First, QoE has been strongly influenced by QoS, because of some technical aspects of cloud service such as performance can influence some dimensions of QoE. One research direction is to find out what these dimensions are. Another one is to evaluate which QoS metrics have an impact on user's overall QoE in different contexts of use.

Second, QoS and QoE can complement each other, although subtle differences between them often lead towards separate policy-based service management. Another research direction is to conduct an empirical study to find out sets of QoS and QoE metrics that can be used to support different policy-based service management approaches.

Third, QoS and QoE metrics can be used to monitor the quality of service at runtime and drive the dynamic adaptation of cloud services. Another research direction is to find out what QoS and QoE metrics can be successfully used to support the dynamic adaptation of cloud service architectures. Finally, in real-time environments such as Fog computing, user interests regarding different cloud services vary from one to another and QoE factors may change very frequently [78]. In addition, fog computing requires QoS to measure and monitor the delivered services efficiently. Therefore, another research direction is to find out which QoS and QoE metrics are more appropriate to support efficient QoE-aware policies in fog environments.

Finally, future research is planned to define a framework with which to support the architectural adaptation of cloud services. The framework will use the metrics collected in order to continuously monitor the quality of services and drive the adaption of the cloud service architecture so as to improve the service quality or the user experience.

APPENDIX A: SELECTED PRIMARY STUDIES

This section provides the primary studies resulting from the selection process, sorted alphabetically by authors:

S01 Abd, S. K., Al-Haddad, S. A. R., Hashim, F., Abdullah, A. B. H. J., & Yussof, S. (2017). An effective approach for managing power consumption in cloud computing infrastructure. Journal of Computational Science, 21, 349–360. https://doi.org/https://doi.org/10.1016/j.jocs.2016.11.007

S02 Abdeladim, A., Baina, S., & Baina, K. (2014). Elasticity and scalability centric quality model for the cloud. In 2014 Third IEEE International Colloquium in Information Science and Technology (CIST) (pp. 135–140). http://doi.org/10.1109/CIST.2014.7016607

S03 Abrahão, S., & Insfran, E. (2017). Models@runtime for Monitoring Cloud Services in Google App Engine. In 2017 IEEE World Congress on Services (SERVICES) (pp. 30–35). https://doi.org/10.1109/SERVICES.2017.14

S04 Alam, A. F. B., Soltanian, A., Yangui, S., Salahuddin, M. A., Glitho, R., & Elbiaze, H. (2016). A Cloud Platform-as-a-Service for multimedia conferencing service provisioning. In 2016 IEEE Symposium on Computers and Communication (ISCC) (pp. 289–294). https://doi.org/10.1109/ISCC.2016.7543756

S05 Al-Jawad, A., Trestian, R., Shah, P., & Gemikonakli, O. (2015). BaProbSDN: A probabilistic-based QoS routing mechanism for Software Defined Networks. In Network Softwarization (NetSoft), 2015 1st IEEE Conference on (pp. 1–5). http://doi.org/10.1109/NETSOFT.2015.7116128

S06 de Oliveira Jr., F. A., & Ledoux, T. (2011). Self-management of Applications QoS for Energy Optimization in Datacenters. In Green Computing Middleware on Proceedings of the 2Nd International Workshop (pp. 3:1–3:6). New York, NY, USA: ACM. doi:10.1145/2088996.2088999

S07 Arumugam, K., & Sumathi, P. (2017). Secure and QoS guaranteed selection resource for storing health care information of cloud users. In 2017 International Conference on Computing Methodologies and Communication (ICCMC) (pp. 1165–1170). https://doi.org/10.1109/ICCMC.2017.8282657

S08 Bao, D., Xiao, Z., Sun, Y., & Zhao, J. (2010). A method and framework for quality of cloud services measurement. In 2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE) (Vol. 5, pp. V5–358–V5–362). http://doi.org/10.1109/ICACTE.2010.5579535

S09 Baranwal, G., & Vidyarthi, D. P. (2014). A framework for selection of best cloud service provider using ranked voting method. In Advance Computing Conference (IACC), 2014 IEEE International (pp. 831–837). http://doi.org/10.1109/IAdCC.2014.6779430

S10 Baranwal, G., & Vidyarthi, D. P. (2016). A cloud service selection model using improved ranked voting method. Concurrency and Computation: Practice and Experience, 28(13), 3540–3567. https://doi.org/10.1002/cpe.3740

S11 Barba-Jimenez, C., Ramirez-Velarde, R., Tchernykh, A., Rodríguez-Dagnino, R., Nolazco-Flores, J., & Perez-Cazares, R. (2016). Cloud based Video-on-Demand service model ensuring quality of service and scalability. Journal of Network and Computer Applications, 70, 102–113. https://doi.org/10.1016/j.jnca.2016.05.007

S12 Bardhan, S., & Milojicic, D. (2012). A Mechanism to Measure Quality-of-service in a Federated Cloud Environment. In Proceedings of the 2012 Workshop on Cloud Services, Federation, and the 8th Open Cirrus Summit (pp. 19–24). New York, NY, USA: ACM. doi:10.1145/2378975.2378981

S13 Bousselmi, K., Brahmi, Z., & Gammoudi, M. M. (2016). QoS-Aware Scheduling of Workflows in Cloud Computing Environments. In 2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA) (pp. 737–745). http://doi.org/10.1109/AINA.2016.72
S14 Bruneo, D. (2014). A Stochastic Model to Investigate Data Center Performance and QoS in IaaS Cloud Computing Systems. IEEE Transactions on Parallel and Distributed Systems, 25(3), 560–569. http://doi.org/10.1109/TPDS.2013.67
S15 Cedillo, P., Jimenez-Gomez, J., Abrahao, S., & Insfran, E. (2015). Towards a Monitoring Middleware for Cloud Services. In Services Computing (SCC), 2015 IEEE International Conference on (pp. 451–458). http://doi.org/10.1109/SCC.2015.68
S16 Cervino, J., Rodriguez, P., Trajkovska, I., Mozo, A., & Salvachua, J. (2011). Testing a Cloud Provider Network for Hybrid P2P and Cloud Streaming Architectures. In Cloud Computing (CLOUD), 2011 IEEE International Conference on (pp. 356–363). http://doi.org/10.1109/CLOUD.2011.52
S17 Costa, C. M., Leite, C. R. M., & Sousa, A. L. (2015). Service Response Time Measurement Model of Service Level Agreements in Cloud Environment. In 2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity) (pp. 969–974). http://doi.org/10.1109/SmartCity.2015.196
S18 de Assunção, M. D., Cardonha, C. H., Netto, M. A. S., & Cunha, R. L. F. (2016). Impact of user patience on auto-scaling resource capacity for cloud services. Future Generation Computer Systems, 55, 41–50. http://doi.org/http://dx.doi.org/10.1016/j.future.2015.09.001
S19 Dou, W., Xu, X., Meng, S., & Yu, S. (2015). An Energy-Aware QoS Enhanced Method for Service Computing across Clouds and Data Centers. In 2015 Third International Conference on Advanced Cloud and Big Data (pp. 80–87). http://doi.org/10.1109/CBD.2015.23
S20 Duggan, J., Cetintemel, U., Papaemmanouil, O., & Upfal, E. (2011). Performance Prediction for Concurrent Database Workloads. In Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data (pp. 337–348). New York, NY, USA: ACM. doi:10.1145/1989323.1989359
S21 Ezenwoke, A., Daramola, O., & Adigun, M. (2018). QoS-based ranking and selection of SaaS applications using heterogeneous similarity metrics. Journal of Cloud Computing, 7(1), 15. https://doi.org/10.1186/s13677-018-0117-4
S22 Faragardi, H. R., Shojaee, R., Tabani, H., & Rajabi, A. (2013). An analytical model to evaluate reliability of cloud computing systems in the presence of QoS requirements. In Computer and Information Science (ICIS), 2013 IEEE/ACIS 12th International Conference on (pp. 315–321). http://doi.org/10.1109/ICIS.2013.6607860
S23 Garcia-Pineda, M., Segura-Garcia, J., & Felici-Castell, S. (2018). Estimation techniques to measure subjective quality on live video streaming in Cloud Mobile Media services. Computer Communications, 118, 27–39. https://doi.org/10.1016/j.comcom.2017.08.009
S24 Garg, S. K., Versteeg, S., & Buyya, R. (2013). A framework for ranking of cloud computing services. Future Generation Computer Systems, 29(4), 1012–1023. http://doi.org/http://dx.doi.org/10.1016/j.future.2012.06.006
S25 Ghafari, S. M., Fazeli, M., Patooghy, A., & Rikhtechi, L. (2013). Bee-MMT: A load balancing method for power consumption management in cloud computing. In Contemporary Computing (IC3), 2013 Sixth International Conference on (pp. 76–80). http://doi.org/10.1109/IC3.2013.6612165
S26 Ghahramani, M. H., Zhou, M., & Hon, C. T. (2017). Toward cloud computing QoS architecture: analysis of cloud systems and cloud services. IEEE/CAA Journal of Automatica Sinica, 4(1), 6–18. https://doi.org/10.1109/JAS.2017.7510313
S27 Gholami, A., & Arani, M. G. (2015). A trust model for resource selection in cloud computing environment. In 2015 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI) (pp. 144–151). http://doi.org/10.1109/KBEI.2015.7436036
S28 Ghosh, R., Longo, F., Naik, V. K., & Trivedi, K. S. (2010). Quantifying Resiliency of IaaS Cloud. In Reliable Distributed Systems, 2010 29th IEEE Symposium on (pp. 343–347). http://doi.org/10.1109/SRDS.2010.49
S29 Gonzales, D., Kaplan, J. M., Saltzman, E., Winkelman, Z., & Woods, D. (2017). Cloud-Trust—a Security Assessment Model for Infrastructure as a Service (IaaS) Clouds. IEEE Transactions on Cloud Computing, 5(3), 523–536. https://doi.org/10.1109/TCC.2015.2415794
S30 Guérout, T., Medjiah, S., Costa, G. Da, & Monteil, T. (2014). Quality of service modeling for green scheduling in Clouds. Sustainable Computing: Informatics and Systems, 4(4), 225–240. http://doi.org/http://dx.doi.org/10.1016/j.suscom.2014.08.006
S31 Hasan, M. S., Alvares, F., Ledoux, T., & Pazat, J. (2017). Investigating Energy Consumption and Performance Trade-Off for Interactive Cloud Application. IEEE Transactions on Sustainable Computing, 2(2), 113–126. https://doi.org/10.1109/TSUSC.2017.2714959
S32 Hassam, M., Kara, N., Belqasmi, F., & Glitho, R. (2014). Virtualized Infrastructure for Video Game Applications in Cloud Environments. In Proceedings of the 12th ACM International Symposium on Mobility Management and Wireless Access (pp. 109–114). New York, NY, USA: ACM. doi:10.1145/2642668.2642679
S33 Hecht, G., Jose-Scheidt, B., Figueiredo, C. D., Moha, N., & Khomh, F. (2014). An Empirical Study of the Impact of Cloud Patterns on Quality of Service (QoS). In Cloud Computing Technology and Science (CloudCom), 2014 IEEE 6th International Conference on (pp. 278–283). http://doi.org/10.1109/CloudCom.2014.141
S34 Heidari, P., Boucheneb, H., & Shami, A. (2015). A Formal Approach for QoS Assurance in the Cloud. In 2015 IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom) (pp. 629–634). http://doi.org/10.1109/CloudCom.2015.36
S35 Hu, Y., Deng, B., Yang, Y., & Wang, D. (2017). Elasticity evaluation of IaaS cloud based on mixed workloads. In Proceedings - 15th International Symposium on Parallel and Distributed Computing, ISPDC 2016 (pp. 157–164). Beijing Institute of System Engineering, Beijing, China. https://doi.org/10.1109/ISPDC.2016.28
S36 Hwang, K., Bai, X., Shi, Y., Li, M., Chen, W.-G., & Wu, Y. (2016). Cloud Performance Modeling with Benchmark Evaluation of Elastic Scaling Strategies. IEEE Transactions on Parallel and Distributed Systems, 27(1), 130–143. https://doi.org/10.1109/TPDS.2015.2398438
S37 Ibrahim, A. A. Z. A., Wasim, M. U., Varrette, S., & Bouvry, P. (2018). PRESEnCE: Performance Metrics Models for Cloud SaaS Web Services. In 2018 IEEE 11th International Conference on Cloud Computing (CLOUD) (pp. 936–940). https://doi.org/10.1109/CLOUD.2018.00140
S38 Joy, N., Chandrasekaran, K., & Binu, A. (2015). A study on energy efficient cloud computing. In 2015 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC) (pp. 1–6). http://doi.org/10.1109/ICCIC.2015.7435661
S39 Kaaniche, N., Mohamed, M., Laurent, M., & Ludwig, H. (2017). Security SLA Based Monitoring in Clouds. In 2017 IEEE International Conference on Edge Computing (EDGE) (pp. 90–97). https://doi.org/10.1109/IEEE.EDGE.2017.20
S40 Karim, R., Ding, C., & Miri, A. (2015). End-to-End Performance Prediction for Selecting Cloud Services Solutions. In Service-Oriented System Engineering (SOSE), 2015 IEEE Symposium on (pp. 69–77). http://doi.org/10.1109/SOSE.2015.11
S41 Katsaros, G., Subirats, J., Fitó, J. O., Guitart, J., Gilet, P., & Espling, D. (2013). A service framework for energy-aware monitoring and VM management in Clouds. Future Generation Computer Systems, 29(8), 2077–2091. http://doi.org/http://dx.doi.org/10.1016/j.future.2012.12.006
S42 Kaur, P. D., & Chana, I. (2014). A resource elasticity framework for QoS-aware execution of cloud applications. Future Generation Computer Systems, 37, 14–25. http://doi.org/http://dx.doi.org/10.1016/j.future.2014.02.018
S43 Kirsal, Y., Ever, Y. K., Mostarda, L., & Gemikonakli, O. (2015). Analytical Modelling and Performability Analysis for Cloud Computing Using Queuing System. In 2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing (UCC) (pp. 643–647). http://doi.org/10.1109/UCC.2015.115
S44 Lee, J. Y., Lee, J. W., Cheun, D. W., & Kim, S. D. (2009). A Quality Model for Evaluating Software-as-a-Service in Cloud Computing. In Software Engineering Research, Management and Applications, 2009. SERA '09. 7th ACIS International Conference on (pp. 261–266). http://doi.org/10.1109/SERA.2009.43

S45 Lim, E., & Thiran, P. (2014). Communication of Technical QoS among Cloud Brokers. In Cloud Engineering (IC2E), 2014 IEEE International Conference on (pp. 403–409). http://doi.org/10.1109/IC2E.2014.92

S46 Lin, Y.-K., & Chang, P.-C. (2011). Maintenance reliability estimation for a cloud computing network with nodes failure. Expert Systems with Applications, 38(11), 14185–14189. http://doi.org/http://dx.doi.org/10.1016/j.eswa.2011.04.230

S47 Liu, M., Dou, W., Yu, S., & Zhang, Z. (2014). A clusterized firewall framework for cloud computing. In 2014 IEEE International Conference on Communications (ICC) (pp. 3788–3793). http://doi.org/10.1109/ICC.2014.6883911

S48 Liu, X., Xia, C., Wang, T., & Zhong, L. (2017). CloudSec: A Novel Approach to Verifying Security Conformance at the Bottom of the Cloud. In 2017 IEEE International Congress on Big Data (BigData Congress) (pp. 569–576). https://doi.org/10.1109/BigDataCongress.2017.87

S49 Lu, L., & Yuan, Y. (2018). A novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects. Journal of Systems and Software, 143, 71–86. https://doi.org/10.1016/j.jss.2018.05.004

S50 Manuel, P. (2015). A trust model of cloud computing based on Quality of Service. Annals of Operations Research, 233(1), 281–292. http://doi.org/10.1007/s10479-013-1380-x

S51 Mastelic, T., Brandic, I., & Jaarevic, J. (2014). CPU Performance Coefficient (CPU-PC): A Novel Performance Metric Based on Real-Time CPU Resource Provisioning in Time-Shared Cloud Environments. In Cloud Computing Technology and Science (CloudCom), 2014 IEEE 6th International Conference on (pp. 408–415). http://doi.org/10.1109/CloudCom.2014.13

S52 Mesbahi, M. R., Rahmani, A. M., & Hosseinzadeh, M. (2018). Reliability and high availability in cloud computing environments: a reference roadmap. Human-Centric Computing and Information Sciences, 8(1), 20. https://doi.org/10.1186/s13673-018-0143-8

S53 Nadanam, P., & Rajmohan, R. (2012). QoS evaluation for web services in cloud computing. In Computing Communication Networking Technologies (ICCCNT), 2012 Third International Conference on (pp. 1–8). http://doi.org/10.1109/ICCCNT.2012.6395991

S54 Pedersen, J. M., Riaz, M. T., Junior, J. C., Dubalski, B., Ledzinski, D., & Patel, A. (2011). Assessing Measurements of QoS for Global Cloud Computing Services. In Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on (pp. 682–689). http://doi.org/10.1109/DASC.2011.120

S55 Preuveneers, D., Heyman, T., Berbers, Y., & Joosen, W. (2016). Systematic scalability assessment for feature oriented multi-tenant services. Journal of Systems and Software, 116, 162–176. https://doi.org/10.1016/j.jss.2015.12.024

S56 Qian, S., Cao, J., Le Mouël, F., Li, M., & Wang, J. (2015). Towards Prioritized Event Matching in a Content-based Publish/Subscribe System. In Proceedings of the 9th ACM International Conference on Distributed Event-Based Systems (pp. 116–127). New York, NY, USA: ACM. doi:10.1145/2675743.2771823

S57 Ran, Y., Shi, Y., Yang, E., Chen, S., & Yang, J. (2014). Dynamic resource allocation for video transcoding with QoS guaranteeing in cloud-based DASH system. In 2014 IEEE Globecom Workshops (GC Wkshps) (pp. 144–149). http://doi.org/10.1109/GLOCOMW.2014.7063421

S58 Ravindhren, V. G., & Ravimaran, S. (2017). CCMA—cloud critical metric assessment framework for scientific computing. Cluster Computing. https://doi.org/10.1007/s10586-017-1384-4

S59 Ravindran, K. (2013). Self-Assessment and Reconfiguration Methods for Autonomous Cloud-based Network Systems. In Distributed Simulation and Real Time Applications (DS-RT), 2013 IEEE/ACM 17th International Symposium on (pp. 87–94). http://doi.org/10.1109/DS-RT.2013.37

S60 Rizvi, S., Ryoo, J., Kissell, J., & Aiken, B. (2015). A Stakeholder-oriented Assessment Index for Cloud Security Auditing. In Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication (pp. 55:1–55:7). New York, NY, USA: ACM. doi:10.1145/2701126.2701226

S61 Rizvi, S., Roddy, H., Gualdoni, J., & Myzyri, I. (2017). Three-Step Approach to QoS Maintenance in Cloud Computing Using a Third-Party Auditor. Procedia Computer Science, 114, 83–92. https://doi.org/10.1016/j.procs.2017.09.014

S62 Roohitavaf, M., Entezari-Maleki, R., & Movaghar, A. (2013). Availability Modeling and Evaluation of Cloud Virtual Data Centers. In Parallel and Distributed Systems (ICPADS), 2013 International Conference on (pp. 675–680). http://doi.org/10.1109/ICPADS.2013.120

S63 Saiz, E., Ibarrola, E., Cristobo, L., & Taboada, I. (2014). A cloud platform for QoE evaluation: QoXcloud. In ITU Kaleidoscope Academic Conference: Living in a converged world - Impossible without standards?, Proceedings of the 2014 (pp. 241–247). http://doi.org/10.1109/Kaleidoscope.2014.6858471

S64 Samet, N., Letaïfa, A. Ben, Hamdi, M., & Tabbane, S. (2016). Real-Time User Experience Evaluation for Cloud-Based Mobile Video. In 2016 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA) (pp. 204–208). http://doi.org/10.1109/WAINA.2016.120

S65 Singh, S., & Chana, I. (2015). Q-aware: Quality of service based cloud resource provisioning. Computers & Electrical Engineering, 47, 138–160. http://doi.org/http://dx.doi.org/10.1016/j.compeleceng.2015.02.003

S66 Slivar, I., Skorin-Kapov, L., & Suznjevic, M. (2016). Cloud Gaming QoE Models for Deriving Video Encoding Adaptation Strategies. In Proceedings of the 7th International Conference on Multimedia Systems (pp. 18:1–18:12). New York, NY, USA: ACM. doi:10.1145/2910017.2910602

S67 Son, S., & Sim, K. M. (2015). Adaptive and similarity-based trade-off algorithms in a price-timeslot-QoS negotiation system to establish cloud SLAs. Information Systems Frontiers, 17(3), 565–589. http://doi.org/10.1007/s10796-013-9432-y

S68 Sousa, F. R. C., & Machado, J. C. (2012). Towards Elastic Multi-Tenant Database Replication with Quality of Service. In Utility and Cloud Computing (UCC), 2012 IEEE Fifth International Conference on (pp. 168–175). http://doi.org/10.1109/UCC.2012.36

S69 Souza, R. H. de, Flores, P. A., Dantas, M. A. R., & Siqueira, F. (2016). Architectural recovering model for Distributed Databases: A reliability, availability and serviceability approach. In 2016 IEEE Symposium on Computers and Communication (ISCC) (pp. 575–580). https://doi.org/10.1109/ISCC.2016.7543799

S70 Taherizadeh, S., & Stankovski, V. (2017). Incremental Learning from Multi-level Monitoring Data and Its Application to Component Based Software Engineering. In 2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC) (Vol. 2, pp. 378–383). https://doi.org/10.1109/COMPSAC.2017.148

S71 Vedam, V., & Vemulapati, J. (2012). Demystifying Cloud Benchmarking Paradigm - An in Depth View. In 2012 IEEE 36th Annual Computer Software and Applications Conference (pp. 416–421). http://doi.org/10.1109/COMPSAC.2012.61

S72 Wagle, S. S., Guzek, M., Bouvry, P., & Bisdorff, R. (2015). An Evaluation Model for Selecting Cloud Services from Commercially Available Cloud Providers. In 2015 IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom) (pp. 107–114). http://doi.org/10.1109/CloudCom.2015.94

S73 Wang, S., & Dey, S. (2012). Cloud Mobile Gaming: Modeling and Measuring User Experience in Mobile Wireless Networks. SIGMOBILE Mob. Comput. Commun. Rev., 16(1), 10–21. doi:10.1145/2331675.2331679

S74 Wen, Z. Y., & Hsiao, H. F. (2014). QoE-driven performance analysis of cloud gaming services. In Multimedia Signal Processing (MMSP), 2014 IEEE 16th International Workshop on (pp. 1–6). http://doi.org/10.1109/MMSP.2014.6958835

S75 Wu, X., Liu, G., & Xu, J. (2015). A QoS-constrained scheduling for access requests in cloud storage. In Industrial Electronics and Applications (ICIEA), 2015 IEEE 10th Conference on (pp. 155–160). http://doi.org/10.1109/ICIEA.2015.7334102

S76 Xia, Y., Zhou, M., Luo, X., Zhu, Q., Li, J., & Huang, Y. (2015). Stochastic Modeling and Quality Evaluation of Infrastructure-as-a-Service Clouds. IEEE Transactions on Automation Science and Engineering, 12(1), 162–170. http://doi.org/10.1109/TASE.2013.2276477

S77 Xiao, Y., Lin, C., Jiang, Y., Chu, X., & Shen, X. (2010). Reputation-Based QoS Provisioning in Cloud Computing via Dirichlet Multinomial Model. In Communications (ICC), 2010 IEEE International Conference on (pp. 1–5). http://doi.org/10.1109/ICC.2010.5502407

S78 Xiong, K., & Chen, X. (2015). Ensuring Cloud Service Guarantees via Service Level Agreement (SLA)-Based Resource Allocation. In 2015 IEEE 35th International Conference on Distributed Computing Systems Workshops (pp. 35–41). http://doi.org/10.1109/ICDCSW.2015.18
S79 Xu, H., Qiu, X., Sheng, Y., Luo, L., & Xiang, Y. (2018). A Qos-Driven Approach to the Cloud Service Addressing Attributes of Security. IEEE Access, 6, 34477–34487. https://doi.org/10.1109/ACCESS.2018.2849594
S80 Yu, N., Gu, F., Guo, X., & He, Z. (2015). A Fine-grained Flow Control Model for Cloud-assisted Data Broadcasting. In Proceedings of the 18th Symposium on Communications {&} Networking (pp. 24–31). San Diego, CA, USA: Society for Computer Simulation International. Retrieved from http://dl.acm.org/citation.cfm?id=2872550.2872554
S81 Zant, B. El, & Gagnaire, M. (2015). Towards a unified customer aware figure of merit for CSP selection. Journal of Cloud Computing, 4(1), 1–23. http://doi.org/10.1186/s13677-015-0049-1
S82 Zheng, X., Martin, P., & Brohman, K. (2013). Cloud Service Negotiation: A Research Roadmap. In Services Computing (SCC), 2013 IEEE International Conference on (pp. 627–634). http://doi.org/10.1109/SCC.2013.93
S83 Zheng, X., Martin, P., Brohman, K., & Xu, L. D. (2014). CLOUDQUAL: A Quality Model for Cloud Services. IEEE Transactions on Industrial Informatics, 10(2), 1527–1536. http://doi.org/10.1109/TII.2014.2306329
S84 Zhou, P., Wang, Z., Li, W., & Jiang, N. (2015). Quality Model of Cloud Service. In High Performance Computing and Communications (HPCC), 2015

REFERENCES

[1] P. Mell and T. Grance, "NIST SP 800-145 the NIST definition of cloud computing," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep., 2011.
[2] T. Hobfeld, R. Schatz, M. Varela, and C. Timmerer, "Challenges of QoE management for cloud applications," *IEEE Commun. Mag.*, vol. 50, no. 4, pp. 28–36, Apr. 2012.
[3] J. Y. Lee, J. W. Lee, D. W. Cheun, and S. D. Kim, "A quality model for evaluating software-as-a-service in cloud computing," in *Proc. 7th ACIS Int. Conf. Softw. Eng. Res., Manage. Appl.*, Dec. 2009, pp. 261–266.
[4] Z. Li, L. OBrien, R. Cai, and H. Zhang, "Towards a taxonomy of performance evaluation of commercial cloud services," in *Proc. IEEE 5th Int. Conf. Cloud Comput.*, Jun. 2012, pp. 344–351.
[5] *Systems and Software Quality Requirements and Evaluation (SQuaRE)—System and Software Quality Models*, Standard ISO/IEC JTC1 SC7, ISO/IEC 25010, 2011
[6] B. Kitchenham and S. Charters, "Guidelines for performing systematic literature reviews in software engineering, version 2.3," Keele Univ., Durham, U.K., EBSE Tech. Rep., 2007.
[7] L. Badger, T. Grance, R. Patt-Corner, and J. Voas, "NIST SP 800-146 cloud computing synopsis and recommendations," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep., 2012.
[8] S. M. Habib, V. Varadharajan, and M. Muhlhauser, "A trust-aware framework for evaluating security controls of service providers in cloud marketplaces," in *Proc. 12th IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun.*, Jul. 2013, pp. 459–468.
[9] N. Herbst, A. Bauer, S. Kounev, G. Oikonomou, E. Van Eyk, G. Kousiouris, and A. Iosup, "Quantifying cloud performance and dependability: Taxonomy, metric design, and emerging challenges," *ACM Trans. Model. Perform. Eval. Comput. Syst.*, vol. 3, no. 4, pp. 1–36, 2018.
[10] F. D. Muñoz-Escoí and J. M. Bernabéu-Aubán, "A survey on elasticity management in PaaS systems," *Computing*, vol. 99, no. 7, pp. 617–656, Jul. 2017.
[11] E. F. Coutinho, F. R. de Carvalho Sousa, P. A. L. Rego, D. G. Gomes, and J. N. de Souza, "Elasticity in cloud computing: A survey," *Ann. Telecommun.*, vols. 7–8, no. 70, pp. 289–309, 2015.
[12] Y. Al-Dhuraibi, F. Paraiso, N. Djarallah, and P. Merle, "Elasticity in cloud computing: State of the art and research challenges," *IEEE Trans. Services Comput.*, vol. 11, no. 2, pp. 430–447, Mar. 2018.
[13] M. Jelassi, C. Ghazel, and L. A. Saidane, "A survey on quality of service in cloud computing," in *Proc. 3rd Int. Conf. Frontiers Signal Process. (ICFSP)*, Sep. 2017, pp. 63–67.
[14] A. K. Bardsiri and S. M. Hashemi, "QoS metrics for cloud computing services evaluation," *Int. J. Intell. Syst. Appl.*, vol. 6, no. 12, pp. 27–33, Nov. 2014.
[15] S. Easterbrook, J. Singer, M.-A. Storey, and D. Damian, "Selecting empirical methods for software engineering research," in *Guide to Advanced Empirical Software Engineering*, F. Shull, J. Singer, and D. Sjøberg, Eds. Springer, 2007, pp. 285–311.
[16] A. Abdelmaboud, D. N. A. Jawawi, I. Ghani, A. Elsafi, and B. Kitchenham, "Quality of service approaches in cloud computing: A systematic mapping study," *J. Syst. Softw.*, vol. 101, pp. 159–179, Mar. 2015.
[17] Z. Li, L. O'Brien, H. Zhang, and R. Cai, "On a catalogue of metrics for evaluating commercial cloud services," in *Proc. ACM/IEEE 13th Int. Conf. Grid Comput.*, Sep. 2012, pp. 164–173.
[18] Z. Li, H. Zhang, L. O'Brien, R. Cai, and S. Flint, "On evaluating commercial cloud services: A systematic review," *J. Syst. Softw.*, vol. 86, no. 9, pp. 2371–2393, Sep. 2013.
[19] S. Lehrig, H. Eikerling, and S. Becker, "Scalability, elasticity, and efficiency in cloud computing: A systematic literature review of definitions and metrics," in *Proc. 11th Int. ACM SIGSOFT Conf. Qual. Softw. Archit. QoSA*, 2015, pp. 83–92.
[20] J. Scheuner and P. Leitner, "The state of research on function-as-a-service performance evaluation: A multivocal literature review," *ArXiv*, vol. abs/2004.03276, 2020.
[21] M. H. Kashani, A. M. Rahmani, and N. J. Navimipour, "Quality of service-aware approaches in fog computing," *Int. J. Commun. Syst.*, vol. 33, no. 8, pp. 1–34, 2020.
[22] M. Petticrew and H. Roberts, *Systematic Reviews in the Social Sciences: A Practical Guide*. Hoboken, NJ, USA: Blackwell, 2005.
[23] N. E. Fenton and J. M. Bieman, *Software Metrics: A Rigorous and Practical Approach*, 3rd ed. Abingdon, U.K.: Taylor & Francis, 2014.
[24] P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, and M. Khalil, "Lessons from applying the systematic literature review process within the software engineering domain," *J. Syst. Softw.*, vol. 80, no. 4, pp. 571–583, Apr. 2007.
[25] L. Chen, M. A. Babar, and H. Zhang, "Towards an evidence-based understanding of electronic data sources," in *Proc. 14th Int. Conf. Eval. Assessment Softw. Eng.*, Apr. 2010, pp. 1–4.
[26] J. L. Fleiss, *Statistical Methods for Rates and Proportions*. New York, NY, USA: Wiley-Interscience, 1981, pp. 38–46.
[27] S. Montagud, S. Abrahão, and E. Insfran, "A systematic review of quality attributes and measures for software product lines," *Softw. Qual. J.*, vol. 20, nos. 3–4, pp. 425–486, Sep. 2012.
[28] *Systems and Software Engineering—Measurement Process*, Standard ISO/IEC, ISO/IEC 15939, 2007.
[29] F. García, M. F. Bertoa, C. Calero, A. Vallecillo, F. Ruíz, M. Piattini, and M. Genero, "Towards a consistent terminology for software measurement," *Inf. Softw. Technol.*, vol. 48, no. 8, pp. 631–644, Aug. 2006.
[30] *International Vocabulary of Metrology—Basic and general concepts and associated terms*, 3rd ed., International Bureau of Weights and Measures, Saint-Cloud, France, 2008.
[31] *Software Engineering—Product Evaluation*, Standard ISO/IEC, ISO/IEC 14598, 2001.
[32] S. Taherizadeh and V. Stankovski, "Incremental learning from multi-level monitoring data and its application to component based software engineering," in *Proc. IEEE 41st Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, Jul. 2017, pp. 378–383.
[33] P. Nadanam and R. Rajmohan, "QoS evaluation for Web services in cloud computing," in *Proc. 3rd Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Jul. 2012, pp. 1–8.
[34] A. Ezenwoke, O. Daramola, and M. Adigun, "QoS-based ranking and selection of SaaS applications using heterogeneous similarity metrics," *J. Cloud Comput.*, vol. 7, no. 15, pp. 1–15, 2018.
[35] G. Baranwal and D. P. Vidyarthi, "A cloud service selection model using improved ranked voting method," *Concurrency Comput., Pract. Exper.*, vol. 13, pp. 3540–3567, Sep. 2016.
[36] S. Schneider and A. Sunyaev, "CloudLive: A life cycle framework for cloud services," *Electron. Markets*, vol. 25, no. 4, pp. 299–311, Dec. 2015.
[37] *Information Technology—Cloud Computing—Overview and Vocabulary*, Standard ISO/IEC JTC1 SC38, ISO/IEC 17788:2014, 2014.
[38] L. Briand, K. E. Emam, and S. Morasca, "On the application of measurement theory in software engineering," *Empirical Softw. Eng.*, vol. 1, no. 1, pp. 61–88, 1996.

[39] G. Poels and G. Dedene, "Distance-based software measurement: Necessary and sufficient properties for software measures," *Inf. Softw. Technol.*, vol. 42, no. 1, pp. 35–46, Jan. 2000.
[40] S. Whitmire, *Object Oriented Design Measurement*. Hoboken, NJ, USA: Wiley, 1997.
[41] (2020). *Mendeley*. [Online]. Available: https://www.mendeley.com/
[42] D. S. Cruzes and T. Dyba, "Recommended steps for thematic synthesis in software engineering," in *Proc. Int. Symp. Empirical Softw. Eng. Meas.*, Sep. 2011, pp. 275–284.
[43] L. Lu and Y. Yuan, "A novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects," *J. Syst. Softw.*, vol. 143, pp. 71–86, Sep. 2018.
[44] D. Preuveneers, T. Heyman, Y. Berbers, and W. Joosen, "Systematic scalability assessment for feature oriented multi-tenant services," *J. Syst. Softw.*, vol. 116, pp. 162–176, Jun. 2016.
[45] K. Hwang, X. Bai, Y. Shi, M. Li, W.-G. Chen, and Y. Wu, "Cloud performance modeling with benchmark evaluation of elastic scaling strategies," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 1, pp. 130–143, Jan. 2016.
[46] S. S. Wagle, M. Guzek, P. Bouvry, and R. Bisdorff, "An evaluation model for selecting cloud services from commercially available cloud providers," in *Proc. IEEE 7th Int. Conf. Cloud Comput. Technol. Sci. (CloudCom)*, Nov. 2015, pp. 107–114.
[47] S. Singh and I. Chana, "Q-aware: Quality of service based cloud resource provisioning," *Comput. Electr. Eng.*, vol. 47, pp. 138–160, Oct. 2015.
[48] X. Zheng, P. Martin, K. Brohman, and L. Da Xu, "CLOUDQUAL: A quality model for cloud services," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1527–1536, May 2014.
[49] P. Zhou, Z. Wang, W. Li, and N. Jiang, "Quality model of cloud service," in *Proc. IEEE 17th Int. Conf. High Perform. Comput. Commun. (HPCC)*, Aug. 2015, pp. 1418–1423.
[50] P. Manuel, "A trust model of cloud computing based on quality of service," *Ann. Oper. Res.*, vol. 233, no. 1, pp. 281–292, Oct. 2015.
[51] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic mapping studies in software engineering," in *Proc. 12th Int. Conf. Eval. Assessment Softw. Eng. (EASE)*, Jun. 2008, pp. 1–10.
[52] H. Zhang, M. A. Babar, and P. Tell, "Identifying relevant studies in software engineering," *Inf. Softw. Technol.*, vol. 53, no. 6, pp. 625–637, Jun. 2011.
[53] H. M. Khan, G.-Y. Chan, and F.-F. Chua, "An adaptive monitoring framework for ensuring accountability and quality of services in cloud computing," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, Jan. 2016, pp. 249–253.
[54] G. Wang, H. Wang, S. Arroyo, R. Rencher, and J. Tjelle, "Compositional QoS modeling and analysis of cloud-based federated ecosystems," in *Proc. IEEE 16th Int. Enterprise Distrib. Object Comput. Conf.*, Sep. 2012, pp. 173–182.
[55] M. Eisa, M. Younas, K. Basu, and H. Zhu, "Trends and directions in cloud service selection," in *Proc. IEEE Symp. Service-Oriented Syst. Eng. (SOSE)*, Mar. 2016, pp. 423–432.
[56] G. Baranwal and D. P. Vidyarthi, "A framework for selection of best cloud service provider using ranked voting method," in *Proc. IEEE Int. Advance Comput. Conf. (IACC)*, Feb. 2014, pp. 831–837.
[57] S. Rizvi, H. Roddy, J. Gualdoni, and I. Myzyri, "Three-step approach to QoS maintenance in cloud computing using a third-party auditor," *Procedia Comput. Sci.*, vol. 114, pp. 83–92, Jan. 2017.
[58] S. K. Garg, S. Versteeg, and R. Buyya, "A framework for ranking of cloud computing services," *Future Gener. Comput. Syst.*, vol. 29, no. 4, pp. 1012–1023, Jun. 2013.
[59] R. H. de Souza, P. A. Flores, M. A. R. Dantas, and F. Siqueira, "Architectural recovering model for distributed databases: A reliability, availability and serviceability approach," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jun. 2016, pp. 575–580.
[60] T. Guérout, S. Medjiah, G. Da Costa, and T. Monteil, "Quality of service modeling for green scheduling in clouds," *Sustain. Comput., Informat. Syst.*, vol. 4, no. 4, pp. 225–240, Dec. 2014.
[61] Y. Hu, B. Deng, Y. Yang, and D. Wang, "Elasticity evaluation of IaaS cloud based on mixed workloads," in *Proc. 15th Int. Symp. Parallel Distrib. Comput. (ISPDC)*, Jul. 2016, pp. 157–164.
[62] W. Dou, X. Xu, S. Meng, and S. Yu, "An energy-aware QoS enhanced method for service computing across clouds and data centers," in *Proc. 3rd Int. Conf. Adv. Cloud Big Data*, Oct. 2015, pp. 80–87.
[63] N. R. Herbst, S. Kounev, and R. Reussner, "Elasticity in cloud computing: What it is, and what it is not," in *Proc. 10th Int. Conf. Autonomic Comput.*, 2013, pp. 23–27.
[64] F. Liu, F. Tong, J. Mao, R. Bohn, J. Messina, L. Badger, and D. Leaf, "NIST SP 500-292 cloud computing reference architecture," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep., 2012, p. 81.
[65] X. Liu, C. Xia, T. Wang, and L. Zhong, "CloudSec: A novel approach to verifying security conformance at the bottom of the cloud," in *Proc. IEEE Int. Congr. Big Data (BigData Congr.)*, Jun. 2017, pp. 569–576.
[66] Y.-K. Lin and P.-C. Chang, "Maintenance reliability estimation for a cloud computing network with nodes failure," *Expert Syst. Appl.*, vol. 38, pp. 14185–14189, May 2011.
[67] *Interoperability and Portability for Cloud Computing: A Guide, Version 2.0*, Cloud Standards Customer Council, Needham, MA, USA, 2017.
[68] B. Stanton, M. Theofanos, and K. P. Joshi, "Framework for cloud usability," in *Proc. 3rd Int. Conf. Hum. Aspects Inf. Secur., Privacy, Trust*, vol. 9190. Springer, Aug. 2015, pp. 664–671.
[69] *Information Technology—Cloud Computing-Interoperability and Portability*, Standard ISO/IEC 19941, 2017.
[70] E. Lim and P. Thiran, "Communication of technical QoS among cloud brokers," in *Proc. IEEE Int. Conf. Cloud Eng.*, Mar. 2014, pp. 403–409.
[71] S. Rizvi, J. Ryoo, J. Kissell, and B. Aiken, "A stakeholder-oriented assessment index for cloud security auditing," in *Proc. 9th Int. Conf. Ubiquitous Inf. Manage. Commun. IMCOM*, 2015, pp. 1–7.
[72] A. Abdeladim, S. Baina, and K. Baina, "Elasticity and scalability centric quality model for the cloud," in *Proc. 3rd IEEE Int. Colloq. Inf. Sci. Technol. (CIST)*, Oct. 2014, pp. 135–140.
[73] *NIST SP 500-291 Cloud Computing Standards Roadmap*, National Institute of Standards and Technology, Gaithersburg, MD, USA, 2011.
[74] E. Saiz, E. Ibarrola, L. Cristobo, and I. Taboada, "A cloud platform for QoE evaluation: QoXcloud," in *Proc. ITU Kaleidoscope Academic Conf., Living Converged World Impossible Without Standards*, Jun. 2014, pp. 241–247.
[75] S. Wang and S. Dey, "Cloud mobile gaming: Modeling and measuring user experience in mobile wireless networks," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 16, no. 1, pp. 10–21, Jul. 2012.
[76] Z.-Y. Wen and H.-F. Hsiao, "QoE-driven performance analysis of cloud gaming services," in *Proc. IEEE 16th Int. Workshop Multimedia Signal Process. (MMSP)*, Sep. 2014, pp. 1–6.
[77] *IEEE Standard for a Software Quality Metrics Methodology*, Standard 1061 TM-1998, R2009, 2009.
[78] R. Mahmud, S. N. Srirama, K. Ramamohanarao, and R. Buyya, "Quality of experience (QoE)-aware placement of applications in fog computing environments," *J. Parallel Distrib. Comput.*, vol. 132, pp. 190–203, Oct. 2019.

XIMENA GUERRON received the M.Sc. degree in enterprise IT management from the Universidad Central del Ecuador (UCE), Ecuador, in 2006. She is currently pursuing the Ph.D. degree with the Universitat Politècnica de València (UPV). She is also an Assistant Professor with UCE, and the IT Expert with the Banco Central del Ecuador. She is also a member of the Instituto Universitario Mixto de Tecnología Informática, UPV. Her current research interests include assessing the quality of cloud services using model-driven engineering and DevOps techniques.

SILVIA ABRAHÃO (Member, IEEE) received the Ph.D. degree in computer science from the Universitat Politècnica de València (UPV), Spain, in 2004. She was a Visiting Professor with the Carnegie Mellon Software Engineering Institute, from 2010 to 2012, the Universitè catholique de Louvain, from 2007 to 2017, and Ghent University, in 2004. She is currently an Associate Professor with UPV. She has (co)authored over 150 peer-reviewed publications. She also leads the Spanish Network of Excellence on Software Quality and Sustainability. Her main research interests include quality assurance in model-driven engineering, the empirical assessment of software modeling approaches, the integration of usability into software development, and cloud services monitoring and adaptation. She is a member of the editorial board of *Software and System Modeling* (SoSyM) journal. She is an Associate Editor of the IEEE SOFTWARE, where she is responsible for Software Quality.

EMILIO INSFRAN (Member, IEEE) received the M.Sc. degree in computer science from Cantabria University, Spain, in 1994, and the Ph.D. degree from the Universitat Politècnica de València (UPV), Spain, in 2003. He worked as a Visiting Researcher with the Université catholique de Louvain, Belgium, in 2017, and the Software Engineering Institute (SEI-CMU), USA, in 2012, and also performed research stays at the University of Twente, The Netherlands, Brigham Young University, Utah, USA, and the University of Porto Alegre, Brazil. He is currently an Associate Professor with the Department of Information Systems and Computation (DISC), UPV. He has had more than 140 journal and conference papers published and has worked on a number of national and international research projects and on several technology, transfer projects with companies. His research interests are cloud service architectures, DevOps, model-driven development, requirements engineering, and software quality.

MARTA FERNÁNDEZ-DIEGO received the European Ph.D. degree in electronics and telecommunications engineering from the Lille University of Science and Technology, France, in 2001. She was, for several years, a member of a software development team for mobile phone applications at an international information technology service company. She is currently an Associate Professor with the Department of Business Organisation, Universitat Politècnica de València, Spain, where she teaches at the School of Computer Science. Her research interests include empirical software engineering, software effort estimation, and project risk management.

FERNANDO GONZÁLEZ-LADRÓN-DE-GUEVARA received the Ph.D. degree in industrial engineering, in 2001. He has worked at several universities and IT companies throughout Europe and Latin America. He is currently an Associate Professor with the Telecommunications Engineering School, Universitat Politècnica de València (UPV). He coauthored several articles published in well-known international journals. He regularly participates in the organising committees of several national and international conferences. He has participated in 27 research projects and contracts with different organisations, and was responsible for seven of them. His research interests include crowdsourcing, ERP systems, and software engineering.