Chapter 3
Quality of Service and Service Level Agreements for Cloud Environments: Issues and Challenges

Inderveer Chana and Sukhpal Singh

Abstract The increasing use of Cloud computing makes the development of high-quality Cloud-based applications a vital research area. Cloud computing, which provides inexpensive computing resources on the pay-as-you-go basis, is promptly gaining momentum as a substitute for traditional information technology (IT)-based organizations. As more and more users migrate their applications to Cloud environments, service level agreements (SLAs) between clients and Cloud providers become a key element to consider. Due to the dynamic nature of the Cloud, endless supervision of quality of service (QoS) attributes is necessary to honor the SLAs. Thus, Cloud computing faces the challenge of QoS, especially in relation to how a service provider can ensure appropriate QoS for its Cloud services. QoS is an inherent element, part of service-oriented architecture (SOA), to direct nonfunctional quality attributes of a service, such as the response time, price, or the supported security rules. Consequently, there is a requirement to grow architectures in order to respond correctly to the QoS requirements. The architecture should be able to change dynamically the amount of resources made available to the applications it hosts. Optimal resource utilization should be attained by providing (and maintaining at run time) each hosted application with the number of resources which is adequate to guarantee that the application SLA will not be violated. This chapter reflects the essential perceptions behind the QoS provision in the Cloud, identifies current and innovative quality attributes based on customers' desires associated with SLA and identifies metrics to measure the deviation of QoS from predictables, with possible resolution in the outline of architecture for spontaneous supervision of QoS without violation of SLA. The existing intent of Cloud SLAs is inspected with a focus on QoS and customer requirements. Further, foremost research problems and scientific challenges in Cloud SLAs have been considered with possible reasons. Autonomic management architecture for dynamic provisioning of resources

S. Singh (✉) · I. Chana
Computer Science and Engineering Department, Thapar University,
Patiala, Punjab 147004, India
e-mail: ssgill@thapar.edu

I. Chana
e-mail: inderveer@thapar.edu

Z. Mahmood (ed.), *Cloud Computing,* Computer Communications and Networks,
DOI 10.1007/978-3-319-10530-7_3

based on users QoS requirements to maximize efficiency and automatic fulfillment of SLA has also been proposed.

Keywords Cloud computing · Service level agreement (SLA) · Service-oriented architecture · SOA · Quality of service · QoS · Autonomic Cloud computing · SLA challenges

3.1 Introduction

Cloud computing is a computing model for permitting omnipresent, suitable and on-demand service access to a common group of configurable computing resources (e.g., networks, servers, storage, and applications) that can be quickly provided and released with minimum management struggle [21]. Public Cloud platforms are usually superior at providing IT services over the open Internet than the on-premise enterprise IT resources. Therefore, the public Cloud can well serve as a workforce that is expected to work at the local region because processing, storage, and enterprise applications to a middle tier between the company and the Cloud consumer can be done easily [31]. The services provided by a Cloud are shown in Fig. 3.1. As a Cloud offers three types of services such as infrastructure as a service (IaaS), or platform as a service (PaaS), or software as a service (SaaS), it requires quality of service (QoS) to efficiently monitor and measure the delivered services and thus needs to follow service level agreements (SLAs) [1, 11]. The complex nature of the Cloud environment requires a cultured means of handling of SLAs as the demands of the service users vary considerably. The QoS attributes that are frequently part of an SLA (response time, throughput, etc.) vary repeatedly and to implement the contract, these parameters need to be carefully controlled [1, 5].

An SLA is part of a service contract where a service is defined based on the agreement between a provider and a customer [19]. In other words, the term SLA denotes the contracted service and its performance. An SLA is a document that specifies the description of the service level parameter, service level objective, agreed service,

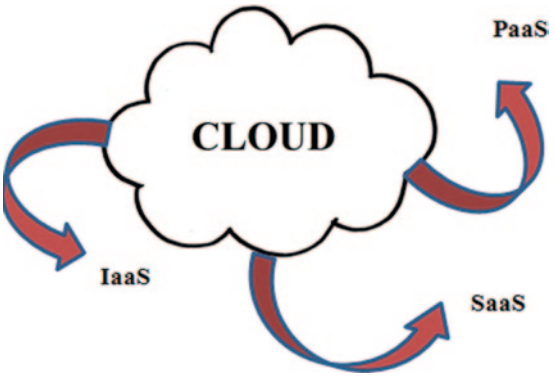

Fig. 3.1 Cloud computing services. *IaaS* infrastructure as a service, *PaaS* platform as a service, *SaaS* software as a service

warranties, and action in case of violation. An SLA is a conveyed bargain that has been documented between two parties which are customer and service provider [2]. The SLA is very significant to define the availability, reliability, and scalability of services. In the literature, the following definitions of SLA are prevalent:

- "SLA is an officially exchanged document that describes (or tries to express) in measurable (and maybe qualitative) terms the service being presented to a customer. Any metrics involved in a SLA should be capable of being controlled on a systematic basis and the SLA should record by whom" [4].
- "A contract is an officially binding bargain between two or more parties. Contracts are subject to particular authorized explanations" [9].

Although, Cloud consumers do not have full supervisory control over the fundamental computing resources, they do require ensuring attributes such as quality, accessibility, trustworthiness, and performance of these resources when users have transferred their fundamental business functions onto their honored Cloud. In other words, it is vital for users to acquire assurances from suppliers on service provisions [18]. Usually, these are delivered through SLAs discussed between the providers and customers [30]. The very first problem is the description of SLA terms in such a way that has a suitable level of granularity, namely the compromises between accuracy and complexity, so that they can ensure most of the user hopes and is comparatively simple to be prejudiced, certified, calculated, and imposed by the resource provisioning mechanism on the Cloud [3, 25]. In addition, different Cloud service models (IaaS, PaaS, and SaaS) will need to express different SLA meta disclaimers [13]. This also increases a number of implementation issues for the Cloud providers. Moreover, innovative SLA mechanisms require to continuously integrate consumer response and customization features into the SLA assessment framework [8].

As the Cloud service models develop and become omnipresent, there is an increase in the probability of clarifying the way the services are provisioned and managed. It, therefore, permits the providers to address the different requirements of their customers. In this perspective, SLAs appear as a significant characteristic which subsequently serve as the establishment for the predictable quality level of the services made available to customers by the providers [38]. Nonetheless, the collection of the recommended SLAs by providers (with marginal overlaps), has directed to manifold different definitions of Cloud SLAs [6]. Moreover, confusions exist on what is (if there is) the difference between SLAs and agreement, what is the marginal quality, what are the terms involved in each one of these documents, and if and how are these associated.

SLAs are a corporate way to officially specify the particular circumstances (both functional and non-functional) under which services are or should be provided. Customers and providers can use top-level SLAs to monitor whether their actual service delivery conforms to the contracted SLA terms [34]. In the case of SLA violations, top-level SLAs permit for penalties or compensations to be paid [16]. In a service-oriented world, services presented are generally self-possessed of or built on a complete set of other services [24]. These services may reside in the domain of the provider itself, or be hosted by external providers. Such services contain

business services, software services, and infrastructure services. The quality of a presented service depends comprehensively on the quality of the services it uses [39]. Service quality also depends on the components used and the structure of the basic IT system appreciating the service. Presently, service providers cannot design their service landscapes using the SLAs of dependent services [4, 28]. They have no means by which to control, why a certain SLA violation might have happened, or how to express an associated penalty. SLA guarantee terms are not unambiguously associated to quantifiable metrics, nor are their relation to lower-level services well defined. As a consequence, service providers cannot define the mandatory supervision required in confirming top-level SLAs. This missing relationship between top-level SLAs and (lower-level) metrics is a main obstacle to effective service planning and expectation or improvement processes in service stacks [15, 36].

Further, Cloud computing allows for organizations to move applications and data to remote servers. Due to virtual computing, Cloud computing can deliver better approach to consumption of available resources. Hosted solutions and on-demand server resources are two cases where the use of external vendors may provide for a lower overall price of computing. As the data is moved to remote resources, the control or governance of the data becomes difficult [29].

In this chapter, we first present the concept of SLA in the context of Cloud computing. The remainder of this chapter is then organized as follows: Sect. 3.2 describes interweaving of QoS and SLA with respect to the Cloud; Sect. 3.3 presents the SLA challenges and benefits with respect to Cloud environments; Sect. 3.4 introduces the Cloud SLA (CSLA) architecture; and Sect. 3.5 presents the discussion of work done. Section 3.6 describes our conclusions and future research directions.

3.2 QoS and SLA: Intertwined in the Cloud

This section presents the background of QoS and SLA, SLA Management, SLA of Cloud provider, SLA levels, Metrics in SLA, and SLA deviation in the area of Cloud computing.

3.2.1 QoS and SLA

QoS is increasingly significant when composing services because a degrading QoS in one of the services can dangerously disturb the QoS of the complete composition. Cloud service providers want to confirm that sufficient amount of resources are provisioned to ensure that QoS requirements of Cloud service consumers such as deadline, response time, and budget constraints are met [36]. Consequently, Cloud service providers want to confirm that these violations are avoided or reduced by dynamically provisioning the exact amount of resources in a timely fashion. The success of next-generation Cloud computing infrastructures will depend on how capably these infrastructures will discover and dynamically tolerate computing

platforms, which meet randomly varying resource and service requirements of Cloud costumer applications [29]. Logically, based on QoS requirements such as scalability, high availability, trust, and security, these applications will be characterized, identified in the so called SLAs. The current Cloud technology is not completely personalized to honor probable SLAs, though industrial and the academic, both the research groups are presenting increasing interest on problems of QoS assurance within the context of Cloud computing. Broadly, an SLA needs a precise assessment of the characteristics of the required resources [19]. Application services introduced in Clouds (e.g., Web applications, Web services) are frequently characterized by great load inconsistency; therefore, the amount of resources required to honor their SLAs may vary particularly over time [8]. An important challenge for Cloud providers is to automate the management of virtual servers while keeping into account both high-level QoS requirements of hosted applications and resource supervision expenses. Cloud market mechanisms are consistently static and cannot react on dynamic variation of consumer desires [26]. To respond to these issues, there is a requirement of an adaptive methodology for autonomically springing SLA patterns based on consumer requirements. The present research in Cloud SLA limits the capability of matching conformation metrics to acceptable benchmarks [1]. These metrics comprise statistical measures such as standard deviation that want to be computed from the expected and actual outcomes of services delivered to customer. Semantic Web technologies can be used to improve the descriptions and therefore increase the quality of these matches.

3.2.2 Cloud and SLA

Resource reservation is one of the main characteristics in parallel and distributed environment like the Cloud. While preserving the services in the Cloud, we require initiating SLAs through settlement. The settlement between consumers and Cloud service providers fundamentally comprise of parameters like price, time, and other QoS parameters. There are presently numerous methods which resolve the issue of expense and time slot settlement mechanism without taking into account the significant characteristics of QoS [23]. Knowingly handling and assigning resources among numerous consumers in a commercial manner is significant for service providers [41]. Thus, SLA shows a chief role in resource provisioning. In practice, the term SLA is occasionally used to mention the limited delivery time (of the service) or performance.

The Cloud is a parallel and distributed system containing a huge collection of interrelated and virtualized resources that are dynamically self-provisioned and offered as one or more merged computing resources based on SLAs [19]. During negotiation/agreement, there are parameters considered like price, time, and other QoS. Since there is an opposing relationship between price and time-slot feasibilities (e.g., a customer desires to pay a higher price to use a service at a more expected time slot—attaining a higher time-slot utility), expense and time slot have to be exchanged suddenly [25].

Another parameter taken into account is about expanding the QoS through supervising the Cloud services by the use of SLA-based Cloud architecture [13, 36]. Cloud supervising environment comprises of measuring the properties of the network to guarantee that the system functions with required parameters. The management station inquires the state of the network in order to respond to alarm circumstances that may develop in the network system parameter, which is defined as a conjunctive predicate on the local properties of different network elements. In such cases, after identifying local variations, each network element has to successively originate alarms in order to ensure that global parameters are not violated. Even though data may be hosted remotely, it is still an organization's accountability to offer for its security. The problem for the organization is to ponder on what mechanisms it has to provide for the safety of data which it may no longer directly control.

3.2.3 SLA Management

SLA management is the element that retains track of SLAs of consumers with Cloud providers and their satisfaction history. Based on SLA terms, the security mechanism preserves the real usage of resources by needs so that the absolute price can be calculated and charged from the consumers [8]. In addition, the preserved past-usage statistics can be utilized by the service request assessor and admission governor mechanism to expand resource distribution assessments.

An SLA is a document that describes the relationship between two parties: the provider and the consumer. This is obviously a very significant item of documentation for both parties. If used appropriately it should: recognize and describe the consumer's requirements, make all the difficult concerns simpler, decrease areas of clash, inspire dialog in the event of disagreements, and eliminate impossible viewpoints [3, 34]. It should resolve an extensive collection of disputes clearly and unambiguously. Amongst these, the following are some of the most frequent services to provide performance, tracking and reporting problem management, legitimate agreement and resolution of disagreements, consumer responsibilities and accountabilities, reservation and trustworthy information termination. Typical SLA substances [3, 4, 15, 16, 19, 24, 25] to be considered are:

1. *Description of services*: This is the most serious section of the contract as it designates the services and the way in which those services are to be provided. Standard services are frequently separated from adapted services but this disagreement is not of serious concern. The information on the services must be correct and comprised through requirements of what is being delivered.
2. *Performance supervision*: An important part of a SLA deals with supervising and evaluating service level performance. Fundamentally, every service must be capable of being measured and the outcomes inspected and informed. The standards, objectives, and metrics to utilize must be quantified in the contract. The two parties must examine the service performance level consistently.

3. *Problem administration*: The determination of problem administration is to reduce the violent influence of occurrences and difficulties. This regularly specifies that there must be a suitable process to control and solve unexpected occurrences and that there must also be preemptive action to reduce happening of unexpected happenings.
4. *Consumer responsibilities and accountabilities*: It is significant for the consumer to understand that it also has accountabilities to sustain the service delivery process. The SLA describes the association, which of course is a two-way unit. Typically, the consumer must organize for entrance, accommodations, and resources for the provider's workforces who require working on-site.
5. *Licenses and cures*: This section of the SLA stereotypically covers the following vital issues: service quality protections, third party claims, and cures for loopholes.
6. *Reservation*: Reservation is mainly a serious feature of any SLA. The consumer must deliver well-ordered physical and logical entrance to its principles and information. Correspondingly, the contractor must respect and obey with the consumer's reservation rules and techniques.
7. *Catastrophe recovery and commercial strength*: It can be of dangerous status. This factor should be conveyed within the SLA. The topic is catastrophe recovery frequently incorporated within the reservation section; though, it is also regularly involved within the problem administration area. At the highest level, both these areas typically state that there must be acceptable provision for catastrophe recovery and commercial strength forecasting to protect the continuity of the services being distributed.
8. *Service termination*: The SLA agreement naturally covers the following fundamental areas: services are finished at completion of preliminary term, finish for suitability, finish for reason, and expenditures on closure.

3.2.4 SLA of a Cloud Provider

Quality attributes play a significant role in SOA environments [23]. An SLA formally describes the level of service. Organizations seek to develop SLAs for numerous causes. From a simple viewpoint, an SLA is developed between two parties to spell out who are responsible for what, what each party will do, and occasionally more clearly what each party will not do [38]. Also an SLA describes the interaction between a service provider and a service consumer. An SLA contains several elements of details [6, 18, 30], viz.:

1. The set of services the provider will offer.
2. A comprehensive, full definition of each service.
3. The responsibilities of the provider and the consumer.
4. A set of metrics to define whether the provider is providing the service as guaranteed.
5. The inspecting mechanism to supervise the service.

6. The courses of action available to the consumer and provider if the terms of the SLA are not fulfilled.
7. How will the SLA vary with respect to time?

A typical SLA of a Cloud provider has the following components [8, 12–14, 17, 20, 28, 29, 32, 35, 36]:

1. *Service assurance*: It specifies the metrics which a provider struggles to meet over a service agreement time period. Failure to attain those metrics will outcome in service recognition to the consumer. Availability (e.g., 99.9%), response time (e.g., less than 50 ms), catastrophe recovery, and fault perseverance time (e.g., within one hour of discovery) are examples of service assurances. Some service assurances can be on a per action basis, such as zeroing out a VM disk when it is deprovisioned.
2. *Service Assurance Time Period*: It describes the duration over which a service guarantee should be happened. The time period can be a billing month or time occurred since the previous advantage was filed. The time period can also be insignificant, e.g., one hour. The smaller the time period, the more difficult is the service assurance.
3. *Service assurance granularity*: It defines the resource scale on which a provider specifies a service guarantee. For example, the granularity can be as per service, per data center, per instance, or per transaction basis. Related to time period, the service assurance can be inflexible if the granularity of service assurance is fine-grained. Service assurance granularity can also be designed as a cumulative of the deliberated resources, such as contacts. For example, aggregate uptime of all running instances must be greater than 99.95%. Though, such an assurance denotes that some instances in the collective SLA computation can hypothetically have a lesser percentage uptime than 99.95% while still meeting the collective SLA. As significant, collective SLA computation leaves provider the room to better accomplish its presented services.
4. *Service guarantee*: Omissions are the instances that are excluded from service guarantee metric calculations. These omissions typically include misuse of the system by a customer, or any downtime associated with the scheduled maintenance.
5. *Service recognition*: It is the amount credited to the consumer or applied towards upcoming expenditures if the service assurance is not met. The amount can be a comprehensive or restricted recognition of the consumer compensation for the miscalculated service.
6. *Service Violation Measurement and Reporting*: It describes how and who measures and reports the violation of service assurance, respectively.

3.2.5 SLA Levels

Cloud SLAs may provide safety at different stages through infrastructure operating systems (OSs) and applications [8, 38]. Some of the significant attention levels that could be included in a Cloud SLA are described in Table 3.1.

Table 3.1 Cloud SLA levels

SLA levels	Description
Facilities level SLA	Here, the Cloud provider will normally deliver an SLA including the data center services necessary to maintain the customer-owned infrastructure. These comprise items such as electric power, on-site generators, cooling, etc
Platform level SLA	The next level of safety in a Cloud usually covers physical servers, virtualization platforms and hardware related to network retained by the provider and used by the Cloud consumer. Usually, the physical server and virtualization software are hidden by a platform SLA
OS level SLA	OS is the subsequent possible area of coverage for a Cloud SLA. Providers proposing an OS level SLA normally deliver some amount of managed services to a client. This extra service permits the provider to guarantee that the OS is suitably sustained so that it is dependably accessible and normally has some warnings
Application level SLA	This category of SLA delivers safety against application level catastrophes up to and comprising the custom application executing on the infrastructure provided by SLA. Under this model, the Cloud provider is ensuring the availability and performance of their Cloud customer software, which is a hard guarantee to encounter
Availability level SLA	The Cloud network (network among Cloud servers) may be covered by a distinct availability level SLA

3.2.6 Metrics in SLA

Realization of Cloud computing requires that both consumers and suppliers can be confident that contracted SLA are supporting their corresponding business accomplishments to their best degree [19]. Current SLAs usually fail in providing such confidence, exclusively when Cloud providers outsource resources to other Cloud providers. These Cloud providers typically provision very modest metrics, or metrics that hinder an efficient misuse of their Cloud resources [2]. We have identified some of the service-level metrics for specifying fine-grain guarantees of QoS. These metrics sanction resource providers to assign dynamically their resources among the executing Cloud services depending on their request. This is accomplished by including the consumer's service usage in the metric description, but avoiding false SLA violations when the consumer's application does not use all its assigned resources [13, 20, 25].

Through metrics, the defects can be easily identified. Assigning a severity type to defects helps prioritize the development of Cloud services [17, 25]. Table 3.2 demonstrates each type of defect associated with it, as well as SLA that describes the time within which Cloud provider promises to fix the defect measured by metrics.

Normally, a Cloud provider approves the QoS with its consumers through a SLA, which is a two-sided agreement between the consumer and the supplier that states not only the circumstances of a Cloud service, but also describes the contracted QoS between them using a set of metrics. Cloud service providers certainly offer service-level metrics (service accomplishment deadline) to their consumers

Table 3.2 Defect types and SLAs

Defect type	Metric description	SLA
Type 1	Business critical features absent or do not function; program may crash	Fix within 4–24 h
Type 2	Business critical features function most of the time. No work around exists	Fix within 1 week
Type 3	Noncritical features absent or do not function; work around exists	Fix within 2 weeks
Type 4	Inconsequential function may not work as expected, typos in documents, etc	Fix for next software release

for specifying the QoS. The Cloud providers must offer service level metrics that can be used to deliver fine-grain QoS assurances. First, the QoS contract can be obviously expressed using general metrics (e.g., number of processors, frequency of processors, etc.), meanwhile underdone resources are the functioned good. Second, having fine-grain metrics, which assures a given resource distribution during a time period, is particularly significant for service providers that outsource resources to Cloud providers, as we have specified before.

3.2.7 SLA Deviation

Customers desire that composed data should be put into expressive perspective. This situation produces the restriction for a procedure which gathers data from different sources and implements appropriate algorithms for controlling expressive consequences. Such metrics comprise statistical measures such as average or standard deviation that want to be computed from the expected and actual outcomes of services delivered to customer [16]. With the rise of the number of Virtual Machines (VMs), the standard deviation of the customer load falls. Due to this unpredictability, the standard deviations of resource utilization and performance are difficult to measure.

At the application's SLA Level, along with the benchmarks, QoS metrics to estimate the performance and SLA deviation are also required [12, 17, 25, 35]. This is appreciated through a distributed supervising framework that is able to combine supervising information coming from several sources and at different stages. For this trend, the assessment method of the platform is capable to evaluate on the cause of the application's performance deviation, i.e., whether it establishes a breach of the application usage terms and if so, whether the application SLA specifies activities to be executed, whether it is an adequate deviation that can be accurately controlled or a real breach of the SLAs. In the previous situation, more evaluation is required in order to accomplish on the particular nature of the SLA breach to recognize the real object or objects that failed to deliver the granted QoS level [36]. An SLA is typically a two-way written contract which outlines the service and principles the providers deliver to their consumers whether these are scholars, supervisor in universities, and/or other central management teams. It also describes what the providers require

from their consumers/service customers in order to provide the service specified. It needs assurance and support from both parties to provision and follow the contract in order for the SLA to work efficiently [6]. In SLA, both the parties (Cloud provider and Cloud consumer) should have specified the possible deviations to achieve appropriate quality attributes. If taking availability as a quality attribute and if it should be 95%, then it means that the system should be available for 22.8 h per day with maximum deviation of 1.2 h per day (5%). In the case of system performance, if the desired deadline is 9 ms with deviation (10%) of 1 ms, then maximum response time should be 10 ms for a particular task without violation of agreement. The Cloud provider's SLA will give an indication of how much actual availability of service the provider views as adequate, and to what amount it is agreeable to require its own financial resources to compensate for unexpected outages. Usually, no Cloud provider considers compensation because 85% resource providers do not actually provide penalty enforcement for SLA violation presently [10]. There should be penalty delay cost or consumers' compensation if the Cloud provider misses the deadline. Moreover, it provides a risk transfer for IaaS providers, when the terms are violated by the Cloud provider. Penalty delay cost is equivalent to how much the service provider has to give concession to users for SLA violation. It is dependent on the penalty rate and penalty delay time period. The effect of inaccuracy could be reduced by two approaches: first, considering the penalty compensation clause in SLAs with IaaS provider and impose SLA violation; second, adding some slack time during scheduling for avoiding risk [27].

3.2.8 Existing SLA Architectures in the Cloud

Not much has been written in the area of Cloud SLA. We have surveyed only three related architectures in this context. Casalicchio et al. [7] presented an architectural model for the autonomic service provisioning system that investigated the problem from the outlook of an application service provider that uses a Cloud infrastructure to attain scalable provisioning of its Cloud services in the respect of QoS restrictions for autonomic resource management of Cloud-based systems. This architecture describes the functional desires of an autonomic service provisioning system and recognized features and services presented by many IaaS providers that might be used to implement such desires [7].

Happe et al. [33] have proposed a reference architecture for multi-level SLA management that provisions the inclusive supervision of possibly difficult service stacks and discussed how SLAs are used for handling the nonfunctional features of the complete Cloud service life cycle. The presented architecture is based on capabilities extended from an SLA framework constructed around a particular reference application. Emeakaroha et al. [14] have presented DeSVi—an architecture for observing and identifying SLA destructions in Cloud computing infrastructures. This architecture is accountable for the provision of resources and for mapping of tasks, accountable for the implementation of consumer applications, and visualizes the execution of the applications and converts low-level metrics into high-level SLAs.

It is used to recognize the intervals for applications with stable resource consumption only.

However, all these architectures do not take into account the dependency of SLA on QoS requirements? Therefore a new architecture is required that considers SLA deviation status, heterogeneous Cloud workloads and their resource consumption dynamically, assigns priority to Cloud workloads and different states of Cloud workloads and also assures the relation between QoS and SLA.

3.3 SLA Challenges and Benefits in Cloud

This section describes the SLA key challenges along with the reasons of their occurrences as well as benefits and potential barriers/issues of SLA in Cloud computing [11, 18, 21, 31].

3.3.1 SLA Challenges

1. SLAs are hard to express in the Cloud in part because areas of the infrastructure (in specific the network) are outside of the scope of either consumer or provider. This hints to the challenge of offering a predetermined contract for something which is only comparatively in the provider's control [36]. Additionally, as the infrastructure is shared (multi-tenanted) SLA's are more challenging to deliver since they rest on capacity which must be shared [22].
2. The consumer accessing services in the Cloud also face a challenge. New Cloud SaaS providers, who are growing their business and attracting more consumers to their multi-tenanted data center, are unlikely to offer serviceably defined SLA for their services as compared to a data-center provider who can bargain where it supervises all fundamentals of the supplied infrastructure [1]. As their business is increasing and an SLA is a massive threat (since it is a multi-tenanted break of one SLA and is possibly a break of lots), the expenditure might look insignificant and unfortunate to the consumer but is great for a SaaS provider). Additionally with each new consumer, the difficulties on the data center, and therefore danger, increase [12].

Every new consumer brings the advantage of growing stress testing of the SaaS platform and improving growth of abilities within the SaaS provider. While the SLA may remain to be neglected, the risk of dissatisfaction of the data center may well reduce as the SaaS transmits [35]. The objective of an SLA is accordingly not just to deliver a predetermined contract but rather to set out the level of service on which the cooperation between customer and supplier is constructed. In this way, an SLA is about the predictable quality demanded of the supplier and with the above model the expected quality may well improve with more consumers—not reduction as is frequently predicted for a Cloud [17]. SLA's for Cloud providers may well be

insignificant and neglected, but the universal risk of using Clouds is not as simple as is often competed. Whereas it is probable that Cloud providers' compromise run-down SLA's, it does not mean that the QoS is, or will stay, underprivileged.

The integration of QoS aware aspects in each Cloud component in order to control and inform the system about its current behavior is required. Further, the optimization of energy consumption in the Cloud computing environment according to user-specified budget constraint is necessary. Thus, maximizing energy efficiency, cost effectiveness, and utilization for applications while ensuring performance and other QoS guarantees, requires controlling important and extremely challenging tradeoffs. These challenges and issues occur due to the following important factors related to the Cloud:

- SLA deviation occurs due to shared nature of the Cloud, and it leads to SLA violations.
- Service quality fluctuations occur due to fluctuations in QoS requirements of different Cloud users.
- Problems in invoices occur due to the various modes of payments along with their own constraints.
- Risk of SLA violations due to urgent execution of Cloud workloads (while assigning priorities to the most urgent workloads), whether the Cloud providers provide the compensation to the user in case of SLA violations or not.
- Difficulty in maintaining the security, due to the multi-tenanted data center, access to the database and type of encryption and decryption.
- Efficient storage is required as memory is wasted due to multiple copies of same data by different or same Cloud users.
- VM migration demands high bandwidth which further leads to complexity.
- Lack of standard QoS-oriented SLA architecture in the Cloud due to heterogeneous nature of Cloud workloads.

The required architecture will focus on developing a resource provisioning and scheduling technique that will automatically manage QoS requirement of Cloud users and would be based on energy efficient usage of the Cloud infrastructure. So, what the customer should deliberate in considering the SLA, in terms of service quality [22, 36, 37], are:

- How does the Cloud SaaS provider determine its progress? The progress of a SaaS service means larger demand on the supplier's data center. Therefore, greater risk that the SLA's will be broken for their multi-tenanted data center.
- How vulnerable is the Cloud SaaS provider in permitting analysis of its services by fresh consumers?
- How well the Cloud SaaS provider engages in planned motivation for service quality alignment with your requirements for service quality?

To address these challenges, SLA can respond to the following issues and questions [2, 3, 6, 8, 9, 13, 16, 19, 25, 38]:

- What are the resources delivered to the consumer? How resources will support the consumer? Are there any limitations to the number of resources?

- How the invoices are created? What are the payment methods? How the services are affected if the customer postpones in compensating invoices? This should comprise refinement period and how the consumer can acquire the services back after the payment when the services are blocked?
- What happens if the SLA is not met? How data is controlled when the service agreement finishes, the sort of data compensated to the company?
- What happens if the service contract is withdrawn? How data is handled and returned to the company?
- How does the service use event logs and who actually has access to the data on the backend?
- Who will check the security of Cloud providers?
- Which of the SaaS employees has root and database access, and will anything prevent them from getting access to your corporate data? What controls are in place?
- Is the held data separated between clients or is it all stored on one huge database out there? How is this data separated? How will the legal question of e-discovery be addressed should it arise as a business concern?
- In terms of service availability, can you get your vendor to sign a service level agreement?
- What security arrangements do you have in place with Cloud service providers that you rely on to deliver your service? What are you doing to build "trust in depth" in the Cloud?

Many significant issues in Cloud computing occur at the boundary between the provider's infrastructure and the Cloud environment [4, 15, 24, 34], e.g.:

- How do you move resources from one side to the other? Is the Cloud application dependent on storage that exists on your side of the boundary?
- What influence will that have on the bandwidth desires? And, how do you perfectly move VMs between the Cloud and your data center as demand raises and failures occur?

These are all legal and motivating problems. But an even larger question forthcoming like a dark Cloud on the perspective is that of the right and authorized grade [8]; i.e., is the matter in the Cloud on the same legitimate footing as the matter in the data center? For example:

- How will the switch occur to a public Cloud when the private Cloud infrastructure gets mixed out? Or would you be using the public Cloud for just executing your services?
- How much confident can be placed on the encryption patterns?
- How safe is the data from natural disasters?
- Is it probable for all of the data to be fully encoded?
- What algorithms are used? Who holds, maintains, and issues the keys?
- And so on.

Thus, it can be construed that SLAs are elements of a quality methodology to help the support teams in classifying and agreeing on what 'good quality' looks like and

deliver a framework for quantifying and supervising the realization of service quality [9, 17].

3.3.2 Prospective Benefits

QoS and appropriate SLA collectively offer huge benefits to Cloud computing paradigm. A few of such benefits are listed below:

- Enables strong understanding of the service and accountabilities of all parties
- Helps you to achieve your service consumers viewpoints
- Encourages clearness, responsibility, and reliability
- Notifies team performance, capabilities, and staffing judgments
- Provisions supportive and collective functioning
- Emphases teams on uninterrupted enhancement

3.3.3 Potential Barriers/Issues of SLAs

Following are some of the potential barriers that hinder the implementation of QoS through SLAs:

- Adequate resources not being available at the desired time.
- Lack of assurance from management to implement the solutions within granted schedule.
- Unavailability of desired staff and momentum, in case of urgency.
- SLA's excessive optimization may become difficult and even may lead to rejection.
- The development of SLAs should be team's strength, and if recommendations made within the team are not appreciated, then it may be difficult to preserve staff commitment in the process.

These barriers can be overcome by deliberating the SLAs as follows: Adjust the work roles and responsibilities to reproduce the necessities of the new structure. Note that stronger work roles and responsibilities can help on specific basis but not in terms of the general service nor will this methodology enable endless improvement, added value, and simplicity of service delivery [3, 18]. Observations and prospects of central services will unavoidably adjust as consumers will search for reasonable service delivery and proof of price/profit/worth of services they use [20].

3.4 The Proposed Cloud SLA Architecture

This section proposes Cloud SLA (CSLA) architecture that can ensure better SLAs for both Cloud provider and consumers, as shown in Fig. 3.2. The objective of the proposed CSLA architecture is to reduce the standard deviation of resource

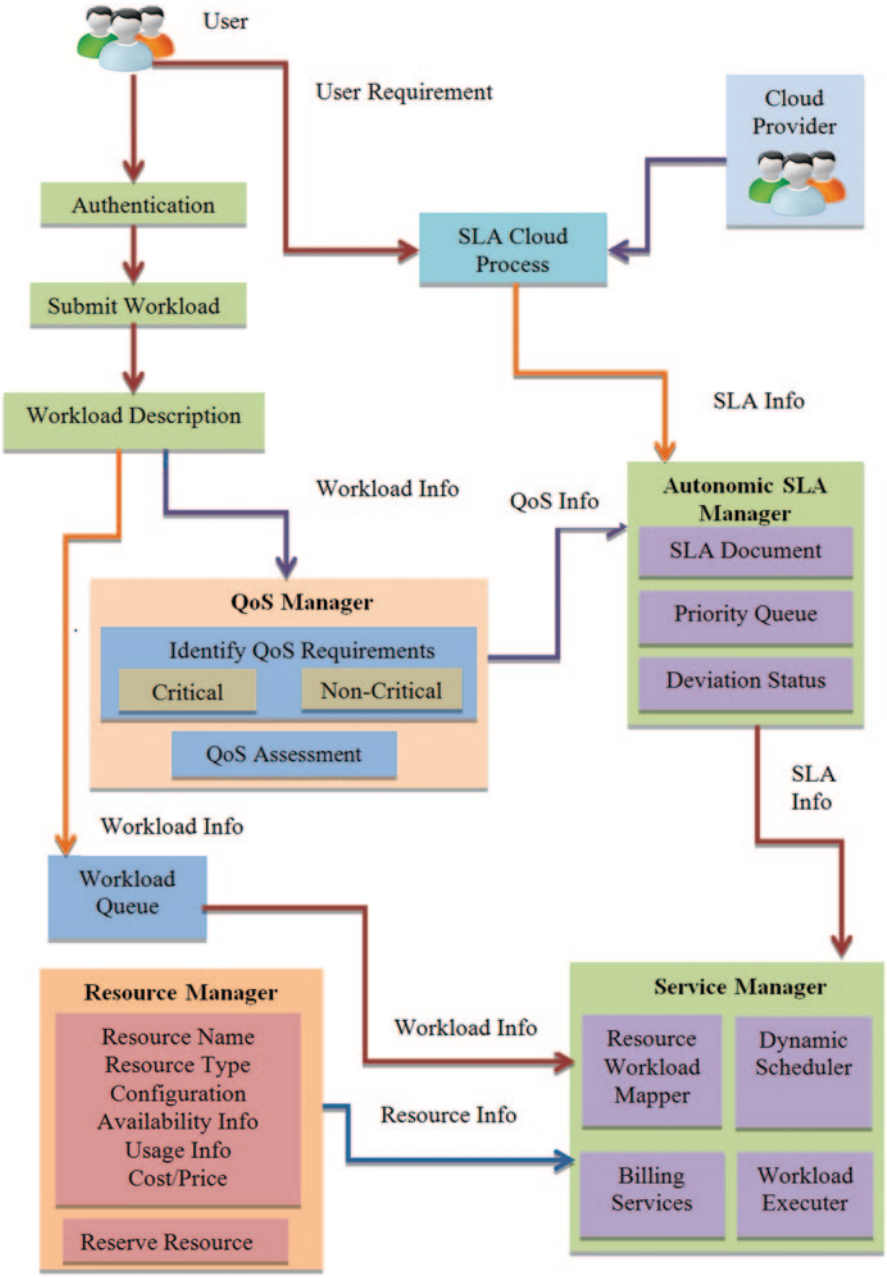

Fig. 3.2 Cloud SLA (CSLA) architecture. *SLA* service level agreement, *QoS* quality of service

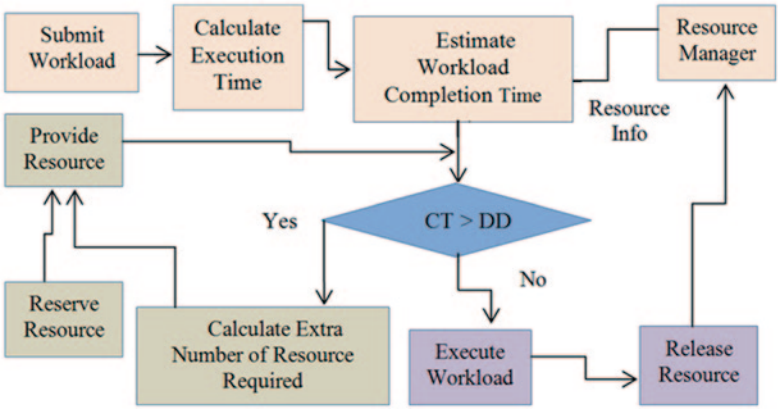

Fig. 3.3 Autonomic service level agreement (SLA) manager in Cloud SLA (CSLA) architecture. *CT* completion time, *DD* desired deadline

utilization and performance to attain a well-proportioned load scattering in the Cloud environments, where the load is characterized as the VM utilization. Furthermore, we define the standard deviation of resource utilization and performance so as, to prevent any hurdle in evaluating the degree of inconsistency. Consequently, the CSLA architecture also targets to reduce the degree of inconsistency. The consideration of standard deviation would aid to avoid the unstable workload of customers during the VMs distribution. The main components of the proposed architecture are as follows:

1. *Authentication*: The user should have valid username and password.
2. *Submit workload*: After authentication, the user will submit their Cloud workload that will be executed in this CSLA architecture.
3. Workload description: All the workload should have their key QoS requirements, based on that the workload is executed with some user defined constraints.
4. *Workload queue*: All the submitted Cloud workloads will be put into a workload queue for execution.
5. *QoS manager*: Based on the key QoS requirements of a particular workload, the QoS manager puts the workload into critical and non-critical queues through QoS assessment.
6. *Autonomic SLA manager*: Based on SLA information, SLA document will be prepared and accordingly urgent Cloud workloads would be placed in priority queue for earlier execution. Deviation status is used to measure the deviation of QoS from predictable with their possible resolution. If the deviation is more than the allowed, then it will allocate the reserve resources to the particular job or workload. Flowchart of autonomic SLA manager in CSLA architecture is shown in Fig. 3.3.
7. *Resource manager*: It contains the information about the available resources and reserved resource along with resource description (resource name, resource type, configuration, availability information, usage information, and price of resource).

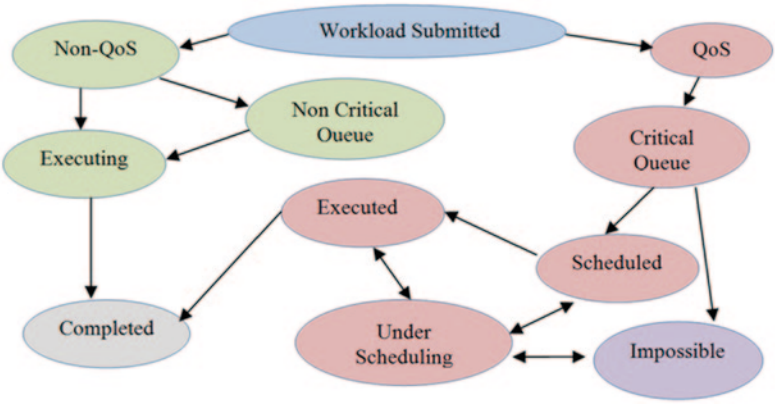

Fig. 3.4 States in Cloud SLA (CSLA) architecture. *SLA* service level agreement, *QoS* quality of service

8. *Service manager*: Based on SLA information, workload information and resource information, the service manger map the workloads to the appropriate resource by taking care of both SLA and QoS. Dynamic scheduler will schedule the workload for execution and billing for that execution will be generated. After payment, the workload executer will execute the workloads.

As shown in Fig. 3.3, the SLA Manager will calculate the execution time of workload and find the approximate workload turnaround time or completion time (CT). If the CT is lesser than the desired deadline (DD), then it will execute immediately with the available resources and release the resource back to resource manager for another execution, otherwise calculate extra number of resources required and provide from the reserved stock for current execution after recreating the SLA document with new user constraints. There are 11 states through which a submitted workload can move as shown in Fig. 3.4.

The first state for every workload is 'workload submission'. Based on key QoS requirements of workload, the next state will be decided either as non-QoS or QoS (quality oriented workloads). After non-QoS state, if there is no other workload pending, then it will execute directly other workload that is waiting into non-critical queue. After successful execution of workload, the workload is completed. On the other hand, all the QoS-oriented workloads are put into critical queue and sorted based on their priority decided by QoS manager and then scheduled for execution. If there is no obstacle (urgency, more resource requirement, etc.), then execute directly with available resources, otherwise put it into under-scheduling state to fulfill the user requirements. If all the conditions meet the given budget, resource, and time constraints, then it will execute, otherwise it will not be executed. CSLA architecture is the key mechanism that ensures that Cloud providers can serve large amount of requests without violating SLA terms. It dynamically manages the resources by using efficient resource scheduling techniques. For instance, when a workload requires low amount of resources, it will assign resources with lower capability, so that new requests can be served.

3.5 Discussion

As designated in the suggested architecture, we observe a very sincere requirement of CSLA architecture to administrate SLAs in the perspective of the Cloud environment. The proposed CSLA architecture recommends a very flexible design for handling SLAs between Cloud providers and Cloud users. We perceive this as one of the strong facets of CSLA architecture where, realistic to the prototype of SOA, each functionality is delivered as a Cloud service that could not essentially come from the similar Cloud provider. One vital remark we make in the framework of Clouds is the absence of standardization. This is especially essential when we try to relate through manifold Clouds. Even though it is possible to provide service for diverse Cloud interfaces through a middleware, there is no general collection of metrics that can be supervised through Cloud providers. There are challenges to organize the Clouds and we highlight the importance of such determinations in the light of observing abilities. As a part of these standardization determinations, we also recommend four types of straightforward metrics for measurements to be recognized. Clouds would not be capable of scaling indefinitely when a resource restriction is faced. A service provider may choose to assign the Cloud workloads or applications or tasks to another provider to avoid important SLA violation penalties. Such a situation generates research prospects in SLA supervision. We proceed to analyze SLA characteristics like accounting, monitoring of QoS restrictions, and condition damage in related situations as upcoming research.

3.6 Conclusions and Future Research Directions

This chapter discussed significant factors that could be considered when developing Cloud SLAs. Four types of metrics have been recognized for specifying fine-grain guarantees of QoS. The defects in the Cloud service can be easily identified and SLA deviation can be measured through these metrics. This work mainly focuses on enhancing the QoS provided by CSLA architecture. The concept and challenges of SLA-based provisioning and QoS for applications and workloads implementation in the Cloud environment have been presented. We have also proposed and presented a CSLA architecture that enables adaptive and dynamic provisioning of the resources based on workload-defined policies for satisfying their own SLA performance requirements, avoiding the price of any SLA violation and governing the budgetary cost of the distributed computing resources. Future research in this area can be recognized in many ways. One such opportunity is based on QoS requirements, which is considered as a vital characteristic of Cloud computing. The work presented here can be extended along several lines. From the research method viewpoint, our investigative method should evolve into theory building and a supposition testing as more experimental data about Cloud computing adoption becomes available. From the research output perception, the work regarding different service and deployment models, the comparative importance of SLA components as

associated to industry-specific features, and new characteristics and perceptions in the innovativeness modeling of the Cloud computing subcontracting judgment can be initiated. Some more QoS parameters can be analyzed and incorporated to find the critical success factors of the CSLA architecture and offer a model that will further help in accomplishing SLA in the Cloud environment using an automated tool.

References

1. Ayadi I, Simoni N, Diaz G (2013) QoS-aware component for Cloud computing. In: ICAS 2013, the ninth international conference on autonomic and autonomous systems (pp 14–20)
2. Bonvin N, Papaioannou TG, Aberer K (2011) Autonomic sla-driven provisioning for Cloud applications. In: Cluster, Cloud and Grid computing (CCGrid), 2011 11th IEEE/ACM international symposium on IEEE, pp 434–443
3. Breskovic I, Maurer M, Emeakaroha VC, Brandic I, Dustdar S (Dec 2011) Cost-efficient utilization of public sla templates in autonomic Cloud markets. In: Utility and Cloud computing (UCC), 2011 fourth IEEE international conference on IEEE, pp 229–236
4. Buyya R, Garg SK, Calheiros RN (2011) SLA-oriented resource provisioning for Cloud computing: challenges, architecture, and solutions. In: Cloud and Service computing (CSC), 2011 international conference on IEEE, pp 1–10
5. Buyya R, Calheiros RN, Li X (2012) Autonomic Cloud computing: open challenges and architectural elements. In: Emerging applications of information technology (EAIT), 2012 third international conference on IEEE, pp 3–10
6. Cardellini V, Casalicchio E, Lo Presti F, Silvestri L (2011) Sla-aware resource management for application service providers in the Cloud. In: Network Cloud computing and applications (NCCA), 2011 first international symposium on IEEE, pp 20–27
7. Casalicchio E, Silvestri L (2011) Architectures for autonomic service management in Cloud-based systems. In: Computers and communications (ISCC), 2011 IEEE symposium on IEEE, pp 161–166
8. Casalicchio E, Silvestri L (2013) Mechanisms for SLA provisioning in Cloud-based service providers. Computer Networks. 57(3):795–810
9. Chazalet A, Dang Tran F, Deslaugiers M, Exertier F, Legrand J (2010) Self-scaling the Cloud to meet service level agreements. In: Cloud computing 2010, the first international conference on Cloud computing, GRIDs, and virtualization, pp 116–121
10. CIO http://www.cio.com.au. Accessed 26 Nov 2013
11. Dillon T, Wu C, Chang E (2010) Cloud computing: issues and challenges. In: Advanced information networking and applications (AINA), 2010 24th IEEE international conference on IEEE, pp 27–33
12. Duong TNB, Li X, Goh RSM, Tang X, Cai W (2012) QoS-aware revenue-cost optimization for latency-sensitive services in IaaS Clouds. In: Distributed simulation and real time applications (DS-RT), 2012 IEEE/ACM 16th international symposium on IEEE, pp 11–18
13. Emeakaroha VC, Brandic I, Maurer M, Dustdar S (2010) Low level metrics to high level SLAs-LoM2HiS framework: bridging the gap between monitored metrics and SLA parameters in Cloud environments. In: High performance computing and simulation (HPCS), 2010 international conference on IEEE, pp 48–54
14. Emeakaroha VC, Calheiros RN, Netto MA, Brandic I, De Rose CA (2010) DeSVi: an architecture for detecting SLA violations in Cloud computing infrastructures. In: Proceedings of the 2nd international ICST conference on Cloud computing (CloudComp'10)
15. Emeakaroha VC, Netto MA, Calheiros RN, Brandic I, Buyya R, De Rose, CA (2012) Towards autonomic detection of sla violations in Cloud infrastructures. Future Gener Comp Syst 28(7):1017–1029

16. Garg SK, Gopalaiyengar SK, Buyya R (2011) SLA-based resource provisioning for heterogeneous workloads in a virtualized Cloud datacenter. In: Algorithms and architectures for parallel processing. Springer, Berlin, pp 371–384
17. Goiri Í, Julià F, Fitó JO, Macías M, Guitart J (2010) Resource-level QoS metric for CPU-based guarantees in Cloud providers. In: Economics of Grids, Clouds, Systems, and Services, Springer, Berlin, pp 34–47
18. Huebscher MC, McCann JA (2008) A survey of autonomic computing—degrees, models, and applications. ACM Comput Surveys (CSUR) 40(3):1–31
19. Kertesz A, Kecskemeti G, Brandic I (2011) Autonomic sla-aware service virtualization for distributed systems. In: Parallel, distributed and network-based processing (PDP), 2011 19th euromicro international conference on IEEE, pp 503–510
20. Kounev S, Nou R, Torres J (2007). Autonomic qos-aware resource management in grid computing using online performance models. In: Proceedings of the 2nd international conference on performance evaluation methodologies and tools. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), pp 1–10
21. Kumar S, Goudar RH (2012) Cloud computing—research issues, challenges, architecture, platforms and applications: a survey. Int J Future Comput Commun 1(4):356–360
22. Li J, Chinneck J, Woodside M., Litoiu M, Iszlai, G (2009) Performance model driven QoS guarantees and optimization in Clouds. In: Software engineering challenges of Cloud computing, 2009. CLOUD'09. ICSE workshop on IEEE, pp 15–22
23. Liu X, Zhu L (2009) Design of SOA based web service systems using QFD for satisfaction of quality of service requirements. In: Web services, 2009. ICWS 2009. IEEE international conference on IEEE, pp 567–574
24. Lodi G, Panzieri F, Rossi D, Turrini E (2007) SLA-driven clustering of QoS-aware application servers. IEEE Trans Software Eng 33(3):186–197
25. Maurer M, Brandic I, Sakellariou R (2011) Enacting SLAs in Clouds using rules. In: Euro-Par 2011 parallel processing, Springer, Berlin, pp 455–466
26. Nathuji R, Kansal A, Ghaffarkhah A (2010) Q-Clouds: managing performance interference effects for qos-aware Clouds. In: Proceedings of the 5th European conference on computer systems, ACM, pp 237–250
27. Ostermann S, Iosup A, Yigitbasi MN, Prodan R, Fahringer T, Epema D (2009) An early performance analysis of Cloud computing services for scientific computing. In: Proceedings of the 1st international conference on Cloud computing (CloudCom 2009), Beijing, pp 1–22
28. Rezaee A, Rahmani AM, Parsa S, Adabi, S (2008) A multi-agent architecture for qos support in grid environment. J Comput Sci 4(3):225–231
29. Rosenberg F, Celikovic P, Michlmayr A, Leitner P, Dustdar S (2009) An end-to-end approach for qos-aware service composition. In: Enterprise distributed object computing conference, 2009. EDOC'09. IEEE international, IEEE, pp 151–160
30. Salehie M, Tahvildari L. (2005) Autonomic computing: emerging trends and open problems. ACM SIGSOFT Software Eng Notes 30(4):1–7 (ACM)
31. Singh S, Chana I (2012) Cloud based development issues: a methodical analysis. Int J Cloud Comput Services Sci (IJ-CLOSER) 2(1):73–84
32. Singh S, Chana I (2013) Advance billing and metering architecture for infrastructure as a service. Int J Cloud Comput Services Sci (IJ-CLOSER) 2(2):123–133
33. Theilmann W, Happe J, Kotsokalis C, Edmonds A, Kearney K, Lambea J (2010) A reference architecture for multi-level sla management. J Internet Eng 4(1):289–298
34. Van HN, Tran, FD, Menaud JM (2009) SLA-aware virtual resource management for Cloud infrastructures. In: Computer and information technology, 2009. CIT'09. Ninth IEEE international conference on Vol. 1, IEEE, pp 357–362
35. Xiao J, Boutaba R (2005) QoS-aware service composition and adaptation in autonomic communication. IEEE J Selected Areas Commun 23(12):2344–2360
36. Xu M, Cui L, Wang H, Bi Y (2009) A multiple QoS constrained scheduling strategy of multiple workflows for Cloud computing. In: Parallel and distributed processing with applications, 2009 IEEE international symposium on IEEE, pp 629–634

37. Yang F, Su S, Li Z (2008) Hybrid QoS-aware semantic web service composition strategies. Sci China Series F: Inform Sci, 51(11):1822–1840
38. Yoo S, Kim S (2013) SLA-aware adaptive provisioning method for hybrid workload application on Cloud computing platform. In: Proceedings of the international multi conference of engineers and computer scientists (Vol 1).
39. Zhang P, Yan Z (2011) A QoS-aware system for mobile Cloud computing. In: Cloud computing and intelligence systems (CCIS), 2011 IEEE international conference on IEEE, pp 518–522
40. Zhang Q, Cheng L, Boutaba R (2010) Cloud computing: state-of-the-art and research challenges. J Internet Services Appl 1(1):7–18
41. Zheng Z, Zhang Y, Lyu MR (2010) CloudRank: a QoS-driven component ranking framework for Cloud computing. In: Reliable distributed systems, 2010 29th IEEE symposium on IEEE, pp 184–193