Cloud Computing: Issues and It's Challenges

KishanKashyap[1](Email:kishankashyap0971@gmail.com
Akash Yadav[2](Email:akashyadav5jul97@gmail.com)
Abhishek Yadav[3],
N.K SHARMA[4] (drnaveenkr101@gmail.com)
Department of MCA
IIMT College of Engineering Greater Noida
Uttar Pradesh, India

Abstract— Cloud computing is widely regarded as a transformative force that will revolutionize the entire ICT industry. This paper focuses on examining the challenges and issues associated with Cloud computing. It begins by exploring two closely related computing paradigms: Service-Oriented Computing and Grid computing, and their connections to Cloud computing. Subsequently, it identifies a range of challenges from the perspective of adopting Cloud computing. Lastly, the paper emphasizes the crucial need for extensive research and development in addressing the issue of Cloud interoperability.

Keywords: Cloud computing; Servcice-Oriented Computing; Distributed Comptuing; Web Services

1. INTRODUCTION

Cloud computing has gained significant attention in the distributed computing community, with many predicting it will revolutionize the IT industry. This paper aims to provide a comprehensive understanding of Cloud Computing, its distinctions from service-oriented computing and Grid computing, as well as the challenges faced by both cloud providers and consumers.

To begin, we define Cloud Computing and highlight its unique characteristics. We then explore the relationship between Cloud computing and Service-Oriented Computing (SOC), as well as Cloud computing and Grid computing (specifically High-Performance Computing). By comparing these three paradigms, we emphasize the mutual benefits they offer in a coexistent environment. Furthermore, we delve into the various service models and deployment models of Cloud computing. This discussion leads us to address critical data-related challenges such as multi-tenancy, security, and other pertinent issues. We analyse the service model and deployment model of a cloud, shedding light on the complexities involved.

Cloud computing is a paradigm that facilitates convenient and on-demand access to a shared pool of configurable computing resources, such as networks, servers, storage, applications, and services. These resources can be swiftly provisioned and released with minimal effort in terms of management or interaction with the service provider [1].

This definition encompasses various aspects of cloud computing, including cloud architectures, security, and deployment strategies. It specifically outlines five fundamental elements that are integral to cloud computing.

Lastly, we tackle the crucial topics of interoperability and standardization within the Cloud computing domain. We recognize the need for seamless interoperability between different cloud platforms and highlight the significance of standardization efforts.Through this paper, we aim to identify key research issues and outline future research directions for Cloud computing. By adopting an outside-in approach, we offer insights into cloud applications, computing paradigms, service models, deployment models, data-related challenges, and the importance of interoperability and standardization.

2. CLOUD: OVERVIEW

A. Definition

Cloud computing is a framework that facilitates easy and immediate access to a shared collection of adaptable computing resources, including networks, servers, storage, applications, and services, through a network. It enables quick allocation and release of resources with minimal need for manual administration or direct involvement with service providers.

On-demand self-service: In cloud computing, consumers have the ability to access computing resources, such as CPU time, network storage, software usage, and more, instantly and automatically. This access is self-serve and convenient, eliminating the need for human interactions with resource providers.

Furthermore, these computing resources are made available over the network, typically the Internet, allowing various client applications on different platforms (such as mobile phones, laptops, and PDAs) to utilize them. This broad network access enables consumers to leverage cloud resources from their own locations, regardless of the device they are using.

Resource pooling. A cloud service provider consolidates their computing resources into a shared pool, aiming to cater to multiple consumers through the adoption of either the multi-tenancy or virtualization model. This pooling approach involves the dynamic assignment and reallocation of diverse physical and virtual resources based on the varying demands of consumers [1]. The establishment of such a pool-based computing paradigm is driven by two crucial factors.

economies of scale and specialization. As a consequence of the pool-based model, consumers are unaware of the specifics regarding the physical computing resources within the cloud. They lack control and knowledge regarding the location, configuration, and origins of these resources, such as databases or CPUs. For instance, consumers cannot determine the exact storage location of their data within the cloud environment.

Rapid elasticity. From the perspective of consumers, computing resources in cloud computing are available instantly and do not require long-term commitments or contracts. Consumers have the flexibility to scale up their resource usage as needed and release them when they are no longer required. Additionally, the provisioning of resources appears to be unlimited, allowing consumers to rapidly increase their consumption to meet peak demands at any given time.

Measured Service. While computing resources in cloud computing are shared among multiple consumers through multi-tenancy, the cloud infrastructure employs suitable mechanisms to accurately measure the usage of these resources for each individual consumer. This is achieved through the metering capabilities of the cloud infrastructure, enabling precise tracking and allocation of resource usage on a per-consumer basis.

B. Servcice Model

Apart from the aforementioned five essential characteristics, the cloud community has widely adopted three distinct service models to classify cloud services.

Software as a Service (SaaS). Cloud consumers deploy their applications on a hosting environment that can be accessed by application users through various client devices, such as web browsers or PDAs, via networks. In this setup, cloud consumers lack direct control over the underlying cloud infrastructure, which typically operates on a multi-tenancy system architecture. This architecture allows multiple cloud consumers' applications to coexist within a single logical environment on the Software-as-a-Service (SaaS) cloud. This approach enables economies of scale and optimization in terms of speed, security, availability, disaster recovery, and maintenance. Notable examples of SaaS include SalesForce.com, Google Mail, and Google Docs.

Platform as a Service (PaaS). Platform-as-a-Service (PaaS) is a development platform that provides comprehensive support for the entire software lifecycle. It enables cloud consumers to develop cloud services and applications, including Software-as-a-Service (SaaS), directly within the PaaS cloud environment. The primary distinction between SaaS and PaaS lies in their scope and functionality. SaaS exclusively hosts completed cloud applications, whereas PaaS offers a development platform that accommodates both completed and ongoing cloud application development. To fulfill this role, PaaS must not only provide an application hosting environment but also possess a development infrastructure encompassing programming environments, tools, configuration management, and more. An illustrative example of PaaS is Google AppEngine.

Infrastructure as a Service (IaaS). Infrastructure-as-a-Service (IaaS) allows cloud consumers to directly utilize fundamental computing resources such as processing power, storage, and networks provided by the IaaS cloud. In the IaaS cloud, virtualization plays a crucial role in dynamically integrating or decomposing physical resources to meet the varying resource demands of cloud consumers. Virtualization achieves this by establishing independent virtual machines (VMs) that are isolated both from the underlying hardware and other VMs. It's important to note that this virtualization strategy differs from the multi-tenancy model, which aims to modify the application software architecture to enable multiple instances from different cloud consumers to run on a single application. Amazon's EC2 serves as an example of IaaS.

Data storage as a Service (DaaS). The provisioning of virtualized storage on demand has evolved into a distinct cloud service known as Data Storage as a Service (DaaS). It is worth noting that DaaS can be considered a specialized form of Infrastructure-as-a-Service (IaaS). The motivation behind DaaS stems from the significant upfront costs associated with on-premise enterprise database systems, including dedicated servers, software licenses, post-delivery services, and in-house IT maintenance. DaaS offers consumers the ability to pay only for the actual storage they use, eliminating the need for site licenses covering the entire database.

In addition to supporting traditional storage interfaces like relational database management systems (RDBMS) and file systems, certain DaaS offerings provide table-style abstractions designed to handle large volumes of data within compressed timeframes. These solutions address scenarios where the data size is too large, the storage requirements are cost-prohibitive, or the retrieval speed is unattainable for most commercial RDBMS systems. Several examples of DaaS can be found in the market Identify applicable sponsor/s here. (sponsors) of this kind of DaaS include Amazon S3, Google BigTable, and Apache HBase, etc.

C. Deployment Model

In recent times, the cloud community has established four distinct cloud deployment models.

Private cloud. A private cloud infrastructure is exclusively operated within a single organization and can be managed either by the organization itself or a third party, regardless of whether it is located on-premises or off-premises. The decision to establish a private cloud within an organization is motivated by several factors. Firstly, it aims to maximize and optimize the utilization of existing resources available within the organization. Secondly, concerns related to security, including data privacy and trust, make the private cloud an attractive option for many firms. Thirdly, the significant cost of data transfer from local IT infrastructure to a public cloud also contributes to the preference for a private cloud. Fourthly, organizations often require complete control over mission-critical activities that are located behind their firewalls. Lastly, private clouds are commonly built by academic institutions for research and educational purposes.

Community cloud. In the collaborative deployment model, multiple organizations come together to collectively build and share a common cloud infrastructure. They share not only the infrastructure itself but also the associated policies, requirements, values, and concerns. This collaborative approach within the cloud community enables economic scalability and fosters a democratic equilibrium. The cloud infrastructure can be hosted either by a third-party vendor or within one of the participating organizations in the community.

Public cloud. The public cloud deployment model is currently the most prevalent form of cloud computing. It is utilized by the general public, with the cloud service provider having complete ownership and control over the public cloud infrastructure. The provider establishes their own policies, values, profit models, and cost structures for the public cloud services. Examples of well-known public cloud services include Amazon EC2, S3, Google AppEngine, and Force.com.

Hybrid cloud. The hybrid cloud deployment model involves the integration of two or more distinct clouds, such as private, community, or public clouds. While each cloud remains independent, they are interconnected through standardized or proprietary technology, facilitating the seamless portability of data and applications. This can enable functionalities like cloud bursting for load balancing across multiple clouds.

Organizations adopt the hybrid cloud model to optimize their resources and enhance their core competencies. They can leverage the cloud for peripheral business functions while retaining control over core activities on-premise through a private cloud. The hybrid cloud model introduces challenges related to standardization and cloud interoperability, which will be further explored in subsequent sections.

Recently, Amazon Web Services (AWS) introduced a novel deployment model known as Virtual Private Cloud (VPC). It serves as a seamless and secure link between an organization's existing IT infrastructure and the Amazon public cloud. VPC can be seen as a hybrid model, combining elements of both private and public clouds.

In terms of its public aspect, VPC utilizes computing resources that are pooled by Amazon and made available to the general public. However, it offers a virtually private environment for organizations. This is achieved through the establishment of a secure virtual private network (VPN) connection, ensuring the same level of security found in a private cloud. Notably, all corporate security policies remain applicable to resources on the cloud, even though they are hosted within the public cloud infrastructure.

Furthermore, AWS provides a dedicated set of "isolated" resources specifically for the Virtual Private Cloud (VPC). It's important to note that users are not required to make upfront payments for these isolated resources. Instead, they can continue to enjoy the "pay-per-use" model for these resources. This approach allows VPC to achieve a harmonious blend of control, resembling that of a private cloud, with the flexibility typically associated with a public cloud.

It's worth mentioning that the service model and deployment model are independent of each other. For instance, Software as a Service (SaaS) can be provisioned on either a public or private cloud, illustrating the separation between the service model and the deployment model.

3. CLOUD, SOC, AND GRID

In this section, we aim to explore the connections and interrelationships among Cloud Computing, Service-Oriented Computing (SOC), and Grid Computing.

A. Cloud and Service-Oriented Computing

The significance of Service-Oriented Computing (SOC) lies in its ability to offer encapsulation, componentization, decentralization, and integration capabilities. These capabilities provide architectural principles and software specifications for establishing connections between computers and devices using standardized protocols over the Internet [3]. It is worth noting that the concept of Cloud Computing has been heavily influenced by the progressive advancements in SOC, particularly in relation to the Software as a Service (SaaS) service model.

The advancements in Service-Oriented Computing (SOC) bring several advantages and benefits to Cloud Computing.

Service Description for Cloud Services. Web Services Description Language (WSDL) and the Representational State Transfer (REST) protocol are commonly employed interface languages for describing web services. These languages have found extensive application in defining the API specifications of Cloud services.

Service Discovery for Cloud Services. Different models for service discovery can be utilized to facilitate the discovery, selection, and verification of cloud resources and service-level agreements.

Service Composition for Cloud ServiceAs web services were originally developed for integrating business applications, a significant amount of research in this field can be applied to facilitate the integration, collaboration, and composition of cloud services.

Service Management for Cloud Service. The research and practices in SOA governance and services management can be repurposed and applied to effectively manage the infrastructure of cloud computing.

When examining Service-Oriented Computing (SOC) from the perspective of small and medium enterprises (SMEs), certain aspects are lacking. While SOC provides a high level of abstraction for integration and business processes, it falls short in providing practical computational models for service execution. SMEs face challenges in determining how to run services with minimal cost and how to efficiently scale their applications built on Service-Oriented Architecture. These computational concerns often require project-specific and ad-hoc solutions, placing additional burdens on SOC developers and IT departments within SMEs. Additionally, incorporating services at different levels into a cohesive organizational structure remains an open question in SOC. Maximizing the utilization of IT services to support business services poses a significant challenge. Consequently, we believe that Cloud computing can offer valuable contributions to Service-Oriented Computing research, addressing these important considerations.

Cloud for Web Service Development. The Cloud offers the opportunity to host service-oriented development through the Platform as a Service (PaaS) deployment model. Small and medium enterprises (SMEs) often struggle to access distributed computing resources necessary for Service-Oriented Computing (SOC) development. For instance, Google's AppEngine provides a comprehensive development platform that includes the necessary software development kit (SDK) and integrated development environment (IDE) for developers to create and deploy Java Web services for their applications. Furthermore, the Yahoo Pipe platform demonstrates how the Cloud can be utilized as both the design-time and run-time environment for service Mashups and Composition. These examples highlight the potential of Cloud computing to support SOC development and enable SMEs to leverage distributed computing resources effectively.

Cloud for Web Service Testing. Web services developers can take advantage of the virtually limitless computing resources offered by Public Cloud to conduct load testing and stress testing for their services. This involves simulating automated machine requests and network flows that mimic real-world scenarios. Traditionally, the ability to simulate network traffic for Web services testing has been challenging and costly, limiting the overall reliability of Web applications. However, the Cloud's accessibility and cost-effectiveness provide an ideal solution by offering extensive computing resources. With the Cloud, developers can replicate real-world usage of their systems by geographically distributed users, executing diverse user scenarios on a scale that was previously unattainable in traditional testing environments. This capability enables more comprehensive and accurate testing, contributing to the improvement of Web service reliability.

Cloud for Web Service Deployment. By leveraging Infrastructure as a Service (IaaS), the deployment process for Web services can be significantly simplified. For instance, with Amazon EC2, service deployers can utilize the Amazon Machine Image (AMI) to distribute their offerings. When client requests are received, a service deployment image is loaded onto a designated virtual machine to handle these requests. Furthermore, any stateful information generated during service interactions can be stored persistently on the AMI, allowing Web services to resume seamlessly even after being suspended for an extended period, such as during a long-lasting transaction. This approach ensures the continuity and reliability of Web services while streamlining the deployment and management process.

Cloud for Service Process Enactment. The integration and composition of services are common challenges, and cloud computing offers a solution by providing a platform to deploy services specifically designed to address these issues. One popular approach is to harness the collective intelligence of the crowd, comprising service users, to enable the reuse of readily available solutions that require minimal configuration and can be easily composed using various algorithms such as Case-Based Reasoning. By leveraging the power of the crowd, organizations can benefit from pre-existing solutions that can be quickly adapted and combined to meet their specific requirements, leading to increased efficiency and effectiveness in service integration and composition within the cloud environment.

The integration between Cloud computing and SOA/SOC poses intriguing questions. Are they operating at the same technical and business level? Do they share the same objectives? Can they be effectively employed in conjunction? If so, what would be the approach for their simultaneous utilization? These challenges warrant further research and exploration, which can be facilitated through the continuous advancement and widespread adoption of cloud computing.

B. Cloud and Grid Computing

Grid computing [4] is an infrastructure, comprising both hardware and software components, that emerges in response to the practical challenges encountered in advanced scientific research. From our understanding, the Grid serves as a distributed computing middleware, facilitating coordinated sharing of computing resources across multiple organizations, particularly for high-performance computational applications in fields like science and engineering. While Cloud Computing and Grid Computing share certain common goals, such as resource virtualization, they also exhibit notable distinctions:

Grid computing places emphasis on "resource sharing" in order to create a virtual organization. In contrast, Cloud computing is typically owned by a single physical organization (unless it is a community Cloud, which is owned by a community), and resources are allocated to different instances as per the organization's discretion.

The primary goal of Grid computing is to provide maximum computing capacity for large-scale tasks through resource sharing. On the other hand, Cloud computing aims to fulfill numerous small-to-medium tasks based on real-time user requirements. This is why multi-tenancy is a crucial concept in Cloud computing.

Grid computing prioritizes re-usability, particularly in scientific high-performance computing scenarios. In contrast, Cloud computing is driven by immediate user needs stemming from diverse business requirements.

While Grid computing strives to maximize computing capacity, Cloud computing focuses on providing on-demand computing capabilities. This includes the ability to scale up and down, as well as optimize overall computing capacity.

C. Cloud and High Performance Computing

High-performance computing (HPC) is focused on utilizing supercomputers and computer clusters to tackle advanced computational problems, particularly in scientific domains. On the other hand, Cloud computing was initially designed to cater to business applications, resulting in different computing paradigms and application areas. While HPC is widely employed for scientific tasks, Cloud computing primarily serves business needs.

HPC extensively utilizes parallelization techniques, leveraging the power of multiple processors to enhance performance. However, in Cloud computing, the complex state and data dependencies present in many business applications make it challenging to fully exploit parallelization approaches. This poses a difficulty in applying parallel computing to business applications within the Cloud environment.

In a study referenced as [5], the authors highlight that the current state of Cloud computing is not yet fully suitable for HPC due to several reasons. Firstly, Cloud infrastructure is still evolving and has not reached maturity for HPC purposes.

Secondly, unlike Cluster computing, Cloud infrastructure focuses on optimizing overall system performance rather than specific scientific applications. Lastly, HPC aims to enhance the performance of a particular scientific application by utilizing resources from multiple organizations. However, the key distinguishing factor lies in the concept of elasticity. In Cluster computing, the capacity is typically fixed, requiring significant human intervention and tuning to match a specific cluster with a predefined number of homogeneous computing nodes. This stands in contrast to the self-service nature of Cloud computing, where the number of physical processors needed or utilized may not be known in advance.

4. CLOUD ADOPTION CHALLENGES

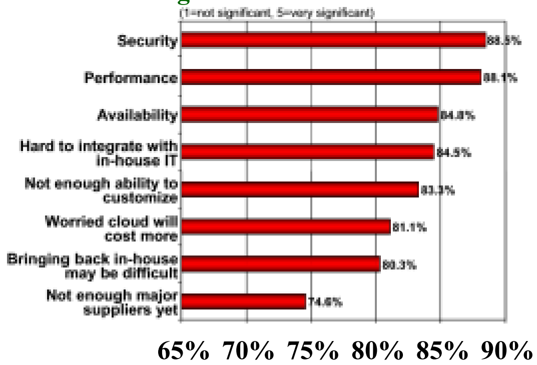
As Cloud Computing is still in its early stages, its adoption is accompanied by various challenges. According to a survey conducted by IDC in 2008, organizations have identified several major obstacles that hinder the widespread adoption of Cloud Computing, as depicted in Figure 1.

A. Security

The security aspect has undoubtedly been the most significant factor impeding the adoption of Cloud computing. It is understandable that entrusting one's data and software to be stored on someone else's hardware and processed on someone else's CPU can be perceived as daunting. Well-known security concerns such as data loss, phishing, and the operation of botnets (remotely controlled networks of compromised machines) pose serious threats to the integrity and confidentiality of an organization's data and software.

Additionally, the multi-tenancy model and the shared nature of computing resources in Cloud computing have introduced new security challenges [6] that demand innovative approaches for mitigation. For instance, hackers are exploring the utilization of Cloud infrastructure to orchestrate botnets, taking advantage of the reliable and cost-effective services offered by Cloud providers to launch attacks [6].

Q: Rate the challenges/issues of the 'cloud'/on-demand model



Source: IDC Enterprise Panel, August 2008 n=244 **% responding 3, 4 or 5**

Figure 1. Adoption Challenges (Source: IDC Survey, Aug 2008)

The adoption of the multi-tenancy model in Cloud computing has given rise to two notable security concerns. Firstly, the sharing of resources such as hard disks, data, and virtual machines on a single physical machine introduces the possibility of unexpected side channels between a malicious resource and a legitimate one. This can potentially compromise the confidentiality and integrity of the data and operations of the unaffected resources.

Secondly, the concept of "reputation fate-sharing" poses a significant risk, whereby the reputation of well-intentioned users sharing computing resources with a malicious co-tenant can be severely tarnished. Due to the sharing of network addresses, any malicious activities conducted by one tenant can be erroneously attributed to all users sharing the same resources. This lack of differentiation between genuine users and those with malicious intent can result in reputational damage to innocent users within the Cloud environment.

B. Costing Model

When considering the adoption of Cloud computing, organizations need to carefully weigh the tradeoffs involving computation, communication, and integration. While migrating to the Cloud can lead to significant cost savings in terms of infrastructure, it does introduce additional costs related to data communication. These costs include transferring data to and from public or community Cloud environments, which can be a substantial expense [7]. Additionally, the cost per unit of computing resources, such as a virtual machine, is likely to be higher in the Cloud.

This challenge becomes particularly relevant when utilizing the hybrid cloud deployment model, where an organization's data is distributed across multiple public, private (in-house IT infrastructure), or community clouds. The concept advocated by Gray [8], emphasizing the importance of "putting the computation near the data," remains applicable in cloud computing. It implies that for optimal efficiency, data-intensive tasks should be performed in close proximity to the data source.

From an economic perspective, it is essential to consider whether the cost savings gained from on-demand computing outweigh the additional expenses incurred by data transfer. While cloud computing is well-suited for CPU-intensive jobs, transactional applications such as ERP/CRM may not be as economically viable due to the potential imbalance between cost savings and data transfer costs.

Furthermore, the cost of data integration in cloud computing can be significant due to the use of proprietary protocols and interfaces across different cloud providers. Cloud consumers are often required to interact with multiple clouds using provider-specific APIs, necessitating the development of custom adaptors to distribute and integrate heterogeneous resources and data assets across different clouds, even within a single organization. This poses a challenge for seamless data exchange and integration.

To address security concerns, cloud consumers, such as the Eli Lilly research lab [9], may need to fragment sensitive data, such as individual patient drug usage information, and distribute it across different clouds. This approach aims to mitigate the impact of a security breach in one cloud by ensuring that the entire dataset is not compromised. However, the process of splitting and distributing data introduces additional financial costs and can negatively impact system performance, leading to increased time costs.

The need to balance security requirements, data fragmentation, integration challenges, and performance considerations further adds complexity to cloud computing adoption. Cloud consumers must carefully evaluate the tradeoffs and costs associated with data distribution and integration to ensure the effectiveness and efficiency of their cloud-based systems.

C. Charging Model

From the perspective of cloud providers, the introduction of elastic resource pools, achieved through virtualization or multi-tenancy, has significantly increased the complexity of cost analysis compared to traditional data centers. In data centers, costs are typically calculated based on the consumption of static computing resources. However, in the cloud, the unit of cost analysis has shifted to instantiated virtual machines rather than the underlying physical servers.

Developing a robust and accurate charging model in the cloud requires considering various factors. This includes incorporating the costs associated with virtual machine usage, such as software licenses, virtual network utilization, and the overhead of node and hypervisor management. Additionally, other factors unique to the cloud environment, such as resource pooling and dynamic provisioning, need to be accounted for in the charging model.

Creating a comprehensive and fair pricing structure for cloud services necessitates considering all these elements and their associated costs. Cloud providers must carefully analyze and integrate these factors into their charging models to ensure transparency and cost-effectiveness for their customers.

SaaS cloud providers face significant costs when developing and implementing multitenancy within their offerings. These costs include the need to redesign and redevelop software that was originally designed for single-tenancy environments. Additionally, there are expenses associated with providing new features that allow for intensive customization, enhancing performance and security to support concurrent user access, and addressing complexities arising from these changes.

Due to these substantial costs, SaaS providers must carefully evaluate the trade-off between implementing multitenancy and the cost-savings it can potentially generate. These cost-savings may include reduced overhead through amortization and a decreased number of on-site software licenses, among others. Therefore, developing a strategic and viable charging model becomes crucial for the profitability and sustainability of SaaS cloud providers.

By implementing an effective charging model, SaaS providers can appropriately balance the costs associated with multitenancy against the potential benefits. This allows them to optimize their revenue generation while providing cost-effective solutions to their customers.

D. Service Level Agreement

While cloud consumers lack control over the underlying computing resources, ensuring the quality, availability, reliability, and performance of these resources is crucial once consumers migrate their core business functions to the cloud. To achieve this, consumers need to obtain service delivery guarantees from cloud providers. These guarantees are typically established through Service Level Agreements (SLAs) negotiated between providers and consumers.

One of the initial challenges is defining SLA specifications that strike the right balance between expressiveness and complexity. These specifications should be sufficiently detailed to encompass consumer expectations while remaining manageable in terms of weighing, verifying, evaluating, and enforcing them through the cloud's resource allocation mechanism. Moreover, different cloud service offerings such as IaaS, PaaS, SaaS, and DaaS will require specific SLA metaspecifications to address their unique characteristics and requirements.

Cloud providers face various implementation challenges in ensuring SLA fulfillment. One such challenge is the need for resource managers to have precise and up-to-date information on resource usage within the cloud. This entails receiving real-time updates on any changes in the cloud environment through event notifications, allowing resource managers to make timely evaluations and adjustments to meet SLA requirements. To achieve this, resource managers must employ efficient decision models and optimization algorithms that enable them to make informed decisions and optimize resource allocation.

In order to maintain the "self-service" promise of cloud computing, these processes should be automated as much as possible. This means that resource managers may need to reject certain resource requests if they cannot be accommodated within the defined SLAs. Additionally, advanced SLA mechanisms should incorporate user feedback and customization features into the evaluation framework, ensuring that SLAs are constantly refined and aligned with user expectations.

E. What to migrate

According to a survey conducted by IDC in 2008, which had a sample size of 244 respondents, organizations were found to be migrating various IT systems and applications to the cloud. The survey identified seven categories of IT systems/applications that were being migrated, along with the corresponding percentages: IT Management Applications (26.2%), Collaborative Applications (25.4%), Personal Applications (25%), Business Applications (23.4%), Applications Development and Deployment (16.8%), Server Capacity (15.6%), and Storage Capacity (15.5%).

The survey findings indicate that organizations still have concerns about security and privacy when it comes to moving their data to the cloud. Currently, peripheral functions such as IT management and personal applications are considered relatively easier to migrate compared to core activities. There is a conservative approach in adopting Infrastructure as a Service (IaaS) compared to Software as a Service (SaaS). This could be attributed to the fact that organizations tend to outsource marginal functions to the cloud while keeping core activities in-house.

Looking ahead, the survey suggests that in three years' time, a significant portion of organizations (31.5%) plan to move their Storage Capacity to the cloud. However, this number is still relatively low compared to the projected migration of Collaborative Applications (46.3%) during that period. This indicates that while there is a growing interest in cloud adoption, organizations are proceeding cautiously and prioritizing certain types of applications for migration.

5. .CLOUD INTEROPERABIOLITY ISSUE

At present, each cloud provider has its own unique approach to how cloud clients, applications, and users interact with their respective cloud services. This situation has resulted in what is known as the "Hazy Cloud" phenomenon, which poses significant challenges for the development of cloud ecosystems. The lack of standardization and interoperability in cloud interfaces and APIs creates vendor lock-in, limiting users' ability to choose alternative vendors or offerings simultaneously. This restriction prevents organizations from optimizing their resource utilization across different levels within their infrastructure.

Moreover, the use of proprietary cloud APIs complicates the integration of cloud services with an organization's existing legacy systems. For instance, integrating an on-premise data center, commonly found in industries like pharmaceuticals, which relies on highly interactive modeling applications, becomes extremely difficult. Interoperability, in this context, refers to the ability to establish connections between different clouds as well as between a cloud and an organization's local systems.

The main objective of achieving interoperability is to enable seamless and fluid data exchange across multiple clouds and between cloud services and local applications. By establishing interoperability standards, the goal is to facilitate the smooth and efficient flow of data, ensuring that it can seamlessly move between different cloud environments and interact with an organization's on-premise systems.

Interoperability plays a crucial role in multiple levels of cloud computing. Firstly, organizations often strive to optimize their IT assets and computing resources by retaining in-house capabilities related to their core competencies while outsourcing less critical functions and activities, such as the human resource system, to the cloud. In this scenario, seamless communication between cloud services (e.g., the HR system) and on-premise systems (e.g., an ERP system) becomes vital for effective business operations. However, poor interoperability, such as the use of proprietary APIs or complex and ambiguous data structures employed by a HR cloud SaaS, can significantly increase integration difficulties, creating challenges for the IT department.

Secondly, organizations may need to outsource various marginal functions to cloud services provided by different vendors to achieve optimization. For instance, a small and medium-sized enterprise (SME) might utilize Gmail for email services and SalesForce.com for HR services. This necessitates the integration of multiple features (e.g., address book, calendar, appointment booking) in the email system with the HR employee directory residing in the HR system. Interoperability between these disparate cloud services is essential to ensure smooth data exchange and seamless functionality across different systems.

By establishing effective interoperability standards and protocols, organizations can overcome these challenges and enable seamless integration and communication between diverse cloud services. This facilitates the efficient flow of data and ensures the smooth operation of interconnected systems, supporting organizations in optimizing their IT resources and achieving their business objectives.

A. Intermediary Layer

Various recent studies have focused on addressing the challenge of interoperability in cloud computing by introducing intermediary layers between cloud consumers and cloud-specific resources, such as virtual machines (VMs). For instance, Sotomayor et al. [11] proposed the concept of Virtual Infrastructure Management using Open Nebula. This approach replaces direct interactions with native VM APIs and enables support for multiple types of clouds (private or hybrid) within an organization. Open Nebula operates at the virtualization level, providing cloud consumers with a unified view and standardized interfaces to interact with diverse underlying virtualization implementations.

In contrast to Open Nebula, Harmer et al. [12] developed an abstraction layer at a higher level. This layer offers a single resource usage model, user authentication model, and API, effectively shielding the heterogeneity of cloud providers. By abstracting away the differences between various cloud offerings, this approach facilitates the development of cloud-provider independent applications, reducing the barriers to interoperability.

Both approaches contribute to overcoming interoperability challenges in cloud computing by providing standardized and unified interfaces that bridge the gap between diverse cloud environments. These intermediary layers enable organizations to work with multiple clouds and develop applications that are not bound to specific cloud providers, promoting flexibility and seamless integration in the cloud ecosystem.

B. Standard

While standardization is seen as a potential solution to tackle interoperability challenges in cloud computing, it has not been a top priority for major industry cloud vendors as the technology is still in its early stages. Notably, both Microsoft and Amazon have not supported the Unified Cloud Interface (UCI) Project proposed by the Cloud Computing Interoperability Forum (CCIF) [13]. The lack of participation from these industry giants makes the standardization process challenging, as achieving consensus becomes difficult without their involvement.

In the academic realm, the Eucalyptus project [14] has gained popularity as a widely used cloud API. It emulates the proprietary Amazon EC2 API, allowing Eucalyptus IaaS cloud consumers to seamlessly connect with the EC2 cloud without significant redevelopment efforts. However, while this compatibility addresses specific interoperability concerns between Eucalyptus and EC2, it does not provide a comprehensive solution to the broader interoperability issue that necessitates an open API adhered to by diverse cloud providers.

Overall, the lack of strong support from major industry players and the absence of a widely adopted open API hinder the progress of standardization efforts. Overcoming these challenges is crucial to advancing interoperability in cloud computing, enabling seamless integration and facilitating the development of cloud applications that can effortlessly operate across different cloud environments.

C. Open API

The Sun Open Cloud Platform, recently launched by SUN under the Creative Commons license [15], introduces a notable contribution in the form of a cloud API. This API, still in development, offers a set of RESTful web service interfaces that are clear and easy to understand. It enables cloud consumers to create and manage various cloud resources, including compute, storage, and networking components, in a unified manner. The API leverages HTTP as the application protocol and utilizes JSON for resource representation.

The open cloud API defines several key resource types, such as Cloud, Virtual Data Center, Cluster, Virtual Machine, Private Virtual Network, Public Address, Storage Volume, and Volume Snapshot. These constructs exhibit similarities with the internal architectural design of Eucalyptus [14], which makes efforts to ensure compatibility between Eucalyptus clouds and the Sun cloud API [15]. This aligns with the ongoing research of DEBII in developing a lightweight PaaS open API using RESTful web services. It is worth noting that the concept of a Virtual Data Center, which serves as the fundamental entity for instantiating the Sun Open Cloud, is analogous to the recently introduced concept of Virtual Private Cloud in Amazon EC2.

The introduction of the Sun Open Cloud Platform and its associated API contributes to advancing interoperability and standardization efforts in the cloud computing domain. By providing a well-defined and openly accessible interface, it promotes compatibility and ease of integration between different cloud services and environments.

D. SaaS and PaaS Interoperability

While many solutions address interoperability issues in the Infrastructure as a Service (IaaS) domain, there has been limited research on interoperability in other service deployment models, particularly Software as a Service (SaaS). SaaS interoperability encompasses various application domains such as ERP, CRM, and others. Our research group at DEBII is particularly interested in the data mining research community.

During the KDD09 panel discussion [16], experts in data mining highlighted the need to establish a data mining standard in the cloud, focusing on practical use cases of statistical algorithms, reliable deployment of models, and integration of predictive analytics across different data mining-based SaaS clouds. A notable development in this area is the Predictive Model Markup Language (PMML), which is gradually being accepted as a standard for exchanging predictive models among different software tools.

In contrast, we have not yet come across significant efforts in providing Platform as a Service (PaaS) interoperability. PaaS encompasses the entire software development lifecycle in the cloud, making it more challenging to achieve uniformity in how consumers develop and deploy cloud applications. Further research and developments are needed to address PaaS interoperability concerns.

6. CONSLUSION

This paper extensively examines the challenges and concerns associated with Cloud computing. It explores the connections between Cloud computing, Service-Oriented Computing, and Grid computing, emphasizing their interrelationships. The analysis delves into various obstacles that organizations face when adopting Cloud computing.

One major focus is the interoperability challenge, which is thoroughly discussed. The paper explores different solutions tailored to address interoperability issues across various cloud service deployment models. The aim is to facilitate seamless integration and interaction between different cloud services.

By shedding light on these challenges and presenting potential solutions, this paper contributes to a deeper understanding of the intricacies involved in Cloud computing. It provides insights into the complexities of adopting Cloud computing and offers valuable recommendations to overcome the identified hurdles.

REFERENCES

[1] P. Mell and T. Grance, "Draft nist working definition of cloud computing - v15," 21. Aug 2009, 2009.

[2] M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, and I. Stoica, "Above the clouds: A Berkeley view of cloud computing," EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2009-28, 2009.

[3] M. N. Huhns and M. P. Singh, "Service-Oriented Computing: Key Concepts and Principles," IEEE Internet Computing, vol. 09, pp. 75 - 81, 2005.

[4] I. Foster and C. Kesselman, The Grid: Blueprint for a New Computing Infrastructure: Morgan Kaufmann, 1998.

[5] J. Napper and P. Bientinesi, "Can cloud computing reach the top500?," in Combined Workshops on UnConventional High Performance Computing Workshop plus Momory Access Workshop, 2009, pp. 17-20.

[6] Y. Chen, V. Paxson, and R. Katz, "What's New About Cloud Computing Security?," 2010.

[7] A. Leinwand, "The Hidden Cost of the Cloud: Bandwidth Charges," http://gigaom.com/2009/07/17/the-hidden-cost-of-the-cloud-bandwidthcharges/, 2009.

[8] J. Gray, "Distributed computing economics," ACM Queue, vol. 6, pp. 63-68, 2008.

[9] M. May, "Forecast calls for clouds over biological computing," Nature Medicine, vol. 16, p. 6, 2010.

[10] M. Nelson, "Building an Open Cloud," Science, vol. 324, p. 1656, 2009. [11] B. Sotomayor, R. Montero, I. Llorente, and I. Foster, "Virtual Infrastructure Management in Private and Hybrid Clouds," IEEE Internet Computing, vol. 13, pp. 14-22, 2009.

[12] T. Harmer, P. Wright, C. Cunningham, and R. Perrott, "ProviderIndependent Use of the Cloud," in The 15th International European Conference on Parallel and Distributed Computing, 2009, p. 465.

[13] "Unified Cloud Interface Project," http://code.google.com/p/unifiedcloud/.

[14] D. Nurmi, R. Wolski, C. Grzegorczyk, G. Obertelli, S. Soman, L. Youseff, and D. Zagorodnov, "The eucalyptus open-source cloudcomputing system," presented at the Proceedings of Cloud Computing and Its Applications, 2008.

[15] "Sun Microsystems Unveils Open Cloud Platform," http://www.sun.com/aboutsun/pr/2009-03/sunflash.20090318.2.xml, 2009.

[16] M. Zeller, R. Grossman, C. Lingenfelder, M. Berthold, E. Marcade, R. Pechter, M. Hoskins, and R. Holada, "Open standards and cloud computing: KDD-2009 panel report," in KDD, Paris, France, 2009, pp. 11-18.