Chapter 1

Introduction

1.1 Problem Statement

Cloud computing is a computing model offering a network of servers to their clients in a on-demand fashion. From NIST's definition [4], *"cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction."* To illustrate how it works, considering a case: a Cloud provider builds a data center which contains thousands of servers connected with network. These servers are virtualized which means they are partitioned into a smaller unit of resources called *Virtual Machines (VMs)*. A web-based application provider can access and deploy their applications (e.g Endnote, Google Drive and etc.) in these VMs from anywhere in the world. Once the applications start serving, application users can use them without installing on their local computers.

Generally, Cloud computing involves three stakeholders: Cloud providers, Cloud users, and End (application) users [3] (see Figure 1.1).

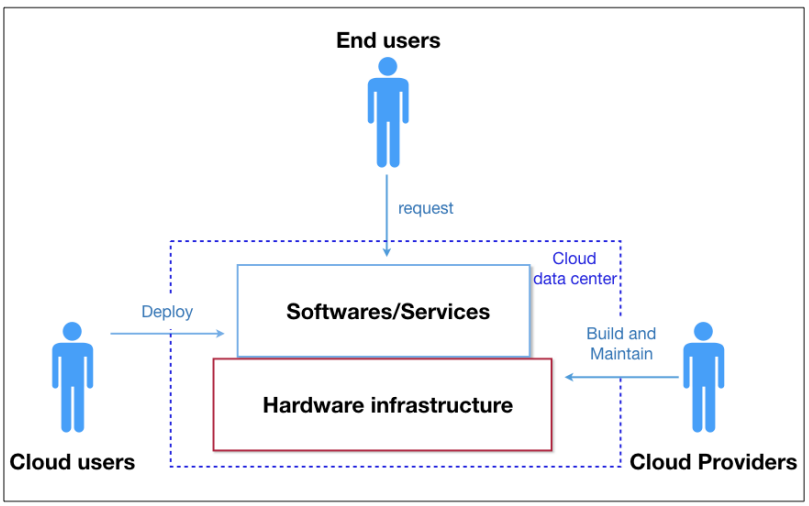

Figure 1.1: Stakeholders of Cloud computing

Each stakeholder has their responsibility, goal, and objectives.

- *Cloud providers* build data centers, provide maintenance and resource management on the hardware infrastructure. Their goal is to increase the profit by boosting the income and reducing the expense. Their income comes from Cloud users' rental of servers

or *Physical Machines (PMs)* in terms of resource quality (e.g 3.5GHz dual-core CPU), quantity (e.g 3 PMs), and time (e.g 1 year). Therefore, Cloud providers objective is to maximize utilization of computing resources. A high utilization brings two benefits, firstly, it increases income by accommodating more applications in limited resources. Secondly, it cuts the expense of energy consumption by packing applications in a minimum number of PMs so that idle PMs can be turned off.

- *Cloud users* develop and deploy applications on Cloud. Each application generates time-vary CPU utilization. Their goal is also increase the profit mainly through two objectives, attracting more End users and reduce the expense of resources. The first objective can be achieved by improving the quality of service as well as lower the fee for End users. Either way depends not only on the software entities but also the quality of service (QoS) offered by Cloud provider. The second objective can be achieved by a good estimation of the reserved resources, so that they do not rent insufficient or too much resources which cause performance degradation or wastage.

- *End Users* are the final customers in this chain. They consume services directly from Cloud users and indirectly from Cloud provider. Their goal is to obtain a satisfactory service. It is achieved by signing a Service Level Agreement (SLA) with Cloud users which constrains the performance of the services.

In this thesis, we focus on an core issue of helping Cloud providers to increase their profits by optimizing the resource allocation in data centers. Specifically, the profit can be improved by reducing the energy consumption. A direct way to reduce energy consumption is to always use a minimum number of Physical machines (PMs) hosting applications. The problem can be described as, a data center has a number of PMs where each of them can be represented as a set of resources such as CPU cores, RAM and etc. The data center received a list of requests for resources which is also represented as resources. The task is to allocate these requested resources to a minimum number of PMs. The decision variable in this task is the location of each requested resource. The assumptions and constraints are distinct in real life service models of Cloud computing.

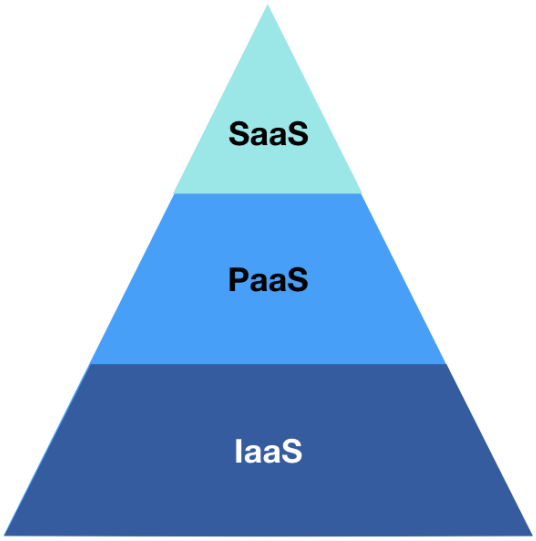

Figure 1.2: Pyramid of service models in Cloud computing. IaaS provides the fundamental resources such as CPU cores, RAM. The resources are usually wrapped with various types of virtual machine. PaaS provides a software platform sitting on top of IaaS for Cloud users to develop applications. SaaS is the application that serves End Users.

Besides the stakeholders of Cloud computing, service models of Cloud computing are the key to describe the relationships and responsibilities among stakeholders. There are three traditional service models [4]: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). The relationship among three service models can be described as a pyramid (see 1.2).

- IaaS, Cloud provider offers the fundamental resources such as CPU cores, RAMs, and network bandwidth. These resources are often encapsulated in virtualized computing units called virtual machines (VMs).

  Cloud providers establish a number of types of VM for simplifying the management. The 'type' means a VM contains a certain quantity of resources such as 2-cores and 1 GB RAM. They are the fundamental unit of resources.

  When Cloud users want to deploy their applications, they must select a type of VM. They estimate the resources that their applications might consume. The Cloud providers, in this stage, cannot help them to make the decision because of lacking knowledge of applications. After Cloud users have made the decisions, they send requests to Cloud providers for a number of VMs. Cloud providers received the request, provisioned and allocated these VMs to PMs. The constraint is that the aggregated resources in a PM cannot exceed the capacity of the PM. After these VMs have been allocated, their types cannot be changed.

  During the life cycle of an application, Cloud providers can dynamically adjust the locations of VMs, provision new VMs (same type) for the replicas of an application, as well as turning on/off PMs.

- PaaS, as shown in Figure 1.2, is sitting above the IaaS which means adding an abstraction over the underlying hardware as a middle-ware layer. This middle-ware layer provides a software development environment which allows Cloud users to build, test and deploy their applications on Cloud. In terms of resource management, PaaS

takes the responsibility of selecting VMs and allows Cloud users to focus on software development.

When Cloud users want to deploy their applications in PaaS, Cloud users need to provide the initial estimation of the quantity of resources instead of types of VM. Cloud providers determines the types of VM for applications according to the estimated resources. Those types of VM are normally just enough to satisfy the requirements. After this step, resource management system conducts the provisioning and allocating as the same steps in IaaS.

During the life cycle of applications, similar to IaaS, Cloud providers can also adjust the location of VMs, add new VMs, and control the status of PMs. Different from IaaS, Cloud providers can change the type of VM for an application as long as the application's performance can be guaranteed.

- SaaS describes the relationship between Cloud users and End users. End users create workloads for applications. Although this service model does not directly related to the resource management, it provides the fundamental reasons for resource management and optimization. Because of the dynamic nature of workloads, the underlying resources must also be dynamic adjusted to meet the requirement.

Our proposed optimization approaches are based on a new service model: Container as a Service (CaaS) [5] which is a variant of PaaS. The reasons for us to establish our approaches on CaaS are listed as followed:

- CaaS uses a new virtualization technology called containers which has shown several important characteristics that overcome the disadvantages of traditional IaaS and PaaS, therefore, CaaS is a promising trend.

- CaaS has a fine granularity level of resource management which has shown opportunities to improve the resource utilization, however, it often proposes new optimization challenges which have not been studied yet.

Firstly, we illustrate the disadvantages of traditional IaaS and PaaS and discuss the reasons of why current approaches cannot completely overcome these problems. From there, we explain why CaaS is a prescription of their problems.

IaaS has three characteristics which naturally lead to a low resource utilization.

- Separated responsibilities of resource selection for applications and resource allocation. As above mentioned, Cloud users have to select the type of resources. This causes a problem. The accurate estimation is almost impossible because of unpredictable workloads; Cloud users tend to reserve more resources for ensuring the QoS at the peak hours [2] than completely rely on auto-scaling, simply because auto-scaling is more expensive than reservation. However, the peak hours only account for a short period, therefore, in most of time, resources are wasted. As the types of VM are a part of the agreement, Cloud providers cannot simply change the type of VMs after provisioning.

- Cloud providers offer fixed types of VM. Because of the fixed size of resources and the one-on-one mapping of applications and VMs, specific applications consume unbalanced resources which leads to vast amount of resource wastage [7]. For example, computation intensive tasks consume much more CPU than RAM; a fixed type of VM provides much more RAM than it needs. Because the tasks use too much CPU, they prevent other tasks from co-allocating. This also causes wastage.

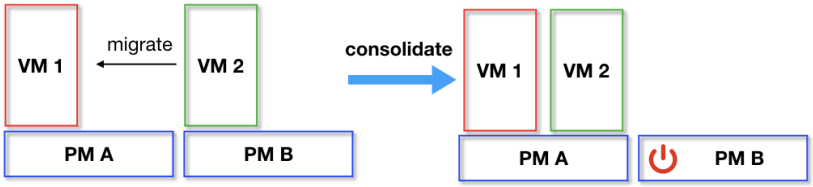

Figure 1.3: A Server Consolidation example: Initially, each PM runs an application wrapped with a VM in a low resource utilization state. After the consolidation, both VMs are running on PM A, so that PM B is turned off to save energy [1].

- Redundant operating systems (OSs) cause vast resource wastage. Since each VM runs a separated operating system, a PM might run many operating systems at a time. However, normal applications do not need specific operating systems commonly used OSs - such as Linux-based: RedHat, or Windows server versions - are well enough for their needs. Therefore, running duplicated OSs in a PM is unnecessary.

For the first two drawbacks, previous researchers and developers have come up with two strategies: server consolidation and overbooking.

Server consolidation [8], as above mentioned, utilized a dynamic migration technique to resolve the low utilization problem by gathering applications into a fewer number of PMs (see Figure **??**), so that the resource utilization of PMs are maintained at a high level. In the meanwhile, idle PMs can be turned off to save energy. Consolidation dramatically improves hardware utilization and lowers PM and cooling energy consumption.

Overbooking strategy [7] is used to overcome the low utilization problem raised by the over-provisioning of VMs. It allocates more VMs in a PM even their aggregated resources have exceeded the PM's capacity. The advantage of this approach is that, indeed, it improves the resource utilization. The disadvantages are also obvious: if one or more applications are experiencing a heavy load; the PM is likely to run out of resources. At this moment, all the applications are suffer from a QoS degradation. In order to avoid the overloading, Cloud users often carefully predict the utilization of applications based on their previous workloads and use a dynamic resource management approach to adjust the VMs' location.

The workflow of utilizing these two strategies in resource management is shown in Figure **??**. It seemingly solves the over-provisioning problem and fixed types of VM, however, the overbooking strategy brings new challenges. Although the overbooking strategy have been studies for years [], the prediction of workload is very difficult [] or even impossible as referred in many literatures []. The improvement of resource utilization is completely based on an accurate prediction of a large number of applications. Its effectiveness is hard to evaluate and justify. It is also very time consuming to predict the utilization of every application. Therefore, it is urgent to provide a strategy without making a prediction. One possible way is to use finer granularity strategy of resource management.

There is no solution for the third drawback in the context of IaaS.

For traditional PaaS, Cloud providers can adjust the applications' location regardless of the type of VM. Therefore, it overcomes the first two drawbacks of IaaS. Because of PaaS is built upon IaaS, the one-on-one relationship between application and VM still exist. Therefore, the OS redundancy problem cannot be solved. In addition, PaaS brings a restriction for the applications deployed on it. PaaS build a software middle-ware to allow Cloud users' development. The middle-ware requires the deployed applications to be compatible with the environment, for example, Google App engine [] only allows certain programming languages and libraries. Therefore, the generality of PaaS is very limited. It is urgent to provide

a environment which supports automatic resource management as well as an editable programming environment.

The recent development of container technology has successfully solved all problems in both IaaS and PaaS. A recent development of Container technique [6] has driven the attention of both industrial and academia. Container is an operating system level of virtualization which means multiple containers can be installed in a same operating system (see Figure **??** right-hand side). Each container provides an isolated environment for an application. In short, a VM is partitioned into smaller manageable units.

Bibliography

[1] BARROSO, L. A., AND HÖLZLE, U. The Case for Energy-Proportional Computing. *IEEE Computer 40*, 12 (2007), 33–37.

[2] CHAISIRI, S., LEE, B.-S., AND NIYATO, D. Optimization of Resource Provisioning Cost in Cloud Computing. *IEEE Trans. Services Computing 5*, 2 (2012), 164–177.

[3] JENNINGS, B., AND STADLER, R. Resource Management in Clouds: Survey and Research Challenges. *Journal of Network and Systems Management 23*, 3 (2015), 567–619.

[4] MELL, P. M., AND GRANCE, T. The NIST definition of cloud computing. Tech. rep., National Institute of Standards and Technology, Gaithersburg, MD, Gaithersburg, MD, 2011.

[5] PIRAGHAJ, S. F., CALHEIROS, R. N., CHAN, J., DASTJERDI, A. V., AND BUYYA, R. Virtual Machine Customization and Task Mapping Architecture for Efficient Allocation of Cloud Data Center Resources. *Comput. J. 59*, 2 (2016), 208–224.

[6] SOLTESZ, S., PÖTZL, H., FIUCZYNSKI, M. E., BAVIER, A. C., AND PETERSON, L. L. Container-based operating system virtualization - a scalable, high-performance alternative to hypervisors. *EuroSys 41*, 3 (2007), 275–287.

[7] TOMÁS, L., AND TORDSSON, J. Improving cloud infrastructure utilization through overbooking. *CAC* (2013), 1.

[8] ZHANG, Q., CHENG, L., AND BOUTABA, R. Cloud computing: state-of-the-art and research challenges. *Journal of Internet Services and Applications 1*, 1 (2010), 7–18.