The Resource-as-a-Service (RaaS) Cloud

Orna Agmon Ben-Yehuda Muli Ben-Yehuda Assaf Schuster Dan Tsafrir
Technion—Israel Institute of Technology
{*ladypine,muli,assaf,dan*}*@cs.technion.ac.il*

Abstract

Over the next few years, a new model of buying and selling cloud computing resources will evolve. Instead of providers exclusively selling server equivalent virtual machines for relatively long periods of time (as done in today's IaaS clouds), providers will increasingly sell individual resources (such as CPU, memory, and I/O resources) for a few seconds at a time. We term this nascent economic model of cloud computing the Resource-as-a-Service (RaaS) cloud, and we argue that its rise is the likely culmination of recent trends in the construction of IaaS clouds and of the economic forces operating on both providers and clients.

> *The single most important proposition in economic theory, first stated by Adam Smith, is that competitive markets do a good job allocating resources.* (Stephen LeRoy)

1 Recent IaaS Trends

Cloud computing is taking the computing world by storm. According to a recent report by Forrester Research[1], the cloud computing market is expected to top $241 billion in 2020, compared to $40.7 billion in 2010, a six-fold increase. What will those clouds look like? Given the current pace of innovation in cloud computing, substantial shifts are bound to occur in how providers design, operate and sell cloud computing resources, and how clients purchase and use those resources.

A transition is beginning in Infrastructure-as-a-Service clouds, from providers selling bundles of resources packaged as server equivalent virtual machines (e.g., Amazon EC2 selling a "virtual machine roughly equivalent to a server with 2-CPU Xeon processors and 2 GBs of memory") to providers continuously selling their clients individual computing, memory, and I/O resources for a few seconds at a time. We call this model of cloud computing the *Resource-as-a-Service (RaaS)* model.

We begin with an overview of three existing trends in the construction, operation, and use of IaaS cloud computing platforms that underlie this transition: the shrinking duration of rental periods (Section 1.1), the increasingly fine-grained resources offered for sale (Section 1.2), and the provisioning of useful service level agreements (SLAs) (Section 1.3). Following each trend to its culmination, and taking into account the economic forces operating on both clients and providers (Section 2), we describe the RaaS cloud (Section 3). We conclude by outlining the challenges and opportunities that the RaaS cloud presents (Section 4).

1.1 Duration of Rent

Before cloud computing, the average useful lifetime of a purchased server was approximately three years. With the advent of web-hosting, clients could rent a server on a monthly basis. With the introduction of on-demand EC2 instances, Amazon radically changed the time granularity of server rental, making it possible to rent a server equivalent for as little as one hour. This trend, of renting server equivalents for increasingly shorter time durations, is driven by economic forces that keep pushing clients to improve efficiency and minimize waste: if you pay for a full hour or any part of it, you will waste half an hour on average over the lifetime of every virtual machine. If you only pay for a full second or any part of it, then you will only waste half a second over the lifetime of every virtual machine.

This trend is moving along, as some providers are already selling virtual machines for less than an hour: Amazon spot-instance prices change as often as every five minutes [1]. CloudSigma[2] also announces new prices once every 5 minutes. Similarly, phone companies have progressed over the years from charging land-lines per several minutes to charging cell-phones by the minute, and then, due to customer pressure, to charging by several seconds and even single seconds.

We expect the cloud trend to continue; eventually, cloud providers will follow the same route as phone companies and sell computing resources for seconds at a time. Such durations are consistent with peak demands which can change over seconds when a site is "slashdotted" (linked from a high-profile website)[3].

1.2 Resource Granularity

In most IaaS clouds, clients rent resources as a fixed bundle of compute, memory, and I/O resources. Amazon calls these bundles "instance types" and GoGrid[4] and Rackspace[5] call them "server sizes". Selling resources this way provides clients with a familiar abstraction of a server equivalent. This abstraction, however, is starting to unravel. Amazon already allows clients to add and remove different "network instances" and "block instances" from running virtual machines, thereby dynamically increasing

or decreasing the I/O-resources available to a virtual machine. CloudSigma offers clients the ability to compose a flexible bundle from varying amounts of resources, similar to building a custom-made server out of different mixtures of resources such as CPUs, memory, and I/O devices.

However, renting a fixed combination of cloud resources cannot and does not reflect the interests of clients. First, as the sizes of servers are likely to continue increasing—hundreds of cores and hundreds of gigabytes of memory per server in a few years—an entire server equivalent may be too large for some customer needs. Second, selling a fixed combination of resources is only efficient when the load customers need to handle is both known in advance and remains strictly constant. As neither condition is likely, the ability to dynamically mix-and-match different amounts of compute, memory, and I/O resources would probably be highly valued by clients.

By extrapolation, compute, memory, and I/O resources will be rented and charged for in dynamically changing amounts and not in fixed bundles. Clients will buy seed virtual machines with some initial amount of resources, and will then deploy an economic agent (described in Section 3) to buy or sell additional resources. The economic agent will make decisions based on the current prices of those resources, the changing load the machine should handle, and the client's subjective valuation of those different resources at different points in time.

1.3 Service Level Agreements

1.3.1 Selling Resources, Not Performance

The prevalent IaaS Service Level Agreements (SLAs) today only guarantee rigid resource availability: either a machine is up or it is down. If a machine is down for long periods of time, the provider might compensate the client with limited service credit for the down time. Amazon, Rackspace[6], and Softlayer[7] work in this model.

Other providers already provide SLA guarantees in terms of minimal actual delivered capacity. CloudSigma guarantees minimal network latency[8], and GoGrid guarantees network performance in terms of packet loss, latency and jitter[9].

The usual SLAs in use today state that the provider provides the client virtual machines with resources equivalent to servers of certain sizes. The performance of the same virtual machine, however, can vary wildly at different times, due to over-commitment [5], interference between virtual machines [11, 15], or other reasons. Thus, there is a discrepancy between what providers provide and what clients would actually like: in practice, what clients care about is their virtual machines' subjective performance.

To bridge this discrepancy, others proposed to base the SLA on client performance instead of consumed resources [6,11,12]. This approach is only applicable where the provider has full visibility into and cooperation from client virtual machines, as is the case in a SaaS, PaaS or private IaaS cloud where all clients are cooperative. Client-performance-based SLAs are not applicable to a public IaaS cloud, where client virtual machines do not cooperate and the provider cannot rely on the clients to tell the truth with regard to their desired and achieved performance.

Therefore, public clouds will have to forsake the approach of charging users a pre-defined sum in exchange for unknown resources and performance, and switch to a market-driven model. In the RaaS market-driven model, clients bid for resources according to their subjective valuations for those resource, thus affecting their prices. Unlike previously proposed models, this economic model can accommodate real-world clients: clients that are rational and cooperate only when it is in their own self-interest to do so.

1.3.2 Providing Prioritized Service

Prioritized service, where different clients get different levels of service, can be found in certain scientific grids. Jobs of clients with low privileges may be preempted (aborted or suspended) by jobs of clients with higher privileges. Although the first clouds did not offer such prioritized service but rather supplied service at only one level, Amazon has since introduced three levels of priority within EC2: reserved, on-demand and spot instances. As in grids, these priorities are relative, so it is hard to explicitly define their meaning. For example, the availability of on-demand instances depends on the demand for reserved instances.

Having clients with different priorities is useful to the provider, since it can provide high-priority clients with elasticity and availability at the expense of lower-priority clients, while simultaneously renting out currently-spare resources to low-priority clients when high-priority clients do not need them. Likewise, different priorities allow budget-constrained cloud clients cheap access to computing resources with poorer availability.

In the next logical step, clients should be able to define their own priority level—their own SLA—individually, choosing from several levels of capacity and availability which are priced accordingly. For example, a private website may settle for 90% availability instead of 99.9%, with longer 95^{th} percentile latencies than a commercial site. This will allow the providers to simultaneously achieve high resource utilization and maintain adequate spare capacity for handling sudden loads.

2 Economic Forces

In the previous section, we surveyed several ongoing trends and tried to surmise where they will lead us next. We now survey the economic forces that are already operating on both providers and clients, causing those trends to continue for the foreseeable future.

2.1 Forces Acting on Providers

As cloud computing continues on the road to commoditization, migration between cloud providers will become easier [9], and competition between providers will continue to increase. Providers will be forced to economize. They will seek to maintain the least amount of supply (computing resources) that can profitably be sold for handling the expected demand (client requests for resources). When the demand exceeds the provider's expectations, it will take advantage of the different client priorities to deprive low-paying clients and sell the same resources to clients willing to pay more.

Increased competition between providers will force them to become more efficient. Constant load changes render static resource allocation inefficient, because as load changes, resources go unused. Providers will therefore seek to sell their clients the available cycles, memory, and I/O-bandwidth that the clients need, when they need it, at market-driven prices, so that no profitable resource goes unused.

Commoditization will also drive the need to differentiate offers. By offering clients the ability to rent resources when they need them, as they need them, at the right price, providers will be able to differentiate their offerings from other providers who only offer server equivalent fixed bundles of resources.

2.2 Forces Acting on Clients

As users move more of their computing capacity to the cloud, they will seek to reduce their costs. Since load and desired performance constantly change, clients will seek to only pay for the resources they need, and only when they need them. The more flexible the provider offerings, the better control clients have over their costs and the resulting performance. Being able to control costs makes it possible for clients to act according to their economic incentives; these economic incentives drive clients to develop methods for efficient resource usage. As providers offer increasingly fine-grained resources and service levels, clients who wish to take advantage of these offerings will start employing computerized agents to negotiate resources and prices on their behalf.

3 The RaaS Cloud

In this section we describe how the trends and economic forces outlined in the previous sections will transform today's IaaS clouds into what we call *Resource-as-a-Service (RaaS) clouds*. The causes and effects leading from IaaS to RaaS are depicted in Figure 1.

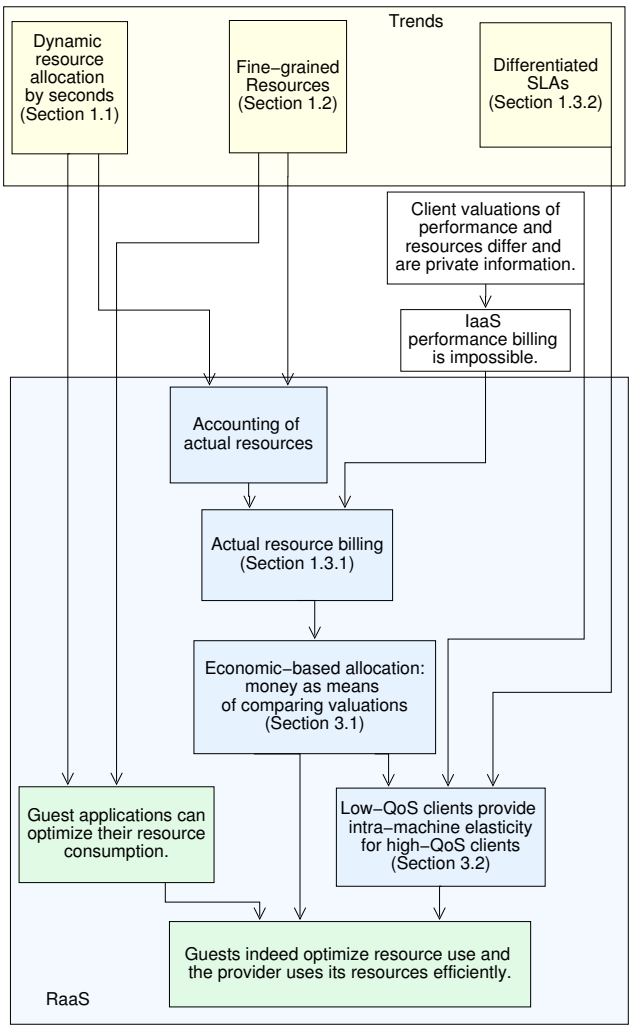

Figure 1: Causes leading from IaaS to RaaS, resulting in efficient resource use

3.1 Trading in Fine-Grained Resources

In RaaS clouds, the client purchases upon admittance a seed virtual machine. The seed virtual machine only has a minimal initial amount of dedicated resources. All other resources needed for the efficient intended operation of the virtual machine are continuously rented and potentially sublet by the client at market-driven prices. The resources available for rent include CPU, RAM, and I/O resources, as well as emerging resources such as accelerators and memristors. CPU capacity is sold on a hardware-thread basis, or even as number of cycles per unit of time; RAM is sold on the basis of memory frames; I/O is sold on the basis of I/O devices with associated I/O bandwidth and latency guarantees. Such devices include network interfaces, block interfaces, and possibly also accelerators such as FPGAs or GPGPUs. Every resource has a dynamically changing price tag attached to it.

To facilitate continuous trading, the provider's cloud software includes an economic agent representing the provider's interests. This agent operates an economic mechanism which defines the resource allocation and

billing (e.g., Kelly [7] and Nathuji et al. [10]). The provider's economic agent decides which client gets what resources and at what price. In addition, it might act as a clearing house for computing resources, as SpotCloud[10] offers to do today for fixed-bundle virtual machines.

To take part in the trade, clients' virtual machines need to include an economic agent as well. This agent represents the client's business needs. It rents from the provider (or from other clients) the necessary resources—given current requirements, load and costs—at the best possible prices. When demand outstrips supply, it negotiates with the provider's economic agent or with other clients' agents, mediating between the client's requirements and the resources available in the system, ultimately deciding how much to offer to pay for each resource at any given time.

Clients can also secure resources early and sublet them if later they do not need them. Such futures markets are nowadays contemplated for virtual machine bundles in the developing IaaS market [2, 14].

Should the provider and clients all belong to the same economic entity (e.g., as might happen in a company's private cloud), then the economic mechanism is not used for actual payments, but still reflects the relative importance of the different clients and the subjective costs of resources (electricity, for example).

For backward compatibility, "dumb" clients without an economic agent are offered fixed resource bundles, same as today, which are purchased upfront at the future-market price. They can easily verify that they meet budget constraints, but they are unable to modify their consumption and thus suffer from either wasted resources or insufficient resources, just as clients do today.

3.2 Prioritized Service Levels

Peak demand in the IaaS cloud can be addressed by bringing more machines online. Therefore, IaaS cloud providers today must hold large amounts of spare capacity (idle machines) to handle surges in load [1]. Moving running virtual machines from one physical machine to another will likely remain less efficient than dynamically balancing the available resources between virtual machines co-existing on the same physical machines. Hence, in the RaaS cloud, fine-grained resource elasticity is limited by the physical resources contained in a single machine. To extend the elasticity boundaries for one client's virtual machine, the spare resources must be taken from another client's virtual machine on the same physical machine. Where providers previously used statically-defined priorities to allocate resources to clients, in the RaaS cloud providers use the willingness of clients to pay a certain price for resources at a given moment in time to decide which client gets what resource; thus market-forces dictate both the constantly changing prices of resources and their allocation.

Market-driven resource allocation can also be used to implement different service levels (SLAs). The provider can cater to the full needs of clients with high-priority SLAs. When supply is insufficient for serving all clients, the provider can starve clients with lower-priority SLAs (e.g., only 90% availability) by raising the price of resources.

Due to the inherent inefficiencies of live virtual machine migration, RaaS clouds must include an algorithm for placing client virtual machines on physical machines. The algorithm composes the right mixture of clients with different SLAs on each physical machine in the cloud, such that high-priority clients always have low-priority clients beside them, to provide more capacity for high-paying clients when their demands peak. The low-paying clients can use the high-paying clients' leftover resources when they do not need them, keeping the provider's machines constantly utilized.

4 Implications, Challenges, Opportunities

The RaaS cloud gives rise to a number of implications, challenges and opportunities for both providers and clients. Broadly speaking, they can be divided into two categories: mechanisms and policies.

The RaaS cloud requires new mechanisms for allocating, metering, charging for, reclaiming, and redistributing CPU, memory and I/O resources between untrusted, not-necessarily-cooperative clients every few seconds. These mechanisms must be efficient and reliable. In particular, they must be resistant to side-channel attacks from malicious clients [13].

The RaaS cloud requires new system software and new applications. Usually, current operating systems and applications are written under the assumptions that their underlying resources are fixed and always available. In the RaaS cloud, virtual machines never know the precise amount of resources that will be available to them at any given second; that requires software running in those virtual machines to adapt to changing resource availability and exploit whatever resources the software has, when it has them. Assume a client application that just got an extra 2Gbps of networking bandwidth at a steal of a price, but only for one second. To use it effectively while it is available, the operating system, run-time layer, and application must all be aware of it.

The RaaS cloud requires efficient methods of balancing resources within a single physical machine, while taking into consideration the different guaranteed service levels. To allow the resource balancer different service levels to work with, workload balancers also require efficient methods of balancing resources across entire cloud data-centers. This is likely to require efficient methods for virtual machine live-migration and network virtualization.

On the policy side, the RaaS cloud requires new economic models for deciding what to allocate, when to allocate it, and at what prices [3]. In game-theoretic terms, these mechanisms should be incentive compatible: truth telling regarding private information should be a good course of action for the clients, so that the provider can easily optimize the resource allocations. The mechanisms should be collusion-resistant: a virtual machine should not suffer if it happens to be co-located with several other virtual machines all of which belong to the same client. They should also be computationally efficient at large scale [4], so that solving the resource allocation problem does not become prohibitive. Ideally, they should optimize the provider's revenue or a social welfare function: a function of the benefit of all the guests. The clients may measure their benefit in terms of starvation, latency, or throughput, but the mechanisms should optimize the impact of those performance metrics on the welfare of the clients, for example by maximizing the sum of client benefits or by minimizing the unhappiness of the most unsatisfied client. In addition, the mechanisms should minimize the price-of-anarchy [8]: the waste incurred by using a distributed mechanism.

In conclusion, making the RaaS cloud a reality will require solving hard problems spanning the entire gamut from game theory and economic models to system software and architecture. The onus is now on us, the cloud computing research community, to lead the way and build the mechanisms and policies that will make the RaaS cloud a reality.

References

[1] AGMON BEN-YEHUDA, O., BEN-YEHUDA, M., SCHUSTER, A., AND TSAFRIR, D. Deconstructing Amazon EC2 spot instance pricing. In *IEEE Third International Conference on Cloud Computing Technology and Science (CloudCom)* (2011).

[2] ALTMANN, J., COURCOUBETIS, C., STAMOULIS, G., DRAMITINOS, M., RAYNA, T., RISCH, M., AND BANNINK, C. GridEcon: A market place for computing resources. In *Grid Economics and Business Models*, vol. 5206 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2008, pp. 185–196.

[3] DANAK, A., AND MANNOR, S. Resource allocation with supply adjustment in distributed computing systems. In *ICDCS* (2010).

[4] DOBZINSKI, S., AND NISAN, N. Mechanisms for multi-unit auctions. *Journal of Artificial Intelligence Research 37* (2010), 85–98.

[5] GUPTA, D., LEE, S., VRABLE, M., SAVAGE, S., SNOEREN, A. C., VARGHESE, G., VOELKER, G. M., AND VAHDAT, A. Difference engine: harnessing memory redundancy in virtual machines. In *Symposium on Operating Systems Design & Implementation (OSDI)* (2008).

[6] HEO, J., ZHU, X., PADALA, P., AND WANG, Z. Memory overbooking and dynamic control of Xen virtual machines in consolidated environments. In *Symposium on Integrated Network Management (IM)* (2009), pp. 630–637.

[7] KELLY, F. Charging and rate control for elastic traffic. *European Transactions on Telecommunications 8* (1997).

[8] KOUTSOUPIAS, E., AND PAPADIMITRIOU, C. Worst-case equilibria. In *Symposium on Theoretical Aspects of Computer Science* (1999), pp. 404–413.

[9] KRIEGER, O., MCGACHEY, P., AND KANEVSKY, A. Enabling a marketplace of clouds: VMware's vCloud director. *Operating Systems Review 44*, 4 (2010), 103–114.

[10] NATHUJI, R., ENGLAND, P., SHARMA, P., AND SINGH, A. Feedback driven QoS-aware power budgeting for virtualized servers. In *Workshop on Feedback Control Implementation and Design in Computing Systems and Networks (FeBID)* (2009).

[11] NATHUJI, R., KANSAL, A., AND GHAFFARKHAH, A. Q-Clouds: Managing performance interference effects for QoS-aware clouds. In *ACM SIGOPS European Conference on Computer Systems (EuroSys)* (2010).

[12] PADALA, P., HOU, K.-Y., SHIN, K. G., ZHU, X., UYSAL, M., WANG, Z., SINGHAL, S., AND MERCHANT, A. Automated control of multiple virtualized resources. In *ACM SIGOPS European Conference on Computer Systems (EuroSys)* (2009).

[13] RISTENPART, T., TROMER, E., SHACHAM, H., AND SAVAGE, S. Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds. In *ACM Conference on Computer and Communications Security (CCS)* (2009).

[14] VANMECHELEN, K., DEPOORTER, W., AND BROECKHOVE, J. Combining futures and spot markets: A hybrid market approach to economic grid resource management. *Journal of Grid Computing 9* (2011), 81–94.

[15] VERMA, A., AHUJA, P., AND NEOGI, A. Power-aware dynamic placement of hpc applications. In *ACM Int'l Conference on Supercomputing (ICS)* (2008).

Notes

[1] "Sizing The Cloud", by Stefan Ried et al., Forrester Research

[2] http://www.cloudsigma.com

[3] "Fifty percent of the time the site is down in seconds—even when we've contacted site owners and they've told us everything will be fine. It's often an unprecedented amount of traffic, and they don't have the required capacity."–Stephen Fry, http://tinyurl.com/StephenFrySeconds.

[4] http://www.gogrid.com

[5] http://www.rackspace.com/cloud/cloud_hosting_products/servers/

[6] http://www.rackspace.com/cloud/legal/sla/

[7] http://http.cdnlayer.com/softlayerweb/SoftLayer_MSA.pdf

[8] http://www.cloudsigma.com/en/platform-details/legal?t=3

[9] http://www.gogrid.com/legal/sla.php

[10] http://spotcloud.com