NSF Workshop on Cloud Economics

Plenary sessions scribed by Supreeth Shastri

1.1 Cloud Competition

David's introduction

Explore the importance of competition in the cloud ecosystem. Topics from this panel include lack of standardization in platforms, customer lock-ins and protection for third-party services.

Ramesh Johari on competition for resource allocation

- Two key questions: can cloud sustain multiple vendors, and what are the financial incentives
- When increasing investments results in increasing returns (e.g. statistical multiplexing in the cloud), economic theories imply industry consolidation. On the other hand, lock-in and increasing switching cost forces diversity of providers.
- Two sides of services on top of services. Incumbent edge (AWS Guard duty competing with third party intrusion detection services) Vs. Acquisition opportunity (AWS/Google buying many service companies). Do these forces balance out or kill innovation opportunities? Unlike in the Apple's app store model, cloud providers here have "data edge" over third-party services that puts them in a better position to build these value-added services by themselves.
- Academic econ research has had limited impact network operations. One option is to move up the stack to work with providers in real-world (quotes his current appointments with Lyft).

David Wentzlaff on economics from architectural perspectives

- Sharing architectures: enabling renting at sub-core level (ALU, caches etc) because de-bundling increases utilization; important to optimize for cost vs. performance (in the cloud)
- Pricing availability: designing flexible interfaces/SLAs and graceful degradation towards increasing everyone's utilization
- Take on competition: Lock-ins are annoying but will solve themselves; chip manufacturers are building wrong chips (performance vs. cost argument earlier) e.g.: Google building TPU

Andrew Chien on lessons from energy markets

- Just as Jeff Bezos says "it is still early days for the Internet", I'd argue don't assume what you see today are going to be what you see in the next 10-20 years.
- Many ways to slice/sell fixed resources: time-based (current on-demand, reserved offerings), continuous auction (Henry George's revocable goods i.e. spot markets)
- Adapting ideas from energy markets: computing and electricity are extremely perishable; electrical energy utilizations in a midwestern ISO ~ 38%; power grid is centralized and its capacities are controlled (because adding new sources or drains alters the balance); energy markets have negative pricing for extra power that gets pushed into the grid; unlike cloud products, electricity is not multi-commodity; demand-response have to adapted in a different way from the energy markets; open (cloud) vs. controlled (electricity);

Follow up questions

Jeff: Energy is fungible but compute should be close to data (and can't be moved easily). How do you solve that?
Andrew: Yes, but many a times, compute is consumed in a more forward looking window.

Hussein: What are the objective functions here (for competition)?
Ramesh: As researchers, we ought to think of overall welfare; how to enable second-best market design (that is grounded in theory but able to accommodate practical constraints);
Wentzlaff: As an engineer, I'm interested in optimizing for customer's cost and provider's revenue.
Andrew: traditional goal of competition is fairness in access and pricing; but in technology, we additionally value support for innovation

Roch: how to design a balance b/w complexity (arising due to competition) vs. ease of use? QoS failed in networks.
David: if there is money people will use it (or someone will build layers/services).
Ramesh: cloud providers have more visibility in their platform's usage and therefore, will increase its complexity to optimize revenue. They would then charge more to make it easier to use.

Bharat: You can't store energy, you can't store computation. Could we store energy using computation? At scale?
Wentzlaff: Is it easier to ship a peta-byte of computation or ship energy required to process a peta-byte elsewhere? The answer is not straight-forward since data is what makes computation valuable.
Andrew: Energy is a low margin business compared to computing. To effectively store energy, you've to find something that is much more valuable later than now. Another angle to look at it, are there computations that we would do only if costs were lower, and then find low energy environment to do it.

Wolski: commodity costs are predictable. As a user, how do you predict your spend in a given cloud platform?
Wentzlaff: any price fluctuation implies that there is opportunity for futures and hedge markets.
Ramesh: it is surprising that this hasn't yet happened in the cloud. You, as a consumer are facing the risk (of unpredictable expenses) and are willing to pay extra to get rid of that risk.

Suchi: To what extent the product differentiation is possible in the cloud? That would improve competition.
Ramesh: Providers have "data edge" and will evolve products to reflect their understanding of the platform. We already see this happening in higher-level services except for IaaS. Another aspect is that of localization.
Andrew: Differentiation is everywhere including IaaS for e.g., Google TPUs. Even applications are driving this to some extent — look at tens of $100M companies building ML/AI chips, they are going to have differences.
Andrew: since we're talking about bringing about changes to enable competition, the biggest impediment is data. NSF should implement a systematic sharing framework like how NIH did with human genome.

Solomon: growth masks all the problems (we're growing triple digit). We got several years before I start caring about the optimal clearing prices for my VMs. Simplicity closes deals faster.
Andrew: Since this is not going to last forever, it is actually a good time to invest in research that would make things optimal.
Ramesh: This also allows academic researchers to experiment more (though I don't see that aspect covered in the topics here).

1.2 Pricing Models and Incentives

George Kesidis on pricing and neutrality in public clouds

- Neutrality becomes an important issue with cloud assuming the status of utility. Competition between cloud providers, between cloud and edge providers. How do users know what they are paying corresponds to the services they are getting?
- Fine-grained pricing is already prevalent with IaaS, PaaS, SaaS, FaaS across different contract types within each. Next steps may be Resource as a Service, bundled pricing, consistent pricing.
- Explains the Sigmetrics 17 paper on burstable instance characteristics, statistical multiplexing of these to realize on-demand instance etc.

Preston McAfee

- Three classic peak load industries (fixed capacity with variable demands) are electricity, airlines and hotel. These are quirky: in electricity, if you don't balance supply and demand, you either blow up a transformer or blow up a consumer appliances. in airlines and hotels, we see integer problems. They have protocols on what to do when they hit the peak: in airlines, they buy back seats (reverse spot market). In hotels, they refuse to host guests.
- In cloud, customers are not as sensitive to price as on reliability and security. In fact, most customers migrating to the cloud just leave their VMs on forever (contrasted that with cloud-native companies like Netflix, who are more frugal with their resource usage).
- Two neglected incentives of public clouds is "spring cleaning", and dramatic increase in security.
- Dynamic pricing does not have to be spot market. Refers to the WWW 2017 paper that shows that time of the day pricing achieves most benefits of the spot markets without the associated complexity.

Shuchi Chawla on algorithmic pricing, and simplicity-vs-optimality

- Can simple pricing structures solve complex allocation problems? For example, can we get the best characteristics of reserved and spot type servers without their associated complexity?
- Proposed scheme: Time-of-the-day pricing with no obtainability guarantees
- Property: if supply is large enough (i.e. if no one customer can overwhelm the provider) then this model can achieve (1-e) of optimal welfare.
- Two things that make our analysis and even system building challenging: (i) in contrast with other peak load industries, cloud allows combinatorial demand. Thus, applications could alter their resource requirements from CPU intensive to memory intensive, or use different resources to achieve same results (ii) varying resource demands over time from the same application.

Follow up questions

Chien: Could you expand on the pipeline problem?

Preston: Some pipelines have regulatory limits on how much utilization they can achieve. Much like speed limits on the cars. This has become cumulative, and for example, the Trans-Alaska pipeline is allowed to operate at only half the capacity of what it used to in the 1950s.

Wentzlaff: How does the pipeline spot market differ from clouds?

Preston: Pipelines follow a sudden death model, where in fixed guaranteed capacities are sold to users but if someone is not using their allocated capacity, it would be re-sold in the spot markets. But when the original purchaser comes back, the spot user experiences sudden death (no warning). It is harder to implement this the cloud since creating VMs take time.

Phillip Afeche: Does the panel think we need better, richer models of demand? Intuitively, it feels different from other industries… is that so?
Kesidis: There are practically nothing on the tenant workloads but some datasets exist for datacenter demand-response systems. Not sure how to alleviate this situation other than simply asking the providers to share data.

Wolski: We are talking about supply-demand. How about externality models that can affect pricing? For example, the US's recent China policy that may alter where customers may buy their cloud compute.
Shuchi: This concern is genuine but it is more of a cloud competition question (for the first panel) because here we're focusing on the stage where customers are fairly locked in, and then designing the pricing models to increase welfare/utility.
Preston: As we are in a growth phase, it is not that much of a problem. Walmart's decision to not use AWS has more to do with them considering Amazon as a retail competitor, than any technical or policy considerations.

Chien: Are there ways to model workload behavior when using IaaS, PaaS, SaaS and FaaS to achieve same thing?
Preston: This is much more rudimentary now. If customers are using a service that just works, they tend to leave their workload there. This is reflected in customers' reluctance to even move to the newer generation VMs that are more efficient, powerful and cheaper. Life is short and everyone is busy. Price is not a driving force at this point.
LaRiviere: My suggestion is not to expect providers to release consumption models at individual user levels. Please simulate customer behavior by taking aggregate demand set, and chopping them up into tiers and making simplifying assumptions.
Solomon: Even if data is anonymized, people may be able to derive information on provider capacities, revenue etc. John's cluster traces clearly shows how big our footprint was at that time.

Timmy Zhu: What do you think about pricing user-defined resources (not just those bundled by the provider)?
Kesidis: This is what RaaS is. Prices will keep going down in the next 5 years. If you're thinking of hedging, please short.
Preston: Most industries either follow human controlled price negotiations, or they will have a fixed set of menus to automatically choose from. However, the cloud may be the first to allow negotiating customized resource procurement (with AI)
Abd-El-Malek: This would result in resource stranding (for e.g., a bunch of cores cannot be rented out since all the associated RAM in the rack is rented already), which in turn need to be factored into loosely offered resources. Neither providers nor customers have strong incentives to go down this rabbit hole, at least for the foreseeable period.
Kesidis: A more realistic way forward on fine-grained resource pricing is through intermediaries and as derivative clouds.

Keith Winstein: What does end-to-end resource utilization looks like in the cloud? How much benefit could one get with sophisticated economic models? Is it 10X? What experiment would help us understand that?
Preston: I don't think anyone has looked at that in the cloud but we know the numbers from airlines. Without dynamic pricing, the airline industry profits would go down by 1-2%. My intuition is that this is because of the law of large numbers.

Preston: End-to-end utilization can be a misnomer. Usually, data and computation are replicated across datacenters for reliability. So, if I have 2 copies running across two datacenters, then maximum attainable utilization per datacenter 50%. I tend to think that 25% is a good target to achieve.
Kesidis: Cloud is different from other industries in that resources are multi-dimensional here (cpu, memory, storage etc as opposed to an airline seat), and the bundling constraints makes it hard to increase utilization.
Preston: I want to challenge this notion that we are somehow more complex than other domains. For e.g., in pipelines, you cannot mix different types of molecules (i.e. fuel types) ‘cause that changes their properties.
Shuchi: Utilization is not necessarily the best metric to optimize on. We should be focused on the value and economic welfare.

Keith Winstein: How much value do you think economists bring in (contrasting with vanilla approaches)? Right now, and in the future, may be 10 years later?
Preston: ~2X for the cloud. Most of the times, huge cost-efficiency comes from minor tweaks. For e.g., in Yahoo, when we switched from archiving the data throughout the day to archiving only at the night, we saved 15% of the overall operating cost. This was because the bottleneck was the network bandwidth overage charges.

1.3 Cost Optimization

Christina Delimitrou on QoS-driven cloud pricing

- How do we correlate the cloud pricing with the performance and quality the users derive from the cloud? This problem is difficult to solve because we expect users to translate their program requirements into cloud resources. Makes this case using Twitter and Google cluster traces, showing their low system utilization.
- Given that the users don’t have great visibility and control over the performance, efficiency and utilization of cloud resources, we can change the interface to that of quality-of-service. This should allow providers to make extra optimizations when lower quality is expected (thereby reducing the prices), and should allow for users to specify their expectations accurately (thereby reducing over provisioning).
- Serverless is a good example in this direction.

Solomon Boulous on real-world cost optimization

- Only the most advanced users are attempting to do cost optimizations. The vast majority of cloud users don’t even take advantage of new generation of servers.
- Google really likes TCO infrastructure, which sometimes mean slower chips, lower density memory etc but it doesn’t affect majority of customers.
- Move to the cloud in itself (i) is a huge savings for customers, (ii) is opening up geographical expansion opportunities, (iii) is making newest technologies accessible that they don’t scramble for complex optimizations (yet).
- Spare capacity product gives most opportunities at optimization. However, early experiments at Google showed that even researchers with no deadline constraints care about getting their results back within a few days. It simply is not worth their time to wait for month for their experiments, even if it is free. Google’s approach has been: don’t be like airlines (hated by people who didn’t buy tickets at the right time). We want to be loved and chosen.
- My request for academic community is to enhance open source frameworks like Hadoop to operate in auto-pilot mode. For e.g., even now, Hadoop cannot take 50% node failure. It simply restarts everything.

Jacob LaRiviere on costly clouds

- Cloud is a "discrete-continuous" model. If you're Dell, you care about whether a customer buys a server or not. But in cloud, once a customer signs up, it is a subscription model. In economics, we have 40 years of research and one Nobel prize on this exact topic.
- On the provider side, customizability of cloud resources results in "packing problem". However, if the economic models don't respect architectural and engineering constraints, they cease to be useful.
- An important takeaway from electricity markets is that automation is the only effective way to scale cost-optimization: programmed thermostats respond to dynamic electricity pricing 3X better than when humans had to intervene.

Follow up questions

Wentzlaff: Fast forward 10-15 years, when the cloud is less emerging. Do you think end of Moore's law will hit hard (i.e. computers stop becoming cheaper) and cloud customers become more savvy?
Solomon: Yes and no (in order). Just like how I don't care about clothes, vast majority of cloud users won't care about complex optimizations. It is possible that we will have sophisticated middleware that automate the process but that's about it.
Wolski: I'd say customer's won't care in the beginning (when they are migrating to the cloud) and then they will start caring at some point.
Christina: End of Moore's law doesn't mean slow down in the demand, it just means that the general purpose chips won't get smaller and faster. But we will likely have specialized chips that do things better.
Solomon: Yes, I don't think the Moore's law ending would alter the growth significantly.

Kamesh: If the packing problem is really hard, and if we could achieve most of the benefits with straight-forward pricing (like we saw in the last panel), then why bother with optimizations?
Christina: Yes, in systems you rarely care about optimality. Since majority of the TCO in a datacenter goes for operational expense, it makes sense to do things efficiently.
Jacob: With good pricing, if you can shave off the extreme behavior then it makes life easier for providers.
Solomon: Yes, cost optimizations like this are second order unless you do something really big.

Roch: I would follow that up with question on trust and interpretability of a complex system. How do customers navigate if things are opaque to them?
Solomon: Yes, we do not offer any SLAs that cannot be independently measured.

Bhuvan: Since it seems like most customers are using their resources poorly, does it makes sense to tell them "hey look here is how your neighbors are faring in the cloud"?
Solomon: It may be a good idea but we don't do that. We do create tools that make it easier for our customers to save more, so that they stay with us in the long run.
Jacob: I'm skeptical that firms would respond to moral signaling. However, we do advise (in an automated way) our customers who have misconfigured or wasting a lot of resources. There are similar quality-of-service in electricity markets as well. For e.g., in Washington DC they offer a 15% discount on your overall bill, if you agree to let the utility throttle your electricity by 20% for up to 5 hours in a calendar year.

Wolski: What are the economic barriers to sharing the data (from a provider)? How may we overcome those?
Jacob: My suggestion would be to parter with cloud customers like Netflix and Walmart, and get their workloads. If you have enough of these, then you could extrapolate the overall cloud workload.
Solomon: I tend to disagree to that. For e.g., Google cluster traces are a bit misleading because it does not represent cloud workloads. Google engineers don't turn on their VM and walk away; there are interesting distributions and tail behavior that is unique to cloud workloads that you may miss if you only rely on sophisticated firms.
Abd-El-Malek: Along with security reasons, it takes a long time to prepare, sanitize and release the data.
Wilkes: Yes, it took about a few months for two engineers to get it out. We never claimed it represents a cloud workload, please read the documentation. We write it diligently. However, it has had good impact with ~400 works citing it. Another aspect is the customer's right to privacy: even accidentally releasing customer's data is antithetical to the business built on trust. For example, GDPR requires us to delete customer's data after a fixed period.
Abd-El-Malek: There are many inadvertent exposures like the Strava heat map that exposed military bases.
Wolski: with all due respect, there is enormous research on how to release data that guarantees privacy and anonymization. My feeling is that it is not just security but also economic tension here having to do with how efficiently you run your datacenters, the power contracts you sign etc that do not want revealed.
Wilkes: One way to think about this is the risk asymmetry. I could put out data from multiple clusters that could, with however low a probability, potentially put a multi-billion dollar business in jeopardy. For what upside actually?
Preston: If we reveal Walmart's use of cloud, we would have broken the SEC laws.
Solomon: I really worry about it when I see Twitter or Snapchat moved to GCE. Any engineer who sees their workload, could estimate how their business is doing before the stock markets actually sees it. This is enormous responsibility.

1.4 Emerging Issues

Jeff Chase on accountable cloud economy

- I want to explore the role of trust in the cloud economy. Not only for the big providers but also the secondary services built on top of the big providers. This is important because we are all tenants now.
- Questions that need to be addressed: (i) how to validate that required security properties are met? (ii) how to assign responsibility in the event of failures? (iii) Could we have an insurance market for this? (iv) how to enable a policy-driven introspection that tenants and regulators can utilize?
- Declarative security and decentralized trust —> cloud attestation and rule-based trust logic. An example is TapCon, a third-party attested Docker containers that include certified build chain and source-code identity.

Rich Wolski on emerging cloud economics

- The talk highlight four things: FaaS, containerization, AI/ML and IoT; and cloud's evolution.
- Emerging trend #1: As a programming environment, the cloud is becoming more heterogenous than unifying. Thought exercise: how many service APIs exist across big providers? ~400 across the big-3.
- Trend #2: Service specialization is moving the cloud away from general purpose hosting to SaaS. This trend definitely help customers who have targeted needs (like chatbots, video tagging etc).
- Trend #3: Open source software is like a gateway drug for the cloud. E.g., Tensorflow, openstack etc. They attract people to the cloud, and in turn makes cloud sticky even if folks move on from their favorite OSS.

- Predictions: (i) public clouds will eventually cannibalize SaaS-only companies (ii) public clouds will become the primary curators of open source (iii) first one to come up with a google search for IoT, will rule the space.

John Wilkes on the big challenges of the cloud

- Two PSA to the research community: (i) please stop fixating on VM rentals in the cloud. The exciting problems are at higher levels, so let's focus on that. As a sub case of this, don't make your students work on EC2 spot markets; it exposes their work to the risk of being irrelevant by internal changes at Amazon. (ii) there were papers (or submissions) at SOSP, where folks are trying game the EC2 lambda system to get cheaper compute than the VMs. They may be fun but are not viable as research contributions. I strongly discourage both.
- This sets us up to the biggest challenge from the provider perspective. If you look at the GCE offerings, there are way more services than VM rentals, and it is not clear how best to offer these services to customers. Pricing, viability, and risks from an economic angle are incredibly hard to get right. We need all your intellectual energy to make it better for both us and the consumers.
- The second problem I want to talk about is building the datacenters. Usually, the building plans start 5 years before the set up goes operational. This kind of timeline is an eternity in the cloud industry, so how to get these decisions right? Google's last 3-year Capex spending was ~$30B. Datacenter infrastructure has many moving parts, different timescale and very big numbers, all of which you've to get right to be successful. So, if you're interested in economics, pricing and forecasting, this offers a rich playground.
- Insight #1: Think about economics of costs and not the economics of prices. The former is fundamental, while the latter is mostly about making profit.
- Insight #2: Relatively small number of customers account for majority of our capacity usage (both internally and externally). We call it the top-6 and the rest. This doesn't play well with Shuchi's problem formulation earlier.
- We would love to give more data; especially if you tell us how would help us benefit from it.

Follow up questions

Kesidis: If the suggestion is not to focus on VM rentals (which easily quantifies the expected performance and capacity), how should users interface with the cloud?

Wilkes: We have internal mechanisms but I'm sure there are better ways to do it. So, instead of answering the question, I would ask you back on what you think. One way to formulate it would be as risk management. Customers have workloads and performance requirements; they could hand it over to the provider and expect them to be met, in which case the provider takes all the risks. Other way is for the provider to simply sell VMs and expect the customer to manage the hosting, in which case the customer takes all the risks.

Wolski: The point made by John was that we're fixated on a QoS bundle called VM because it gives us a direct analogy with physical machines, and has isolation property. The challenge is to figure out how to get the applications to reason about the SLAs and performance without requiring a physical machine to do so.

Roch: Couldn't this be solved by adding profilers?

Wolski: Definitely works for a number of applications, especially scientific which are run repeatedly. But as the Pokemon go application (which tested Google's scalability limits) showed it is inadequate for many others. The promise of the cloud has been that it could serve both types, so it has to be something more than simply profiling.

LaRiviere: How should the cloud enable OSS as an alternative to internal development? What are your thoughts on vendorization in the cloud?
Wolski: Cloud and OSS are two sides of the same coin. Cloud gives a mechanism to distribute, test, and make OSS accessible to a large community. Cloud is accelerating due to OSS and vice versa too.
Wilkes: I wouldn't worry about vendorization. Most interesting things happen on the cloud within the customer applications, we just provide them the tools to do their job. If we start exposing all the tiny internal details, the customers will trip themselves. Besides, it takes away the provider's flexibility to innovate and upgrade underlying infrastructure.
Chase: vendorization is happening because providers want to differentiate from others. But I hope that the model doesn't stifle new companies to build intelligent services on top of the basic offerings from the providers.
Wolski: how much vendorization exists today in the search business? There is one good search engine. But that has not prevented higher-level service providers working on value-added search services.

Klimovic: Could provider's help users determine their utility function in choosing services/resource?
Wilkes: We already provide services like automatically sizing customers' jobs, programmatically suggesting that better fit VMs exist for the workloads etc. If customer's have an idea on the performance they want, we help them identify resources that will get them that. But customer's utility is a hard metric that they have to solve for themselves.

Roch: this looks like a chicken-and-egg problem to me since customer's don't know upfront what kind of resources they need to get the performance they want in the cloud, and providers can't guarantee performances without knowing user applications inside out. Do you propose some form of dynamic profiling and pricing?
Wilkes: It is a hard problem. The current model of charging the customers for instances (in IaaS), puts the risk completely in their hand, or the model of providers taking on users workload (in SaaS), puts the risk in our hand so we have to charge enough to cover worst cases. What I'm encouraging folks to explore is to think about how to manage this better for both parties, what interfaces you'd have us provide, what information could be known up front etc. Don't stop at the first solution you arrive it, try to make it generic so that interfaces could be designed in a way that increases welfare for everyone.
Wolski: I'm hopeful that there are as answers as we move up the stack. If you're running a stochastic gradient application and your data domain is say farms, then I can imagine something like Google autoscaler taking this knowledge and combining with its AI to better manage resources than what you could do by yourself.
Wilkes: I like that answer. Think about how we characterize latency in the cloud. We have zones, and regions. We give you options to choose the redundancy model: no redundancy (i.e. runs within a zone), or N+1, or N+2 etc. Now, how do I publish the latencies at the application level ('cause you could have multiple configurations) and how do I charge for it?

Timothy Zhu: Should customers have the ability to negotiate prices for higher-level services?
Wilkes: Let me draw an analogy. Say, you walk into a local phone store. You won't have much leverage to demand a customized pricing on your phone bills. However, if a fortune 500 firm decides to use our services, we are willing to talk about the prices.
Chase: This clearly reflects the current state of market — it is not commoditized.

Kesidis: You had a graph on cloud providers. How much of Microsoft's rise do you attribute to things like Office 365?

Wolski: I don't want to read too much into the market state of individual provider. However, I will say (from the publicly available information) that Microsoft has made a concerted effort to push its cloud services across enterprises and mobile backend, and are beginning to see its payoff. It is much more healthy than hockey stick growth, which is well suited for open source but not a large scale provider.

Chien: Rich, could you clarify your comments on the connection between SaaS and open source?
Wolski: These two are not strongly related. The first thing is that cloud APIs are going in the direction of SaaS. So, the pure SaaS businesses are going to feel the pressure from large cloud providers. Distinct from that, is the role of OSS. They reap benefits from integration with the cloud and in turn contribute to cloud's wider adaption.

Wilkes: I want to ask Jeff for his thoughts on how to add auditing to the cloud? As a way to build more trust for the customer and to showcase their compliances to regulatory agencies but without having to force the provider to open up about their internal structures.
Chase: There is always tension in auditing. One way would be to have multiple third-party auditors (trust anchors) provider certifications and guarantees. We have understand that these cannot tell how something will work; instead it can only inform how certain things are configured and set up. This doesn't even have to be humans doing it. It is easier to accomplish at the lower levels, say VM monitors and containers by starting at the hardware level (say using Intel SGX).

Solomon: In VM rental, you pay per minute. However, when it comes to higher-level services, say language translation, pricing gets trickier. A Chinese character is one unicode-byte; so if you dump me a 200 MB file and say translate it, I do not know how much CPU/memory I will need to finish the job.
Abd-El-Malek: There are standards for this. Every time we go through lengthy review meetings to determine the pricing for new product launches. This is based not only the internal costs but also on how much value it brings to the customers.
Preston: Yes, this is common in other service industries. When you get your car repaired, they charge a set number of minutes to replace your carbonator. Despite its inefficiencies, it got the biggest advantage that is cost predictability.