CTO Roundtable: Cloud Computing

Our panel of experts discuss cloud computing and how companies can make the best use of it.

Many people reading about cloud computing in the trade journals will think it's a panacea for all their IT problems—it is not. In this CTO Roundtable discussion we hope to give practitioners useful advice on how to evaluate cloud computing for their organizations. Our focus will be on the SMB (small- to medium-size business) IT managers who are underfunded, overworked, and have lots of assets tied up in out-of-date hardware and software. To what extent can cloud computing solve their problems? With the help of five current thought leaders in this quickly evolving field, we offer some answers to that question. We explore some of the basic principles behind cloud computing and highlight some of the key issues and opportunities that arise when computing moves from in-house to the cloud. Our sincere thanks to all who participated and to the ACM Professions Board for making this possible.

PARTICIPANTS

WERNER VOGELS is the CTO of Amazon.com, responsible for both e-commerce operations and Web services. Prior to working for Amazon he was a research scientist at Cornell University, studying large, reliable systems.

GREG OLSEN is the CTO and Founder of Coghead, a PaaS (platform-as-a-service) vendor on both sides of the cloud equation. Coghead sells cloud-based computing services as an alternative to desktop or client/server platforms and is also a consumer of cloud services. The company built its entire service on top of Amazon's EC2 (Elastic Compute Cloud), EBS (Elastic Block Storage), and S3 (Simple Storage Service). Previously, Olsen founded Extricity, a company that provided business-to-business integration.

LEW TUCKER is CTO of cloud computing at Sun Microsystems. In the 1980s he worked on the Connection Machine, a massively parallel supercomputer, which got him interested in very large-scale computing. He then spent 10 years at Sun as VP of Internet services running Sun's popular Web sites. Tucker left Sun to go to Salesforce.com, where he created AppExchange (<http://www.salesforce.com/appexchange/>), and afterward went to a start-up called Radar Networks. Recently he returned to Sun to lead its initiative in cloud computing.

GREG BADROS is senior engineering director at Google, where he has worked for six years. Before that he was chief architect at Infospace and Go2Net. He earned his Ph.D. from the University of Washington in constraint algorithms and user experiences.

GEIR RAMLETH is CIO of Bechtel, where he provides cloud services for internal company use. Prior to his current job, Ramleth started a company inside Bechtel called Genuity, which was an early ISP and hosting company. Genuity was later sold to GTE.

STEVE BOURNE is CTO at El Dorado Ventures, where he helps assess venture-capital investment opportunities. Previous to El Dorado, Bourne worked in software engineering management at Cisco, Sun, DEC, and Silicon Graphics. He is a past president of the ACM and chairs both the ACM Professions Board and the ACM *Queue* Editorial Board.

MACHE CREEGER (moderator) is principal of Emergent Technology Associates, where he provides marketing and business development enterprise infrastructure consulting for large and small technology companies. Beginning his career as a research computer scientist, Creeger has held marketing and business development roles at MIPS, Sun, Sony, and InstallShield, as well as various start-ups. He is an ACM columnist and moderator and head wrangler of the ACM CTO Roundtable series.

CREEGER Let's begin the discussion with a general question and then dig down into some of the deeper issues. How would you define cloud computing?

TUCKER Cloud computing is not so much a definition of a single term as a trend in service delivery taking place today. It's the movement of application services onto the Internet and the increased use of the Internet to access a wide variety of services traditionally originating from within a company's data center.

BADROS There are two parts to it. The first is about just getting the computation cycles outside of your walled garden and being able to avoid building data centers on your premises.

But there's a second aspect that is equally important. It is about the data being in the cloud and about the people living their lives up there in a way that facilitates both easy information exchange and easy data analysis.

The great search tools available today are a direct result of easy access to data because the Web is already in the cloud. As more and more user data is stored in the cloud, because there is a relatively high-bandwidth connection to all those bits, there is a huge opportunity that transcends just computation being off-premises.

TUCKER Tim O'Reilly's definition of Web 2.0 was that the value of data significantly increases when a larger community of people contribute. Greg [Badros]'s characterization complements that nicely.

VOGELS It's not just data. I also believe that clouds are a platform for general computation and/or services. While telcos are moving their platforms into clouds for cost effectiveness, they also see opportunities to become a public garden platform. In this scenario, people can run services that either extend the telco's services or operate independently. If, for example, you want to build an application that has click-to-call or a new set of algorithms such as noise detection in conference calls, then you can run those services connecting to the telco's platform. The key is having execution access to a common platform.

Because we have a shared platform, we can do lots of new things with data, but I believe we can do new things with services as well.

RAMLETH We never defined the Internet, and it became extremely successful. Cloud computing is the computing side of the Internet available on a public or private basis. Let us not define it and limit its possibilities.

TUCKER I see it as three layers: SaaS (software-as-a-service), which delivers applications such as Google Apps and Salesforce.com; PaaS (platform-as-a-service), which provides foundational elements for developing new applications; and IaaS (infrastructure-as-a-service), which is what Amazon has led with, showing that infrastructure can also be accessed through the cloud. I believe it is in this infrastructure layer—in which we've virtualized the base components of compute and storage, delivering them over the Internet—where we have seen the fundamental breakthrough over the past two years.

RAMLETH It doesn't matter who provides software services. What software-as-a-service means to me is that the guy who is actually writing the software is also running and operating it. It could be Google, Salesforce.com, or Bechtel if I have some custom applications, running inside or outside the company.

VOGELS I see software-as-a-service as a precursor. Tim Chou, former president of Oracle Online, wrote a great book called *Seven* about different business models related to, among other things, SaaS. The title comes from seven multibillion-dollar, publicly traded companies that have been delivering software-as-a-service for decades. The book is especially interesting around the business models it explores—specifically, how business models evolve, how companies both build and provide their software, and how customers use their software. Understanding cloud computing requires a look at its precursors, such as: SaaS before it became this platform-like environment; SOA (service-oriented architecture); virtualization (not just CPU virtualization but virtualization in general); and massively scalable distributed computing.

These were technologies that we needed to understand fully before cloud computing became viable. We needed to be able to provide these services at scale, in a reliable manner, in a way that only academics thought about 10 years ago. Building on this foundation, we have now turned these precursors into the commercial practice of cloud computing.

TUCKER A handful of companies, such as Amazon, Google, and Yahoo, demonstrated the advantage of very, very large scale by building specialized architectures just to support a single application. We have started to see the rest of the world react and say, "Why can't we do that?"

CREEGER Or, "Why can't we leverage what they have built?"

RAMLETH Not only leverage, but how can they teach us. At Bechtel we studied 18 of those companies to figure out if we could apply what they learned to our business. We wanted to understand how to build new economies of scale to support our service offerings in a much more flexible, reliable, and cost-efficient manner.

BADROS While I agree that the emergence of the massive scale of these companies plays a critical part, I also think that the development of client-side technology such as HTML, CSS, AJAX, and broadband connectivity is very important.

CREEGER What about virtualization? It provides an encapsulation of application and operating system in a nice, neat, clean ABI (application binary interface). You could take this object and put it on your own premises-based hardware or execute it on whatever platform you choose. Virtualization makes execution platforms generic by not requiring the integration of all those horrible loose ends between the application and the operating system every time you want to move to a new machine. All that is required for a virtualized application/operating-system pair to execute on a new platform is for that platform to support the VM (virtual machine) runtime.

TUCKER An important shift has been to use basic HTTP, in the form of REST (representational state transfer) APIs (http://en.wikipedia.org/wiki/Representational_State_Transfer), as an easier-to-use SOA framework. Everything that made services hard before, such as CORBA (<http://www.omg.org/gettingstarted/corbafaq.htm>), IDL (http://en.wikipedia.org/wiki/Interface_description_language), etc., went away when we said, "Let's do it all over HTTP."

RAMLETH We broke timeshare into either private or public, because people wanted the user experience. We broke it by introducing PCs to get it. Now with a thin client you get that user experience back again. It's user experience that is driving this. AJAX, HTML, and other new

protocols have brought back the user experience.

TUCKER We computer scientists got out of the way. The most interesting APIs that started to emerge on the Web were put together for very application-specific purposes. The Flickr photo-sharing service APIs were very simple and pragmatic so that anybody who knew a little bit of JavaScript could easily consume the service. This pragmatic approach toward a service-oriented architecture was an important precursor to cloud computing.

VOGELS While applications driven by AJAX and HTML are important, cloud services are just as much integrated into fat-client applications. I can use Adobe Lightroom and publish to Flickr, Picasa, or wherever. It doesn't force me to use either a fat or thin client.

RAMLETH But that enables you to get the user experience. It doesn't matter what technology; it's user experience that people want.

BADROS That is what is really important in contrast with timesharing and thin clients of the past. Now we get to balance how much of the computation is done at different tiers of the hierarchy that either ends in the cloud or extends it out. No longer is it just a dumb thin client but an additional computational platform. We are just at the beginning of developing the programming model that facilitates the transparent movement of applications throughout this hierarchy of platforms up to and including the cloud.

VOGELS The location where things happen is important. If you start developing an application, you want to minimize risk. One interesting application that I recently saw from the data group of Nasdaq called Market Replay (<https://data.nasdaq.com/MR.aspx>), recalculates earlier market movements. A client would note that the billing for a trade executed six months earlier was 2 percent higher than the quote made at the time of the trade. Investigating a past trade required Nasdaq to run expensive and unprofitable ad-hoc queries against the database.

The stock exchange received so many requests over time that it wondered whether the recalculation of earlier market movements could be a profitable service. To minimize development risk, however, it did not want to make any capital investment in infrastructure to support that effort. It chose to start storing all exchange information into Amazon S3 next to its own databases.

Nasdaq's goal in providing this service was only to bear development costs. It wanted the client's desktop to do all the computational work and the data had to reside somewhere other than the exchange. It developed a Flash application out of components it already had and made the application available for download. Clients would run the application, and the data to do the recalculation was made available from S3 through a fee arrangement. Given the goals of zero additional investment, Nasdaq thought really hard about where the CPU cycles should execute. It decided that the application should run on the client's desktop because it didn't want to invest in the required infrastructure to execute those cycles at the Nasdaq server site.

BADROS Yes, that's our point. You get to pick all the sites. Thin clients and timesharing didn't give you that choice. That is one of the reasons cloud computing is so powerful.

TUCKER Using the cloud as a powerful back end for devices opens up whole new areas for innovation. An interesting example of this is Shazam, an application that runs on my Apple iPhone. I can hold my iPhone up to a radio playing a song. Shazam samples that song and sends the sample up to the cloud where it is matched to a large library of songs. Shazam then tells me what song I am listening to so I can then go and buy it. This is where we are being smart about where computation occurs. We're using a lot of heavy-duty computation on the server side coupled with a smart device.

BADROS As devices get smarter, the fingerprint may well be done on the client side, and then you'll just send the fingerprint up to the server to save bandwidth.

TUCKER Sometimes you might want computation to move to the data to minimize bandwidth issues.

CREEGER What you just said is very astute and appropriate for cloud computing: Where does computation need to occur in relation to data?

RAMLETH I don't think it necessarily has to be close to the data. It has to be close to where that data is massaged. We have some applications where we do project controls. Large multibillion-dollar projects produce a ton of information. While I can analyze the data from the server side, sometimes when client/server bandwidth is limited, it is better for me to download the data and extract it to the client so analysis can be done locally.

For a project in Equatorial Guinea, everything had to be done over a satellite. For that situation it made sense to move the data to a location that afforded better accessibility. We write applications so they can execute where it is most effective for the end user. If end users feel like they get a better rate of response by doing it locally, then they do it locally. If there is no constraint on the bandwidth, then you do it where the data resides.

BADROS Our challenge is figuring out how we can build the application once so it can configure itself automatically in realtime to operate most effectively based on its deployment parameters. We can't build it to be local and have the user or someone make an ad-hoc compensation for a specific deployment scenario. We need to figure out how to build the application so that the right things happen automatically during deployment.

OLSEN The granularity of the things that live in the cloud is much coarser than envisioned in some of the earlier distributed Internet infrastructure approaches such as Sun's DOE (Distributed Objects Everywhere; http://en.wikipedia.org/wiki/Distributed_Objects_Everywhere) CORBA effort. Instead of many types of interacting distributed objects, with complex state relationships arbitrarily invoking each other, we see very loosely coupled services and simple interactions.

When I first heard about the Amazon SQS (Simple Queue Service; <http://aws.amazon.com/sqs/>), I was just baffled by how it would be useful to folks. I had been living in the world of middleware and IBM WebSphere with my prior company, where the focus was on completeness of capability. All of the excitement we are seeing in cloud computation has been driven by a simple, pragmatic model of component interaction.

CREEGER Those of us who came from an academic computer science background (that might be all of us) were taught to define algorithms and architectures to be theoretically complete. While I do not want to dismiss the importance of completeness, I do think that in some cases it has been pursued at the expense of practicality. In business today there are certainly some things that don't need to be theoretically complete. They just need to provide specific practical services required by the user.

VOGELS We can only build very large-scale services based on very solid principles, and simplicity is one them. That list includes symmetry, asynchrony, and many other things, but of all those principles, simplicity is probably the hardest one to implement.

As soon as complexity increases, things become much harder to scale and generalize. If you start with something simple, everyone may not be happy, but people are pretty resilient and will build stuff on top of the basic platform to compensate for what's missing in the framework.

TUCKER A revolt against bloat develops over time in a lot of very large, long-lived software products.

This results from the addition of evermore features in an attempt to satisfy every customer, making applications grow bigger and bigger. Emerging in cloud computing today is a reductionist principle, going back to simple, understandable elements that can be composed in a robust, scalable way.

BADROS The interaction between complexity and scale is at the heart of the issue. If you've got something that's going to fail one in a million times, it's fine if it's a single-user system: it's just one person, and one day in a million it acts a little wonky. The minute you deploy it for tens of millions of Gmail users, however, it becomes a real problem. If a service is failing even one in every 10 million times, you now have thousands of screaming users to deal with.

VOGELS We see this in S3. If you do trillions and trillions of operations, then a bug that has even the smallest probability is a certainty.

BADROS We talk about the number of hard drives that are failing per minute in our data center. It's not about the MTBF (mean time between failure); it's the rate of hard-drive failures in the data centers.

TUCKER How do you train people to build applications in the cloud? I think we have just answered it. We need to learn that component failure is inevitable, yet the application needs to stay up. Developers designing scalable applications should be aware that functionality will disappear or malfunction, and this must be planned for and protected against.

One interesting aspect we haven't yet talked about is the economic advantage that we are seeing in cloud computing. End users and developers are taking advantage of a pay-as-you-go model to avoid upfront capital costs. When you start using cloud services, whether it's at the application level or infrastructure, you start to get a closer correlation between actual service use and its associated cost.

BOURNE OK, so let's be practical. What are the economics of clouds? What's the CapEx (capital expenditure) and what is the OpEx (operational expenditure)? At the end of the year, did I spend more or less?

VOGELS I'm not convinced it's either more or less. CapEx forces you to make massive investments. In the past 10 to 15 years it has been very hard to predict software product success. Previously, as an enterprise, you could kind of predict what your next-generation products were going to be—the same set of customers, things like that. Today there's a proliferation of products and intense competition for customers. In the past, you had some measure of control over your customers; these days your customers have control over you. They know what to choose and have perfect information. So if you build products today as an enterprise, but also as a young business, you have no idea whether you're going to be successful or not. The less investment you have to make upfront, the better.

The important thing is that you build your architectures and systems such that your expenses follow your income. If your expenses are going to be in the number of videos delivered, but you make money on some completely unrelated metric, then you're toast.

I find that younger businesses are more focused on cost, but cost is only one of the factors. The feedback I get from most enterprise customers is that flexibility and access to resources in a very short time frame are actually as important as cost.

OLSEN What inspired me about the cloud was that I could start a company and not buy any servers, phones, or software licenses. We were dedicated to using cloud services from day one. We started our company relying solely on services for e-mail and Internet and went from there to putting our source control on as a service. I wrote an article titled "Going Bedouin" where I expressed these

views in more detail (<http://www.webworkerdaily.com/2006/09/09/going-bedouin/>).

I see examples everywhere in which software and hardware infrastructure are an unnecessary burden. My local bike shop is a good example. The owner puts significant energy into dealing with an ancient version of Windows to support an ancient version of some Microsoft DOS-based application that manages inventory for bike shops. For medical care, I go to a large clinic that is affiliated with a larger health-care company. It has a Microsoft Windows server with Citrix on top running on some ancient VB (Visual Basic) legacy application. Cloud computing provides a path to avoid all that inconvenience.

BADROS Clouds are clearly a huge win to get started with a business or product offering. At Google, we see internal people using the GAE (Google App Engine; <http://code.google.com/appengine/>) as a means of deploying something very quickly before they worry about scaling it on our base infrastructure. People do this because it is so much faster to get going, even inside Google where you have lots of infrastructure available.

Today's developer has a decision to make: after I am a success, am I going to switch off of this initial platform? That's the trade-off. Once it's obvious that something like an Amazon S3 is able to outperform the best that the vast majority of companies can ever deploy, then it's obvious you should just work entirely within the cloud. In this way you never have to suffer the replacement CapEx for the initial infrastructure.

TUCKER At the end of one or two years your overall costs might be roughly equivalent, but you will realize significant time savings, lower opportunity costs, and direct savings in staff and management costs.

VOGELS For many customers, using our cloud products requires new expertise. You are no longer looking for a typical system administrator. You're looking for someone, if you have a large company, with the specific expertise to support 50,000 internal customers. Using the cloud, you no longer have to deal with things at the physical level. You no longer need to have people running around the data center replacing disks all day.

RAMLETH You can get your smartest guys to work on what matters most rather than having them work on mundane stuff. That's a huge benefit.

TUCKER Don MacAskill, CEO and founder of photo-sharing start-up SmugMug says he doesn't want to run data centers anymore (<http://blogs.smugmug.com/don/>). He would much rather have his best engineers focus on highly valued product issues than on undifferentiated data-center operations.

CREEGER What about the people who need to run a flat-load, basic accounts receivable package? Once they get their software and hardware in place and get their operational process down, it's pretty straightforward and they can amortize the capital expenditure over a very long time period.

TUCKER Every three years they've got to upgrade the software and the hardware.

CREEGER Do they?

RAMLETH You have to because the software vendors are forcing you. We spent \$5 million last year on an upgrade that did nothing for our business processes or end users. The software vendors told us that if we did not upgrade, they would stop supporting us. That doesn't happen in other industries, such as the car industry. Chevrolet, even with all its problems, will not say, "You have to buy a new car because we're not going to support your existing car any longer."

OLSEN I always wondered why we think software is so different from anything else. If a restaurant was growing its own food, slaughtering its own animals, generating its own power, collecting

rainwater, and processing its own sewage, we would all think they were idiots for not using ready-made services. For a long time people built their own stack from the ground up, ran their own servers, etc., because they could. Viewing the state of our industry, any student of economics will tell you that you have to start layering. You have to take advantage of efficiencies of scale and build value-added on top of what other people have produced.

TUCKER The market typically drives people to specialize in being the most efficient deliverer of some service, whether it's supplying groceries, meat, etc. In Nicholas Carr's (<http://www.roughtype.com/>) argument about the movement toward utility computing, it is as if we are back in the days when everybody is running their own power generator. It's undifferentiated and it doesn't mean you can brew better beer. Running your own power generator has nothing to do with the quality of your final product.

VOGELS There are restaurants that do not buy their own herbs; they grow them on-site. They would argue that it contributes to the quality of the end product. They will never generate their own electricity, however, because that will not produce better food.

RAMLETH They might have a restaurant that sits on the top of a mountain where they have no power and they have to generate their own because they want people to have that experience. The thing is, it all depends.

OLSEN Realistically, however, software is really extreme in terms of how many people are doing undifferentiated tasks, on their own, at all kinds of levels. Look at the auto industry: there are many tiers of subcontractors, each providing specialized services and products. We just haven't evolved to that same level of efficiency.

RAMLETH We have dramatically reduced our data-center capital expenditures as a direct result of virtualization, allowing us to reuse our capital many more times than we ever could before. Before we started our effort, the average server utilization in our global server park was 2.3 percent. Going to virtualization has increased it to between 60 and 80 percent.

Regarding capital expenditure reduction, when we started, the core side of our central data centers, not including peripheral things, ran 35,000 square feet. In the early 2000s, we consolidated and got it down to 20,000 square feet. Then we virtualized everything, and the equivalent of those 35,000 square feet is now operating in less than 1,000 square feet. We are utilizing our hardware in very different ways than we could ever do before.

We had to go through the earlier painful way of doing things. Companies such as Google and Amazon got started at that level so they didn't have to go through that kind of pain. The lesson we learned is that a very big part of building these public and private clouds is to be sure that you can get utilization factors significantly better than traditional company operations.

VOGELS If you run your services inside the company, privately, utilization becomes an issue. It amortizes your costs over a number of cycles. If you run services outside, on a public service, it is no longer an issue for you.

RAMLETH If you are in a transitional company, you cannot say that you want to do A or B; you have to do both. At Bechtel there are applications that must be executed internally for legal or other requirements. We want to learn from Google and Amazon how to run them at those high levels of utilization—even inside the company, if that is the requirement.

VOGELS One of the first things that customers who have been successful with our platform do is think how they can become more horizontally scalable, incrementally scaling their standard mode

of operation as their demands grow. As they get more customers and their data set grows, they can apply more resources and support the next level of demand.

The second thing that I see with successful customers is automation. One of the costs of running data centers is having lots of people managing legacy applications. When folks move into the cloud, they start thinking about how to invest in their software so they can better exploit the properties of the cloud. The smart ones go to a much higher level of automation.

Before they move to the cloud, they may have automated the placement of virtual machines on the hardware, but they still had folks running around manually tweaking things. When they move into the cloud, customers truly start thinking about how they can be really efficient and automate the heck out of everything. Since you no longer have an IT person to call, you automate that entire process.

RAMLETH We are operating hundreds of servers that are processing data for projects that no longer exist and are no longer generating revenue. We do this because there may be a time and place where we would need this information, as in a warranty situation.

Amazon taught us that we can move these programs from our data center to EC2, get them operational, capture that image, and then shut it down. At this point we have incurred very minimal costs. When conditions arise that require the execution of one of those programs, we can do it. We don't need the results that instant; if we have it available in two or three minutes, or even in six hours, it probably would be OK. By using Amazon EC2, we can transform what used to be a fixed cost of allocating a dedicated in-house server—regardless of whether we need the information—to a variable cost that is incurred only when the business case requires it.

BOURNE That is a very interesting point. What is the contract for binary compatibility from vendors like Amazon? If I give you a binary today and I run it today on EC2, what is the commitment that it will still execute in the future?

VOGELS You don't give us a binary. We offer a number of virtual machines to choose from: a number of popular Linuxes, including Red Hat Enterprise Linux and Oracle Enterprise Linux, OpenSolaris, and Windows Server 2003. First you run the software on your own virtual machine at home or in your business. When you are completely happy with it, you freeze it, make an image of it, and send it over to us.

RAMLETH Our scenario is first to move it to Amazon, make sure it runs well on EC2, and then shut it down.

BOURNE Is Amazon going to guarantee to run that in 10 years?

VOGELS We have chosen a virtual machine model. If you can run on this virtual machine, it is of no importance to us what you do inside.

RAMLETH We rely on being told when a major change occurs so we can do the proper conversion in a timely way. The cost savings of using Amazon is quite compelling. A basic server, operating internally, that sits and does nothing costs us about \$800 to a \$1,000 per month to run. We can go to Amazon and get charged only for what we use, at a rate of 10 to 15 cents an hour.

This is an opportunity for my department to provide platforms to applications that have a clear requirement to remain in-house. For needs without those requirements we can go to outside vendors such as Google and Amazon and purchase applications, as well as platforms as is practical.

TUCKER This is the promise of utility computing. Users will be able to move their applications and their platforms off-site, and they will have more choices. There will be many different kinds of cloud

service providers and, ultimately, opportunities for arbitrage. We are moving to a scenario where it will not matter where things execute, and where choosing an execution platform will be based on a number of different characteristics such as cost, security, performance, reliability, and brand awareness.

CREEGER So before, you had to build to peak demand and put that capital expenditure in place, and you could never shut it down. You had to bear that peak-demand expense.

RAMLETH And if we put it up there and a higher peak came, we had to enhance its capacity to that new peak.

CREEGER And you had to continue to staff that infrastructure, support its software, hardware, and everything else. With cloud computing you can go over to Amazon and say, “OK, I prototyped it, it’s set, and it works fine. Whenever I need it I can run it for pennies on Amazon’s infrastructure environment.”

TUCKER The great thing is that self-service has now moved into the provisioning of virtualized compute, storage, and networking resources. Without even talking to anybody at Amazon, you can use its service with just a credit card. Enterprise customers are looking at their internal customers the same way. If the marketing department now wants to run a new kind of application, traditionally you had to get the IT department to agree to help you build and deploy that application. Now IT departments are able to say, “You’ve got your own developers over in your area. If they want to develop and run this, fine, go ahead. Here are the policies for infrastructure services.”

RAMLETH We talked about the services that are running to support things that might be in warranty. We want people to be able to go to a Web site and say, “Fire up this project.” They choose their service, get provisioned on Amazon, and then get a little hourglass that has them wait a few minutes—and my internal IT department is never involved.

TUCKER The way that you get economic advantage is by sharing the resources and purchasing power of very large-scale, multitenant data centers.

CREEGER One guy’s trough is another guy’s peak.

BADROS One of the key benefits is that not only is it easier to get going at start up, but also there is no discontinuity as things grow. It’s never the case that you are debating internally whether you should buy that extra server, invest in a more sophisticated infrastructure, or be able to scale to that second machine. There are none of those discontinuities.

TUCKER We need to be a little careful. Not all applications scale easily. While there is a whole class of applications that have very easy scaling characteristics, others do not. Databases are part of this class unless you are using something that has been set up to scale, such as Amazon’s SimpleDB (<http://aws.amazon.com/simpledb/>). If you’re running your own database, unless it has been designed to be scalable, don’t count on it happening. We don’t want to lead people into thinking that all applications scale without question.

CREEGER How does that poor person sitting at a small to mid-cap company make a decision to invest in clouds? What is he going to do next quarter or next year when the CEO comes in and says, “I read this thing in the *Wall Street Journal* stating that all the smart companies are going to cloud computing. What exactly are we doing in cloud computing and how can we get rid of all this CapEx that you’ve got here?” How is this guy going to respond?

OLSEN First, adopt a philosophy of buy first, build second—even at the basic level of *I’m going to start a company, I need IT services. Do I go look to hire engineers and buy equipment or do I assume that there’s*

some outside service that might meet my needs? To me that's half of it. I'm going to assume that services that meet my needs are already available or are going to evolve over time. I take a philosophy that says, "I'm all about my core business. I buy only infrastructure that directly supports my unique contributions to the marketplace."

RAMLETH I agree with you if you are rational. You're dealing with humans, however, and they are often not rational. When a CEO goes down to his IT manager's office and asks, "How are we utilizing cloud computing?", the first thing that manager asks is, "What will this mean to me?" The biggest obstacle to change at our company was our own IT guys trying to protect their jobs. The change we have done at Bechtel has been 20 percent technology and 80 percent managing the change.

TUCKER There are certainly different approaches for different businesses at different points in their life cycles. A start-up has a certain set of needs. I completely agree with Greg Olsen to look for all the services that you can purchase before you think of building it yourself.

OLSEN As computer professionals, we need to learn what other industries have learned: look to buy things that are already mature and layer on top of it to be more of a system integrator than a creator. As a business, our key focus in selling to an enterprise is how to find transitional paths. What are the things that you can't do now that a cloud computing solution can? We look for applications that make no sense being locally hosted on a server or desktop that provide value in a short time.

One example is doing supply-chain departmental applications in a large enterprise. In this scenario, a buyer needs to manage material sourcing from a bunch of very small vendors for a manufacturer. Right now they have an ad-hoc process using spreadsheets and phone calls to vendors. Eventually they enter the data in the ERP (enterprise resource planning) system so that the purchase order has the right quantities and the right people get paid.

We built a little application that filters the key information out of the ERP system, lets the buyer do his dynamic updates with the partner, and provides the ability to sync that back to the core ERP system. We give users only the core information they need and let them do something that's pretty agile. We let them make that information available to the supplier so that those guys can also update.

RAMLETH I believe an important part of your value proposition should be to explain to both the decision maker as well as the user how this tool enhances their professional futures. If it does not, those folks are going to be your obstacles.

TUCKER Animoto is a new company that makes movies out of photographs synced with music. It started with 50 instances running on Amazon. They launched it on Facebook and had very high success. In a matter of three days they went to 3,500 instances.

Can you imagine going to your IT department and saying, "We're running on 50 servers today, and in two to three days we want to go to 3,500 servers."? It just would not have been possible. To me, being instantly able to take advantage of an opportunity and easily scale as the demand increases is one of the prime benefits of elastic or scalable computing. Because of this, cloud computing is actually going to drive a lot more computing usage as it more closely tracks economic benefit with the resource being expended.

CREEGER So, for the zero- to a million-miles-an-hour overnight business plan that is stalled because of up-front CapEx costs, cloud computing is going to be your answer.

What other types of criteria can we give to people to evaluate how effective their internal infrastructure is in supporting business goals? How do they identify what services they should start

pushing over to the cloud, and what criteria do they use to make those decisions?

BADROS Replacing existing organization structure or IT functionality is harder in larger companies. Often you have a better chance of success if you introduce something that provides new value, perhaps by enabling a new type of collaboration, rather than replacing or modifying existing functionality. In this way you can avoid the risk of encountering resistance resulting from complexity or politics. In today's tougher economic times, you may also want to make your proposal more compelling by showing that operational TCO (total cost of ownership) can be significantly lowered when using a cloud.

RAMLETH While I agree that scalability is an important attribute of cloud computing, for us flexibility is also very important. For every 100 people who retire from our company, we can replace them with only 60. There just aren't enough qualified folks available. Because qualified people are so scarce, we have to bring the jobs to where the talent lives, so we go to Shanghai, Taipei, Bangkok, Mumbai, Warsaw, and so on.

This means that with very little warning we get requests to set up new engineering resources in totally new places. We don't have the time to build new data centers and new infrastructures; we need to turn them on very quickly. In Warsaw, we had to get a new engineering center up and going in less than 30 days.

In these scenarios cloud computing makes a lot of sense. It's not only scalability, but also breadth. Scalability to me is vertical. It's being able to handle a light or heavy load of the same mix of applications. I also need a wider range of applications, which to me is horizontal. Look at today's vendors of cloud services: Amazon can sell me the scalability, and Google can sell me the range of applications. They both provide complementary solutions to my problem.

TUCKER Is IT obsolete?

RAMLETH You don't want IT to be obsolete. You want it to be a business enabler, not an obstacle.

TUCKER I want IT to be my business partner. IT should be doing things such as capacity planning. If I'm projecting some level of demand, I want IT's advice on how I meet that demand. That's what I think cloud computing is going to start to do. I also want them to be setting corporate policy. They set policy around what we can do and what we can't do, so you have uniformity.

RAMLETH I don't need to do capacity planning; I want to have capacity availability. Planning means that I have some knowledge of what will happen in the future, and I don't have that. I want to be sure that I have scalability and a breadth of applications ready for deployment.

TUCKER For the organizations that cannot go outside their four walls, I believe the cloud-computing model is appropriate for the delivery of IT services even when hardware is limited. By using the cloud-computing model, IT can take on the role of capacity planning by tracking utilization and say, "We are growing at 20 percent per year and need to expand capacity." You lose some of the advantages of scaling to high levels that you have at Amazon, but you can use the public cloud as an overflow capability. The best result is that you have the flexibility to dynamically assign jobs to be run either as overflow or in-house as your internal capacity dictates.

RAMLETH You call that overflow, but it is not. It's pushing back the demarcation line of what you have to host yourself.

CREEGER This goes back to the issue of how to help people make those decisions. What types of properties should they be looking to in order to make the assessment of where to place that demarcation line?

We haven't talked a lot about whether Amazon's offerings are good for some very basic IT services such as accounts payable and accounts receivable—services that are straightforward and that do not grow or change.

TUCKER Running your own Microsoft Exchange server on Amazon is not the most cost-effective way of providing that service when there are providers that are specialized for this service.

CREEGER That's a good point. What other criteria are there to distinguish between running services in-house or off-premises?

RAMLETH We benchmarked around 18 companies when we started thinking about building our internal cloud. I can buy a gigabyte of storage on Amazon S3 for between 10 and 15 cents per month. Our internal corporate charge to support a gigabyte is \$3.75 per month. While it is not an apples-to-apples comparison, the delta between 15 cents and \$3.75 is too big to ignore. Also during this time our internal cost for fully managed network bandwidth was \$500 per megabit per month, down from \$3,500 two years earlier. Because YouTube can send millions of messages a day for free with a little bit of advertising, we assumed that they must be paying in the teens. Why was the delta so large between us at \$500 and YouTube in the teens?

When we changed our network philosophy and started doing the networking differently, we achieved, at a minimum, a 50-percent latency reduction worldwide. Our bandwidth costs went from \$500 to YouTube levels. When our network costs dropped to one-fiftieth of what it was earlier and latency improved, we were able to do business in fundamentally different ways. How did we do it? Simply put, we brought the data to the network rather than bringing the network to the data. We started placing the data very close to the network provider's Internet exchanges. Both Amazon and Google have done it from the beginning, but that is not what the enterprises are doing.

CREEGER What recommendations would you make as to how a company should evaluate whether or not this could work for them, since a lot of enterprises are not going to be as big as Bechtel?

RAMLETH You can be very small and still do what we did.

VOGELS If I look at really successful companies, whether they're Bechtel or much smaller, they all start at a very small scale. You don't need much to get started.

OLSEN The assumption that it's central IT making decisions about other technologies is wrong. Cloud computing has become successful not because a whole bunch of central IT groups proclaimed that cloud computing is good. Cloud computing has become popular from grassroots acceptance, from IT decisions made by small businesses, new providers, or at the departmental level. Cloud computing is coming into IT only at the end of this. My company does not sell to CIOs. We don't even try.

CREEGER That's fine, but there are CIOs who will have to provide plans after their CEOs read that one can realize massive savings with cloud computing.

VOGELS There are many first steps that corporations take into this world. Engineers can start by experimenting with these services, using them for small projects and comparing cost savings. I find that many of the first steps that enterprises take are just something small, easy, simple, and cost effective.

The *New York Times* scanned images covering a 60-year period in history and wanted to place them online. These guys moved four terabytes into S3, ran all the stuff on a Sunday, spent \$25, and got the product done. Another example is from the U.S. government where the expected prototype

was to cost \$30,000, and doing it with Amazon was something like \$5.

OLSEN Today, the majority of commercial cloud-computing projects do not originate through the CIO. Yes, we have to get there, but most of what is happening right now is from other points of entry. Part of the question is who is the target? IT is one target, but there are many, many others. Most computing needs to me are not about central IT. Millions of small and medium businesses, our primary customers, are little solution providers that don't have IT departments.

BOURNE So who should pay attention to cloud computing? You just gave one example. Is that the only one?

OLSEN I'm either a consumer of information technology needs: I need applications, I need storage; or I'm a producer: I'm somebody who's going to provide a service. Both of those audiences need to know what they can build from and how they can sell what they have. To me, it's not primarily about central IT. Central IT is an important constituent, but all these little system integrators, consultants, little ISVs, VARs—these are the folks who actually deploy computation on a broad scale to businesses and people. Any person who is in that space, either as a producer or a consumer of IT, needs to understand how to use cloud services.

TUCKER If I am a developer in the sales or marketing department of a corporation and I am given the task of creating a Web site, a collaboration environment, a test and deploy environment, or whatever, cloud computing is the ideal candidate platform.

Instead of taking the traditional approach, now you start by thinking about what you are trying to deliver to your customers and focus on the application. One of the last questions should be: Where does this need to execute? That can be answered based on the economics.

BADROS To me, the value proposition of cloud computing is so broad that the beauty of it is you can sell to almost anybody in the organization. Different aspects of the solution appeal to different sets of folks. Depending on whom I'm talking to, the story is different in order to let them see how it's going to be better for them.

The individual who has been using consumer e-mail and Google Calendar is excited about having the home experience at work and about the rich search capabilities and collaboration of Calendar. We see people using docs and spreadsheets to manage their wedding on the docs collaboration suite. Then when they are doing a similar type of project at work, they don't understand why they are stuck in early '90s-style thinking with a set of applications that don't talk to one another. For that person, the collaboration story is the value proposition.

If an enlightened CIO comes to us and is wondering how this thing helps his organization, then cost of ownership, ease of scaling, and simplicity of starting new geographically distributed offices are really rich selling points.

To the CEO, it may be the fact that the IT department doesn't need to be as large as it is. The CEO is often scratching his head asking why he is spending 20 percent of his people budget just so the rest of his people can get their e-mail. So, it really depends on the audience to understand what the best value proposition is. The beauty of cloud computing is that there is a story for everyone—it's that compelling.

RAMLETH If you look at the past 15 years, the effectiveness of an enterprise user has remained pretty much flat. On a relative basis, what users could do in the office 15 years ago is no different from what they can do today. If you then compare what consumer users could do 15 years ago, what they can do today has skyrocketed and there is a huge gap.

Eight to nine dollars of every 10 from the venture-capital community is spent on that consumer curve, not on the enterprise curve. As enterprises we have to ask how we can follow the money and learn from the consumer curve. I have employees coming to me and saying, “I’m more effective when I work from home than when I work at the office.” While as an IT manager I feel hurt by that, you have to see it as an opportunity.

You need three things to effect this type of transition.

First, you have to set the vision and get your people to believe that collaboration, being part of the open world, is better than walling yourself off in your own smaller world. Second, you have to get your people to trust that they are not going to get hurt, laid off, or otherwise experience a negative professional result if they actively participate in making IT more effective. Solve the social problem with your staff. Third, don’t ever deviate and don’t ever let your staff down.

BOURNE So, I’m the guy and I have decided to go and do it. What types of support can I expect?

VOGELS In terms of cloud infrastructure services, one of the things we provide are tools and mechanisms for developers to address the particular requirements of the jurisdiction that their applications operate in. If the Canadian government has a law that Canadian companies cannot put information about Canadian citizens on servers controlled by American companies, then the infrastructure provider has to provide appropriate data-location choices to the application so that users can be in compliance with applicable law.

The EU (European Union) has privacy laws that are two-tiered. First, there is the EU-wide law that applies to all 22 member states; then each of the 22 member states has its own overriding laws. As an infrastructure provider, we offer abstractions to developers so that they can build applications that do the right thing for each of the member states.

BADROS One of the challenges is that we are way ahead of the government and legal systems. They will catch up but it’s going to take more time and energy.

CREEGER Does cloud computing enable new types of functionality that were not feasible under more traditional IT architectures?

VOGELS In the past, I always thought that you could not build data warehouses out of general components. It’s highly specialized, and I thought being really fine-grained precluded you from doing scatter-gather of lots of data operations. I think MapReduce (<http://labs.google.com/papers/mapreduce-osdi04.pdf>) has shown us that brute force works, and while it’s not the most efficient approach, it allows you to get the job done in a very simple way.

A number of small companies now provide data warehousing as a service. If you look at their storage resource usage, it’s about 5 percent more than if they would just run off their own set of specialized servers. The data movement is a little more inefficient than it used to be, but they’re getting access to much smarter, much easier-to-use computational components.

It turns out that we have many customers who do not need a data warehouse 24 hours a day. They need it two hours a week. In the worst case, they’re willing to spend a bit more on computational resources just to get these two hours. They are still ahead on cost, given the alternative of having to purchase the hardware outright and build it up to support a peak load.

CREEGER So, the analogy would be to analyze the cost of either purchasing a car or taking taxis to meet personal transportation needs?

VOGELS Engineers are not well trained to think about end-to-end cost. MapReduce and other examples have shown us that the end-to-end picture of cost looks very different from what

you would normally expect. We have to learn to think about the whole package—at storage, computation, and what the application needs to do—and really reason about what the axis of scale and cost really is.

BADROS We have lots of challenges in the way we teach cost to IT folks because we oversimplify the model. The old model, before cloud computing, was a single unit of cost—a server and the guy to keep it up—and you didn’t get any smaller than that. To run bubble sort or quick sort costs the same for anything less than a million items because you still needed one computer to do it on and one guy to keep it running. Cloud computing turns that model on its head.

CREEGER Let’s try to identify some places where the owner of a small- to medium-size business can begin to get some traction on how to take advantage of the benefits of cloud computing.

TUCKER If I were a small or medium business and needed to automate a business process, I would look at Greg [Olsen]’s Coghead. If I needed a CRM application to better understand my customers, I would look at Salesforce.

CREEGER What about PaaS? What issues does an SMB need to look at and how do they make a decision?

RAMLETH We try to pull everything to a very macro level. We ask, “What is the unit of output that matters most to the company?” If you produce widgets, it’s how many widgets you produce; the same is true for cars. Making the company more competitive is what matters most to us. A good strategy is something that creates value at a lower cost than your competitor. If you can find an output measurement that matters, ask what your IT cost is against that.

In 2002, we established our revenue metric as the inverse of the hours it took to complete our projects. Over the past seven years, our cost per unit of output has gone down by 55 percent, increasing our capacity and overall satisfaction. What has been required to achieve these milestones has been higher infrastructure utilization. If you are a traditional IT shop, that might mean that you have to go to cloud computing or some other form of service platform and delivery. From an IT standpoint, taking the steps we need to make service delivery more efficient and flexible is the way we improve our company’s competitiveness in the marketplace.

TUCKER An entrepreneur choosing cloud computing gets to eliminate the need for CapEx almost entirely. You no longer have to buy computers to put in your closet to run your e-mail system. You use on-demand applications to provide traditional IT services. You also put your corporate Web site and your product offering up on a cloud-computing site. You recognize that much of your real value is going to be based on the intelligence and data you can collect about your customers. Suddenly, you have to manage an enormous amount of information; having a cloud-computing provider manage it is better than trying to do this yourself.

RAMLETH In an entrepreneurial world, you can start a new company for hundreds of thousands of dollars rather than millions of dollars because you don’t need to raise money to build infrastructure.

VOGELS The benefits of cloud computing are beyond just start-ups. I like companies such as Mogulus that stream 120,000 live TV channels over the Internet. They own no hardware except for the laptops they use. Mogulus did all the election coverage for most of the large media sites streaming 45 gigabytes per second out of EC2 without impacting the other customers. The CEO of Mogulus states that he could not be in business without infrastructure-as-a-service. The business would not exist if it had to purchase tens of millions of dollars of its own capital infrastructure. Moreover, under those circumstances, he would not want to be in business.

CREEGER I'd like to go around the room once and give some final recommendations to the folks who are struggling to try to make sense of all this.

RAMLETH This is not a technology game but a change-management game. The goal is to get people to understand that it is not dangerous to think this way. We have three rules:

Think about what you can do that can benefit service delivery in aggregate; don't focus on the small subcomponents that can lead to suboptimal solutions.

Don't think about how you're going to distribute your costs before you start any effort. Make sure that internal charging mechanisms (allocations) are not obstacles for change and progress.

Don't think about and design future organization changes. Base decisions on organizational benefit and not on increased power to you as a manager or to your organization.

If you think about these three things, it's amazing what an organization can actually do.

BADROS The beauty of what we're talking about is that it's so easy to try. You don't need a big budget or approvals to get started. The fact that you can do this so simply enables innovation that would be unavailable if you needed to purchase a big piece of hardware ahead of time.

The types of things that emerge in this culture of innovation and bottom-up thinking are just amazing. To me, the key point of all this is that you do not have to make a big investment to get going and show value. That is a huge change from where things have been until now.

TUCKER As services move into the Internet, they become easier and more cost effective. This also means a shift in power in IT away from those who control capital resources to the users and developers who use self-service to provision their own applications. When FedEx went online, people were taken out of the support loop and customers could find their package status information themselves whenever it was needed. You can now apply the same principle to the provisioning of computing resources. A developer can have a server provisioned to run an application without having to contact a human. That cuts the most costly aspect of computing out of the equation.

OLSEN Cloud computing presents a compelling opportunity for consumers of information technology and producers of information services. Application builders should take advantage of existing functionality they can buy as opposed to the past practice of building their own and focus their resources on the unique capability they alone can deliver. Consumers of information technology have got to rethink where they look for functionality. If they don't adapt their service delivery models, then they will quickly become obsolete.

VOGELS To emphasize the point that Greg Olsen made, all you need is a credit card and the Amazon URL. I'm especially proud of CloudFront, the CDM (content delivery manager) that we have just launched. As a consumer, the CDM business is a hard business to be in. You have to have all these negotiations and minimum requirements in place, and you have to commit to tons of stuff. It's hard. Before CloudFront, there were no self-service CDM offerings. Now there is a true revolution in the business model as we are making this functionality available to folks for the first time on a pay-as-you-go basis.

BADROS That is the exact same trajectory that Google AdWords took when we enabled retailers to get their messages out to worldwide audiences and totally revolutionized the Internet.

TUCKER Self-service is tremendously powerful because it saves money for both the buyer and seller of a service.

RAMLETH We managed to be successful by either minimizing or eliminating the need to make a complicated business case for change.

CREEGER Reducing cost and enabling overall agility are what I believe you all are trying to say. Cloud computing has the potential for removing business friction to make more services possible and to do so much more easily, with less risk and capital outlay. I think that is as good a summary as any for something as transformative as cloud computing. Thank you all very much for your time, talent, and wisdom. ☐

LOVE IT, HATE IT? LET US KNOW

feedback@queue.acm.org

© 2009 ACM 1542-7730/09/0600 \$10.00.