Credit Suisse Technology Conference

Company Participants

- Jason Taylor, VP of Infrastructure

Other Participants

- Stephen Ju, Analyst, Credit Suisse
- Unidentified Participant, Analyst, Unknown

Presentation

Stephen Ju {BIO 6658298 <GO>}

All right, think we're going to go ahead and get started. Stephen Ju from the Credit Suisse Internet equity research team. Joined by Jason Taylor, who heads the infrastructure development effort at Facebook. So without further ado, take it away.

Jason Taylor {BIO 18251157 <GO>}

Great! So my name is Jason Taylor and I run a group called Infrastructure Foundation at Facebook. We are responsible for server design, server supply chain, overall capacity management. So capacity engineering, performance reviews, things like that. And then also the long-term infrastructure plan. So today I am going to walk through a little bit of our infrastructure and talk about a few efficiency programs that we are excited about and look toward the future of efficiency at large-scale computing.

So Facebook is large. 82% of our monthly active users are outside the United States. We are a global deployment. We have international data centers, one in Lulea and several in the United States. So 1.35 billion connect with us monthly, 1.2 billion on mobile and a stunning 930 million photos are uploaded to the site every day. That's a lot of media, a lot of content distributed on Facebook. 6 billion likes, 12 billion messages per day. It's a very active site, very dynamic. And we built an infrastructure to accommodate that.

Now, for the last five years really efficiency has a top priority at the Company. And initially I would say that it was really about necessity. We were facing a huge uptick in adoption of Facebook and usage. And efficiency has always been core just to be able to scale. And as we reached a large-scale it became necessary just for long-term financial viability and also just our ability to build platforms that scale well.

Now, from a cost perspective really efficiency breaks down into three areas. For data centers, heat management is really one of the most important things that we do in terms of core efficiency. In a poorly designed facility, a facility that doesn't concentrate on heat very much, you could easily pay 50% or 90% additional electricity bills for every watt that you deliver to a server.

Now at Facebook, because we've designed our own servers and because -- both servers and data centers, that heat tax is only 7%, which means that we are using cold air from the outside. We are not chilling air at all; we are passing it across the servers, mixing it in a hot aisle and then evacuating out the other side of the building. So in terms of raw thermal efficiency, our data centers are second to none.

Now, with servers we pride ourselves and having a vanity-free design. And we really focus on supply chain optimization. In 2011 we released our first data center and our first set of servers. We also started the Open Compute Project, which I'm sure many of you are familiar with, where we give away the designs to our servers and are very open about how we design, what our approach is. And how we think about our efficiency on the server level.

The other main efficiency win really comes from software. And horizontal wins like HHVM or HPHP, wins in cash, database. And web are all absolutely critical and continuing to deliver really efficient infrastructure. So during a peak time where one of our front-end clusters is really pretty piping hot, we can run for 10 hours of the day at about 90% to 93% server utilization. So we really work a tremendous amount on making sure that not only are the individual servers and the software optimized but also the whole data center is optimized to provide content.

We like blue. So all of our servers have blue LEDs. What you see here is -- so these are the fronts of the servers here. And that enclosed space is hot aisle containment. And so cold air comes in from the ceiling, it's sucked through the servers. Then inside that hot aisle containment that temperature can reach up to 100 degrees. That hot air is then evacuated out of the building or potentially mixed during winter times. So it's thermally a very efficient system.

And the other thing you will notice is all of our servers look the same. And that's because we really work hard on having a very homogeneous footprint so that you get good wins in terms of serviceability and maintenance, drivers, everything else.

Now, we have also been very open about all of our efficiency wins. So not only have we talked publicly, released data center designs, not only have we released server designs but we have also released most of the core software that powers Facebook. HHVM is our core PHP web server. It is, ballpark, five to six times more efficient than a traditional Apache stack.

Flashcache is the service that we use on databases that trades off flash caching and access to slower hard drives. Presto is one of our data processing/data warehouse pieces of software. Rock Stevie [ph], Proxy, Thrift and Folly is a general library we use.

So in all cases we really try to, as we are able to support open sourcing a project, we try to keep it out there.

And the reason for this is that we really believe that the entire industry can benefit from efficiency work that we do and that we can benefit from the industry feeding back and contributing new ideas and designs. And fundamentally our Company is going to win or lose based on our product, the cost of our infrastructure. And cost efficiency wins on infrastructure is something we would like the entire industry to benefit from.

So in terms of our architecture we keep it pretty simple. We have front-end clusters. Front-end clusters are synonymous with network clusters. These are large. They are about 12,000 servers per cluster. And this is the stamp of capacity that we push out in order to serve hot requests from users. Service clusters contain many of our dedicated services -- search, photos, messages. And others. Then our back-end clusters are all built and optimized for database storage. So you think a lot about the power redundancy in those clusters.

So to take one of our services, it's useful to think through a little bit about how one of these large services works. So if you are on Facebook and you are viewing that main feed. So on desktop, it is the center; on mobile, it is the main experience. All of that is the reason activity of all of your friends on Facebook.

That data is all kept in an index called the News Feed rack. All of the reason activity for the last few days of all people using Facebook is kept on each one of these racks. And the design is the leaf aggregator. So the leaves contain all of the data, all of the storage of recent activity -- it's all in RAM. And the aggregator is the thing that does the ranking algorithm and that consolidates all of the information to respond to a request.

So a web hit comes in. It says, I need some stories, goes to a News Feed rack, picks any one of the aggregators and says, give me some stories. The aggregator then blasts a query out in parallel to the other 40 servers in the rack, gathers that subset of data, ranks it based on your interests. And then since it off to be displayed.

Now, if you are an engineer at Facebook you can have any server you like as long as it's one of these five servers. We don't allow any variations. I have one of our main teams at Facebook infra is capacity engineering. They wear black T-shirts that say no on the front. They are very good at saying no to all kinds of engineering requests because fundamentally software is far more flexible than hardware. And you pay for hardware. Your hardware becomes inefficient when you have a lot of variation. And so the more homogeneity you can keep in your infrastructure, the easier it is to optimize, the better supply chain you have, better efficiencies up and down the stack in terms of operating an understanding servers.

And so at Facebook each year we have -- we only have five types of servers. So there's web, which is our main compute workhorse. Database, over the years it has

evolved from purely a disk thing to entirely flash. Hadoop is main data warehouse for data processing. And that's a lot of compute and a lot of disks. Photos -- it's all about the lowest dollars for gig. We certainly store a lot of media at Facebook. And so optimizing for dollars per gig is really important. Then that last feed rack, that's the News Feed service we talked about. So most engineers do like a lot of memory and a lot of compute. And that's what that rack gives you.

So the advantages of five server types and really constraining it to that are pretty classic. You get volume pricing. If you are putting in an order, a large order, you can really work deeply with your suppliers to work in a way that's beneficial to their supply chains, that's very predictable. And they can pass on savings to you, which is important.

It's also important for repurposing. So if we have five services that all have very large projections in terms of how well something could launch were when something could launch and if they are all using the same type of server, then we can be very flexible and reallocating servers from one service to another. And what that means is that we don't have servers or infrastructure that just lays fallow waiting for products to launch. It's -- well, you guys do mutual funds. But it's like a mutual fund. It's all dollars and some are up and some are down. You can really manage it well.

The other key advantage is easier operations. So in a typical data center facility you might have data center tech to server ratio of one to about 400 or 450. In our facilities it is somewhere between 1 to 15,000 and 1 to 20,000. So all of our servers are the same. They are all optimized for serviceability and we do work hard to make that easy. That also translates into operation and software benefits, just efficiency all up and down all of the consumers of the devices.

So some of the drawbacks. And this is drawbacks intrinsic in any hardware -- as soon as you allocate hardware, the hardware lands and lives for three or four years. However, the software needs change over time. And so at any point in time, your hardware and your software doesn't fit very well.

Now, because at Facebook the rack is a computer, we are really thinking about not just how does a piece of software fit an individual server but, because we allocate in racks, the question is how does the software live on the whole rack. And what I want to talk to you at this point is really an idea that we have mentioned before. And I want to talk about a few fun results. A disaggregated rack -- so here we are shooting for a better component service fit over time and we are also looking to extend the useful life of servers.

Now, if you think of that rack of News Feed servers and you ignore the fact that it's a bunch of servers and you think about, well, what is it really? You've got a bunch of compute, you've got a bunch of RAM. And you've got some flash. Now, exactly where the computer is and where the RAM is you shouldn't matter if it's all within the rack and if you got a nice healthy network. And so if you break down this disaggregated rack idea into a few major components, you got processors or

compute servers. You've got RAM or just kind of RAM servers. You might have storage server and you might have Flash.

So rather than put all of that in one server, where at any time you are going to hit a weakest link, you are going to have not enough computer or not enough RAM, let's break that all up, put it on a high-bandwidth backplane and then switch in and out resources as the service needs. And so at any time, you are not wasting services -- or you are not wasting resources. And so, your server and service fit can be better both across services and over time. And you can also accommodate a longer hardware refresh.

So if we have a type 6 server. And this is a News Feed server -- so up and down is CPU on the left, right and left is RAM. The News Feed server fits CPU and RAM very well. And it should, because we designed the servers for News Feed. However, if you go to another service, maybe search or one of the other index services, they might need more RAM than CPU. And what that means is that they are terminally not using that CPU resource.

The other thing that can happen is at the beginning of the service's life, maybe during year one, they are a perfect fit. But along year two they need more RAM or the need more flash. And so being able to allocate that just in time or allocate that hardware along with the needs of the service provides a huge benefit because otherwise you are buying more servers when all you really needed was more RAM. Now, you can't open the cases on 10,000 servers and upgrade the RAM. That doesn't work. So being able to add in sleds of RAM is a huge benefit.

The other and third benefit is that you can really keep the hardware for as long as it will physically last. Again, many times when you are doing computers at scale, you are deprecating them based on the critical resource that's no longer good enough. And that means you are throwing away other resources that are perfectly fine. Compute really doesn't get old. RAM -- it's a solid-state device; just a pure RAM device can operate forever. Disks can wear out over time. Then flash, depending on your write volume, might wear out over time. But it can actually live for a while.

And so if you think of a disaggregated rack for, say, graph search, rather than have computers that have all three resources, let's have compute servers, things that have just compute -- some RAM but are mainly focused on compute -- that would be a type 1 server for us -- a flash sled, which would be anywhere between 30 or 256 terabytes of flash on a single sled. A RAM sled might have 256 or 512 gigs of RAM and then storage.

Now, in year one the ratios that we pick might be perfect. But then in year two they might discover -- they might have an efficiency win. The index might grow over time. And the best thing to do would be to just give that service more flash. And so rather than allocate an entire separate rack, thereby doubling the cost, you just allocate that one resource and you just slam in another Flash sled. And that kind of flexibility really leads to some pretty nice efficiency wins.

So we still maintain all of our core strength -- our volume pricing, custom configuration, all of those sorts of things. But what it really allows us to do is much smarter technology refreshes and the hardware, the resources -- because we are thinking on that rack level we can evolve the hardware with the service.

Now, last year I talked to you all about the approximate TCO wins. And so over a three; or six-year period, we were looking at between 12% and 20% OpEx savings, on the conservative side and, more aggressive, between 14% and 30%. Using this different approach -- keep in mind nothing has fundamentally changed about the computers that we are allocating. We are just allocating them in a different way. And we are helping our software team be a little bit more flexible in how you bring on resources. And this works for pretty much anybody at scale.

Now, we -- at the time we talked last year we were working on this project. And we've now landed it for one of our services. Now, we've got 20, 30 services, maybe 40 major ones. But for the one that we started with we were actually able to realize a 40% savings in the total cost of operating this equipment. So by doing nothing more than just bringing the resources necessary online at the right time and by customizing the rack in a scalable, flexible way that maintains all of our supply chain wins we were able to realize a 40% reduction in costs on this one service. This is something -- this is an approach and a technique that we think pretty much anyone can use.

Now, if we think about computers and resources and we think about the last 20 years, when -- one of our ideas about disaggregated rack is to be able to adopt different types of resources. And if you look at the last 20 years, the types of things that are in servers that scale are pretty much the same. It's -- the server below is -- Amir Michaels is holding one of our first Facebook-built servers. That server, from an architecture perspective, is almost identical to that 386 in a tower case from 20 years ago. There have a few new technologies -- math coprocessors, two processors for server, multicore. All of those are good. The only game changers in the last 20 years have really been GPUs, which are great at vector math. And flash memory, which is a phenomenal win over the last four or five years.

What we are looking forward to is really major advancements in the network. So 100-gig NICS, 400 gigs between switches -- those are all perfectly reasonable now, given the state of technology. Flash has been going and all of the Flash providers have been really pushing for higher and higher IOPS and heavier-duty flash. We actually feel that at the at-scale environment we want to look in the other direction, going for lower IOPS because you really don't need that many. And then also we can be much more careful about how we use flash. And what that means is that we can get much denser flash sleds and realize nice benefits there.

So we think that in the flash there's always going to be a nice market for high-performance flash. But we think in the data center world a lot of the flash interest is going to shift towards lower and lower flash. Last year we did a talk and asked the industry for the -- please make the worst flash possible. And really, we can work with

very -- not necessarily low-quality flash but lower-endurance flash, TLC or even beyond in terms of bit-density.

Now, there's also a number of RAM alternatives we think are coming up and are very interesting. Phase-change memory, I think, is going to be -- if it works out it's going to be pretty good. And resistive memory is on par with that in terms of technology. Then also cold flash or WORM solid-state storage. So not all storage needs to be on spinning media. It's perfectly reasonable for very immutable data to be put on solid-state devices.

So if you look at our eye chart for technologies over the next few years, it is an eye chart. There's a lot of detail there. But if we simplify a bit -- give you one more second to take a photo. Sorry. We are down to some pretty basic evolution. So over the last -- from, say, 2009 to, say, 2020, we think that computer in the data centers is going to be much more focused on one processor, system-on-a-chip processors. That whole efficiency ecosystem that's developing there is really strong. Yes. We will still have RAM. But we think phase-change memory and resistive memory will also be strong players.

WORM -- so solid-state storage. This is the technology that might evolve over the next, say, three to five years. But permanent, immutable storage in solid-state is completely reasonable. And we think that the bit densities can get to the point where they can be superior from a TCO sense for thin hard drives compared to hard drives.

Optical data storage has a great future just in terms of the ability to densify media over time. We've seen a number of announcements in terms of archival disks and taking what looks like Blu-ray discs and taking them from 100 gigs to 500 gigs or even up to a terabyte. Then 100-gig fiber to the servers -- we think we are only a couple of years away from that.

So these are the technologies that we are interested, these are the technologies that we are excited about. And with that, I'd like to open up for questions.

Questions And Answers

Q - Stephen Ju {BIO 6658298 <GO>}

Yes. We probably have time for a couple of questions here. So poll for questions from the audience.

Q - Unidentified Participant

There's a lot of talk about Mesos and its implications for computing and software infrastructure stack. What is your perspective on how Mesos gets adopted and the implications of the change there?

A - Jason Taylor {BIO 18251157 <GO>}

Mesos is the virtualization thing? Yes. We don't do that. So all of our infrastructure is based on bare metal scaling. So virtualization in the cloud is excellent when you are managing a heavy idle workload. So if you have 20 idle servers and you need to compact them into two idle servers, that's obviously a win. You didn't buy 18 idle servers. When you're building for scale and when you are building for throughput, the best efficiency wins come from, one, just really balancing the utilization of the gear, which is a lot of what the disag work is about. But then also really looking deeply at how does the software and hardware work together and then look for big performance wins there.

The first, I would say, 3X of HPHP's wins really just came from going from an interpreted language to a compiled language. And that was a very solid win. Most of the wins since then have really come from both optimizations and how the code works. But also how does it work on the specific hardware we run. And as soon as you start virtualizing, as soon as you start putting some layers of abstraction in there, you decouple the engineers, who are fantastic, from -- you essentially don't allow them to make those kind of optimizations anymore. And so for Facebook, because we have such a large workload, we really focus on using each piece of hardware as much as we can.

Q - Unidentified Participant

To what extent are the efficiencies that you discussed this morning going to allow Facebook to have more capital-efficient growth as the Company continues to grow its revenues over the next three to five years?

A - Jason Taylor {BIO 18251157 <GO>}

I think fundamentally efficiency has been a top priority for us for a long time. And I think the most important aspect of efficiency to us is the coherence that it brings to all of our software engineers in terms of unifying them in thinking about how does the software and hardware work together. So if you want to talk actual capital spend you would have to talk with Deborah. But in terms of efficiency its core to the way that we work.

Q - Unidentified Participant

You talked a lot about 100-gig [ph] fiber. Can you talk about your (inaudible) activity?

Q - Stephen Ju {BIO 6658298 <GO>}

And the expansion of 100-gig fiber within that footprint.

A - Jason Taylor {BIO 18251157 <GO>}

Sure. So I can say that when I have talked about 100-gig fiber and 400-gig fiber, all of that is within the data center. So what has happened is that the telecommunications industry has delivered fantastic technology over the last bunch

of years in what you talked about, dark fiber, major facility two major facility, running hundreds of miles.

Well when you look at that core technology, it is ready to be adopted into the data center space. And that kind of bandwidth within a data center is very possible. There's a number of great technologies that are developing right now in terms of silicon photonics. There's several companies working and essentially fully integrated chip and optics packages which deliver 100-gig performance at really very low cost. So it's the largest thing that has happened.

The biggest thing that happened two or three years ago was flash at the data center. The thing that's happening right now is the amount of network that you can buy for a reasonable price is climbing dramatically. When I joined Facebook we had 1-gig NIC's everywhere. And 1 gig was the standard. In 2011 we shifted to 10 gigs. Pretty soon, in the next couple of years we will have 25 gigs for the servers. I'd say within three years we will have 100 gigs for the service.

So over around a six-year period going 100X up in the amount of bandwidth that's available -- that's transformative in both how the data center operates but also how do you write your services. So network -- I think that work and improvements in networking is going to be the largest driver towards changes in how large-scale Internet companies work.

Q - Stephen Ju {BIO 6658298 <GO>}

I think with that we are actually out of time. So thank you very much.

A - Jason Taylor {BIO 18251157 <GO>}

Thank you.