cloudscaling

in association with

NEC

Infrastructure-as-a-Service Builder’s Guide

Network Edition: The Case for Network Virtualization

v1.0.4 – Q4 2010

TABLE OF CONTENTS

INTRODUCTION

Cloud computing offers a truly revolutionary paradigm shift in the model for how IT services are consumed and managed. Much more than a set of technologies or even an outsourcing model, cloud computing provides a new way for IT to "do business." This whitepaper focuses on helping cloud providers build Infrastructure-as-a-Service (IaaS) clouds. In particular, this edition of the IaaS Builder's Guide explores how to deliver cloud scale networking for 10,000+ physical servers.

As a technical guide this paper makes an attempt to strike a balance between a purely academic resource, a pragmatic reference, and enough business information to allow a broad range of readers to derive value. Therefore, the target audience for this document is assumed to be network architects, cloud engineers, CTOs, and technical business leaders looking to understand how to build networks inside data centers at cloud scale.

Some information here is intentionally simplified to reach a broader audience, but there is more than enough detail to point experienced network engineers and architects in the right direction when trying to understand how to build cloud scale IaaS networks.

EXECUTIVE SUMMARY

Cloud networking is not a trivial task. The Internet itself is a piece of infrastructure designed to run at massive scale. In its history and evolution, a large number of challenges had to be overcome to run a large, global network of millions upon millions of connected devices. Modern data centers designed to provide cloud service offerings, such as Infrastructure-as-a-Service (IaaS) face similar challenges to building the Internet itself due to their size. At its simplest, when providing virtual machines on demand, like Amazon's EC2, we are talking about data centers that need to provide as much as 1 million—and potentially much more over time—networked devices in a single facility[1].

Building a data center network that can manage one million networked devices means using techniques very similar to those used in large ISP backbones and the Internet at large. In fact, many of the largest cloud providers, such as Amazon, Google, Yahoo!, and Facebook, prefer Layer 3 (L3) networking techniques over typical Layer 2 (L2) networking techniques because of their proven scalability. L2 networking models, prevalent in typical enterprise data centers, are known to have significant scaling problems and most enterprise data centers are in the 10,000-server range, not the 100,000- or even 1,000,000-server range.

[1] *At the time of this writing Amazon EC2's size is estimated at ~600,000+ virtual servers.*

In addition to the need to network massive numbers of physical servers together, the rise of hardware virtualization means that the number of connected devices in a data center, particularly those of IaaS clouds, is growing exponentially. For example, our founder Randy Bias estimated that Amazon's EC2 had as many as 40,000 physical servers and perhaps 300,000 virtual servers in June 2009 with an average of 8 virtual machines per physical server. Newer clouds are seeing densities of 40:1, which leads us to conclude that a similar sized deployment could contain over 1.6 million virtual servers in the near future[2]. Only L3 networking techniques are designed for this scale.

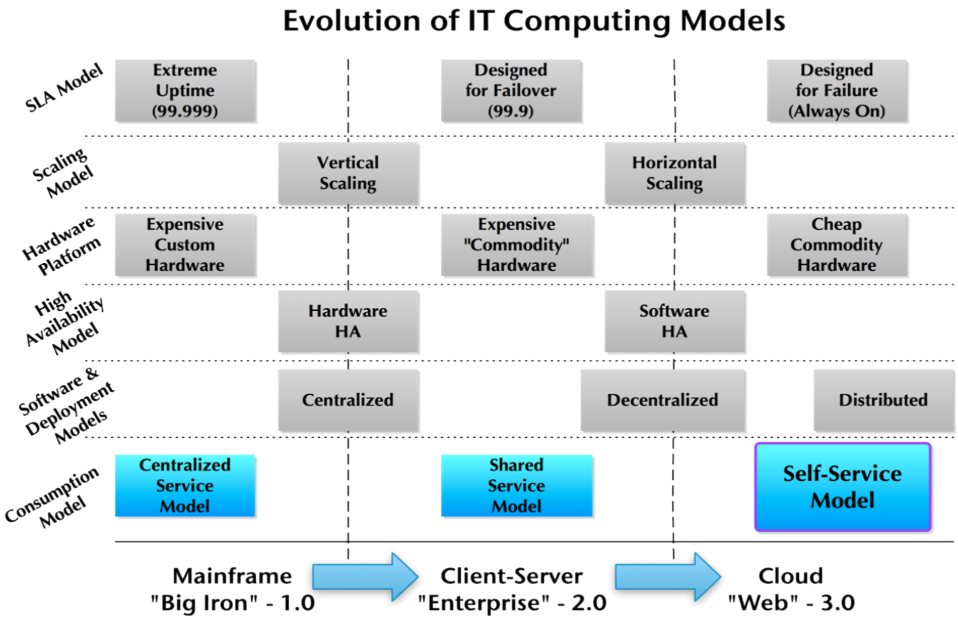
While L3 networking techniques are excellent from the point of view of the cloud provider, they restrict how cloud customers can use their virtual networking between servers. For example, Amazon's EC2 does not easily allow the use of broadcast traffic, multicast networking, or allow customers to pick their own IP address range. Many use cases, including support of legacy applications, require this functionality, which is provided by allowing each customer to have their own L2 network(s).

An ideal arrangement, therefore, would be to allow cloud providers to build and operate their networks using L3 networking techniques, while allowing customers to use L2 for their own purposes. This is now possible using data center network virtualization. As the methodology and technology matures, it will become the prevalent model for building cloud scale networks.

[2] *We know of at least one cloud looking at densities even higher, such as 90:1.*

EVOLUTION OF CLOUD COMPUTING

Cloud computing is a new way of delivering IT services that is only a few years old, but is seeing rapid adoption. This new methodology includes changes to technological, architectural, operational, and service models. Cloud computing can be thought of as “Computing 3.0.” It will displace the current computing paradigm, client-server Enterprise computing (Computing 2.0), which displaced big iron mainframe computing (Computing 1.0). This diagram depicts the aspects of the evolution to this point:



In this diagram you can see a clear evolution from mainframe to client-server and finally to cloud. There is a very clear change in values for both those building IT services and those consuming them. For the builder, concerns have largely moved towards using distributed systems techniques at a massive scale through automation, decentralization, more and cheaper hardware, and valuing software solutions over hardware solutions. Companies like Google, Amazon, and Microsoft typify this trend today.

For the consumer, there is now a distinct difference in how they experience IT services they consume at home (e.g. GMail) and those they consume at work (e.g. Exchange). At home, the experience is low friction, easy to use, and requires no intervention or help from an IT person, while at work, it is very much the opposite. Simultaneously, online web services are steadily gaining in terms of features and functionality.

Another way to think about cloud computing is to consider it as the IT equivalent to large scale robotics factories for manufacturing cars. Since the 1970s, when introduced into the automotive industry, robotics and automation have significantly transformed that industry. Fully automated automobile factories provide significant efficiency improvements through increases in quality and production output for cars. In effect, robotics and automation allowed the car industry to 'scale' more effectively. Unfortunately, one side effect of increased efficiency in this case is that it requires retraining and adjustment of workforces. Cloud computing will have a similar impact on the IT industry as it provides massive automation and efficiency gains for IT services.

CLOUD COMPUTING TODAY

Cloud computing evolved from the way that Google, Amazon, Microsoft, and Yahoo! operate their online IT services. The massive scale implied in delivering these services to huge audiences, necessitated building IT in completely new ways. These new technologies, techniques, and operational changes allow businesses to achieve cloud scale. For example, Bechtel's CIO details the difference between staffing levels for Bechtel and Google in a recent CIO.com article:

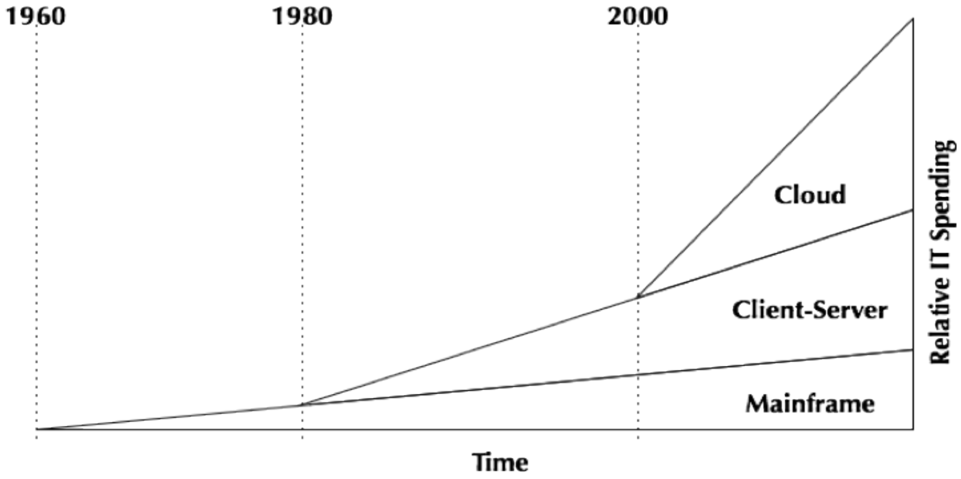
> *Next, Bechtel studied how Google's servers operated. Google had one system administrator for every 20,000 servers, while Bechtel had one for every 100 servers. "What we learned is that you have to standardize like crazy and simplify the environment," Ramleth says. "Google basically builds their own servers by the thousands or gets them built in a similar fashion, and they run the same software on it. So, we had to get more simplified and standardized."*

Only massive automation, standardization, and cloud computing techniques can provide this kind of leverage, a difference of 200 times. Cloud pioneers like Google and Amazon get similar leverage for all operational and capital expenses: hardware, power, labor, space, cooling, and development. This is the level of efficiency that typifies cloud computing.

We witnessed a similar progression when client-server computing displaced the mainframe model. Two developments from this earlier shift are noteworthy. First, a long-term 10- to 20-year shift produced a fundamental impact in the way that IT was leveraged inside of businesses. Second, mainframes never died; they simply became isolated into a smaller niche of the IT landscape. Even today, mainframe computing continues to grow at a steady clip.

It is safe to say that the evolution into cloud computing from client-server (and mainframe) will be much the same with most new applications and services being deployed using cloud computing techniques "in the cloud," while client-server continues on a much slower growth trajectory. As with mainframe to client-server, many applications and services will migrate to using cloud computing over time.

With the proliferation of mobile computing devices, build-outs of very large scale cloud data center capacities, and the ease of entry into mass markets made possible with platforms such as Facebook, we are entering a new era. We will see many more uses of IT technology than ever before, and this will largely drive the adoption of cloud computing. In effect, we will see a Cambrian explosion of applications and services, much as client-server created all new opportunities for businesses. This means we're at the very beginning of a long term change, as shown in the diagram below.



CLOUD COMPUTING AND THE NETWORK

Like industrialized robotic automobile factories, bigger is better for cloud data centers. Larger data centers can achieve economies of scale that are impossible to achieve otherwise. Everything from space to power, cooling, servers, storage, and networks can be effectively designed to maximize efficiency. Service providers who build public utility cloud services are driven to maximize efficiency because every IT dollar spent or saved directly relates to profits earned. The networks inside these data centers are no exception.

Cloud networks, much like the Internet because of size, have special concerns for two key groups: providers and consumers. Providers of cloud scale data centers must be able to achieve efficiency and scalability in their networks. An increase of just one percentage point in efficiency can save millions of dollars. Scalability directly impacts a service provider's ability to grow effectively in the face of demand while keeping operational costs low.

Meanwhile, consumers find themselves with two kinds of applications they must support on cloud provider infrastructure: legacy (client-server or mainframe) and cloud. As expected, new cloud applications can be designed for the special environments of cloud providers to maximize their own scalability and elasticity. Simultaneously, however, cloud consumers have huge numbers of legacy applications that make assumptions about operating in traditional networking environments found in client-server-based data centers today.

Because today's clouds mainly support either—but not both—types of applications, any complete solution for cloud networking needs to address these two very separate concerns:

1. Scalability and efficiency for cloud applications and cloud providers
2. Support for legacy enterprise applications

For example, EC2, the market leader, was largely designed for new cloud applications and optimized for scalability and efficiency. By using a Layer 3 (L3) networking design, they are able to grow to a massive size. Amazon's success in this regard is already well known, with an estimated 500,000+ virtual servers running on 60,000+ physical servers today.

Other clouds, such as Savvis or Hosting.com, focus on enterprise support for legacy applications. These are typically VMware-based vCloud providers whose architectures are designed for supporting legacy client-server applications. Their networks are a Layer 2 (L2) networking design, which simplifies consumer issues with legacy applications but increases complexity, cost, and scalability for the provider.

The need for both solutions can be inferred from Amazon's Virtual Private Cloud (VPC) service, which provides a simplistic L2 capability on top of their existing L3 network.

NETWORK SCALABILITY & NETWORK VIRTUALIZATION

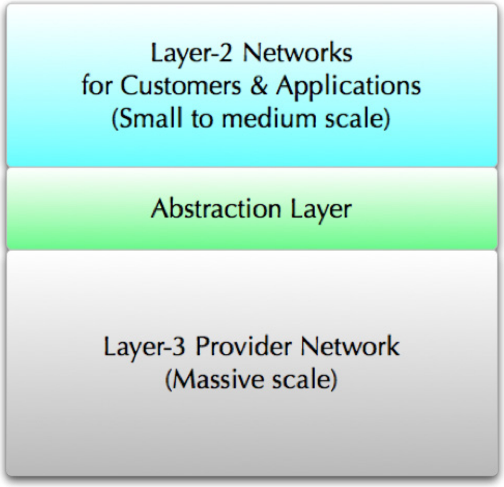
It is well understood today that Layer 3 (L3) oriented network designs can scale to massive size. This is the design of the Internet. Layer 2 (L2) network designs are known to have significant scalability issues. Most enterprise data centers are oriented on L2 design principles, while most Internet Service Providers (ISPs) and large cloud data centers use the L3 model. Besides Amazon, Facebook is known to use the L3 model inside its data center and it is widely believed that Google and Microsoft do as well.

A number of techniques and technologies attempt to solve some of the scalability, performance, and security issues in L2 network designs. For example, VLANs are a mechanism used to provide both scalability and security (isolation). Use of VLANs allows for isolation between network segments without using routing (L3). Newer techniques, such as RBridges and SEATTLE, attempt to provide additional mechanisms for scaling L2 oriented networks. Unfortunately, these techniques continue a time-honored tradition of being “bolted-on” to the existing L2 methodologies.

An emerging technique that offers far more promise is that of *network virtualization*. Network virtualization is a combination of technology and methodology that allows for the use of L3 network designs for the service provider, but L2 network designs for the consumer of the cloud. This approach is sometimes referred to as “L2 over L3” or “L2oL3”. The advantage of this is that it uses well understood network designs for both the provider and consumer.

Network Virtualization Explained

Network virtualization comes into play when an abstraction layer is created between the provider network and the consumer network(s), providing a separation of concerns. The service provider can operate the L3 network topology very efficiently, while providing a traditional enterprise network data center view for consumers. Additionally, the L2oL3 model allows customers to pick between the L2 and L3 models. Customers with a new application that can be designed in the cloud computing model, by choosing to use an L3 network design for that application and eschewing less scalable L2 networking, gaining the scalability inherent in the former design.



Network Virtualization

Network virtualization also focuses on providing programmability of network configuration, particularly the virtualized L2 network layer that cloud consumers will see in a typical Infrastructure-as-a-Service (IaaS) deployment. This programmability is key to allowing for scalability on the provider side. Large clouds like Amazon Amazon's Elastic Compute Cloud (EC2), Rackspace Cloud, and GoGrid, are highly automated systems. Each new customer deployment requires that networking is allocated appropriately. Well designed L2oL3 solutions provide an API that can integrate to provisioning and scheduling systems.

In the following sections, we will discuss L3 and L2 network designs as deployed in cloud provider data centers today, followed by a deeper dive into some detail on how L2oL3 works.

Finally, we'll make recommendations on emerging providers and technology stacks that can allow you to leverage network virtualization.

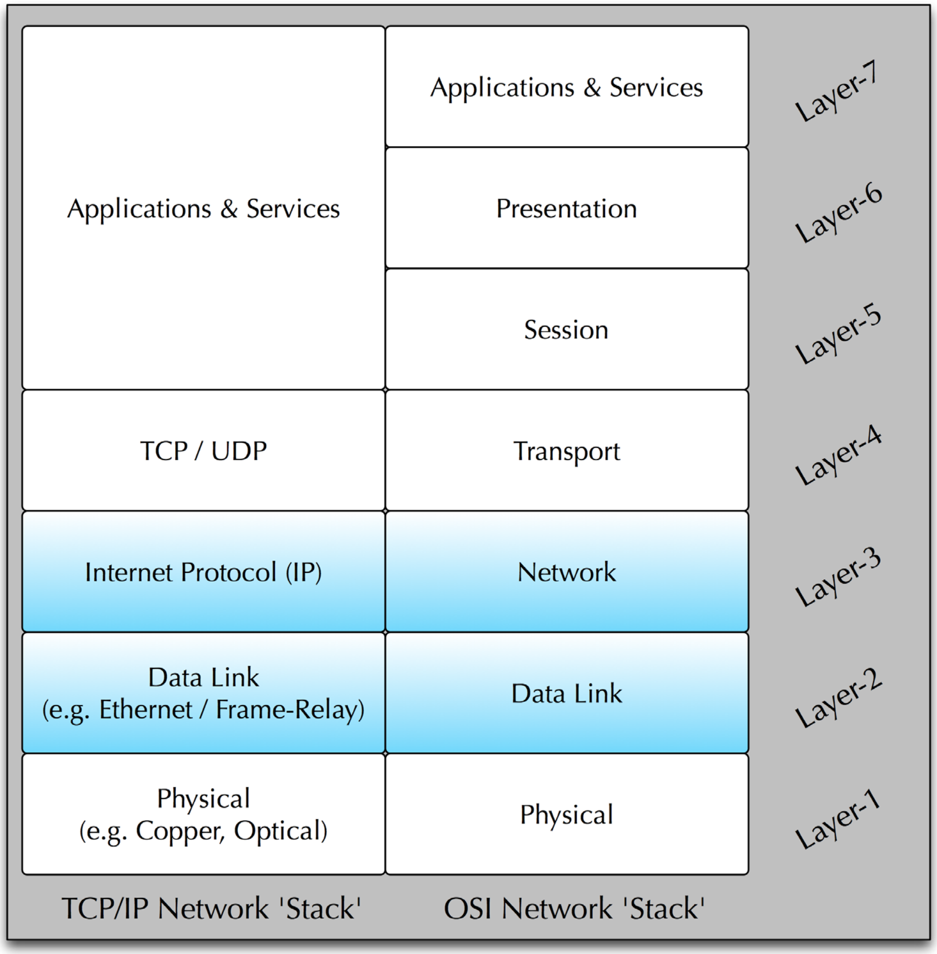
IaaS Networking Approaches

There are three major approaches to building IaaS networks today: L3, L2, and network virtualization. We will first provide an overview, then look at each of the three techniques, how they are used, their pros and cons, and where cloud networking is headed in the future.

LAYER 3 ROUTING VS. LAYER 2 SWITCHING OVERVIEW

If you are not familiar with the TCP/IP "stack" (aka "model"), we recommend that you familiarize yourself with the basics.

The TCP/IP and other networking models (e.g. OSI Reference Model) are "stacks", where each layer of the stack provides different functionality. The following diagram shows both TCP/IP and OSI stacks side-by-side:



TCP/IP Network 'Stack'	OSI Network 'Stack'	
Applications & Services	Applications & Services	Layer-7
	Presentation	Layer-6
	Session	Layer-5
TCP / UDP	Transport	Layer-4
Internet Protocol (IP)	Network	Layer-3
Data Link (e.g. Ethernet / Frame-Relay)	Data Link	Layer-2
Physical (e.g. Copper, Optical)	Physical	Layer-1

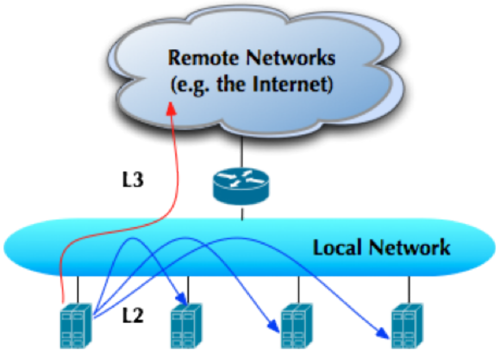
The two "layers" of the stack we care about are #2 and #3. In most modern data centers, Layer 2 (the Data Link Layer) is Ethernet. There are a variety of alternatives to Ethernet, but most are used in Wide Area Networks (WANs), not Local Area Networks (LANs), where Ethernet is king. Directly above Layer 2 is Layer 3, the Internet Protocol (IP) Layer. Every computer on the Internet uses an IP address, which allows you to communicate with that computer. When we discuss moving data at Layer 2 in Ethernet-based environments, we talk about “switching”, while in Layer 3, we talk about “routing”. As implied by the diagram above, switching (L2) and routing (L3) are not entirely independent. One depends on the other.

In this document, when we say an “L2 oriented” network design, we mean one in which the focus of effort is on scalability, performance, and security for the switching layer. When we say an “L3 oriented” network design, we mean the focus of effort is on the routing layer.

The reason to use an L2 approach instead of an L3 approach is improved simplicity and usability. At the lower end, an L2 network design is much, much simpler than an L3 network design. Usability is also much easier. With Ethernet, each device has a unique Ethernet address. When moving between physical locations in the same data center or campus, the Ethernet address tells where that device can be found. This means that the IP address doesn’t have to be changed when the location is changed. With IP, however, most devices must have updated IP addresses when their locations are changed.

On the other hand, L2 network designs simply don’t work on larger scales, such as the Internet. The need for every switch to understand the location of every Ethernet address on the network makes this impossible.

Somewhere in the middle is a large gray area where L2 network designs have a number of “bolt-on” technologies and protocols that attempt to allow them to scale up. Although there have been varying degrees of success, L2 still does not scale as might be desired. Perhaps more importantly, these L2 scaling techniques do not sufficiently acknowledge fundamental customer requirements: L2 networks are usually designed around a single application or customer and hence individually don’t need to be very large. VLANs that provide isolation and protocols like Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), and Multiple Spanning Tree Protocol (MSTP) all attempt to manage very large multitenant networks, in which each individual network is manageable, but the aggregate is not.



Layer-2 and Layer-3

In this regard, network virtualization fits the requirements since it is designed so that each tenant receives its own set of L2 network domains, while the underlying physical layer is scaled based on L3—not L2—network scalability.

CLOUD NETWORKING: THE LAYER 3 APPROACH

L3 networking is the approach used inside large cloud providers today such as Amazon, Google, and Facebook. As a proven approach to cloud scale infrastructure, L3 can operate at arbitrary scale. In other words, when built properly, it should be possible to build a L3 network of any size. This is how the Internet is designed today.

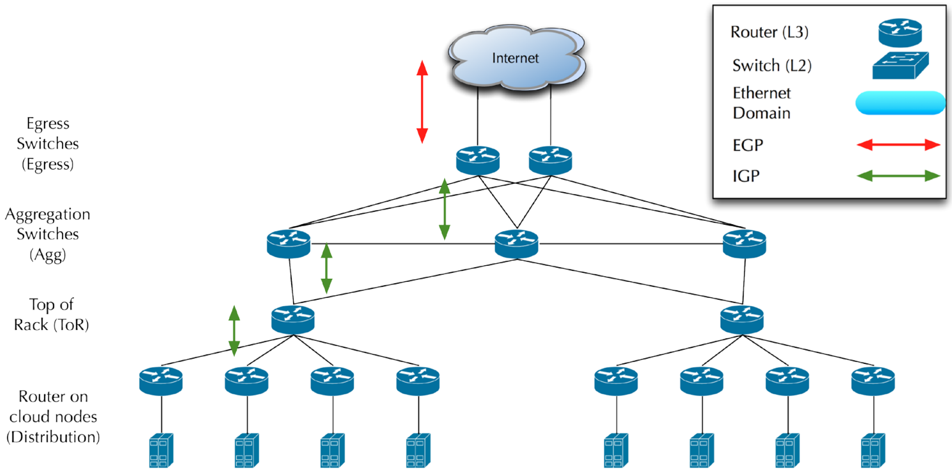
Put simply, L3 networking allows for any scale network by hiding all of the details of the network depending on the topological location. By aggregating route information, using a hierarchical addressing scheme, and only storing what is necessary to understand whether data is local or remote, each system in the L3 network can make relatively simple decisions on a small dataset. In contrast, L2 networking requires that core systems know a lot about every system in the network.

For example, if we attempted to use L2 networking to run the Internet today, the back-bone routers of ISPs would have to hold hundreds of millions of entries in their routing tables and make decisions across large datasets. Not only would hardware capable of efficiently handling such data be prohibitively expensive, but the ever increasing size of the Internet itself would eventually make it impossible to keep up.

L3 networking techniques solve this problem elegantly.

L3 In Cloud Data Centers

The following diagram shows how L3 networking works in practice in cloud data centers today. Each node or system in the infrastructure acts as an L3 router and has only enough information to make a local routing decision, punting the data up to the next tier of router as necessary.



The most obvious difference in an L3 routed network topology for a cloud data center is that its routing protocols are run all the way down to each individual server (cloud node). Routing protocols are generally broken into two kinds: Interior Gateway Protocols (IGPs) and Exterior Gateway Protocols (EGPs). IGPs are used inside of a single business network. Most L3 solutions that run networking down to the cloud nodes probably use Open Shortest Path First (OSPF) as the IGP, but Intermediate System to Intermediate System (IS-IS) is also possible[3]. EGPs are used exclusively for Internet backbones and are not relevant here.

The next thing you may notice is that each virtual server is directly connected only to its default gateway—in this case, the physical cloud node—and there are no other virtual servers on the same L2 network as the virtual server. This is why broadcast traffic is impossible, multicast networking is hard, and why applications that make assumptions about servers being physically co-located can become confused.

L3 Pros and Cons

There are pros and cons to the L3 approach and, while there could be a vigorous debate about whether this approach is better or worse, we know for certain that L3 routing has several key advantages:

Manageable networking: Route aggregation allows L3 to scale. Because each router needs only a subset of all network information to make a decision where to send data, large networks are extremely manageable.

Efficient network utilization and improved bandwith: Equal Cost Multi-Pathing (ECMP) means that data traveling from one point to another can use multiple paths simultaneously, allowing for very efficient network utilization and greater amounts of aggregate bandwidth.

Increased utilization and efficiency: Shortest-Path First (SPF) routing means that in larger networks data will take the shortest and quickest path to its destination, resulting in increased utilization and efficiency.

The primary disadvantage of L3 networking is that locality matters. Unlike L2 Ethernet, it's not possible to simply move a device from one L3 network to another without adjustments. Moving a server requires changing its IP address to an address on the new network. Despite this disadvantage, L3 is clearly the technique of choice for large scale networking, so let's take a look at each of the advantages in some detail.

Route Aggregation

For L3 networking to scale properly, route information is aggregated at each layer. In the previous diagram, for example, the egress routers have simple routes to a small

[3] *In fact most folks use OSPF. The notable exception is Google who uses IS-IS like many Tier-1 Internet Service Providers (ISPs).*

number of networks that point to the next layer down (aggregate routers). The aggregate routers then have information about the networks in each rack that point to each top-of-rack (ToR) switch, but do not necessarily know where each virtual server is located inside the rack. Finally, the ToR switches know which servers have which networks that virtual servers reside on.

This table shows how the networking might be aggregated at each layer using an example network block (10.1.1.0/24[4]):

SWITCH LAYER	NETWORK/MASK	NUMBER OF NETWORK ROUTES
Egress	10.1.0.0/16	1 65,536-IP address network (/16) containing 256 * 256 /24 networks
Aggregate	10.1.1.0/24	1 256-IP address network (/24)
Top-of-Rack	10.1.1.0/26 - 10.1.1.192/26	4 64-IP address networks (/26)
Cloud Nodes	10.1.1.0/30 - 10.1.1.60/30	16 4-IP address networks (/30)
Virtual Machines	10.1.1.0/30	1 4-IP local area network with 2 IPs usable; one for the VM and one for the gateway (i.e. cloud node)

Route aggregation means that when and if a host in the network needs to move locations, its IP address also must change. Inside a constrained and controlled environment such as Facebook, for example, the application itself can be told about changes to host IP addresses. However, most IaaS customers will be more familiar with an L2 paradigm, where they simply move their servers around from place to place without any thought to IP addressing. Amazon's EC2 is built predominantly around L3 networking, which explains why a newly reallocated virtual machine that is running on the same physical machine usually has the same IP address.

In addition to issues with IP addressing, L3 networking does not allow broadcast traffic. Broadcast networking is inherent to L2 networking, but is not available in an L3 environment. Similarly, multicast traffic can also be challenging, although it is possible to provide multicast in an L3 network. In many modern applications, broadcast and multicast traffic are used for discovery. An example of this is Microsoft's NetBIOS protocol. Windows servers look for each other on the same network using NetBIOS. Without L2 connectivity between servers, a different method has to be used to allow Windows servers to discover each other.

[4] *An explanation about how Classless Inter-domain Routing (CIDR) notation and network segmentation works may be too technical for some audiences and is beyond the scope of this document. Please see the Wikipedia article for a more in-depth explanation.*

ECMP

Equal-cost multi-path (ECMP) routing is a technique used to maximize bandwidth between two routers over a single hop by using multiple equal-cost links. It is a load-balancing technique, and comes with native support in both the OSPF and IS-IS routing protocols. Generally, individual flows are restricted to a single path since per-packet decisions can create problems when maximum transmission units (MTU) or latencies differ across paths. Nevertheless, ECMP offers a straightforward way to increase aggregate throughput between routers when a single link can not accommodate the traffic volume.

Shortest-Path First

For any graph with nodes, and weighted links between nodes, finding the shortest weighted path between a single source and any destination node on the graph is an important problem. Imagine a typical road or travel map with cities, roads, and distances. A shortest-path first (SPF) algorithm[5] will tell you which roads to choose when driving from one city to another minimizing the total mileage driven. This elegant algorithm was discovered by Edsger Dijkstra in 1959 and underlies the dominant IGP link state protocols IS-IS (Intermediate System to Intermediate System) and OSPF (Open Shortest Path First).

SPF is fast and reliable in selecting the shortest path from a source node to any node on the graph. Routing protocols implementing SPF algorithms also provide operators the ability to adjust the weight of each link for factors such as latency, throughput, reliability, and cost, thus granting strong control over traffic engineering across the network. In practice, this provides highly tunable, fast converging, and reliable loop free routing. This is in contrast to L2 switching, which in practice uses Spanning Tree Protocol (STP) to remove loops and the ability to shut down multiple paths and the use of ECMP. In order for STP to be truly fast converging like SPF-based protocols, a significant amount of tuning must happen at all switches. Manually telling the switch fabric what its topology is instead of discovering dynamically (as in L3) is error prone and brittle.

The relatively minor down side is that these L3 link state protocols require an accurate map of the graph to make these decisions, and all routers in the same area need to work from identical maps in real time to avoid routing loops.

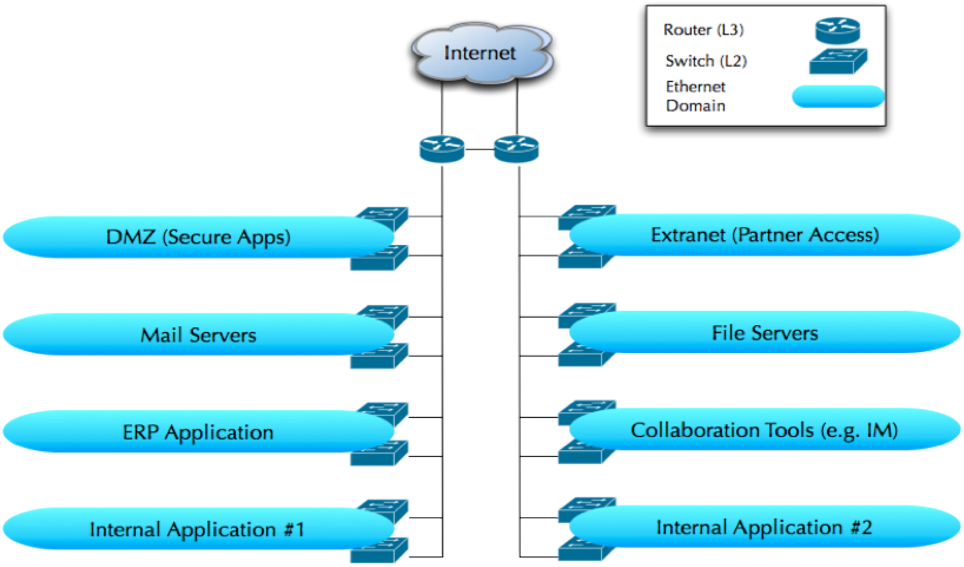
[5] *Also called* Dijkstra's algorithm.

Another challenge arises from the fact that every router in the area must know about every other router and the links that connect them. This creates size constraints, which these protocols manage by dividing the internal network into areas or instituting hierarchies. The largest single OSPF area we are aware of contained 1000 routers, but this is generally considered a poorly designed and needlessly large network design. These downsides are considered facts-of-life for link state protocols, and are well managed by a combination of the protocol implementations and reasonable network design.

CLOUD NETWORKING: THE LAYER 2 APPROACH

Unlike L3 networking, L2 switching is very easy to understand and use at a small scale. Simply attach any two servers to the same L2 network segment and they can find each other instantly, without involvement from L3 routers. Inside most data centers, L2 networking is the dominant method for connecting devices. While not able to operate at cloud scale, L2 does work well for small and medium deployments. When operating less than 1,000 servers together—most applications and data center needs are far less than 1,000 servers—L2 networking is easy to use, configure, deploy, and manage. As previously mentioned, many modern applications make assumptions about being able to use L2 protocols for discovery and easy networking.

It should be apparent at this time that at the application level L2 is a desirable networking technique. From a customer perspective, it's easiest to simply have lots of L2 networks, one for each application or functional area of a data center. The following diagram depicts how L2 networks are created inside enterprise data centers today:



Applications in Layer-2 Silos

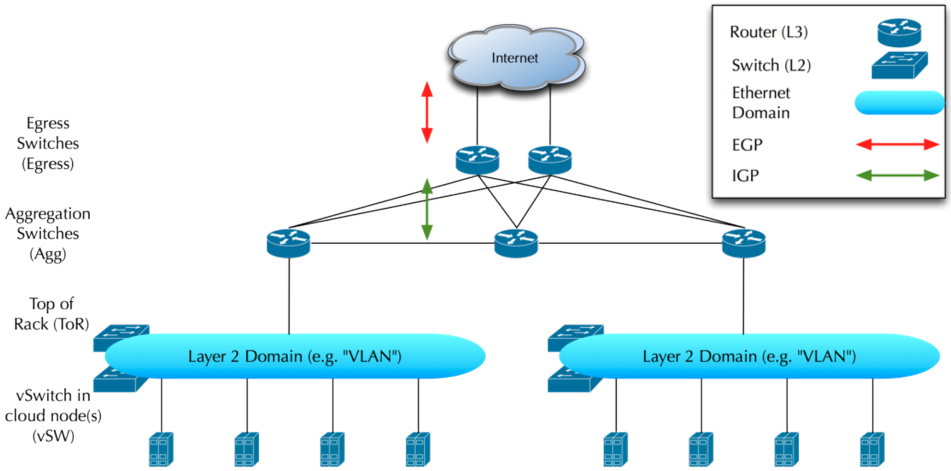
Unfortunately, L2 networking has significant complexity and scaling issues once you reach a certain size.

L2 in Cloud Data Centers

In contrast to L3, L2 networking means that each host knows how to talk to local hosts on the same network segment. A designated router (aka *default router* or *default gateway*) is usually assigned for the host to send all data that is not local.

It is convenient for a distributed application on multiple servers to have the other servers that make up the application on the same L2 network segment. This makes discovery easier, reduces the chances of an impact from a router failure, and implies that security devices such as firewalls are not interfering. In fact, there is an implicit contract between network operators and their customers. When an L2 network block (e.g. 192.168.0.0/24) is assigned to a customer, it is assumed that the customer can do what they want with the assigned network. This includes adding new machines or changing IP addresses. The only requirement in this contract is that the switch sends all nonlocal network traffic to their default router.

The following diagram depicts an L2 oriented networking topology:


Internet
Router (L3)
Switch (L2)
Ethernet Domain
EGP
IGP
Egress Switches (Egress)
Aggregation Switches (Agg)
Top of Rack (ToR)
vSwitch in cloud node(s) (vSW)
Layer 2 Domain (e.g. "VLAN")
Layer 2 Domain (e.g. "VLAN")

This is a simplified diagram, but it shows how L2 switching hides the topology. For each L2 domain all of the devices appear to be local, meaning here the top-of-rack switch (ToR) and the virtual switch (vSW) are invisible to the virtual machines. This simplifies networking. It also means that if a virtual machine is moved from one cloud node to a different cloud node in the same rack, it just works. On the other hand, if a virtual machine had to move between these two L2 domains, its IP address would have to change and it would now be routed (L3), requiring updates to ensure it can continue to speak with the other devices on its old network.

You will also notice that no network can be completely L2. This is because L2 networks are used only for local networks. L3 networking is required for reaching remote networks or the Internet. In other words, L3 networking can never go away.

L2 Pros and Cons

L2 networking (primarily Ethernet) has significant advantages over L3 (IP) in its simplified addressing and ease of use.

The shortcomings for large scale networks, however, are glaring. In particular, L2's flat addressing, which makes simplified addressing possible also makes having large scale networks difficult. IP's hierarchical addressing, in comparison, allows simplifying the problem through route aggregation.

While L2 has a flat address scheme, this also enforces a very strict tree topology where each switch must look for loops. This is the problem that Spanning Tree Protocol (STP) was designed to solve. Unfortunately, STP must shut down links in order to avoid loops, which means that ECMP is not possible. It also means that in L2 deployments most traffic takes the longest route possible, moving from the bottom of the tree, to the top of the tree and back down.

For example, in a large L2 data center network using gigabit Ethernet, it is not unusual to see "core links" between switches at the top of the tree that are 10 or 100 times the size of "edge links" (to hosts). This makes core links expensive as they are usually "fat pipes" such as 10Gig, 40Gig, and 100Gig Ethernet because all traffic must transit these links between core switches. This puts an undue burden on those switches, making them more expensive, and making L2 difficult to manage and scale for large networks.

Because L2 switching is inherently hard to scale, most attempts to make it more scalable appear to be "bolt-ons" or require the addition of L3 networking techniques. For example, RBridges (formerly TRILL) is an attempt to provide better topology and L2 switching information between switches using IS-IS. Other companies' attempts to make L2 networking scale, such as SEATTLE and Cisco's Layer 2 Multipath Protocol (L2MP), share the same characteristic of stretching L2 networks to behave like L3 routed networks. An ideal solution, of course, would be L3's scaling properties with L2's ease of use.

Despite its issues, a number of large cloud providers today do attempt to use L2 networking to provide each customer their own L2 network (aka "VLAN"). Most of these attempts work sufficiently well at small or medium size, but begin to fall apart rather quickly once significant size is achieved.

SUMMARIZING L2 VS. L3

This table summarizes the two approaches:

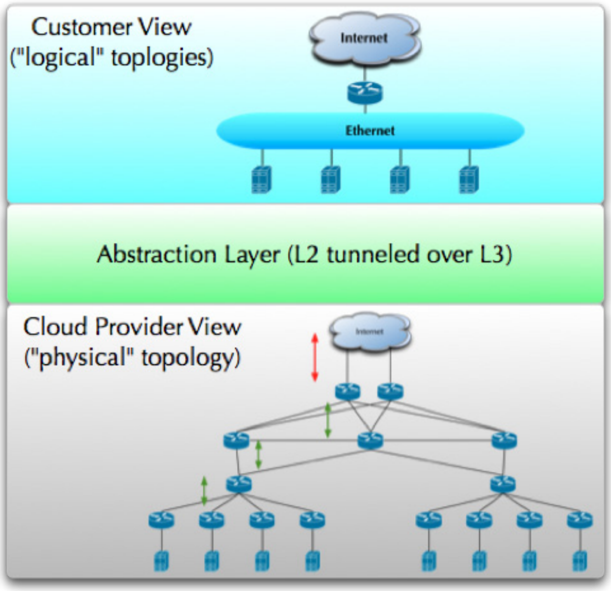
L3	L2
Fast convergence times	Fast convergence times only when tuned properly
Use all available bandwidth (via ECMP)	Simple to configure
Proven scalability	Tree topology works at small to medium scale only
The Internet is L3	A typical data center is L2

CLOUD NETWORKING

Network virtualization attempts to meld the L2 and L3 approaches. New approaches to cloud networking overcome the L2 scaling issues by either creating virtual L2 networks across L3, sometimes referred to as "L2 over L3"[6] or by eliminating the L2 scaling issues altogether.

With these approaches, we provide a layer of abstraction between the "physical" and "virtual" network topologies. Network customers may have as many L2 networks as they like, wired however they like, while the underlying physical networking is run as cloud scale L2 network or multiple L2 networks connected across L3 networks. This allows the best of both worlds. The tradeoff is some additional complexity and concerns around performance, most of which are being addressed today.

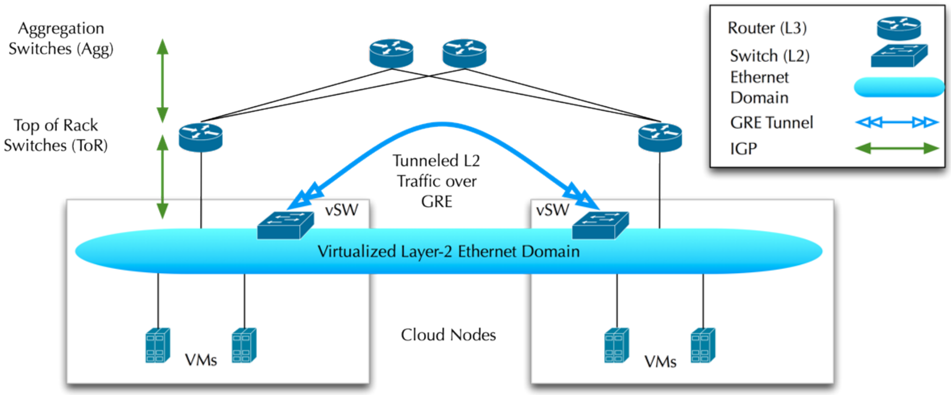
The diagram below shows how network virtualization works. A key characteristic of true network virtualization is the abstraction layer between the "physical" and the "virtual". This is akin to the hypervisor in a virtualized server. The hypervisor acts as an abstraction, hiding the physical hardware and providing a virtualized set of hardware for the guest operating systems.



Network Virtualization

[6] *Layer-3 is 'over' Layer-2 in the TCP/IP stack.*

This is still a bit abstract. Let's take a look at a more detailed network topology that shows how L2 over L3 actually works. The following diagram shows how each cloud tenant's vSwitch can provide a dedicated GRE tunnel to every other tenant, thereby creating a fully virtualized (and dedicated) L2 domain, much like a VLAN:


Aggregation Switches (Agg)
Top of Rack Switches (ToR)
Tunneled L2 Traffic over GRE
vSW
vSW
Virtualized Layer-2 Ethernet Domain
Cloud Nodes
VMs
VMs
Router (L3)
Switch (L2)
Ethernet Domain
GRE Tunnel
IGP

L2oL3 Datacenter Example

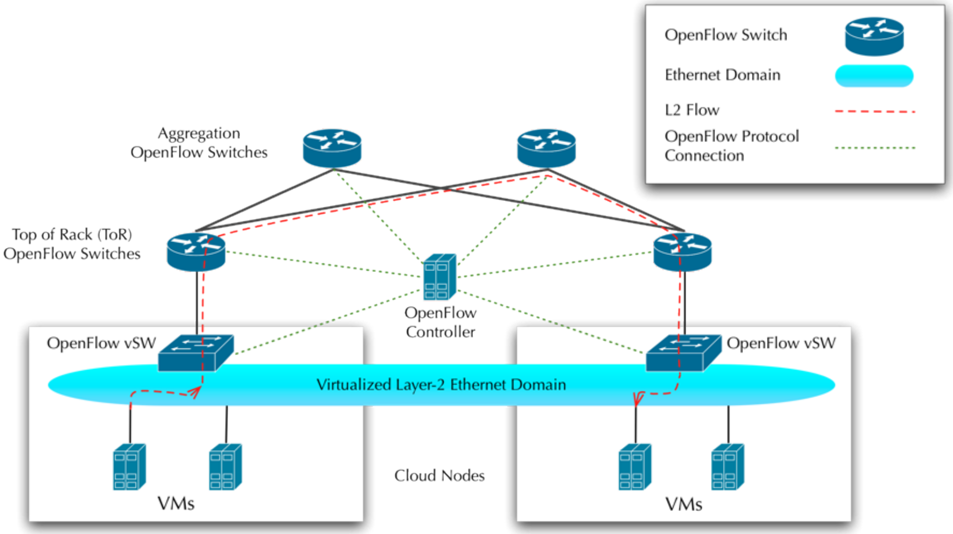
Seen here is that invisibly to all of the customer virtual servers traffic that is not on the local cloud node (shown as a white box) is tunneled over L3 using the GRE protocol.

OPENFLOW NETWORKING

OpenFlow is a new approach to providing scale-out virtualized L2 networks in the datacenter. In an OpenFlow network the switches do not use traditional L2 or L3 protocols to determine the links traversed or routes followed by traffic. Instead, an OpenFlow controller configures the switches, explicitly building L2 forwarding paths by directly populating the packet matching and forwarding tables in the switch hardware. The controller has full knowledge of the physical and virtual network topology and orchestrates the provisioning of all virtual networks across the complete set of connected OpenFlow devices.

OpenFlow enabled switches are essentially dumb "configless" devices which rely on the external controller to build virtual networks. This control is performed using the OpenFlow protocol, an open standard. The protocol defines a mechanism for defining flows by manipulating the forwarding tables of switches and for performing actions on these flows. Packets that match table entries are forwarded directly in hardware, while the first packet of an unmatched flow is sent to the controller, enabling new flows to be installed dynamically.

The following diagram illustrates a simple example where OpenFlow has configured a forwarding path from a VM in one cloud node to a VM in another:


OpenFlow Switch
Ethernet Domain
L2 Flow
OpenFlow Protocol Connection
Aggregation OpenFlow Switches
Top of Rack (ToR) OpenFlow Switches
OpenFlow Controller
OpenFlow vSW
OpenFlow vSW
Virtualized Layer-2 Ethernet Domain
Cloud Nodes
VMs
VMs

OpenFlow Datacenter Example

In this example the VMs that are part of the same virtualized Ethernet can transmit L2 frames to each other as if on a simple switch. OpenFlow dynamically builds the necessary forwarding entries to pass the frames from VM to VM across the network.

An all-OpenFlow network builds on the scale-out advantages that the L3 approach has over L2. With routed L3 networks the configuration is typically static, with routing and load balancing planned and configured in advance and stored in non-volatile memory on each L3 device. The path taken by L3 traffic will only change in response to link or device failure when the SPF algorithm is triggered to recalculate paths. In an OpenFlow network the network elements need to have very little static configuration, leaving the controller to make intelligent load balancing decisions at the time flows are created. The OpenFlow can plot optimal flow paths based upon the current state of the changing virtual network configuration.

A standard fat tree topology can be used for deployment however a flatter topology or mesh will also work well since OpenFlow will make use of all links at lower levels. Flat topologies can reduce the requirement for very fat expensive links at the top of the tree.

NETWORK VIRTUALIZATION IN CLOUD DATA CENTERS

Network virtualization is being converged on by many. Microsoft has written whitepapers detailing the L2oL3 technique from the perspective of a large scale data center operator, while folks like NEC and Nicira have been working on the programmatic and management aspects. For more information, we recommend reading Microsoft's original whitepaper on Valiant Load Balancing, and its update on 'VL2'.

There are a number of ways to implement network virtualization, and although it's relatively early days, we'll certainly see more. Examples include:

- Generic Routing Encapsulation (GRE)
- Virtual Distributed Ethernet (VDE)
- Custom solutions (e.g. L2 tunneling over UDP)
- OpenFlow (standalone)
- OpenFlow + GRE tunneling

All of these techniques allow the service provider to build a highly scalable network while giving each customer its own set of simple to use and easily configured L2 networks. In addition, because customers are completely isolated from one another, it is possible to run overlapping IP address ranges and provide advanced features such as quality of service (QoS) and distributed firewalls.

This is why we particularly like using OpenFlow; either standalone or along with GRE or UDP tunneling. GRE or UDP handles tunneling and OpenFlow can be used as a programmatic interface to switches (physical and virtual) and for advanced features like QoS.

OpenFlow

A brief aside on OpenFlow. Although OpenFlow is very new technology, it is already appearing in solutions available now or in the near future. The default vSwitch in the upcoming Citrix XenServer 6.0 release uses OpenVSwitch, an OpenFlow enabled virtual switch, NEC is now shipping OpenFlow enabled switches, and the Nicira team is working on providing OpenFlow enabled solutions.

Besides the advanced networking capabilities mentioned above, OpenFlow can be loaded as firmware onto physical switches, something the NEC team has developed, and others are developing. By running the same protocol for configuration across both physical and virtual switches it is also possible to run mixed cloud environments in which the tenant has both virtual and physical servers. This will be an important concept in the future, as some intensive workloads, such as databases and file servers, may not be virtualized.

CLOUDSCALING RECOMMENDATIONS

We are a vendor agnostic consultancy, but we would be remiss if we didn't point out a number of solutions that provide network virtualization that are already in use today. Some of these are solutions focused solely on network virtualization, while others are more like full cloud stacks that incorporate network virtualization in them.

NETWORK VIRTUALIZATION OPTIONS

NEC

NEC is a $35 Billion global leader in networking, offering a broad portfolio of IT and communications solutions. They are making significant investments in this space, actively working on network virtualization solutions that combine their physical switches running OpenFlow with innovative control and management systems. Partnering with Stanford and other leading organizations – including commercial enterprises - on early OpenFlow implementations, NEC is large, but moving fast. We consider NEC to be a good choice for enterprises looking to be early adopters in network virtualization while minimizing risk.

Nicira Networks

One of the pioneers and leaders in this space, the Nicira team is a fast-moving startup that is managed by the people who developed OpenFlow at Stanford, as well as team members who worked on virtual switching at both VMware and Cisco. Nicira has been running early pilots with large clouds that use network virtualization, making it an attractive choice.

Cloud.com

Cloud.com's[7] CloudStack is one of the most productized of the Infrastructure-as-a-Service (IaaS) software packages. CloudStack uses network virtualization to provide L2 domains between clients. Cloud.com chose a custom UDP tunneling solution to optimize performance. By writing their own solution, integrating it tightly with the hypervisor, and removing certain functionality (e.g. TCP checksumming which is usually handled by NICs), they built a higher performing network virtualization solution. CloudStack also provides a way to do L2 VLAN networking using physical hardware, although we don't generally recommend this technique for large providers due to the inherent issues around scalability and lack of flexibility.

OpenNebula and Eucalyptus

Both of these offer strong open source projects and both support VDE. We generally prefer OpenNebula over Eucalyptus for its ease of use and modular nature, and because Eucalyptus pulled away from its open source roots, while OpenNebula has

[7] *Formerly called VMOps*

embraced them. If you wanted to roll your own using GRE or OpenFlow+GRE, we recommend starting with OpenNebula.

SUMMARY

Providers need scalability and cloud users need the ease and familiarity of simple Ethernet networking. Network virtualization provides this. It is truly virtualization in that it provides a clean abstraction layer that creates a separation of concerns between the cloud provider and cloud user. There are potentially some downsides in terms of throughput efficiencies, but with 10GE on the horizon, technologies such as OpenFlow and the tremendous amount of power now available in a single 1U rackmount server, small inefficiencies are more than overridden by major gains in scalability.

We believe network virtualization is the way of the future of large cloud data centers. It simplifies networking for all and avoids complex bolt-on technologies. Best of all, it plays inherently to the strengths of commodity systems. Instead of buying increasingly expensive networking gear, it is much less expensive to scale-out using L3 networking techniques on cheap equipment and then use the abstraction layers to hide it all beneath.

Cloudscaling is available for engagements to assist you with your cloud data center needs. Please inquire further if you have questions.

ABOUT CLOUDSCALING

Cloudscaling builds the world's largest Infrastructure-as-a-Service clouds for telecom companies, service providers, and enterprises. We offer a complete suite of services from initial strategy and planning, through implementation and support. The Cloudscaling team is comprised of cloud veterans from industry leading companies such as Amazon, GoGrid, RightScale, Opscode, Canonical, Reductive Labs, Engine Yard, BitTorrent, VMware, eBay and Microsoft.

Want to learn more? Here's how you can reach us:

Website	cloudscaling.com
Twitter	@cloudscaling
Email	info@cloudscaling.com
Phone	(877) 636-8589
Int'l	+1 (415) 508-3270