EANTC

EANTC's Independent Test of Cisco's CloudVerse Architecture

Part 1: Cloud Data Center Infrastructure Including Business Applications

LIGHT READING

IXIA®

Photos taken by Kim Ringeisen (Flying Monkey Photography)

Introduction from Light Reading

In December, Cisco Systems Inc. (Nasdaq: CSCO) introduced CloudVerse, its approach to building and managing cloud-based networks. In his story about the announcement, Light Reading's Craig Matsumoto listed the key components of CloudVerse:

- Cisco Intelligent Automation for the Cloud (CIAC), tools for the provisioning of services. CIAC is part of a wider framework called Cisco Unified Management, which includes automation and orchestration software, including technology from two Cisco acquisitions: newScale (service catalogs and service-provisioning portals) and Tidal Software (tool for monitoring application performance to detect problems ahead of time).
- Cisco Network Services Manager, which handles virtualization of the data center's networking elements (routers, switches, load balancers, firewalls, etc.). It can set up provisioning and policy universally, so that all these pieces don't have to be configured individually.
- Cloud-to-Cloud Connect, a way of letting data centers connect to the cloud more dynamically. A key ingredient here is Cisco's Network Positioning System (NPS) being added to the ASR 9000 and 1000 routers. NPS, originally introduced on the CRS-3 core router, searches the network/cloud for alternative resources when capacity limits get reached.

Pre-tested applications: Cisco has about 50 of them ready to add to CloudVerse, the company said at the time. These are meant to help kick-start a carrier or enterprise's cloud offerings. The examples Cisco intends to emphasize on its webcast deal with enterprise collaboration.(see video links in online version)

With the scene and the context suitably set, we'll now include some terms we'll be using in this report below for you edification and then we'll hand the baton to our testing partner European Advanced Networking Test Center AG (EANTC) to explain its tests of Cisco's CloudVerse.

Table of Contents

Introduction from Light Reading 2
Cisco's Unified Data Center 2
Test Methodology 3
BMC Cloud Lifecycle Management Integration 5
Multi-Tenancy Isolation 6
FabricPath 7
Tiered Cloud Services 9
Unified Fabric (UF) – UCS Manager 10
Virtual Machine Fabric Extender Performance ...11
Virtual Security Gateway 13
Locator/ID Separation Protocol (LISP) 14
Provisioning: Cisco Network Services Manager 14
Enterprise Applications: Siebel CRM 15
Cisco's Hosted Collaboration Solution.....16
Conclusion: Unified Data Center Test 17

TABLE 1. Acronyms Used in This Report

Terminology	Description
ACE	Application Control Engine
BMC CLM	Cloud Lifecycle Management
CGSE	Carrier-Grade Services Engine
FC	Fiber Channel
FCoE	Fiber Channel over Ethernet
HCS	Hosted Collaboration Solution
Multi-tenancy	Multiple users accessing virtual servers
SAN	Storage Area Network
SLA	Service Level Agreement
SLB	Server Load Balancer
SDU	Systems Development Unit
UCS	Unified Computing System
UF	Unified Fabric
VM	Virtual Machine
VNMC	Virtual Network Management Center
VMDC	Virtualized Multi-Tenant Data Center – Cisco's cloud architecture
VM-FEX	Virtual Machine Fabric Extender
VPC	Virtual Port Channel
VSG	Virtualized Security Gateway
VSM	Cisco Nexus 1000V Virtual Supervisor Module

Cisco's Unified Data Center

When Light Reading asked us to conduct a suite of tests for Cisco's converged data center we were far from surprised – these days clouds are everywhere and Cisco has all the components one would need to offer cloud services.

Cisco sees itself as a one-stop shop for every service provider interested in rolling out cloud services or upgrading such services. They offer both networking infrastructure elements and the actual components of the data center.

In addition to the components, we knew from our work with service providers that managing cloud services and data center components can come at a high cost. Cisco's answer is a comprehensive system. The company has worked hard to merge the various server and network elements that typically exist in a data center in order to present a unified

system to the market. One result of this effort is Cisco's Unified Computing System (UCS). This article documents our attempt to quantify and verify some of Cisco's UCS latest features and capabilities. Of course we cannot forget about the supporting network infrastructure, which has undergone changes of its own. We look at data center infrastructure in later articles. Some believe that this process of unification Cisco described means Fiber Channel over Ethernet, but to Cisco this is a very small piece of the story, so much so that we didn't even test much in that area.

Intriguing, but we had questions: What do we feel is important? For over six months we brainstormed internally and discussed with Cisco before spending yet another month pre-staging and conducting the testing in Morrisville, N.C. We set out to answer the following questions: How does the data center infrastructure scale? Which services and applications are cloud ready? Is the infrastructure ready to be migrated to IPv6? Were any key pieces – security, virtualization, scale, multi-tenancy, prioritization, performance, server, software and network components – all there? As we broke these questions down into more specific tests of particular components we also built an understanding of how Cisco answers these questions, and how we could put them to test. We hope you find the answers you're looking for as well.

About EANTC

The European Advanced Networking Test Center (EANTC) is an independent test lab founded in 1991 and based in Berlin, Germany that conducts vendor-neutral proof-of-concept and acceptance tests for service providers, governments and large enterprises. EANTC has been testing MPLS routers since early 2000s for both online publications and interoperability and service providers.

EANTC's role in this program was to define the test topics in detail, communicate with Cisco, coordinate with the test equipment vendor (Ixia), and conduct the tests at the vendors' locations. EANTC engineers then extensively documented the results. Cisco submitted their products to a rigorous test in a controlled environment contractually defined. For this independent test, EANTC exclusively reported to Light Reading. Cisco did not review the individual reports before their release. Cisco had a right to veto publication of the test results as a whole, but not to veto individual test cases.

— Carsten Rossenhövel is Managing Director of the European Advanced Networking Test Center AG (EANTC) , an independent test lab in Berlin. EANTC offers vendor-neutral network test services for manufacturers, service providers, governments and large enterprises. Carsten heads EANTC's manufacturer testing and certification group and interoperability test events. He has over 20 years of experience in data networks and testing.

Jonathan Morin, EANTC, managed the project, worked with vendors and co-authored the article.

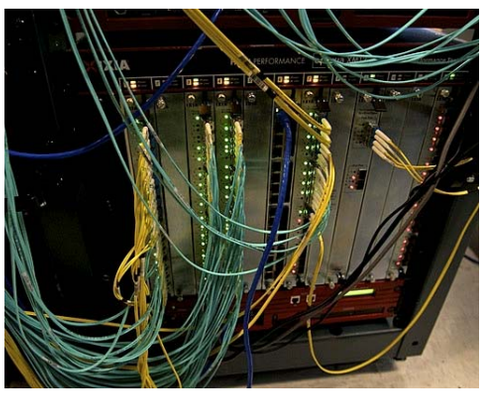
Test Methodology

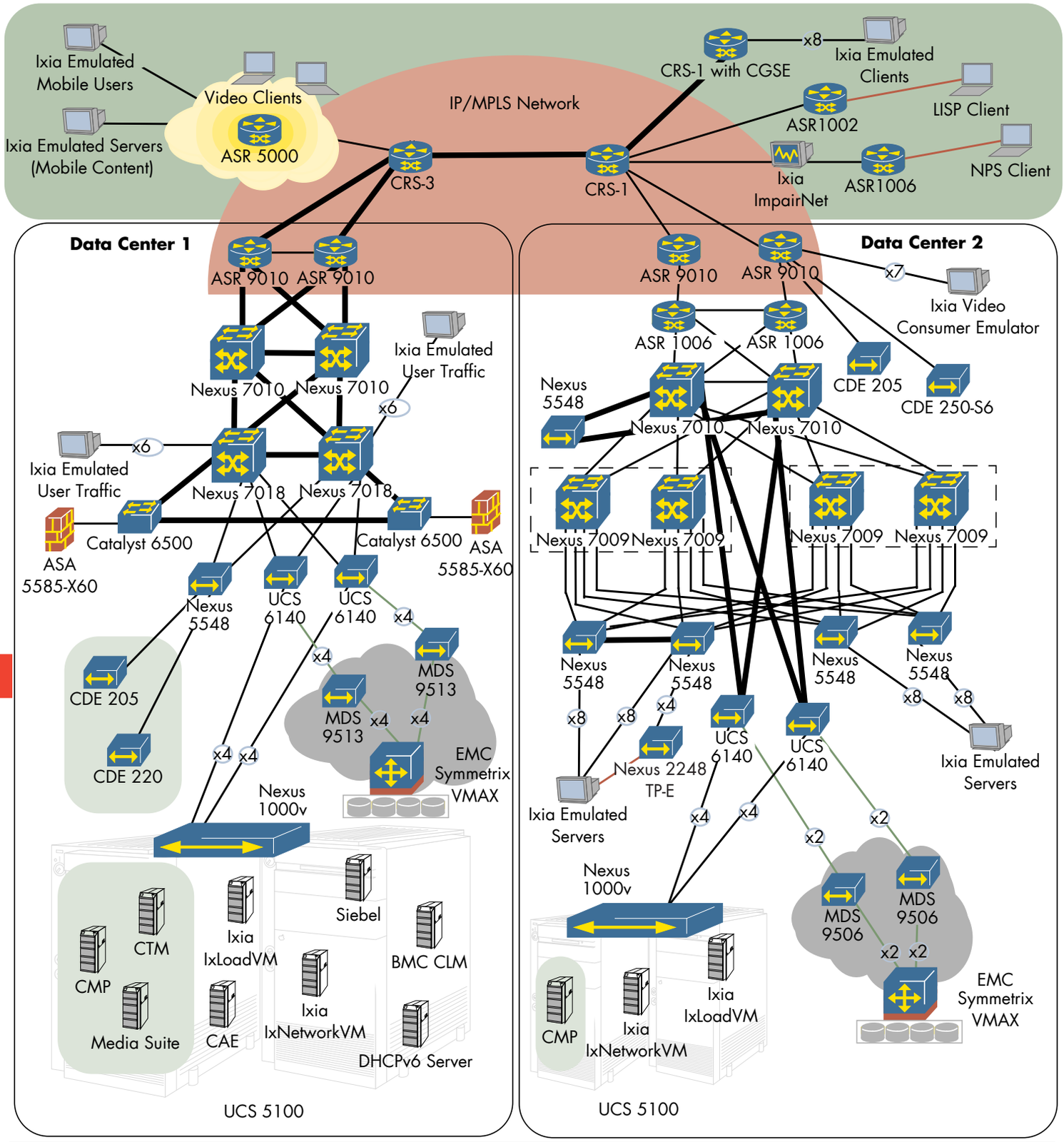
Testing cloud solutions is complex for a several reasons. Cloud testing includes infrastructure tests that could be evaluated using standard network test equipment. But cloud testing also includes the virtual server space that standard testing tools cannot easily explore, not to mention the added complexity of shared memory, CPU, network and storage resources. Luckily Ixia (Nasdaq: XXIA) was able to support both type of tests offering hardware to test the infrastructure, virtual Ixia tools to test within the virtual space, and configuration assistance to put it all together.

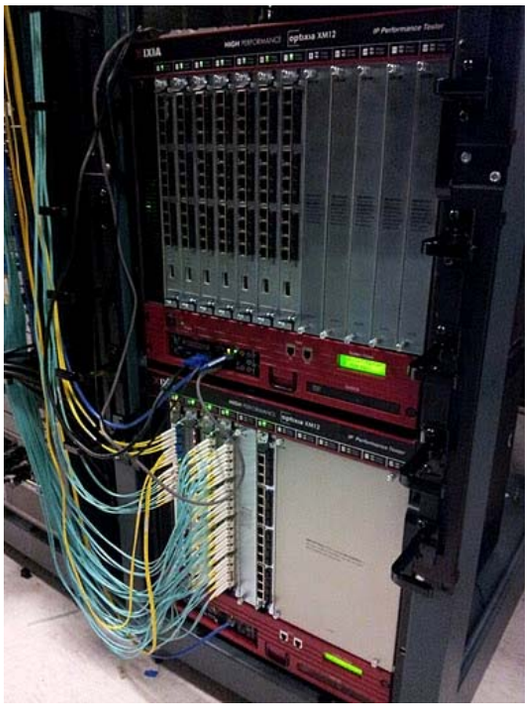
All together we used two XM12 chassis and one XM2 with the modules listed below:

- Xcellon-Ultra NP Application Traffic Testing Load Module: Used for Layer 4-7 testing with 10 Gigabit-Ethernet interfaces, such as for our Videoscape or PCRF tests (to be seen in an upcoming article)
- Xcellon-Flex Accelerated Performance and Full Emulation Load Modules: Used for Layer 2 to 3 testing with 10-Gigabit Ethernet interfaces, such as for our Fabric Path test
- LSM1000XMVDC16-01: Used for Layer 2 to 3 testing that required Gigabit-Ethernet interfaces
- Xcellon-Ultra XT/XTS Application Traffic Testing Appliances

We made use of both IxNetwork software for tests that required Ethernet/IP (Layer 2/3) traffic, and IxLoad when state-full traffic like HTTP, streaming video or emulated mobile traffic was required. Here is a photo of the XM12s:




Ixia Emulated Mobile Users
Ixia Emulated Servers (Mobile Content)
Video Clients
ASR 5000
IP/MPLS Network
CRS-3
CRS-1
CRS-1 with CGSE
x8
Ixia Emulated Clients
ASR1002
LISP Client
Ixia ImpairNet
ASR1006
NPS Client
Data Center 1
ASR 9010
ASR 9010
Nexus 7010
Nexus 7010
Ixia Emulated User Traffic
x6
Ixia Emulated User Traffic
x6
Nexus 7018
Nexus 7018
ASA 5585-X60
Catalyst 6500
Catalyst 6500
ASA 5585-X60
Nexus 5548
UCS 6140
UCS 6140
x4
x4
MDS 9513
MDS 9513
x4
x4
CDE 205
CDE 220
x4
x4
EMC Symmetrix VMAX
Nexus 1000v
CTM
CMP
Media Suite
Ixia IxLoadVM
CAE
Siebel
Ixia IxNetworkVM
BMC CLM
DHCPv6 Server
UCS 5100
Data Center 2
ASR 9010
ASR 9010
x7
Ixia Video Consumer Emulator
ASR 1006
ASR 1006
Nexus 5548
Nexus 7010
Nexus 7010
CDE 205
CDE 250-S6
Nexus 7009
Nexus 7009
Nexus 7009
Nexus 7009
Nexus 5548
Nexus 5548
Nexus 5548
Nexus 5548
x8
x8
x4
x8
x8
UCS 6140
UCS 6140
Nexus 2248 TP-E
Ixia Emulated Servers
Ixia Emulated Servers
x4
x4
x2
x2
Nexus 1000v
MDS 9506
MDS 9506
x2
x2
EMC Symmetrix VMAX
CMP
Ixia IxNetworkVM
Ixia IxLoadVM
UCS 5100
Videoscape
SAN
Mobile Core
Access Network
IP/MPLS Network
Aggregated Physical Links
Physical Chassis
xN
N x Physical Links
10 Gigabit-Ethernet
Fibre Channel
Gigabit-Ethernet



For the virtual tools, we made use of both Ixia IxLoad VM and Ixia IxNetwork VM, for state-full and stateless traffic, respectively. Finally you will also see that in some cases we are looking at Cisco's applications. In these cases we had to evaluate Cisco applications that did not allow us to use an Ixia tool and thus we had to come up with new methodologies.

BMC Cloud Lifecycle Management Integration

EXECUTIVE SUMMARY: BMC Cloud Lifecycle Management successfully provisioned new tenants and their network container services throughout the data center, as well as their respective virtual machines (VMs), all from a single user interface.

Deliberations about new investments, for network operators and especially for cloud providers, typically focus more on operational cost than capital expenditure. The questions that come up include: How will a new system be operated, and will this ultimately mean an increase or decrease in operational efforts? Will administration of the systems become a complex task once resources are shared within a virtual space?

Given the scalability and cost effectiveness that service providers expect from cloud service platforms, automated service creation (provisioning) is a key aspect. Provisioning is literally the first activity of a network operator when a service has been commissioned. Thus it was on the top of our list of things to evaluate.

Cisco explained that they partner with BMC Software Inc. (NYSE: BMC) to provide provisioning software, with the goal of easing the pain for the administrator and reducing the ramp-up time for new cloud tenants and services.

BMC's Cloud Lifecycle Management (CLM) promises to provision tenants' data center services (cloud customers) and virtual machines across the data center, all through a single interface. As an umbrella system, CLM makes use of other tools to manage individual elements – BMC BladeLogic Network Automation (data center network provisioning) and BMC BladeLogic Server Automation (VM and application provisioning, which interfaces directly with VMware's Vcenter). Our interest was to put this solution to use in the lab, and to take a peek under the hood to see how we could use it to provision services, providing readers with an idea of the administrator's experience.

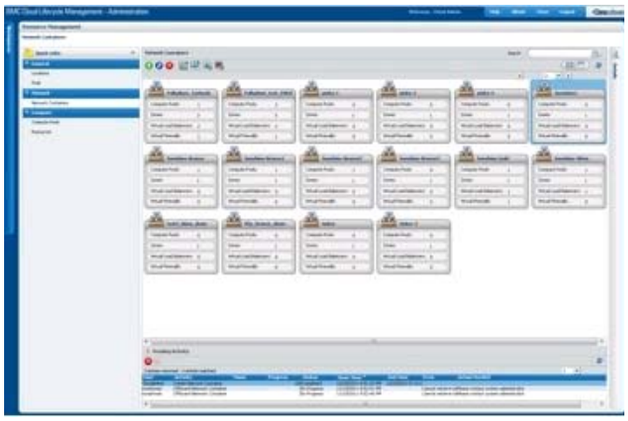


Figure 2: BMC Cloud LifeCycle Management Tenant View

In a cloud services data center, even management applications such as BMC CLM can run on virtual machines. We sat down with Cisco's technical marketing engineers and went through the two tasks at hand – tenant provisioning and VM provisioning. We started provisioning a single tenant/VM pair, then moved to bulk amounts. To provision a new tenant, CLM in fact opened telnet sessions to configure each component via command-line interface (CLI), just as an administrator would.

The Cisco routers, switches and firewalls we configured were two ASR 9010s, four Nexus 7000s, two Catalyst 6500s, ACE 30 and Firewall Services Module (FWSM), the UCS system and the Nexus 1000v virtual switch. BMC CLM added all the necessary VLANs and IP subnets, all taken from a pool that had to be configured initially. The Cloud LifeCycle Manager automatically logged in to the Cisco equipment via telnet protocol, using the command-line interfaces (CLIs). Once this was completed, we compared the configurations before and after, and sent some quick pings to see that they were active.

Then we moved on to provision a Virtual Machine for this tenant. BMC CLM created a VM from its predefined template, connected it to the appropriate tenant, and turned it on, all from our one user interaction. We were able to open a remote desktop session through the network – it worked. Finally, we repeated the whole process but enabled bulk jobs. It took some time to get started after we had some failed jobs due to other (human) administrators accidentally stepping on CLM's toes – an inherent

problem with multi-user configuration management.

Cisco confirmed that it would be possible to configure how CLI sessions are maintained/interrupted, but they spared effort for this test. Once this was cleared up, we were able to successfully provision five tenants and then 10 VMs for each tenant. The single tenant took less than 25 minutes to create – all through a single job request from the administrator. The bulk tenant job for five tenants took just over an hour, and the bulk VM job just under an hour – the exact time taken of course depends on management hardware variables like computer power. The BMC CLM tool saved us as admins from opening a CLI session to the many devices in the infrastructure, having to copy each and every VM, and coordinating the whole process.

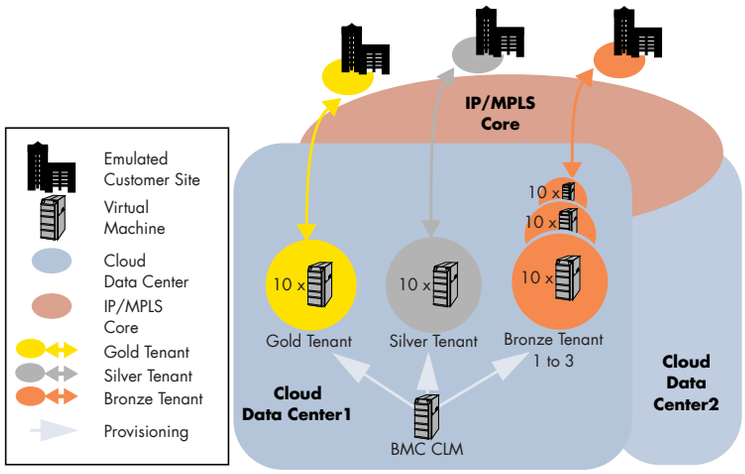


Figure 3: BMC CLM Logical Test Setup

The tool did the job we expected it to. In addition, we witnessed the "tenant portal" – the part of BMC CLM the customer would see. We provisioned some VMs, and even shut some down administratively while they were used by other users in the lab. The "others" didn't appreciate that, but it worked.

Multi-Tenancy Isolation

EXECUTIVE SUMMARY: All tenants were completely isolated from each other on virtual servers, throughout the data center network, to core network VPNs.

One of cloud service customers' main concerns is security in terms of isolation: If I can access this application, virtual machine or service from my network, who else can?

Firewalls are certainly a big part of the security story in the cloud, but at least for enterprise users, they are not sufficient. The service must be completely isolated from other services – much like a VPN. In fact, VPNs, amongst other technologies such as VLANs and virtual switching instances, are used in Cisco's Virtualized Multi-tenant Data Center (VMDC) reference architecture, which was leveraged for this test program. Since there are several ways to design the network, with countless combinations of how the various systems are configured, the architecture helped both to provide a reference for the test program, and to define conventions – for example, how gold tenants will get firewall services, and how gold and silver tenants will get load-balancing services.

To verify tenant isolation we pulled out a legacy test methodology for MPLS VPNs. Using Ixia (Nasdaq: XXIA) virtual tools, we deployed 54 Ixia IxNetwork VMs – one in each of the fifty four tenants – and attempted to transmit traffic in a full mesh. We expected 100 percent loss. In parallel, we sent traffic between each tenant and the outside core network, which we expected to work resembling acceptable use. For this traffic, we defined a Cloud Traffic Profile with Layer 3 traffic resembling a series of realistically emulated applications using Ixia hardware. In addition we set up yet 24 more Ixia IxNetwork VMs to emulate normal traffic within the data center (so-called "east-west" traffic) sent in a full mesh pattern at 500 Mbit/s per

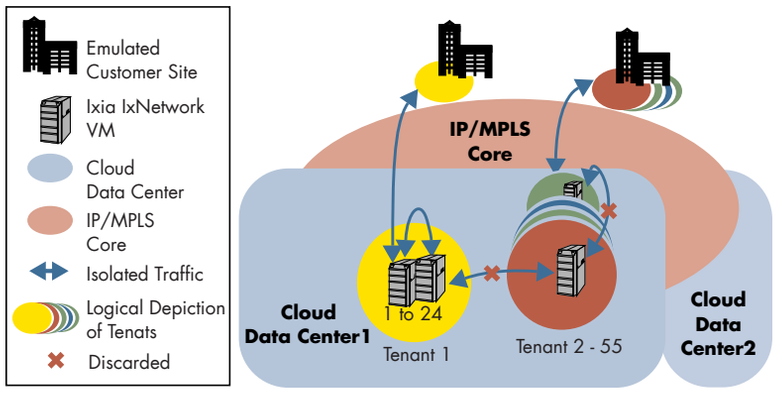


Figure 4: Logical Tenant Isolation Test Setup

VM. The 24 VMs were distributed across three UCS chassis – each chassis configured as a single ESX cluster, eight blades per cluster, one Ixia VM per blade. The pie charts below show the traffic distribution toward the fifty four tenants' users, which was also used for our QoS test. (See Tiered Cloud

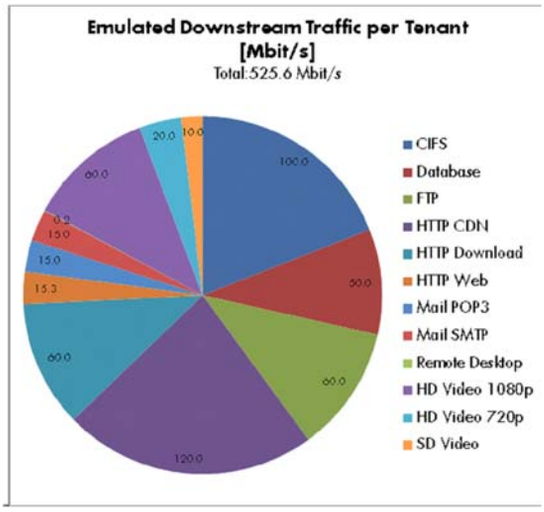


Figure 5: Emulated Downstream Traffic per Tenant

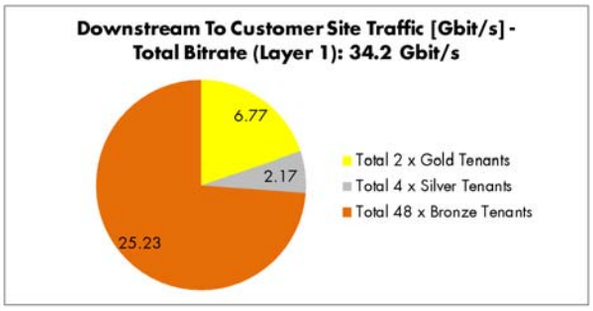


Figure 6: Total Downstream Traffic

Services.)

After running all traffic configurations simultaneously for 209 seconds (each Ixia configuration was running in a separate system, some ran longer), we correctly observed 100 percent loss on traffic between tenants, zero loss for traffic toward the customer. There was a very minor amount of 0.00014 percent loss on the traffic from the 24 Ixia IxNetwork VMs within the single tenant that we had expected to pass. The team explained that it is possible to achieve a lossless virtual environment with software switching, but with all the services we were running for this test, and the services for other tests still running in parallel, this was a very low amount of loss. Additionally, we pinged the different gold firewalls from different tenants' VMs, and correctly only received responses for those we expected access.

FabricPath

EXECUTIVE SUMMARY: FabricPath, using 16-by-10 Gigabit Ethernet links throughout the topology, forwarded 292.8Gbit/s of net traffic in the data center while also providing resiliency with sub-200-millisecond outage times and allowing for bursty traffic.

We're seeing more and more services looking to be hosted in the cloud while more and more tools are enabling those services to do so. The New York Times recently reported that, while the economy continues to be slow, data centers are booming and companies are reporting cloud-related growth each quarter. (See Cisco Sees 12-Fold Cloud Growth.)

How will the infrastructure support all this growth? Virtualization is only one part of the story. What about the network? Will standard bridging, link aggregation and Spanning Tree do the trick?

Not really. Cisco and other interested parties are contributing to new standardized protocols like TRILL – Transparent Interconnection of Lots of Links. In our test bed, we calculated massive traffic requirements to and from the virtual machines. Cisco configured its solution, FabricPath, which incorporates TRILL and other Cisco technology in order to scale the number of paths in the network, scale the bandwidth in the data center and lower out of service times in the case of a failure. Furthermore, Cisco wanted to quantify the buffering power of their latest "fabric extender," which goes hand in hand with their FabricPath architecture. We looked at each solution one at a time.

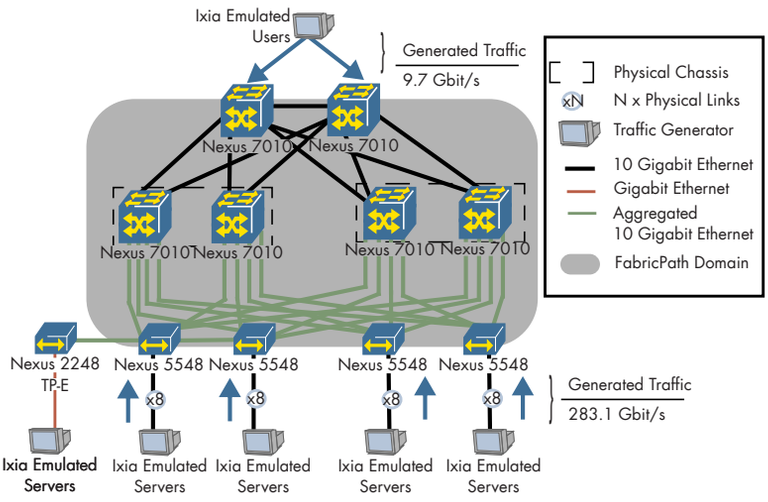


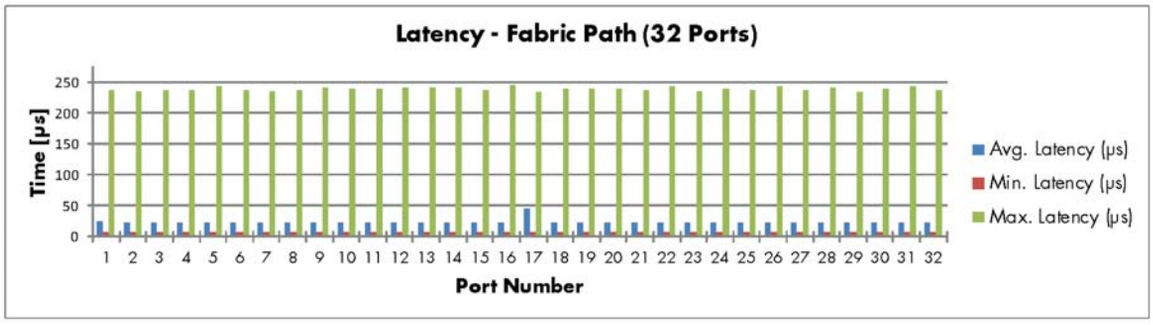
Figure 7: FabricPath Setup

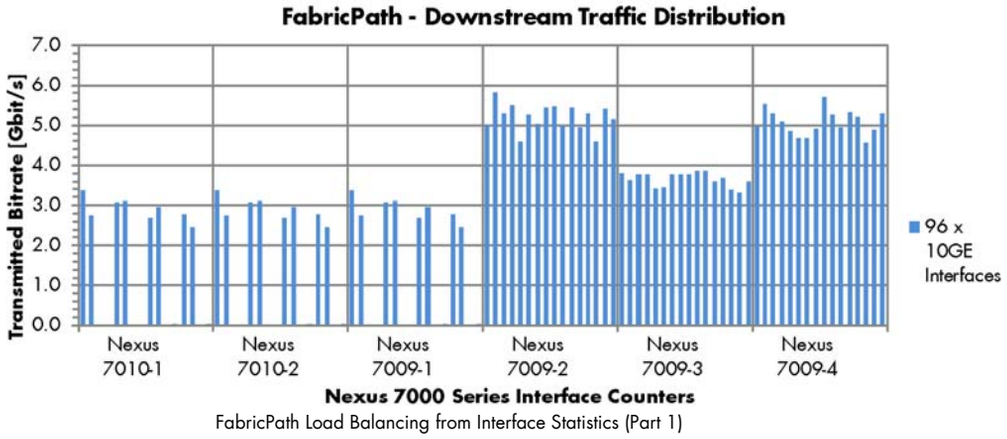
Given the scale of the test, and that there was no UCS included in the setup, we used hardware-based Ixia tools running IxNetwork to emulate all hosts. The Ixia test equipment was directly connected to Nexus 5548 switches. Each of these switches had sixteen

physical connections to each upstream end of row switch – the Nexus 7010. Most traffic was transmitted from within the data center, as would normally be the case. We ultimately emulated 14,848 hosts spread across 256 VLANs behind the four Nexus 5548 switches, transmitting a total of 273.9Gbit/s of traffic to each other in pairs, while also sending 9.2Gbit/s of traffic toward the emulated users located outside the data center (283.1Gbit/s in total), who in return sent back 9.7Gbit/s to emulate the requests and uploads. This added up to a total of 292.8Gbit/s traversing the FabricPath setup, for ten minutes, without a single lost frame.

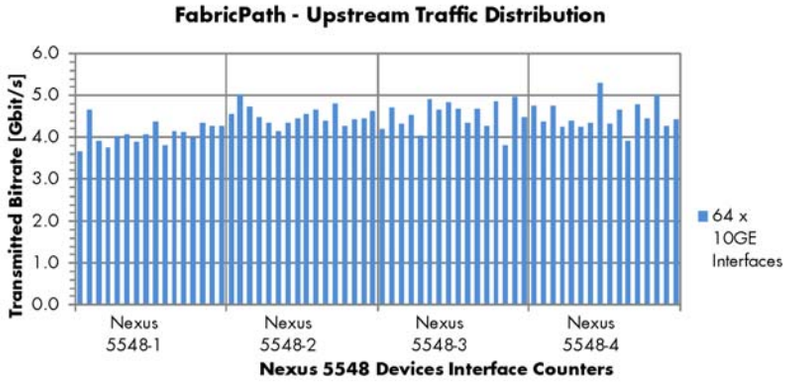
The total FabricPath capacity per direction was 320Gbit/s, given the number of links that were hashed across. Our traffic was unable to fill the 320Gbit/s completely, but it was still indeed a hefty amount of traffic. Below we have graphed the latency as well as the load distribution within the network (as reported via Cisco's CLI) to show how evenly the hashing algorithm distributed the load.

Now that we had measured the performance, what happened upon link failure? Cisco claimed we should see shorter outages compared to those experienced during failures in spanning tree networks. Measuring the out-of-service time from a failure in our scenario was much less than straightforward. The major testing problem was FabricPath's strength – traffic distribution by hashing – which was sort of unpredictable looking from the outside. We created an additional traffic flow of minimal load – one user – at 10,000 frames per second, and tracked its associated physical path in the FabricPath domain. Once we found the link, we physically pulled it out, and plugged it back in, while running traffic, three times. The link failure results are shown below. When we replaced the link, in all three cases, zero frames were lost.





FabricPath Load Balancing from Interface Statistics (Part 1)



FabricPath Load Balancing from Interface Statistics (Part 2)

Figure 8: FabricPath Latency Results

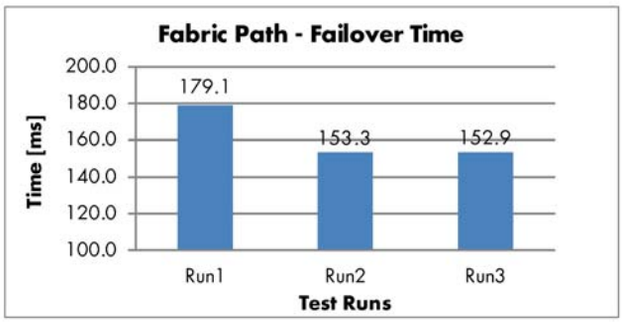


Figure 9: FabricPath Failover Times

Finally, we wanted to validate one of Cisco's claims regarding their Fabric Extender, or FEX, a standard part of their installation when data centers are keeping legacy Gigabit Ethernet links.

Cisco explained the FEX is an interface card that must not be located in the Nexus 5548 chassis, but rather in its own chassis – in this case the Nexus 2248. This allows the card itself to be placed at the top of a data center rack, for example, as an extension of the end-of-row switch. This way when more ports are needed across a long distance, operators need not invest in a new top-of-rack switch, just a new card, thus extending other top-of-rack or end-of-row switches. Although this card is designed for Gigabit Ethernet-based servers, it is likely to be used in data centers that also have 10-Gigabit Ethernet. Thus, Cisco explained, it was important to design the card with large buffers, accommodating for bursty traffic coming from a 10-Gigabit Ethernet port toward a Gigabit Ethernet port.

How bursty could that traffic be in reality?

We connected the appropriate Ixia test equipment as shown in the diagram. Different burst sizes were configured on the Ixia equipment until we found the largest burst size that just passed FEX without loss. We set the inter-burst gap to a large value – 300 milliseconds – so we could send constant bursts, but the individual bursts would not affect each other. We repeated this procedure twice, once using IMIX (7:70, 4:512, 1:1500) frames, and once using 1,500-byte frames. The burst size that observed no loss with IMIX was 28.4 MB and 13.4 MB for the 1,500-byte frames. Both tests ran for three minutes without any loss. Latency was as high as expected, since we expected the buffers to be used: from 3.0 microseconds to 98.8 milliseconds for the IMIX bursts ranging from 3.2 microseconds to 204.5 milliseconds for the bursts of 1,500-byte frames.

Tiered Cloud Services

EXECUTIVE SUMMARY: Application-specific quality-of-service prioritization across the network is crucial for cloud services. Cisco's ASR9010 managed quality under congestion well, maintaining low latency for high-priority cloud service patterns while also not starving out low-priority cloud customer traffic.

Delivering quality in the cloud is a task that bears many building blocks. When considering the packet network alone, quality of service (QoS) is already a broad term that raises a number of design questions. What algorithms and policies are used to prioritize traffic? What traffic characteristics need to be expected? Is latency or loss more critical? In the cloud, one would expect three tiers (Gold, Silver and Bronze) of application traffic at minimum. In order to come up with realistic traffic patterns, we made a few assumptions.

In almost all cases – whether business, mobile or residential services – the traffic patterns are typically asymmetric. As compute platforms, cloud services are likely to generate more traffic than they receive. In effect, there will be little congestion on the switches connected towards the servers. The other direction – from the virtual machines to the Internet – will surely be used more heavily.

Generally, a healthy traffic mix is about one-third Gold traffic and 10-to-30-percent Silver traffic, with the remaining traffic filled by emulators.

In the short term, some cloud operators might have much configured gold (a.k.a. high-priority enterprise) traffic. Typically gold services are under-booked and in turn over-provisioned so this should not be a concern. It is more of an issue if there are too many lower-paying customers, who sign on easily, and become overpopulated.

Reviewing the standard network topology, we figured the congestion point would often be the ASR 9010, and the cloud traffic profile we designed – with an overload of Bronze user traffic – would fit just right for our use case. We only had to remove links between our data center and our core network in order to emulate the typical situation of having more bandwidth available in the data center than upstream toward the network.

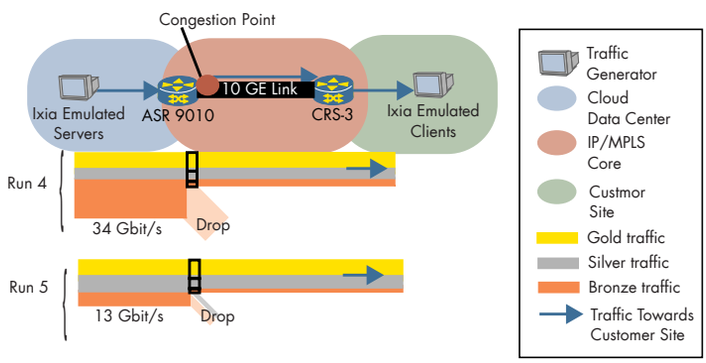


Figure 10: Traffic for Runs 4 and 5

In addition to prioritizing Gold tenants over Silver and Silver over Bronze, Cisco configured a percentage of bandwidth for each class to be guaranteed: 70 percent for Gold, 20 percent for Silver and 9 percent for Bronze bandwidth, so that no single class of customers could get completely blocked even if higher priority traffic aims to monopolize a link.

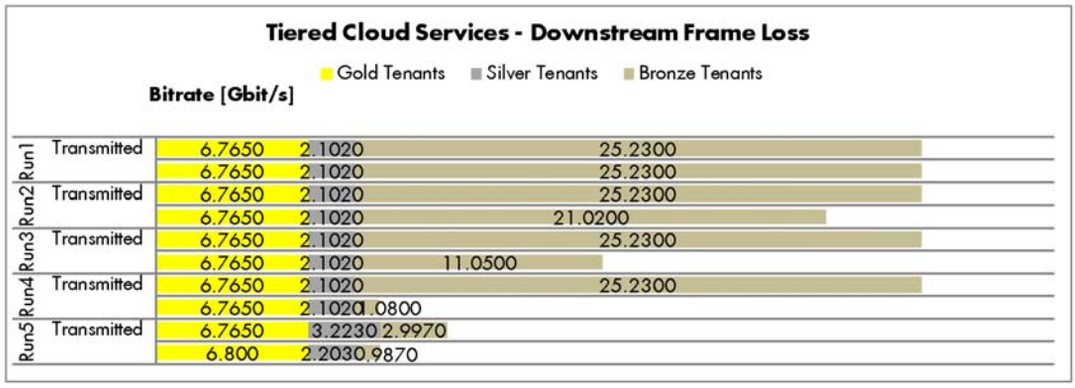


Figure 11: Traffic Loss per Test Run

In order to evaluate the QoS prioritization efficiency, we undertook the following test steps.

Step 1: Transmit all north-bound and south-bound bidirectional traffic between the users and the data center from our data center traffic profile and verify that there is no loss, as a baseline. Check.

Step 2: Reduce the bandwidth available between the data center and the core network by removing all links between one of the two ASR 9010s, and removing one link from the second ASR 9010, to create a minimal amount of congestion. Only bronze customers were affected. Check.

Step 3: Remove another link from the remaining ASR 9010, leaving two links left, or 20Gbit/s upstream and downstream. This still left enough bandwidth for Gold and Silver traffic, and only affected Bronze customers, as expected. Check.

Step 4: Remove yet another link between the ASR 9010 and its upstream CRS-3 peer, leaving only a single 10-Gigabit Ethernet link remaining. At this point the Cisco team told us the router was experiencing fabric congestion due to the high congestion, thus not necessarily guaranteeing the dedicated percentages explained above, which are applied after the fabric. Nevertheless, Gold and Silver were forwarded within their dedicated percentages. We observed loss only on Bronze traffic.

Step 5: We then decreased bronze traffic to avoid fabric congestion, and increased Silver traffic to see if Bronze tenants would still get their dedicated 1 percent of traffic. They did.

We also checked to see if latency increased, particularly for Gold tenants, during the various congestion scenarios. The Gold latency remained consistent, as shown in the graph below:

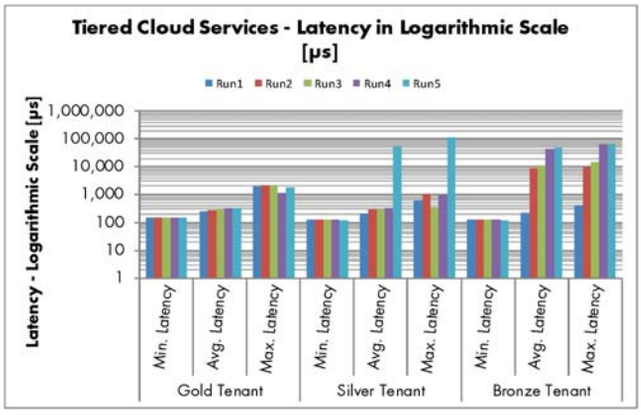


Figure 12: Latency per Service Tier Throughout the QoS Testing

In summary, the ASR9010 passed the three-tier prioritization tests.

Unified Fabric (UF) – UCS Manager

EXECUTIVE SUMMARY: The UCS Manager successfully brought up new cards and replaced failed cards automatically through the unified fabric.

Earlier we underlined the importance of reducing operational costs for cloud operators. Due to the number of components in the data center, its management has historically been quite complex. In recent years several advances such as the adoption of Fibre Channel over Ethernet (FCoE) and the virtualization of servers have helped simplify data center management. Cisco contributed to the simplification of data center operations with the introduction of its Unified Fabric solution – a system that aggregates server connections and reduces the amount of cabling and resources needed. Unified Fabric is one of the cornerstones in Cisco's Unified Computing System (UCS), which also includes server blades, fabrics and the focus of this test: UCS Manager.

Simplifying data center operations and reducing the duration of tasks are two ways to control and decrease the operational costs of running a data center. In our test we looked at both aspects. Specifically, we looked at Cisco's UCS service profiles – a saved set of attributes associated with a blade. The service profile or template is configured once with definition for VLAN pools, World Wide Names (WWNs), MAC addresses as well as pointers to the appropriate boot disc image sitting in the Storage Area Network (SAN). Once this profile is applied to a blade the operator can expect that the services will be brought up automatically. In the case that a blade failed, the profile could be automatically moved to a new physical blade speeding up the failure recovery. Through the UCS Manager the operator could see the status of the various blades and create the configuration.

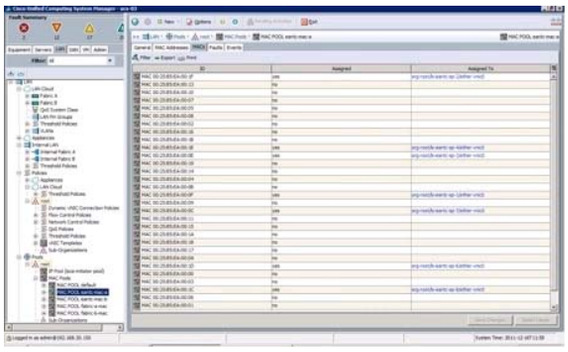


Figure 13: UCS Manager GUI

We started this test by configuring service profiles. The service profiles are typically stored in the Cisco UCS 6100 Fabric Interconnect, which is running the embedded UCS manager. We then set up two test scenarios. In the first test, we associated a service profile to a slot within the Cisco UCS 5000 Blade Server Chassis. Out of the eight blades installed in the chassis, six were being used by other applications, one was active and was running the profile we just created, and one blade was completely shut down. We then went down to the lab, and pulled our active blade out of the chassis. Before we went down though we initiated ping messages, both to the blade's IP, and to the IP of a VM running on that blade. Our expectation was that the UCS manager would load the same profile to the new blade without our involvement, since the profile was associated to the slot.

We came back to the lab and checked our ping messages. The replacement blade took 595 seconds to boot and respond. The UCS manager had applied the same profile to the new blade, and activated it. The ping to the VM, however, started getting responses after 101 seconds (we sent one ping per second). This was achieved thanks to a VM level failover recovery mechanism that moved the VM to another blade altogether.

.

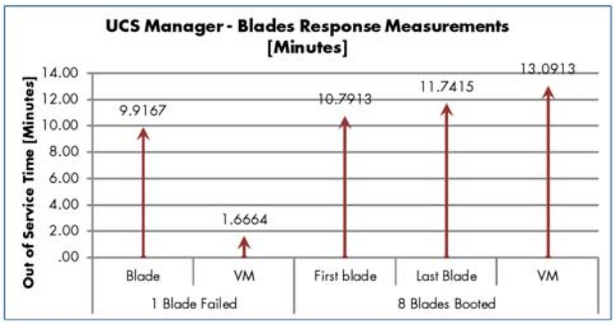


Figure 14: Time Taken to Respond in Minutes

In the second test we wanted to see if we could bring up a full chassis of UCS blades through the automatic service profile. We started by emulating the full scenario as much as possible. We put ourselves in the administrator's shoes. We went to the lab and made sure that there were no blades in our chassis, as if we were waiting for them to be shipped. We then created service profiles for each blade, requesting some randomization on the fly to ensure that they were new profiles. After we associated the service profiles to the empty blade slots, we went back down to the lab, pushed the blades in, and waited. Before pushing the blades in, we again initiated pings to the first and last of the eight blades. Since the blade's IPs were DHCP based, we had to configure the DHCP server to bind IP addresses to the service profile MAC addresses, which in turn served as another verification point that the service profiles were used.

Indeed, as we saw the blades come up through the management tool, the ping messages started receiving responses. The first of the chosen blades responded to pings after 647 seconds, and the last blade after 704 seconds. The VM we started on the eighth blade was responding to pings when just over thirteen minutes had passed since we began inserting blades. Had the UCS Manger service profiles not been available to us, we would have had to boot the cards; look through our system to see which MAC addresses, VLANs, etc., should be used; connect to each blade physically and configure these variables; ensure that they could reach the SAN and boot from it; and of course debug any issues that came up along the way. The UCS Manager reduced these steps to a simple "verify on the GUI that all blades were up and working," and worked quite smoothly from the get-go.

Virtual Machine Fabric Extender Performance

EXECUTIVE SUMMARY: The Cisco UCS's Virtual Machine Fabric Extender (VM-FEX) offers consistently increased network performance operations compared to virtual distributed switch installations.

In a standard Local Area Network, various hosts, laptops and PCs typically connect to a Layer 2

switch that aggregates the physical stations before handing them off to a router. Communication between two hosts on the same LAN can be done directly without reaching the router. Similarly, a virtual switching instance passes traffic either to a VM sitting in the same hardware, or pushes it out the physical port. Virtual Switches (such as the Cisco Nexus 1000v or VMware's vNetwork Distributed Switch) operations are done in software and Virtual Switches (such as the Cisco Nexus 1000v or VMware's vNetwork Distributed Switch) operations are done in software and therefore take resources away from the virtual machines hosted on the blade. Reducing the amount of resources available to the VMs.

Cisco's Nexus 1000v has a rich set of capabilities such as VLAN aggregation, forwarding policies and security. Cisco, however, found that not all VM installations require these features, and in such cases it makes sense to save the resources taken by the virtual switch and appropriate them to the customer needs.

Cisco claimed that their Virtual Machine Fabric Extender (VM-FEX) in VMDirect mode replaces the switch and shows a significant increase in CPU performance for network intensive applications. VM-FEX, installed on VMWare ESX 5.0, enables all VM traffic to be automatically sent out on the UCS's Virtual Interface Card (VIC). This meant more traffic on the physical blade network interface, but reduced CPU usage, which is typically the VM bottleneck. To verify that the VM-FEX really frees up CPU resources, we ran a series of tests comparing a VM-FEX-enabled UCS blade to a Nexus 1000v virtual switch setup. Both UCS blade installations were identical in all aspects apart from the use of the VM-FEX in one and Nexus 1000v in the other.

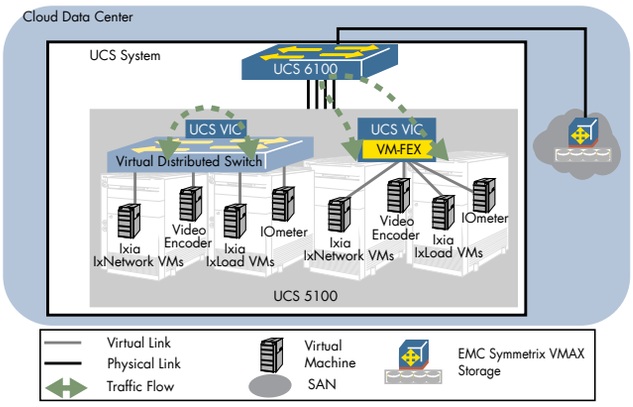


Figure 15: VM-FEX setup

We started comparing the performance between the two setups using Ixia's virtual tools. We installed four Ixia IxNetwork VMs on each of the two UCS blades and sent 3,333 Mbit/s of traffic from each of the first three VMs, toward the fourth for 120 seconds using 1,500-byte frames. In the VM-FEX case we recorded 2.186 percent frame loss, while in the distributed switch environment we recorded 16.19 percent frame loss.

We expected loss in both cases, given the almost 10Gbit/s load we were transmitting in the virtual space. The load was required in order to really keep the CPU busy. We deduced from this initial test result that in the VM-FEX environment less resources were used, which is why the frame loss we recorded was smaller than the loss recorded in the virtual distributed switch setup.

For the next test setup we installed one IxLoad VM on each of the two blades. We configured both IxLoad VMs as HTTP clients that requested traffic from a Web server Cisco configured. The IxLoad emulated clients were configured to try and use as much bandwidth as possible by requesting 10 different objects from 10 URLs repeatedly. The VM-FEX setup reached 9.87 Gbit/s while the distributed switch reached 7.78 Gbit/s. The CPU usage was also significantly higher in the virtual distributed switch setup when compared to the VM-FEX setup.

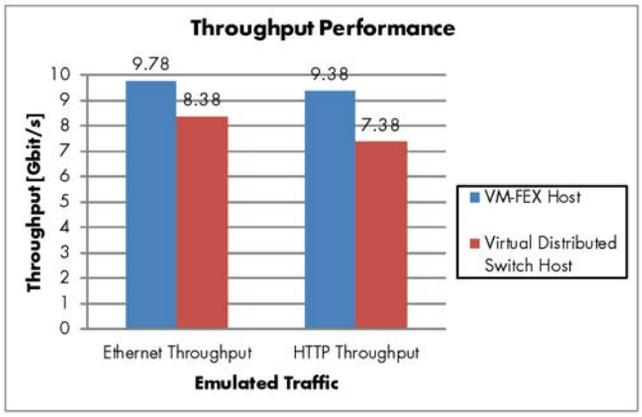


Figure 16: Performance Comparison Using Ixia Tools (higher values indicate better performnce)

Using the Ixia test tools we recorded the performance difference we expected. Cisco recommended that we perform a test that relies more heavily on the Storage Area Network (SAN). For this test, Cisco helped us to set up 10 VMs on each of the two setups, and install IOmeter on each virtual machine. IOmeter was configured to read blocks from an iSCSI based SAN as fast as it possibly could. We manually started each of the twenty IOmeter instances, and after 10 minutes we manually stopped each of them. At the end, we looked at three statistics – Input/Output Operations per Second, Data Rate, and Average Response Time – all three averaged across the 10 VMs in each setup. The VM-FEX performance was indeed higher for all three metrics. The data is shown in the graph below:

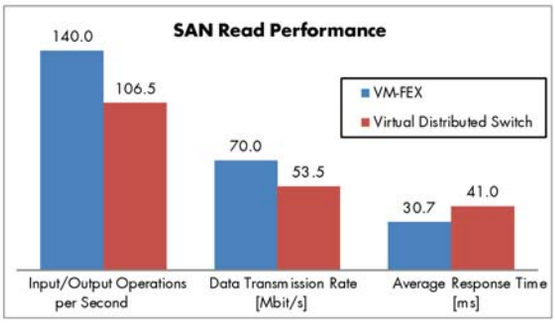


Figure 17: SAN Read Performance

We were still curious what the difference would be when someone is running a common task on a single VM. We wrote a script to use the open source program mplayer to encode a DVD image file that was stored in the SAN into mpeg (for private use of course). We wrote two versions of the script – one performed an additional round of encoding. The results of this test run actually showed that the act of fetching blocks off the network-attached DVD were not too resource intensive as the VM-FEX setup required only marginally less time to perform the encoding than the virtual distributed switch setup.

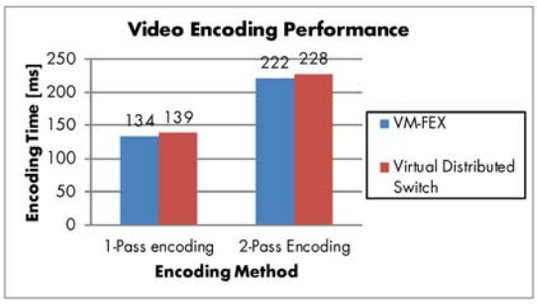


Figure 18: Video Encoding Performance (lower values indicate better performace)

Perhaps the most interesting metric was not the performance, but rather the CPU utilization. How much of the CPU was used for the operation, and how much was left over for other operations and other users? As shown below, the VM-FEX setup used far less of CPU resources in all cases. This was expected, since the CPU was skipping an entire layer of virtual switching, and this was, after all, exactly what Cisco wanted to demonstrate.

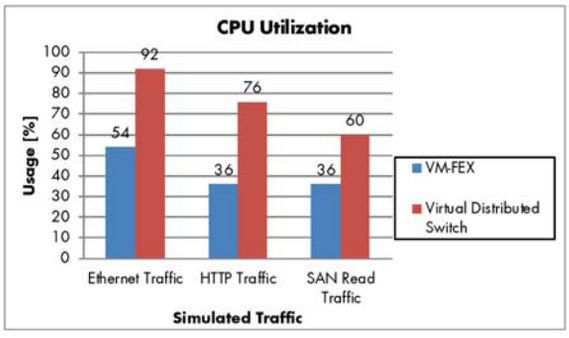


Figure 19: CPU Utilization (lower values indicate beter performance)

Virtual Security Gateway

EXECUTIVE SUMMARY: Cisco's Virtual Security Gateway (VSG) successfully applies policies between virtual machines, and continues to do so as VMs are migrated from hardware to hardware.

Earlier in the Tenant Isolation test (link Tenant Isolation test) we discussed the undeniable need for a comprehensive solution to the security concerns associated with cloud services. The isolation of tenants answers the question of security between private customers, but what about public or shared cloud services? Typically data centers use firewalls to block all traffic except for the specific types of traffic that are allowed, for the specific servers that need them. And if those servers are virtual? Well, then you need a virtual firewall of course.

Cisco's Virtual Security Gateway (VSG) integrates with the Nexus 1000v via a module called vPath, which is embedded into the distributed virtual switch. As Cisco explains it, most of the intelligence of the policies are off-loaded to the VSG component, which tells the vPath component how to treat traffic, thus reducing the complexity of the forwarding decision. Our interest was to a) verify that standard realistic policies would work in a realistic environment, and b) verify that the appropriate policies remain associated with the appropriate VMs even when those VMs are moved around.

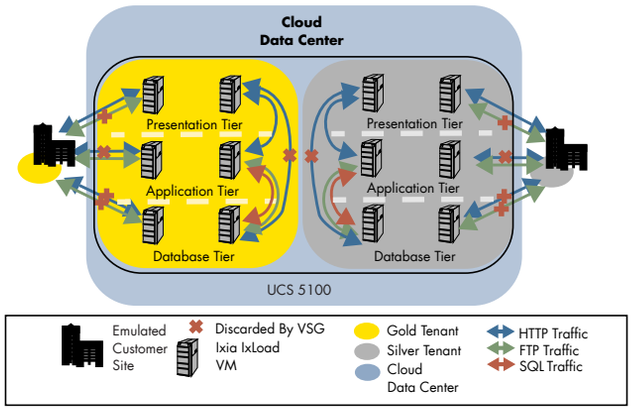


Figure 20: Virtual Security Gateway Test Setup

We set out to emulate a typical three-tier Web server scenario. For those less familiar, websites are delivered by splitting the main functions into three parts: the "presentation" or "Web" tier that users access directly; the "application" or "logic" tier, which runs the intelligence for the service that website is delivering; and "database" tier to store information. We started by creating three Ixia IxLoad VMs, each one emulating one of the three Web server tiers. Using Ixia hardware to emulate the outside users, we verified these policies with the appropriate traffic:

- Users could exchange HTTP traffic with the presentation tier VM, but other types of traffic would not work. We used FTP to verify that other traffic types were discarded.
- Users only had FTP access to the application tier. HTTP access was expected to fail.
- Users had no access to the database tier – verified with both HTTP and FTP traffic.

All traffic was passed or blocked as expected. We set up an additional three IxLoad VMs to verify that the policies among VMs would work in parallel to policies to the outside. Between the servers:

- The presentation tier VM could exchange HTTP traffic with the application tier, but not with the database tier

- The application tier could exchange both FTP and SQL traffic with the database tier. Ixia helped us to write a script to run the SQL traffic in parallel to the IxLoad generated traffic.

Again, all traffic was blocked and forwarded appropriately, and the behavior we had observed with outside customers remained the same, also in parallel. In fact, this entire setup was duplicated – one setup running within a Silver tenant, and one within a Gold tenant. With all these traffic flows running to all twelve emulated servers in parallel, we were pretty convinced. In fact Cisco also showed us that there were different ways to configure the policies down to the VM – one exemplified by the Silver tenant setup, and one by the Gold tenant setup. The Silver tenant used VM attributes (in fact the name of the VM) to match the policy to the VMs, while Gold tenants used IP-based mapping. Yet, one feature remained to be verified – what happens when a VM is migrated? That is, what is when a VM is moved by an administrator to different hardware?

Cisco promised that the behavior would remain the same as VMs were migrated. We tested this out by leaving the traffic running and performing a "vMotion" (migration) on different Ixia IxLoad VMs while they were still responding to clients (still sending traffic). Indeed, the same behavior was witnessed as described above. HTTP was blocked where expected, and forwarded where expected, as was FTP. We randomly chose to move the outside-user-facing presentation tier IxLoad VM to a new blade, and in addition we also moved the outside-user-facing application tier IxLoad VM. We were left feeling pretty secure.

Locator/ID Separation Protocol (LISP)

EXECUTIVE SUMMARY: Using Nexus 7010 and ASR 1002, Virtual Machines were successfully migrated seamlessly from one data center to another without the need for IP reconfiguration.

The folks at Cisco have supported the development of Locator/ID Separation Protocol (LISP) in the IETF for some time now. Sometimes referred to as a protocol, and sometimes an architecture, LISP is a mechanism to optimize network flows by managing address families. LISP was originally devised to abstract network areas in order to "divide and conquer" scale, but ended up having the consequence of enabling mobility. The latter ends up being a pretty useful tool for data centers – this is what we tested.

In a nutshell, using an additional level of mapping, LISP abstracts exceptions in IP routes to avoid long routing tables and reconfiguration of routers. Thus, if a virtual machine would move from one data center to another, it would not have to change its IP address and the routers in the new data center would not have to be configured for the VM's IP subnet. The LISP-aware nodes would dynamically learn about this new VM with its misfit IP address. Users who cached that VM's IP address would not have to relearn a new IP, their services would be transparent to the location change, at least from an IP addressing perspective. Without LISP, the administrator would have to change the VM's IP address, affecting customers much more.

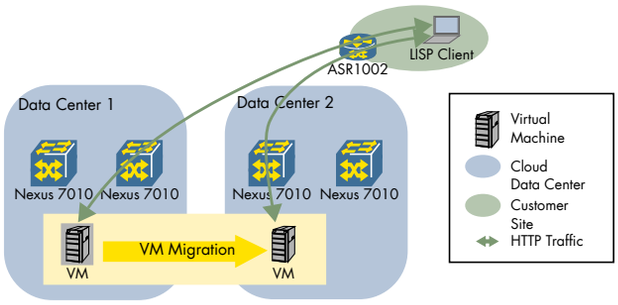


Figure 21: LISP Setup

We started by establishing a simple website, hosted on a virtual machine in our "Data Center 1." We connected to that website via a laptop host connected to an ASR 1002, which was positioned to serve as that host's Customer Edge (CE) – LISP was enabled here. LISP was also enabled on the Nexus 7010s within each of the data centers. These systems were both configured to serve the LISP mapping database. We ensured that caching did not affect our test.

As the next step, we migrated the VM. There are different tools on the market to administer such an operation, but this was not the focus of this test. We used vCloud Director, which Cisco had installed. Once the move operation was completed, we first pinged the server from the ASR 1002, which sent five echo requests per default. The first did not receive a response, as it triggered the LISP to poll and update its database. As a note, this process was perfected during the pre-staging through a software patch, thus the Nexus 7010s were running effectively engineering code. All further echo requests from that same ping operation, were responded to – just as expected. We refreshed our demo website, and it returned the contents to the client immediately.

In summary, the ASR 1002 and Nexus 7010 automatically detected the new location of the route, and the user could continue to access the website from the new data center – all without any reconfiguration.

Provisioning: Cisco Network Services Manager

EXECUTIVE SUMMARY: Cisco Network Services Manager automatically configured multiple tenants throughout the data center.

After we completed the BMC CLM tests, Cisco asked us to test another provisioning tool: Cisco Network Services Manager, formerly known as OverDrive Network Hypervisor. Cisco explained that this tool,

acquired in 2010 together with Linesider Technologies, is focused more on the network side of the data center, whereas BMC CLM also provisions virtual machines.

Cisco walked us through the Network Services Manager Graphical User Interface (GUI) as we created our first tenant, but at the same time they explained that the plan is for the admin to never see this GUI in the future. They plan for the tool to integrate with higher layer orchestration tools.

From the GUI, we chose a so-called "Metamodel" with which we could associate our yet-to-be-built tenant with policies and resources such as which IP/VLAN pools to use, and we chose which data center to create it in. The only manual interaction requiring specific information of the administrator was when the tool prompted us for the tenant's public IP address – understandably, some may not want to have this metric set dynamically. Once we had hit "Go," we compared configurations throughout the data center before and after the action. We were able to see the appropriate configuration changes on the ASR 1000s, Nexus 7000s, UCS 6100s and the Nexus 1000v.

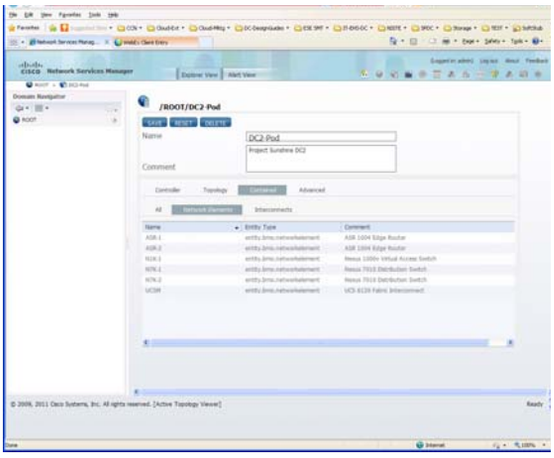


Figure 22: Cisco Network Services Manager

Following that, a Cisco team member showed us a script he had written in Ruby to simulate the higher level tool plugging in to Cisco Network Services Manager. We used the script to provision ten tenants. After less than 20 minutes the script was complete, and we again captured and observed the network configuration changes.

Enterprise Applications: Siebel CRM

EXECUTIVE SUMMARY: Cisco's Siebel-based call center application was successfully installed and run by multiple users within the UCS-based cloud infrastructure.

Cloud services are much more than throwing data to some off-site storage. The real value added by cloud services comes from applications. In the next weeks we'll be releasing the next sections of this report, one of which focuses on Cisco cloud applications. To give a preview of what's coming we picked two such applications to highlight today. Cloud operators focusing on enterprise markets will be maximizing their revenues as soon as they can convince their customers to focus on their core business and leave the corporate IT to the cloud. One essential application run used by every modern enterprise is Customer Relationship Management (CRM) – the software enterprises use to maintain sales, clients and customer contacts. One common CRM software is Oracle Corp. (Nasdaq: ORCL)'s Siebel CRM.

For the test, Cisco provided us with access to the Cisco corporate Siebel CRM. Cisco programed its own customized version of Siebel for its call center, what it calls Sales and Marketing Call Center (SMCC) Application. Like any CRM software should, SMCC helps manage customers, the appropriate contact people and their contact information, the current state of a sales lead and other customer and sales related information. Cisco's flavor is particularly designed for call centers. Call center employees can bring up a contact and then open a script associated for that contact leading them through the conversation with the customer.

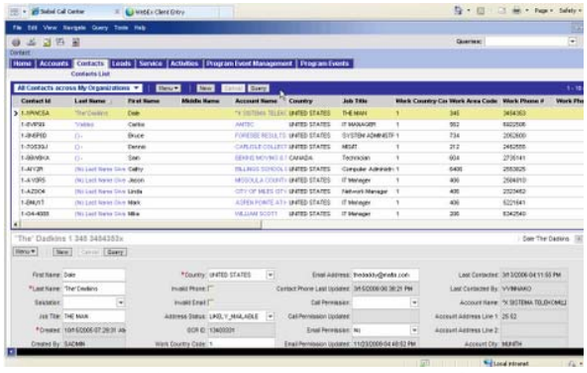


Figure 23: SMCC Screen Shot

Such CRM applications have been typically located locally within an enterprises IT infrastructure, but since the advent of the cloud, have moved away. Our goal was to see how Cisco's SMCC worked when set up in a cloud environment. We verified that the presentation and application tiers of the service were running on VMs in the test bed. For the database tier, rather than copying an immense amount of data, we took advantage of a Siebel database at a remote site – one of Cisco's main national corporate data centers in a completely different U.S. state (the tests ran on a database copy). Since we were investigating an application, there was not much to learn by pumping traffic into an infrastructure. With the help of a team proficient in HP's LoadRunner VuGen software programmers, we created a list of automated actions that a call center employee typically would execute.

These actions included:

- Searching for a customer
- Adding a customer
- Associating a customer to a sales program

- Displaying and stepping through the pages of a script (normally to be read out loud by call center employee)
- Entering a predetermined set of customer answers
- Passing the lead to a sales team

1200 emulated call center employees were configured to repeatedly execute the list of actions for a total of 250 repetitions, five at a time. The process of stepping through the list of actions took four hours, 16 minutes and 13 seconds. At the end of the run 200 of the transactions were evaluated as successful. The rest of the runs were mostly successful – in the last step, in which the operator was to pass the lead to the sales team, a database request took longer than 60 seconds to complete, a delay that we defined as too long, which is why the software considered these runs as fail. Since we defined 60 seconds as a fail condition we could not really hold this against Cisco.

At the same time as the action list was executed, we monitored the interface statistics between our data center test bed and the interface toward Cisco's corporate site to verify the setup we understood was indeed the one being used. We found no hat tricks here and were able to verify that the test bed was really used for this test. .

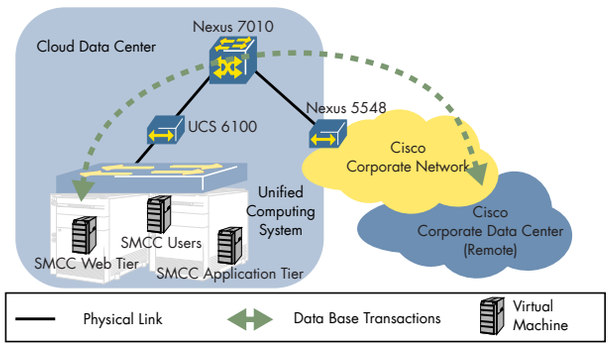


Figure 24: Logical Siebel Setup

In summary, aside from a few delays in certain actions, the Siebel installation ran smoothly on the UCS system

Cisco's Hosted Collaboration Solution

EXECUTIVE SUMMARY: Cisco's Hosted Collaboration Solution application successfully enabled phone calls and instant messages between customers, while running from the cloud.

In the previous section, Enterprise Applications: Siebel CRM, we reminded ourselves how effectively running enterprise applications is the key to winning and serving enterprise cloud solutions. To drive the point home we investigated another cloud-based application: Cisco's Hosted Collaboration Solution (HCS).

What does it actually mean? Hosted – again, the idea is to run the application on Cisco's UCS platform. Collaboration – the key word indicating that HCS is a solution for communications: managing phone numbers, instant messaging users, voice mail, with additional features planed. At first we were looking for a tangible single graphical user interface, but we quickly learned that HCS is in fact a suite of applications. In our discussions we used the parallel of Microsoft Office – not a program itself, but a suite of programs. So what could we do with it?

To explain what HCS is, Cisco started by walking us through the Cisco Unified Communications Domain Manager (CUCDM), which is a simplified version of the familiar Cisco Unified Communications Manager (CUCM). For the reader not initiated to the world of Cisco acronyms we could simplify the story a little. The demonstrated we received included using a piece of software to setup VoIP phones and then associate an instant messenger ID with a phone number. This demonstration also used a different setup to our data center test bed used for most of the other tests we report here. We went to the HCS team's lab to inspect the setup, and found a very similar setup there – UCS 5100, UCS 6100 and Nexus 7000. We also removed a cable during one of our test calls explained below, to prove that the new setup was really being used to make the calls.

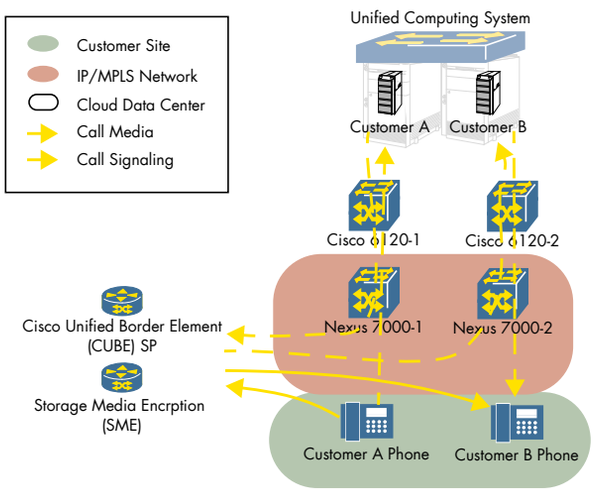


Figure 25: HCS Call Flow

We used the software to add fifty customers to the system, each with a number of users between 500 and 10,000, for a total of 91,048 users according to the system's user interface. To see HCS in action, we connected some phones to our setup, and stepped through registering a new caller which took approximately five minutes. Once the user/caller was entered we associated the user to one of the phones we setup. Before the association we just picked up the phone and verified that there was no dial tone. After the association, we repeated the exercise to verify that a dial tone was present. We then tried making and receiving calls to and from different customers. By default the user did not have voice mail services, so using the CUCDM GUI we enabled voice mail, and left ourselves messages. Finally, using the same CUCDM GUI, we could associate that user to a Jabber instant messaging ID

and test it out by logging in using that ID and having a quick chat with the team.

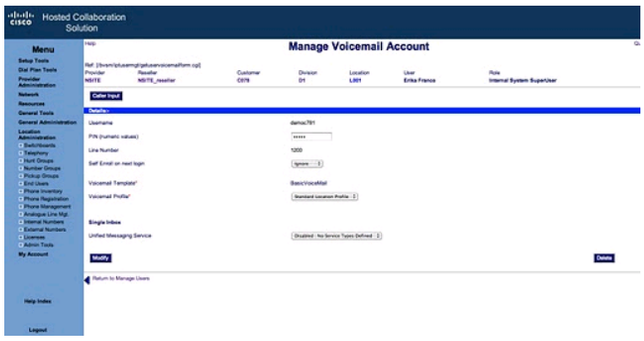


Figure 26: Adding Voicemail

We've attached a screen shot to give a feel for the GUI. Overall the demonstration worked smoothly without issues, and we learned a bit about how Cisco call centers work in the process.

Conclusion: Unified Data Center Test

There you have it – Cisco's data center. You'll notice that we focused on Cisco's newest bells and whistles and made the assumption that the basics – disks, virtual machines and data center switches – work. We focused on the essential stops along the end-to-end road: the comprehensive infrastructure that covers all corners in the data center; the unified management solution; and the security and agility of the solution. All elements were not only demonstrated, but passed our detailed inspection.

We were relieved to find that Cisco's Virtual Security Gateway (VSG), while differing quite a bit from standard files on implementation, still had the familiar configuration and user interface, and enforced its policies in a realistic virtual machine setup.

We found that VM-FEX is certainly not for every deployment, but for those willing to sacrifice network functionality for performance, it is right up their alley. By sending traffic both in the virtual and physical space of the data center, we effectively verified that no tenant's traffic will be seen by one another. (See Virtual Machine Fabric Extender Performance.)

Cisco Tiered Cloud Services architecture of Gold/Silver/Bronze successfully prioritized appropriately without starving anyone out, as long as the load was not overwhelming. FabricPath showed us that we could theoretically have had a lot more servers in that data center, forwarding 292.8Gbit/s through a fully meshed topology.

Cisco found a clever use for their Locator/ID Separation Protocol (LISP) implementation in the cloud, enabling mobility by bending the rules of IP subnetting through abstraction.

TABLE 2. Cisco Devices Tested

Cisco Hardware Tested
Cisco ACE30
Cisco ASA 5585-X60
Cisco ASR 1002
Cisco ASR 1006
Cisco ASR 5000
Cisco ASR 9010
Cisco Catalyst 6500
Cisco CDE 205
Cisco CDE 220
Cisco CDE 250-S6
Cisco CRS-1
Cisco CRS-3
Cisco MDS 9506
Cisco MDS 9513
Cisco Nexus 1000v
Cisco Nexus 2248 TP-E
Cisco Nexus 5548
Cisco Nexus 7009
Cisco Nexus 7010
Cisco Nexus 7018
Cisco UCS 6140

Furthermore, operators have been pretty loud about the need for streamlined management tools, so we believe they will be relieved to see Cisco prioritizing operations with tools like UCS Manager and its partnership with BMC. (See BMC Cloud Lifecycle Management Integration.)

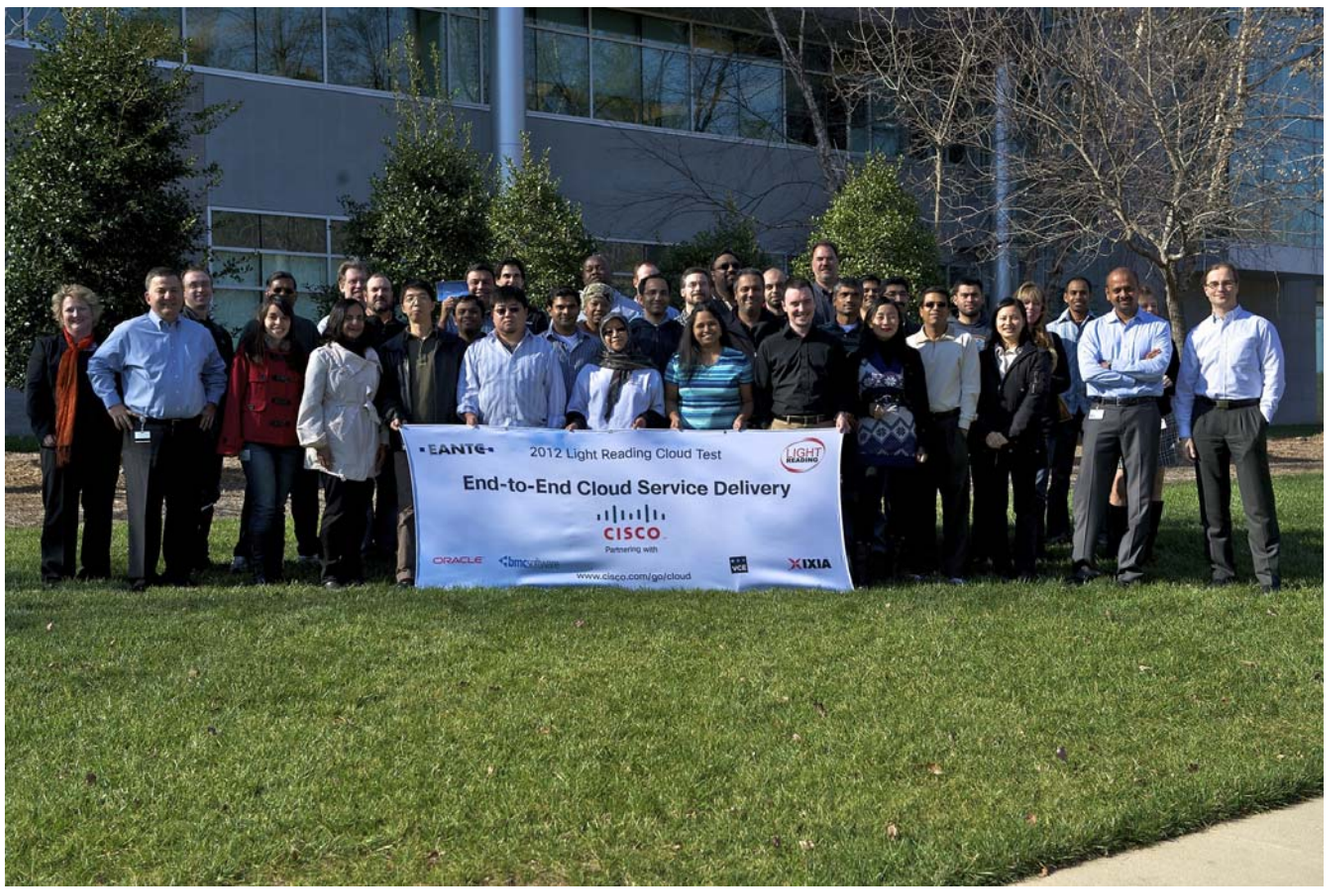
Data centers these days also open up a series of new variables, and it's not always clear how testing methodology should manage them and keep experimental control. Take the VM-FEX test, for example. It was important to ensure that both the hardware configuration and the virtual machine activity are identical on both blades, so we could compare their virtual network interface configuration.

When we tested the Cisco's Virtual Security Gateway (VSG) it was crucial that we run all tiers of the test in parallel – multiple tenants, multiple Web tiers and multiple traffic directions were all flowing at the same time. This puts to rest the common security concern that shared resources lower the ability to control and activate policies. The flows in the data center are also important to understand, and much different from LANs or WANs. In our FabricPath, tenant isolation and VSG test, we knew it was key to have so-called "north-south" traffic (between the data center and the customer) and "east-west" traffic (within the data center) in the appropriate proportions and flow models, which were mostly pairs and partial meshes.

We also had firsthand experience at the level of commitment that Cisco has to the data center and cloud. In past tests of massive scale, Cisco built the

test bed from scratch specifically to the project. In this test, we took advantage of Cisco's Virtualized Multi-Tenant Data Center (VMDC) lab, where Cisco engineers spend their days (and often nights) designing, building and testing the Cisco data center solutions. The same team that is responsible for verifying that different chapters of the data center story will work for the customers also supported our tests. Along with the lab came the equipment and knowledge, and the team was open to our persistent questions and calls for retests. That is not to say that it was not a very long three and a half weeks – it was.

As we move on to finishing up the next report on Cisco's network for the cloud, we hope those who were looking to become familiar with Cisco's cloud data center solution have done so. For those wondering about IPv6, stay tuned for the next report.



EANTC AG
European Advanced Networking Test Center

Salzufer 14
10587 Berlin, Germany
Tel: +49 30 3180595-0
info@eantc.de
http://www.eantc.com

Light Reading
A Division of United Business Media TechWeb

240 West 35th Street, 8th floor
New York, NY 10001, USA

http://www.lightreading.com