Network Services Automation for Cloud—Practical Overview
Content

Network Services Automation for Cloud—Practical Overview............................................. 1
About This Paper .................................................................................................................. 3
PaaS/IaaS and NaaS Recap............................................................................................... 4
Challenges of Network Services Automation........................................................................ 5
Cloud Network Services Appliance Types............................................................................. 6
Network Services Interconnect Options............................................................................... 10
The Virtual Data Center Service........................................................................................... 12
The Data Model for Building a VDC Using Its Atomic Objects............................................. 14
Logical Rules for Service Appliances Placement................................................................. 21
End-to-End Isolation and Security....................................................................................... 22
Full Redundancy in any VDC.............................................................................................. 23
The Relationship Between IaaS and PaaS........................................................................... 25
Lesson Learned and Key Takeaways.................................................................................... 29
About This Paper

Cloud services or IT-as-a-Service (ITaaS) is now becoming a reality; it promises significant cost savings and more streamlined management of mission critical information technology, data processing, networking, security, and storage needs. Automation of the network is one of the biggest (and often most challenging) customer requirements to help this tremendous market shift – and specifically, the automation of network services such as virtual private networks (VPNs), load balancers, and firewalls.

Still there is confusion and misunderstanding about the implementation, best practice design details, and the actual use cases of network services automation. In reality many cloud service providers still struggle to provide the right network services automation for their cloud environments.

The purpose of this paper is to provide a simplified review of the challenges around network services automation for the cloud and to clear out some of the common misunderstandings. The paper also tries to provide a clearer best practice design and a view of how network services automation should be implemented to create a virtual data center (VDC) that spans virtual and physical domains and that is replicable, cost effective, and highly dynamic.

Cisco has been driving innovations around cloud technology for the last few years; this paper also attempts to aggregate several learning from network services automation implementation projects.

The goal of this paper is to simplify the challenges and describe a practical model to be used as the basis for network services automation with our service provider and large enterprise customers.
PaaS/IaaS and NaaS Recap

Cloud service providers and large enterprises both started the journey into cloud services with a deployment model called IaaS (infrastructure-as-a-Service). For many of these organizations, this was an evolution of their existing data center hosting model. In order to save infrastructure costs, this hosting model was evolved to provide infrastructure in an automated and self-service fashion. This provides lower administration overhead and faster service, with self-provisioning of infrastructure resources—that is, mostly compute, storage, and networking resources for basic operations.

More recently, on top of IaaS, these organizations (and their developers) have begun looking for certain types of commonly used platforms and applications to be available and automatically installed on top of the already available automated infrastructure. They want those application and platform options to be available through the same self-service model. This evolution is known to some as enterprise Platform-as-a-Service (PaaS); Forrester Research refers to it as “IaaS Plus”. It’s somewhat different than the traditional PaaS definition for cloud services – but there are common attributes. Bottom line, it’s an evolution of IaaS that delivers a standard application stack (not just the infrastructure) through automation and self-service.

This paper does not discuss the SaaS (software-as-a-Service) model as in most cases the underlying infrastructure in such a model is not necessarily automated. Figure 1 shows the models for those three cloud service types:

Network-as-a-Service (NaaS) is another business model for delivering network services virtually over the Internet on a pay per use or monthly subscription basis. This model is required mostly by service providers and large enterprises. In this model the focus is specifically on networking features and communications provided automatically on demand. This model is now known as networking as a service. The use cases for NaaS range from self-servicing connectivity details,
like changing service-level agreement (SLA)/quality of service (QoS) parameters on the network, adding more bandwidth to user connections, making user connections redundant or more secure, redirecting user connections to a different location, and so on in order to automate large-scale networks using Border Gateway Protocol (BGP), Multiprotocol Label Switching (MPLS) VPNs, and other technologies. Software defined networking (SDN) is yet another possible implementation path for NaaS where an application’s networking requirements are met by a set of network provisioning tasks fulfilled on networking devices through their APIs in order to use a set of networking capabilities. Those SDN use cases again can be changing SLAs/QoS, redirecting traffic, providing more throughput, and so on.

The network services automation part in the IaaS/PaaS models has the following characteristics:

- Mostly uses basic and common networking features and capabilities
- Highly focused on data center hosting services use cases
- Requires highly flexible networking services, ability to add/remove/modify the networking infrastructure dynamically per application requirements
- Requires highly scalable, reusable protocol capabilities and networking solutions

Network services automation for data center cloud hosting is the focus of this paper. The paper will provide a suggested networking data model for allowing highly scalable, reusable, and dynamic data center networking services. This model is now known as Flexible Topologies for Dynamically Automated Virtual Data Centers (VDCs). The model was successfully implemented in service provider facilities using both Cisco and third-party automation tools. The model is now made more dynamic and highly secure by the Cisco Prime™ Network Services Controller.

**Challenges of Network Services Automation**

As Cisco has been implementing customer’s network services automation requirements, it has become apparent that there are three major challenges for network services automation:

- Ability to instantiate, create, and automatically configure common data center service appliances:
  
  - Switching services for Layer 2 network connectivity (now referred to as overlay networks)
  - Routing services for Layer 3 network connectivity (routing service appliances)
  - Firewall services for traffic protection and authentication services, using multiple layers of firewalls (known as three-tier application security models)
  - Load-balancing services for a more scalable application deployment
  - IP services: Network Address Translation (NAT), Dynamic Host Configuration Protocol (DHCP), Domain Name System (DNS), and so on
  - Application layer network services
In this paper we will refer to this challenge as “network services configurations,” or NSC.

- Ability to interconnect many different service appliances to form a virtual data center topology, potentially using all or a couple of the above data center services; for example:
  - Networks that are routed or switched and/or firewalled according to requirements
  - Networks connected to the public Internet or to a private network or both
  - Networks with/without load balancing and IP services
  - Network with/without application networking services

In this paper we will refer to this challenge as “network services interconnect,” or NSI. Cisco is known as one of the few companies that is able to automatically as well as dynamically interconnect all of the above data center service appliances through a common API, while doing so using best-in-class data center service appliances, both virtual and physical. This will be detailed in the latest release of Cisco Prime Network Services Controller by Cisco.

- Ability to use a unified data model for all data center resources and its required connections. This resource management model (like the Flexible Topologies for Dynamically Automated VDCs model describe in this paper) should allow create, remove, update, and delete (CRUD) operations for all potential virtual data center topology use cases for the automated cloud hosting facility. This resource management model is the most important component as it provides the ability to scale, to provide multitenancy, consumption monitoring, and assurance.

In this paper we will refer to this challenge as “Service Resources Management,” or SRM.

Cloud Network Services Appliance Types

The evolution of cloud computing and IaaS created several types of networking devices and appliances differentiated by their deployment model, scalability, feature sets, and integration capabilities (APIs). Cisco was the first to develop and innovate around most of those networking services appliance types and continues to heavily invest in producing all those appliance types and capabilities per customer use cases and requirements.
As Cisco has been implementing our customers’ network services automation requirements, it has become apparent that there are three major cloud network services appliance types. Following are the characteristics of those device types:

- Physical form-factor appliances:
  - Purpose-built hardware, focused on performance and scale (see Figure 2); for example, Cisco® ASA 5585-x firewall appliances, Cisco Nexus® switches/routers, Cisco IPS appliances, and others
  - Purpose-built operating system; for example, Cisco IOS® Software, NX-OS, and ASA OS
  - Support multiple virtual instances of their function; for example, Cisco virtual contexts inside ASA 5500 appliances, virtual routing and forwarding instances inside Cisco Nexus appliances
  - Deployed in-band or out-of-band as needed
  - Require dedicated fiber links, dedicated networks, use dedicated bandwidth

Figure 2: Physical Form-Factor Appliances with Virtual Instances
• **Virtual machine form-factor appliances:**
  
  o Run as virtual machines (VMs) on a generic (x86) hypervisor server (see Figure 3); for example, the Cisco ASA 1000V appliance, Cisco Services Router (CSR) 1000V, and others  
  o Consume its resources (memory, CPU) from the hypervisor server  
  o Consume its bandwidth from the hypervisor server virtual network interface cards (vNICs) they run on  
  o Deployed in-band, typically one service appliance per network or few networks; requires dedicated connectivity from users’ VM vNICs across all hosts (overlay connectivity needed)

![Figure 3: Virtual Machine Form-Factor Appliances on Generic x86 Servers](image)

• **Hypervisor-based appliances:**
  
  o Run on a generic (x86) hypervisor server (see Figure 4); for example, Cisco VSG/vPath  
  o Consume its resources (memory, CPU) from the hypervisor server  
  o Deployed out-of-band, one service appliance per hypervisor; do not require connectivity to user VM vNICs (no overlay connectivity needed)  
  o Typically implemented inside the virtual switch code in the hypervisor
Hypervisor-based appliances like the Cisco VSG are sometimes referred to as “compute firewalls” and an in-line firewall device is sometimes referred to as an “edge firewall” (see both described later in this paper).

Customers may choose to implement one or more network services appliance types. Cisco provides all those options; for higher scale and less overhead the out-of-band option is best practice.

Cisco recommends customers to do the proper cost-effectiveness calculations and evaluate the scalability as well as the capabilities of each services appliance type, taking into account resource consumption (CPU, memory, traffic throughput requirements) as well as the total number of host consumption and licenses in order to scale a certain appliance type. The total cost of ownership includes:

- Bandwidth consumption
- Device traffic overhead (how much traffic a device will need to handle)
- Traffic pattern predictability (that leads to stability)
- Total number of appliances per server/host, total number of hosts
• Compute resources, licensing
• Management cost and management overhead

Network Services Configurations (NSCs) are device specific. The actual configuration details of each device depend on the device type and its deployment model. The challenge and total cost of owning a certain set of devices and their different deployment models should drive the decision to go with a certain deployment model and a certain device type.

Cisco Prime Network Services Controller can provide a unified provisioning platform with a dynamic and scalable northbound API. Cisco Prime Network Services Controller is built with policy templates. Each device inherits a certain policy that handles the necessary configurations supported by that device. Using a template of policies provides an extremely scalable architecture. Please refer to http://www.cisco.com/en/US/docs/net_mgmt/virtual_network_mgmt_center/3.0/programming/PNSC_3_0_XML-API-Guide.pdf for more details on the Cisco Prime Network Services Controller architecture and northbound API.

Network Services Interconnect Options

The evolution of cloud computing and IaaS created several types of network services interconnects, most of them developed by Cisco and now referred to as overlay networks. Those overlay networks are a type of Layer 2 packet encapsulation that provides connectivity between endpoints and service appliances and among different service appliances. The overlay network also provides isolation across different tenants and different local area networks. The expected growth of such services in cloud environments may require the adoption of specific Layer 2 encapsulation methods to support both multitenancy and scale. The typical and most common solution to provide NSIs is virtual local area networks (VLANs), and those are based on the 802.1q standard (see Figure 5). This encapsulation can scale up to ~4000 NSIs on a single physically interconnected Layer 2 domain in the data center. Several domains (routed/isolated) can allow further scale of the 802.1q NSI. Several technologies (data center interconnects) developed by Cisco can provide additional scale for multiple Layer 2 physical locations in the data center. Those do not solve the inherited limit of 802.1q in the same Layer 2 domain.
Cisco also developed and innovated several Layer 2 tunneling mechanisms or overlay networks to scale the Layer 2 local domain, scaling Layer 2 NSIs much more, the most recent of them being VXLAN (see Figure 6). This encapsulation is used between different hypervisors running the Cisco Nexus 1000V virtual switch. The primary goal of VXLAN is to extend the VLAN address space by adding a 24-bit segment ID and increasing the number of available IDs to ~16 million.

Being a fairly new technology VXLAN has not been adopted as a standard. As with many other networking technologies, a formal standard based on VXLAN models and concepts is expected.

Cisco’s VXLAN will be further developed on new platforms and services, providing the needed NSI for data center cloud use cases. Those NSIs are to be automated by the overlay management systems and create the necessary connectivity between different cloud services.
The Virtual Data Center Service

With the review provided so far in this paper one can summarize the networking automation requirements for building a virtual data center (VDC) hosted in the cloud:

- Dynamically provision network service appliances and their configurations (NSCs)
- Establish the needed interconnects (NSIs) for end hosts and their network services
- Manage and monitor all those resources as they are being consumed by users

The three major challenges of network services automation, described in this paper, as well as the development of the underlying NSIs, had so far delayed service providers from offering many of their end-customer requirements. However, Green-Field service providers have been building their own platforms and products to support those requirements while developing their own solutions for NSCs and NSIs. Many cloud service providers transition to building a static and highly basic definition of the network services in their cloud catalog, avoiding the need to build a unified data model of those resources.

Cisco is now changing that paradigm by solving all three challenges with its cloud network automation offering:

- The Cisco Prime Network Services Controller, for managing top-of-the-line service appliances of all types and providing a policy-based configuration model with a northbound API that supports its entire feature set
- Automating NSI configurations across all platforms (using either Cisco Prime Network Services Controller directly or through a northbound orchestration layer)
- Using a unified data model for the networking objects

In contrast to the compute resources orderable in the cloud, which offer great flexibility and hit many customer variances (many different machine resource options, many different operating systems, and so on), cloud network services capabilities are so far rigid in their nature, preventing the offering of more complex applications and virtual data center requirements like multiple public and private links and three-tier security models.

Virtual data centers are known as the collection of virtual machines, their compute resources, their applications, but also their networking resources and networking services discussed so far in this paper (NSCs, NSIs, and the unified data model that ties them together).

Service providers will be differentiated by the service catalog offered to their IaaS customers. Their success will be measured by the number of end-customer deployments on their infrastructure, and therefore, in order to attract as many customers as possible, service providers will need to support all the end customers’ different application needs and underlying virtual data center types.

Experience shows that end customers have different ways to logically build their virtual data centers, depending on their application needs. The physical topology of the hosting facility is
however built very much the same. The physical hardware chosen for such a facility might have different resource capacity and different feature sets, but the interconnects of those physical devices in a point of delivery (PoD) are made according to best practices and are well known; PoD is the collection of certain sets of physical network devices, physical compute nodes, storage arrays, and application components that work together to deliver IaaS virtual resources.

Per the total virtual resources scale needs, Cisco offers different types of PoDs (see Figure 7 for a typical example). Those hardware resources can be coupled into well-defined consumable types, making the design easier, more reliable, predictable, and scalable.

Experience shows that managing and assuring both virtual and physical VDC components using a single unified resource management platform is crucial for successful cloud services.

Just like the virtual platforms, the physical infrastructure, including network, network-based services, compute, and storage will be shared among tenants—creating a shared multitenant environment. Security is important in the multitenant environment as sharing physical resources increases the risk of one or more tenants negatively affecting other tenants. With well-designed and secure architecture, these risks can be mitigated and tenants can enjoy the same benefits of dedicated physical infrastructure.
The Data Model for Building a VDC Using Its Atomic Objects

To facilitate the creation of a truly dynamic VDC with its CRUD operations, the network services automation for that VDC can follow some basic rules:

- VDC is created by connecting different data center components or atomic objects that are controlled by the NSCs.
- Each atomic object has a consistent configuration template that is replicable for that object (policy-based configuration templates).
• Those atomic objects can be changed or deleted/added to a VDC without causing downtime to the VDC services still in place.
• Resource management is deployed to track each atomic object usage and allocations per device (device-based tracking).
• The collection of the VDC atomic objects ordered by a certain tenant belongs only to this tenant; they are managed and tracked for cost models (tenant-based tracking).

The rules above should be abstracted from the underlying device type and specific hardware.

The following atomic objects are the basic set of objects needed to construct the network services in a VDC. Those are used by several cloud facilities and by several automation products; this model allows the flexibility for building a dynamic yet stable VDC:

**Network:**

A network is an IaaS atomic object to define a communication path that servers, VMs, and services appliances can be connected to, in order to communicate directly with each other at Layer 2. A network is either attached to an endpoint or used as an NSI to connect different network service appliances. In an IaaS environment it is common to use a network or several networks dedicated to a specific tenant. Networks can be added per zone (see below).

**User:**

The entity that connects into the virtual data center. Users can connect to the VDC through the public Internet or through private interconnection (like MPLS/VPN or a dedicated IPsec tunnel).

**Zone:**

A zone is an IaaS atomic object defined by a group of networks that share the same security level. It is a VDC area where many networks are routed to each other with no restriction (by default). The most important aspect of a zone is the fact that it is separated/isolated from others and has a certain type attached to it. Zones are the points of interaction between IaaS and PaaS (see below). Users and zones are separated by a security mechanism like a firewall.

The zone is then mapped to a single routing domain. In IaaS environments it will probably be a virtual router (VR) that consists of one or more networks that are routed to one another. Other Layer 3 mechanisms to interconnect different Layer 3 networks can apply (virtual distributed routing).

Servers or VMs are always placed inside a network and inside a certain zone and use the zone’s VR as their default gateway.

Zones are divided into types. The zone type defines the connectivity characteristics (where it is connected to) and the security level provided for the zone’s networks (which services apply to that zone). It is common to define a dynamically constructed VDC with four zone types:
- Public zone: Connected directly to the public Internet, no restrictions by default, communication can be blocked by compute firewall rules
- Public protected zone: Connected to the public Internet through a firewall, communication restricted by default, can be allowed by edge or compute firewall rules
- Private zone: Connected directly to a private network accessed through tenant premises, no restrictions by default, communication can be blocked by compute firewall rules
- Private protected zone: Connected to the private network through a firewall, communication restricted by default, can be allowed by edge or compute firewall rules

Intrazone traffic is allowed by default but can be protected by compute firewall like VSG.

Interzone traffic is blocked by default but can be allowed through a firewall device.

Users are divided into public users and private users and are made a point of reference for firewall rules.

Zone isolation rules:

Public zone: Can only connect to public users and to public protected zone

Private zone: Can only connect to private users and to private protected zone

Public protected: Can connect to public users, public zone, and private protected zone

Private protected: Can connect to private users, private zone, and public protected zone

Public zone and private zone are then kept isolated. Load-balancing and IP services are provided per zone.

Those zone definitions can define multitier and multihierarchy networking automation for building up to 45 different VDC topologies for each tenant with the different services variances per zone and across zones.

Shared zone: Mostly in service provider environment a fifth zone type might be introduced; this zone type defines an area where multiple tenants share the same routed networks.

**Virtual Router:**

A VR is an IaaS atomic object that defines an isolated per tenant private router. The VR should provide routing services for different networks in the same zone. A VR can be created using the VM form factor; physical appliance form factor or hypervisor kernel based (hypervisor distributed routing using overlay networks) or network based (like MPLS/VPN). In all cases it needs to be assigned to a certain zone for a certain tenant.
**Virtual Firewall:**

A virtual firewall (VFW) is an IaaS atomic object that defines an isolated per tenant firewall instance that facilitates security policies for a certain zone type. Cisco uses compute or edge firewall types (see appliance types). An edge virtual firewall instance can connect to multiple zones and isolate them by default, allowing only specific defined application flows across different zones. Firewall rules defined on the VFW will specify how traffic is handled based on originating application attributes (compute-based attributes like name of the host, name of the application, and so on), traffic type, or source and destination flow details (IP TCP/User Datagram Protocol [UDP]). Firewall rules apply dynamically from zone to zone or/and from users to zones. Virtual firewalls use a link to each zone it is protecting or to external connections. Connectivity details are embedded into the zone type.

**Virtual Load Balancer:**

A virtual load balancer (VLB) is an isolated load balancer instance that can contain multiple load-balanced pools, multiple VIPs (Virtual IPs), multiple server pools. A VLB is associated to a zone and can perform server load balancing functions for any of the networks in that zone, using VIP addresses from any IP pool assigned to that zone. The VLB is using the zone’s VR as a default gateway (this is a crucial design best practice to allow the adding and removing of a VLB without affecting the other VDC services running at the time of that change).

**Load Balanced Pool:**

A load balanced pool is a group of servers and/or a group of VMs that are part of the same application or the same service in load-balanced server resources.

When a load balancer instance is used, clients connect to a single destination IP (the VIP) and then get redirected to the most available real server resource on the server pool.

**Route:**

A route is an IaaS atomic object that defines a traffic path for traffic packets that are sent through the VDC on a specific IaaS device, commonly the VFW or the VR. It is used to define the next-hop destination for particular traffic flow. Each zone can have multiple routes, and multiple routes can point to different links on zones. The devices can use dynamic routing protocols to allow automatic route provisioning by the networking devices themselves; however, running multiple instances of dynamic routing protocols on many Layer 3 devices can be less scalable than defining static routes. Automatic provisioning of routes (static route automation) might provide better scale as well as still maintaining control on the VDCs in the IaaS facilities. This can be referred to as dynamically automated static routes (DASR).
**IP Address Space:**

This IaaS atomic object is associated with servers or VMs, networks, and links. IP address space is assigned for link endpoints and for VMs and their VR gateway in the specific network.

Each zone defines an isolated address space, used for the zone’s networks. Private zones can have overlapping IP address spaces across different tenants. In some cases, such as the private zone connection to an enterprise using a private VPN link, the end user will want to define the address space to use per network per zone. Public zones will have a pool of public IP addresses that may be allocated based on the IaaS service provider’s policies, or can be assigned by the customer for their public zone networks. IP address space is allocated for links and for networks. Each IP address space is tagged with the link name and the network name per zone.

**Remote Access Resources:**

This IaaS atomic object is associated with servers or VMs that needs to be accessed through VPN devices (SSL VPN and other forms of overlay VPNs). Once a VDC has been constructed for a specific tenant, remote access should be provided to the tenant’s specific resources (VMs and network components). Normally a VPN device will be used to accomplish this access in a secure manner. The VPN device requires information about the list of VMs and other resources (VFW and vSLB) built for the user on his or her VDC. Per tenant resources are handed to the VPN device to configure an isolated path from the public Internet to the tenant’s VDC. In an SSL-VPN device it is translated to a list of bookmarks or HTML links to point into the specific tenant VMs (using Remote Desktop Protocol [RDP], VNC, HTTP, SSH, or other protocols). Another example of a remote access resource is an isolated drive (NFS or CIFS for example) that can be accessed through the VPN in order to upload and/or download files (like ISO, OVF files) to be used by the cloud provider or the tenant to install on their VMs as needed.

Figure 8 represents the set of objects that construct the dynamically automated virtual data center and the relationship between them. The actual implementation details of such a model as well as the language used to describe the relationship between those objects are out of scope for this paper.
Many tenants can utilize one or more of such a dynamic data center. As explained above, the resource management will then need to assure and monitor the physical environment for those resources consumed from the infrastructure, both per device and per tenant (for all atomic objects).

The PoD then becomes a resource pool (physical and virtual) from which to consume those VDCs (Figure 9).
Deploy this VDC In Pool 1

Resource-pool 1

Resource-pool 2

Resource-pool 3

Figure 9: VDCs and PoD Resource Pools
Logical Rules for Service Appliances Placement

As best practice for a predictable and stable VDC, Cisco recommends a smart placement for the network service appliances in the certain PoD. It makes more sense to run them on dedicated hardware (a services cluster). Just as the physical appliances run virtual instances and are placed in a common predefined and known place, this is the only way to support QoS and guarantee SLAs for those services. Several vendors and customers now follow that guidance.

On top of the smart placement, anti-affinity rules make sure High Availability (HA) is actually implemented and virtual appliance HA pairs don’t end up running on the same host. This capability, which has long been built-in with physical appliances, now needs to be verified in the virtual data center. Making sure that a pair of network service appliances doesn’t end up running on the same host guarantees device failover but more granular placement rules need to be implemented to make sure certain network service appliances (the non-hypervisor-based types) are placed in a more robust cluster that can utilize traffic and device resources to guarantee QoS. The implementation details for virtual services cluster and QoS is out of scope for this paper.

A stable cloud infrastructure design should include network service appliances smart placement (Figure 10).

HA and smart placement on virtual appliances

![Diagram of network service appliances placement](image)

Figure 10: Smart Placement of Virtual Appliances
End-to-End Isolation and Security

A virtual data center for a certain tenant should be private, isolated from other tenants. This is made possible by using dedicated overlay interconnects (NSIs) and dedicated network services devices. Many times the tenants’ IP address schemes will overlap across different tenants (private IP schemes), while the shared network offered by the provider needs to be unique. In enterprise facilities this depends on the business unit’s deployment requirements; in general overlapping IP is possible even in enterprise cloud facilities.

The challenge of keeping tenant isolation remains an issue on shared networks (shared storage, shared management access, and so on). For example, the cloud provider will still need a way to access all customer VMs and their services through an out-of-band network, and the tenants themselves will need to use this network to access their virtual data center resources. This network might be shared across different customers in order to provide unique IP schemes for the cloud provider network itself. Privacy of the tenant resources needs still to be provided through this provider network, while allowing the provider to access all tenant resources (for maintenance, automation, shared services, and so on).

Overlay networks are great for total isolation but may have challenges with multitenant shared networks for use cases like shared storage areas and shared networks for management purposes where IP uniqueness is required by the cloud provider itself.

To keep tenant resources secure even on shared IP domains, a set of network isolation technics and authentication, authorization, and accounting (AAA) technologies needs to be deployed. Typically shared networks are needed also for allowing remote access services for tenant users, in order to allow them actually to use their VMs hosted on the cloud facility. Apart from allowing this access on the front-end self-service portal, it needs to be secure on the infrastructure, physical and virtual side separated and fully authenticated across all devices.

Cisco has provided those solutions for many cloud providers, using several technologies embedded into Cisco Nexus 1000V virtual switches, SSL-VPN security appliances, as well as on the Cisco UCS® servers and the Cisco Nexus physical networking appliances.

The out-of-band management network should be deployed on each hosted VM on a dedicated secondary vNIC, while the port profile attached to that vNIC should support:

- Dynamic Address Resolution Protocol (ARP) inspection, DHCP snooping, and IP Source Guard
- Gateway services for that profile that would deploy NAT to avoid the need for static routes on the VMs’ OS level
- DNS guard with technical hostnames used for cloud provider communication with VMs
Protecting OOB Management Domain

In the logical creation of a VDC, the implementation of high availability is crucial. Every component of the infrastructure (network and services) has a redundant mate on the underlying physical devices; this should be accommodated by automation tools—building HA configuration into the logical VDC for both redundancy and better utilization of links in the provider infrastructure. From our experience active-active deployment of the HA configuration is highly appreciated as it allows utilizing all available bandwidth on the infrastructure.

HA should be an attribute of a VDC at the zone level.

In order to allow service providers to utilize all available links and all available bandwidth on a particular PoD, the automation will be building, by default, an active-active logical construction of a VDC. We recommend doing that using Tenant-ID:VDC-ID for load-balancing between different redundant devices.
The creation of an active-active logical design will be done by using the device’s HA mechanism, such as Hot Standby Router Protocol (HSRP), Virtual Router Redundancy Protocol (VRRP), or any other First Hop Redundancy Protocol (FHRP) technology used by networking and services devices like switches, routers, firewalls, and load balancers. Each device might use different ways to configure the HA feature. In all cases there are methods to define who will be the primary for a specific network and what is the IP address to use as the next hop.

We suggest that Tenant-ID:VDC-ID be divided into odd and even IDs so different VDCs will use different paths accordingly. The odd VDC will be built using primary devices on one side of the infrastructure; the even VDC will be built using primary devices on the other side of the infrastructure. Detailed templates for this design will be introduced in such a way that the ID to be used for path selection could be changed (into active-passive setup or into a different mechanism for path selection) by allowing service provider administrators to change the VDC IDs and the priority of run-time variables that are created for it.

Figure 12 illustrates a redundant VDC construction with single zone that implements an active-active design for the PoD level, using Cisco VDC components:
Figure 12: Active-Active VDC Construction at the PoD Level

The Relationship Between IaaS and “Enterprise PaaS”

As referenced at the introduction to this paper, organizations are beginning to evolve beyond traditional IaaS to what may be referred to as “Enterprise PaaS” or “IaaS plus” – to provision entire application stacks (including the middleware, application server, database server, etc.) in addition to the underlying infrastructure.
To build a dynamically configurable virtual data center in this model, a provider can decide on a certain “blueprint.” This might be provided through a UI canvas or through a form-based blueprint buildup wizard. The provider can then define a certain “superset” virtual data center based on the zone model described in this paper:

- The “superset container” includes the maximum available objects and their interconnects.
- All objects that construct the VDC are managed through its lifecycle.
- All objects that construct the VDC can be included or excluded.
- VDCs are created on a certain PoD; the PoD becomes a single resource pool for both virtual and physical services.

The virtual data center conceptual model described in this paper has the following characteristics:

- An orderable service that is a collection of many network services objects, virtual and physical.
- It provides network connectivity and network services: Routing, firewall, load-balancing, app acceleration, and so on.
- It’s a collection of the networking components, networking services, and VM resources, as well as the application resources.
- The placement of all those objects can be distributed but needs to be predictable and scalable without losing performance.
- VDC is a baseline for resource management, modeling, and assurance.
- VDC is abstracted from the actual application definition and supports multilayered and multitiered applications.
- Add, delete, and modify functions on a VDC should not affect the availability of the applications installed on top of it.
- Applications get attached to a VDC, forming a PaaS layer on top of the IaaS components.

From our experience, the end users for cloud services can vary significantly – they may be non-technical developers or even business-oriented staff or they may be more technical IT infrastructure architects, infrastructure engineers and application engineers, as well as security engineers. The more business-oriented, application-oriented end users and developers would want to be able to simplify the network details and use their service for their application, with an application view only. They don’t want to – or do they need to – understand the technical details of the network services and networking infrastructure. In reality this enterprise PaaS model cannot be provided without the networking services automation detailed in this paper; however, it can be easily abstracted. For this abstraction the PaaS view can use only the underlying zones to refer back to the IaaS detail. Figure 13 shows an example.
Furthermore, a security layer can be created on that same canvas, with the same zone model, to represent the details of the security-related objects in the VDC (Figure 14).
Figure 14: Security Details of the VDC
Lessons Learned and Key Takeaways

Network services automation challenges span the network services instantiation, configurations, interconnections, and the resource management of both physical and virtual resources. To overcome those challenges a cloud services provider should consider:

- A set of predictable and replicable PoDs
- A well-defined, flexible object-based virtual data center model
- Deciding on the NSI preferred technology
- Deciding on the NSC to be replicable and stable
- Deploying VDCs on PoDs as resource pools
- Maintaining, managing, monitoring, and assuring virtual data center resources by tracking consumption per device as well as per tenant

Cisco Prime Network Services Controller and northbound orchestration and automation platforms will provide the needed stability and agility. In upcoming releases customers will be able to utilize Cisco’s years of experience in cloud facilities through the use of those products.