Ericsson Review
284 23-3184 | Uen

Enabling the network-embedded cloud

Growth in the number of smartphone and tablet applications deployed in public data-centers, and the rising use of cloud services by enterprises can lead to stretched resources, suboptimized networks and, ultimately, an inferior user-experience. Maybe it's time to reshape the cloud.

DAVID ALLAN, JAMES KEMPF AND TORBJÖRN CAGENIUS

Making better, more dynamic use of cloud resources requires a whole new innovative architecture: the network-embedded cloud. This concept is based on a variety of computational and storage resources being embedded in the network, interconnected via provisioned WAN links, and distributed closer to the network edges to provide the right QoE and more flexible connectivity. Naturally, a change in architecture places new requirements on the way clouds integrate with networks. To implement the network-embedded cloud, some key enablers are required, one of which is elastic networking.

Elastic networking is a technique for dynamically managing network connectivity between data centers, in a way that is complementary to their computational and storage resources. This technique can reduce provisioning connectivity time from days to minutes. However, for it to work, a direct relationship must exist between the data-center architecture and network connectivity to maintain tenant isolation, ensure provisioned bandwidth and uphold prioritization within the distributed cloud.

Market situation

Cloud computing combines the collective benefits of rapid fulfillment and multiple business models with the elasticity of computational and storage resources. It is the next phase in the automation evolution that began with batch processing, continued with early time-sharing and virtualization, and has resulted in the massive expansion of warehouse-scale computing.

Until now, the networking component of cloud infrastructure has been confined primarily to intra-data-center communication. However, there are a number of factors that are driving the development of inter-data-center networking, including:

- increased deployment of applications over private and public clouds, which is known as the hybrid-cloud model;
- storage-capacity issues at existing facilities, requiring seamless interconnect between multiple data centers, which is known as cloud bursting;
- geo-redundancy, to minimize the impact of a major disruption to normal operations; and
- the ability to personalize the location of computational and storage resources in the network to conserve overall bandwidth, or minimize latency between critical components.

The requirements imposed by the seamless interconnect and the network-embedded cloud need to be analyzed to understand: the differences in the resource environments of intra- and inter-data-center communication; and the characteristics of data-center traffic patterns throughout the life cycle of a given workload.

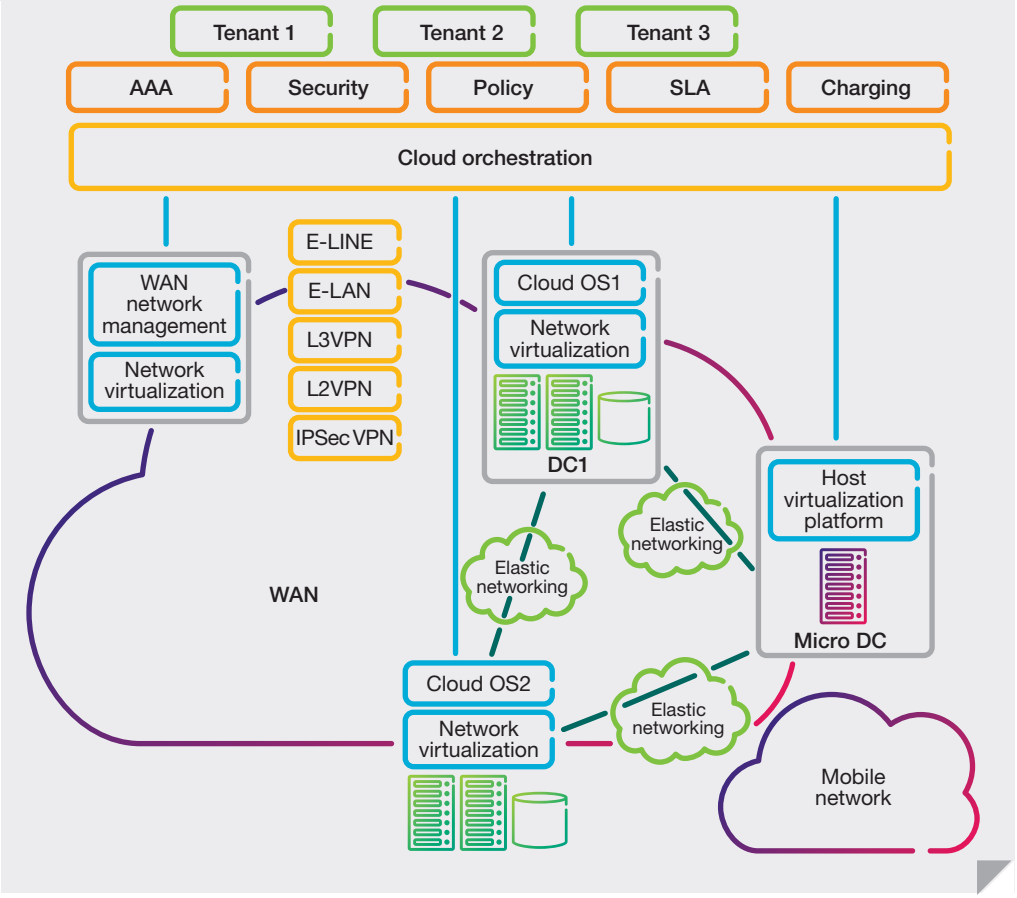
The trend in architecture design is shifting towards the provision of non-blocking connectivity within the fabric of the data center. Statistically speaking, connectivity is non-blocking – assuming perfect load balancing across the bisection bandwidth. However, the scaling capability of such designs or clusters is limited; once the maximum has been reached, inter-cluster connectivity becomes blocking. Current operational practice to minimize inter-cluster bandwidth requirements is to place workloads that need to interact with each other within the same cluster.

Similar operational practices could be applied to the sets of clusters, irrespective of physical location, if requirements for geo-redundancy and data-mirroring could be ignored – allowing the network to continue to function without further intervention. But even

BOX A Terms and abbreviations

AAA	authentication, authorization and accounting
API	application programming interface
CDC	central data center
CDN	content distribution network
CIR	committed information rate
DC	data center
DDC	distributed data center
EIR	excess information rate
E-LAN	Ethernet LAN service
E-LINE	Ethernet line service
GW	gateway
IPsec	IP security
IPT-NMS	IP Transport Network Management System
MEF E-LINE	Metro Ethernet Forum Ethernet Line Service
MPLS	Multiprotocol Label Switching
NaaS	Networking as a Service
NMS	network management system
OS	operating system
OTN	optical transport network
RNC	radio network controller
SLA	Service Level Agreement
VLAN	virtual local area network
VM	virtual machine
VPLS	Virtual Private LAN Service
WAN	wide area network
WDM	wavelength division multiplexing

FIGURE 1 **Vision of a network-embedded cloud**



within a non-blocking cluster, additional computational support is needed to manage collisions of exceptionally large data flows. Adopting an intra-cluster or intra-data-center approach requires enhanced resource management. Grouping computing resources geographically to manage overall network bandwidth or customer latency will place even greater demands on the integration of cloud data-centers and the WAN.

Ericsson vision

Management of computational and storage resources has, until recently, been the primary focus of building and operating clouds. Unfortunately, this focus tends to result in suboptimal designs as factors such as network latency, the cost of high bandwidth over long distances and the lack of service guarantees are ignored, leading to poor application and network performance or high costs for certain applications.

Ericsson's key insight into the network-embedded cloud has been finding the proper balance between computational, network and storage resources to create optimal cloud-infrastructure designs. Ericsson's vision covers the full range of deployment scenarios, from mega data-centers, smaller regional micro data-centers, clusters of individual servers, and embedded-service blades on routers – sometimes referred to as pico data-centers.

Network virtualization provides tenants with secure, isolated, elastic network-slices with associated SLA guarantees – in essence, Networking as a Service (NaaS). Cloud orchestration and elastic networking act as the glue that binds computational, storage and networking resources together into a single entity that can be dispatched – as shown in **Figure 1**. The resulting service provided to tenants is a seamless, secure and isolated elastic slice of the networking, computational and storage resources – a slice that has outward connectivity to the internet.

Elastic networking provides tenants with an abstraction called a flash network-slice – a logically isolated virtual network that may span several data centers and network domains, with service-level guarantees defined in timescales of less than a second and up to a few minutes. This is a considerable improvement on, say, the time it takes to bring up a VPN connection in the WAN today – a process that can take from a couple of hours up to several days to complete. Tenant traffic within the flash network-slice is logically isolated from other traffic and can be further isolated with the addition of a firewall or through encryption. Each slice will have associated bandwidth guarantees, which can be met on a flexible basis so the tenant can save on unused bandwidth for times of greater need. The flash network-slice brings to networking the same model of on-demand, elastic resource-allocation that data centers and virtualization bring to computation and storage.

Cloud network-orchestration aggregates computational, storage and network resources, providing the tenant with a single logical view for the deployment and provisioning of applications. Tenant applications are constructed by:

- selecting various components – from a catalog;
- specifying the service-level guarantees for network connectivity between applications; and
- adding location constraints on the deployment of each chosen component.

Storage resources are allocated to the application to provide access to critical data, and the properties of the connection from the application to clients are specified. The developer describes the application at a high level, and it is the job of network orchestration to refine this description into a physically deployable system – a process that involves filling in the details of where to deploy virtual machines, how to provision them, the exact network links and their bandwidth, and what type of

load balancing, if any, is necessary for client access. This orchestration model involves a multi-tiered approach, which enables developers to create applications with a wider scope than that allowed by the currently popular three-tiered approach.

Use cases

When considering cloud operations, and in particular inter-cloud transactions that could benefit from network-resource management, a number of use cases emerge. All are various forms of the larger problem of mapping workload to computing resources while taking network connectivity into account. The architecture of the network-embedded cloud distributes computational and storage resources geographically and integrates these resources operationally throughout the operator network. As shown in **Figure 2**, resources vary in size and are placed at different locations in the network, which in turn imposes different requirements for elasticity on each resource.

The typical central data center (CDCs) has a high capacity to process requests, host many applications and support a large number of tenants. As this type of data center tends to be served by dedicated high-capacity links, the requirements for fully elastic transport connectivity are often trivial. On the other hand, a typical distributed data center (DDC) has fewer tenants, fewer statistical multiplexing gains, and shares connectivity over access and aggregation links. Consequently, DDCs will have a proportionately greater need for additional elasticity at the transport-layer level.

Central to central

For the CDC-to-CDC use case, the bandwidth for transport between the two centers is assumed to be high. This type of link could be implemented through a dedicated lambda on a WDM network, OTN, MPLS or an MEF E-LINE service. This level of bandwidth allows the DC operator to provide high-availability or geo-redundancy services to its cloud users as well as for application mobility, such as follow-the-sun scenarios – where workflows are available to teams collaborating across time zones. Elasticity for these slices is provided on a per-tenant basis and managed dynamically. In this way, the bandwidth per tenant can be managed – enforced, policed and monitored – dynamically at the DC border gateway, as long as the total bandwidth required is within the established shared-transport capacity.

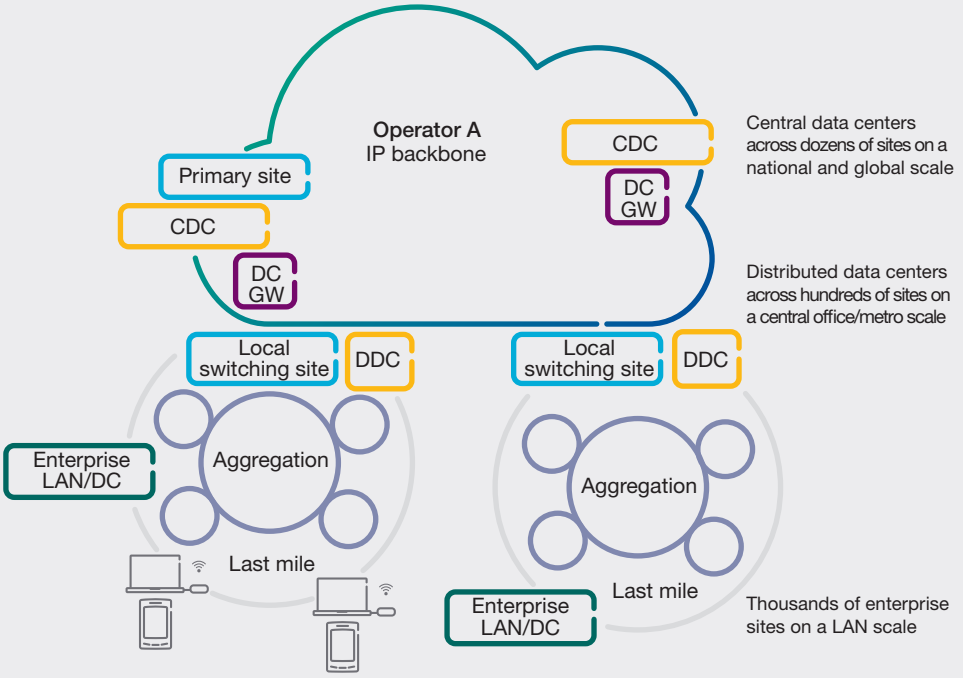
The migration of an application from one center to another exemplifies the CDC-to-CDC use case. Consider an application that resides in CDC_A. For reasons related to redundancy, the user wants to move the application to CDC_B. To do this, the user orders additional VMs in CDC_B and an inter-DC bandwidth, which may contain QoS properties, for a certain period of time. The connectivity between the application in CDC_A and the copy in CDC_B is established by configuring a new virtual connection on top of the existing inter-DC transport connection. This request can be enforced by the border gateway of each data center.

Distributed to enterprise

For the DDC to enterprise-site use case, the need for dynamic-bandwidth connectivity at the transport level is assumed to be greater – as significant fixed-facility capacity may not be available 100 percent of the time. Even though access and aggregation links may support up to 1Gbps or 10Gbps, they may be shared by other services. So the need to access or move applications dynamically between the enterprise network and the DDC creates a need for elastic networking-connectivity. Bandwidth must be managed dynamically throughout the access and aggregation network as well as in the DC border gateway.

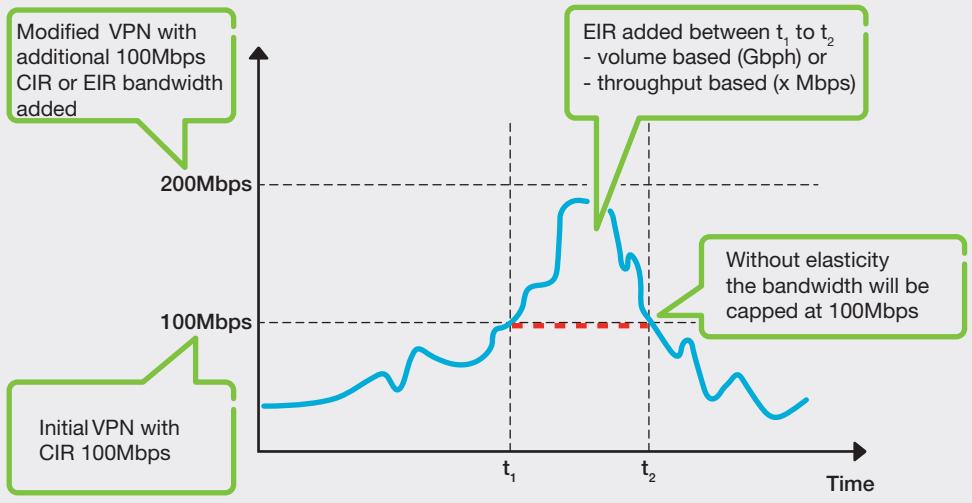
To use cloud services provided by an operator, enterprises first need to establish initial connectivity from their site to the operator DC. This step is performed automatically through the operator's customer portal. An existing or new VPN service ordered

FIGURE 2 **Distributed data-centers in the operator network**



Connectivity type	Transport elasticity	Per tenant elasticity
Central DC to central DC	No	Full
Central DC to distributed DC	Limited	Full
DC to enterprise site/DC	Limited/full	Full

FIGURE 3 **Properties of an elastic network-connection**

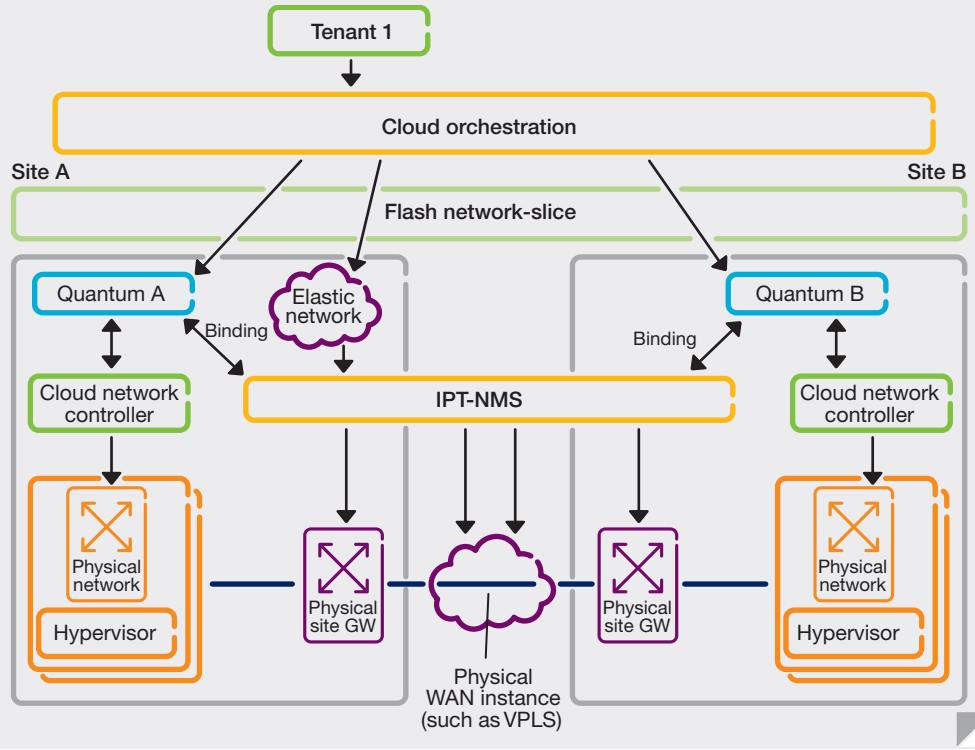


in this way could include a basic connectivity service, with say 100Mbps CIR bandwidth.

Once operational, an enterprise may need to modify connectivity dynamically. This can occur, for example, when an application running at the enterprise requires additional computational resources from the cloud service. Flexible connectivity is achieved through the cloud network orchestration and, in this case, connectivity bandwidth might be increased from the existing 100Mbps to say 200Mbps for a specific time period. The requested connectivity is provided by modifying the existing VPN, and is enforced in the border gateway and intermediate network. As highlighted in **Figure 3**, the additional bandwidth may be offered with properties that are different from those included in the basic VPN service. It may, for example, be offered as an EIR or as a measured service where the user is charged per megabyte.

In this particular use case, the elastic connectivity impacts the allocated resources inside the DC as well as in the network, which requires coordination between the two.

FIGURE 4 **Elastic networking with IPT-NMS**



Missing technology

To enable the network-embedded cloud at the infrastructure level, a number of key technologies need to be added to today's cloud deployments, one of which is support for network virtualization. To achieve this, the cloud operating system must support virtualized networks as first-class objects in the API, exactly as for virtual machines and storage blocks.

OpenStack, an open-source cloud operating system currently under development, contains support for virtualized networking in the form of a virtual Layer 2 network management API called Quantum[1], which supports the following objects:

- network – a virtual isolated Layer 2 domain in the data center for the exclusive access of a tenant;
- port – a logical Layer 2 port on a virtual switch; and
- attachment – a Layer 2 interface providing network services to a virtual machine.

The Quantum API allows a tenant to create and delete these objects, plug attachments into ports, unplug them as well as perform other operations. It is the job of the data-center network manager to implement a Quantum Layer 2 network on top of the physical data-center network. To export this API as a plug-in there are a variety of technologies available to the network manager, such as VLANs and IP tunnels.

While this API supports virtual networks inside the data center, implementing NaaS in the network-embedded

cloud requires additional support to link virtual network resources inside the data center with those outside the data center (in the WAN). Ericsson's elastic-networking extension to Quantum provides this support.

Elastic networking supports the creation of a VPN over the WAN by adding virtual links and connecting the VPN to Quantum networks in multiple sites. The resulting VPN is an implementation of the flash network-slice abstraction in the OpenStack Quantum cloud operating system[1].

As with Quantum, an elastic-networking agent is required to tie the API into the physical network. For that purpose, Ericsson's implementation supports an internal API that binds the external API implementation to a specific network management system. The concept behind Ericsson's IPT-NMS[2] supports managing wide area network connectivity, so a system design using this component as part of cloud orchestration can leverage this. The resulting architecture is illustrated in **Figure 4**.

The cloud orchestration system deploys tenant applications in the different data centers depending on their requirements for computational, storage and networking resources. The system handles internal resources, including networking via the Quantum API, and calls the IPT-NMS via the elastic networking API to handle the details of WAN network-establishment between data centers – relieving tenants of the need to be concerned with such details.

Being able to make the most of the integration between cloud orchestration and NMS gives network operators with their own data centers a clear advantage. However, in the case where the data center and the network are managed by different administrative entities, coordination between cloud orchestration and the NMS will be required to establish a flash network-slice, stitching each part of the slice over the domain boundaries and over the multiple domains involved.

The final piece of missing technology in the network-embedded cloud is support for diversely deployed computational resources that are often present in networks, such as unclustered servers, RNCs and routers with service blades embedded into the network. Ericsson's prototype cloud orchestration system allows operators to manage these resources in the same way as they do large homogenous data centers.

Routers running a hypervisor on blades also run an instance of the cloud orchestration agent on those blades. The agent insures that the VM instances are running the latest software by managing upgrades when new versions are released. Cloud orchestration can also manage content-distribution software from multiple vendors, as content from certain sites can require a particular type of content distribution network (CDN). Attempting to manually manage the complexity created by different vendors' software, patch releases and upgrades is a challenging task, with a high probability of errors and resulting in misconfigurations.

Conclusion

Enterprises have started to move applications to the cloud. However, to fully adopt the cloud as a central technology, additional network considerations such as bandwidth, latency and privacy need to be addressed. Current VPN technology takes these parameters into consideration in the WAN but comprehensive integration into the cloud technology base is lacking. The same level of flexibility in allocating and managing flash network-slices needs to be provided by WAN networks as the cloud provides for allocating and managing computational and storage resources. To cope with the increased traffic generated by smartphones and tablets, the cloud will have to get closer to users – deploying computational and storage resources in small, micro or pico data centers embedded in the network.

As they have control over resources in both the network and the data center, network operators managing data centers have a clear advantage when it comes to supporting enterprises in establishing connectivity. To combine NaaS with data center computational and storage services, closer cooperation is needed between cloud orchestration and the NMS. Extensions to Quantum and OpenStack to connect data-center virtual networks over the WAN can fulfill that need while pico data-centers can be incorporated into the architecture in a seamless manner.

David Allan

is a distinguished engineer at DUIB Systems and Technology, Business Unit Networks at Ericsson Silicon Valley in San Jose, California. He joined Ericsson in 2009 and his current role focuses on carrier and cloud infrastructure based on MPLS and Ethernet. He holds a B.Eng. from Carleton University, Ottawa, Canada.

James Kempf

has worked at Ericsson Research in Silicon Valley on software defined networking (SDN) OpenFlow and cloud computing since 2008. He graduated in 1984 from the University of Arizona in the US with a Ph.D. in systems engineering, after which he worked with various blue chips in Silicon Valley, primarily in research. He has also worked with the IETF for 10 years.

Torbjörn Cagenius

is an expert at BNET System Management, Business Unit Networks. He joined Ericsson in 1990 and has worked in a variety of positions and with different technology areas. His current role is system area driver for distributed cloud architecture. He holds an M.Sc. in electrical engineering from the Royal Institute of Technology (KTH), Sweden.

References

1. OpenStack, 2012, Quantum API Guide v1.0, available at: http://docs.openstack.org/api/openstack-network/1.0/content/index.html
2. IP Broadband Network Management, available at: http://www.ericsson.com/ourportfolio/products/ip-broadband-network-management