Ethernet Activation Goes Digital

The next-generation of automated service-activation systems and tools for MSOs

A Technical Paper prepared for SCTE/ISBE by

Mark Gibson
Director of Product Management, Amdocs Network Group
Amdocs
Mark.gibson@amdocs.com

Table of Contents

List of Figures

Introduction

Metro-Ethernet activation involves much more than just turning up the connectivity –it's also about turning up the support and active-monitoring functions of the service (such as fault management) to ensure MSOs can track and monitor the service over time once it's live – and that's not something that's easily performed manually.

Activating a service on DOCSIS might not be simple but at least it offers a standard interface. On the other hand, Ethernet – partly due to its ubiquity –shows up across a range of equipment type, from optical switches to edge routers, each offering a variety of provisioning techniques.

It used to be a choice of TL1 or command line interfaces, but now these are augmented by YANG, XML/SOAP, MTOSI and a plethora of other methods, each aligned to a domain-specific standards organization. And as if this weren't enough, modern Ethernet services are expected to come with active reporting on the health of the connection, giving in-life updates. In fact, frequent configuration of the monitoring actually results in more commands than the service itself.

If a network is to operate flexibly and support the rate of rapid service change that MSOs want to offer customers, automation is crucial. In a world where customers can press a button to change their service bandwidth, it's unacceptable that this should result in a network engineer typing commands.

This paper shows how activation must be driven from a service design that reflects both the connection, monitoring, and operational KPIs for the service so that the whole lifecycle is activated.

It also shows how linking a service design to command generation enables quick command generation and removes human error.

Referencing a variety of customer use cases, the paper explains the required features of an Ethernet activation system for a digital customer experience, and also shares Amdocs' new initiatives in MEF to harmonize network provisioning and reporting.

1. The problem for MSOs

Step into a modern office building and it can resemble something like a science fiction movie from the 90s. Workers sit in front of two, three, or even four screens, looking at colleagues wearing headsets from across the globe, while a live demo whirs with ghostly precision in the background and a chat stream critiques the content in a closed group. Other employees look at social-media postings of colleagues at an off-site meeting, while yet more are at the coffee station reading the latest WhatsApp group video post from their kid's foreign holidays. Someone, somewhere, is even reading an email.

For customers looking for providers to support these work-environment services, there is great news – in many regions, there are tens of providers who will promise to hook up your premises and deliver you "unbeatable connectivity" at a market-leading price. But once the customer is signed up, keeping them happy and preventing them from looking elsewhere for a better deal critically depends on the MSO showing value to the customers they serve.

There are six main areas that an MSO needs to address to be competitive in the enterprise space:

- A flexible network foundation capable of delivering up to gigabit speeds.
- You must have a large geographical reach so that you can serve a customer wherever they have an office
- The connection that you offer must be reliable and when this is not the case, the customer must be made aware of this quickly
- The solution provided must be future proof and able to adapt to the changing needs and operational models of the customer
- The solution should be capable of outsourcing IT operations of the customer and providing a seamless work environment.
- The solution must be able to adapt to the customer's own environment including the network infrastructure that they operate.

While each of these relies to a greater or lesser extent on an operational process, they all result in a service or a device being activated.

In the following sections, we'll look at how the role of activation varies, and how, in many cases, it's more than just sending commands to a target device.

For customers, it's no longer good enough for an MSO to activate a service at the start of its life, and to only change it when there is planned maintenance. For an MSO to differentiate themselves in the marketplace, they don't only need to provision a service, but they also need to manage and maintain that service over its lifetime, so that they can reduce outages by either reacting as they occur, or predicting them *before* they occur. This then gives the MSO a toolkit that can be applied to more interesting and flexible offerings.

The days when an activation operation pushed commands on to the network and then handed over maintenance almost context-free to the network operations team are gone.

1.1. Strong Foundation

There's little doubt that Ethernet has won the race to be the Layer 2 interconnect of choice. And while as recently as 10 years ago, you might have heard any of a number of rival technologies being mentioned, **in today's networks, there is only Ethernet**.

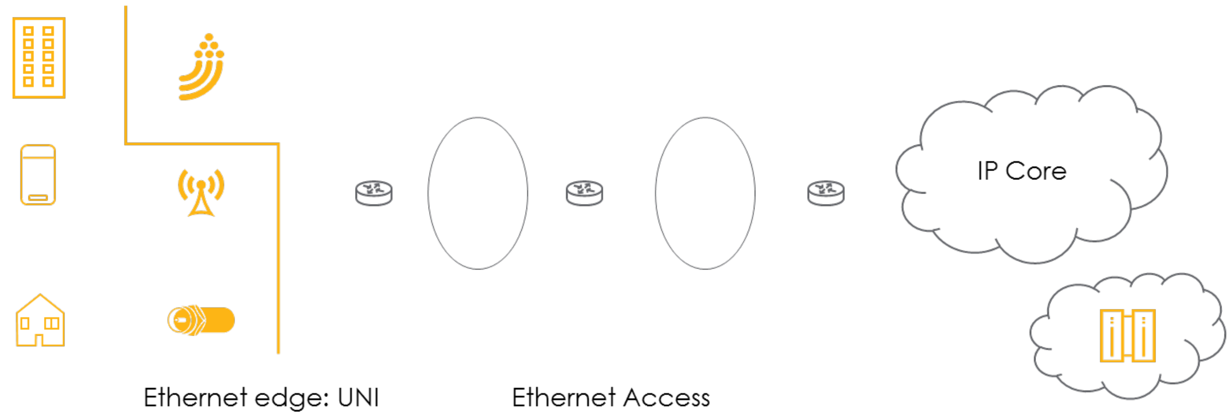

Figure 1 – Ethernet is now a ubiquitous aggregation technology

Except… that's not quite the case in reality.

Nearly all Ethernet Services are described according to the service types laid out by the MEF (www.mef.net). The customer end of the service is described as the UNI (User to Network Interface) and it is at this port that customer receives their Ethernet traffic. However, there can be many types of Ethernet Service between different UNIs.

For example, in some cases, you see a L2VPN network which only needs edge configuration to enable the port-to-port connection, sending commands to the edge devices of the MPLS network to map the Ethernet traffic onto an MPLS connection.

Anyway, regardless of the technologies, there are many ways to enable the service, ranging from a command line interface (CLI) to a RESTful service-level interface that abstracts the service from the network details. There may even be some flavor of YANG being used.

In many cases, there are multiple generations of network equipment being used to deliver similar end customer services over dissimilar infrastructures – this means that the activation platform needs to adjust to each of the network systems that it finds. But however this is abstracted, it's a complexity that still needs to be addressed, and the orchestration layer above shouldn't have to be aware of how the network is implemented.

MEF's Third Network Vision sets out a candidate blueprint for how Ethernet can be used as the common foundation to support the diverse needs of an MSO and ensure normalized and predictable services which will leave MSOs free to concentrate on the value-added IP services that create more profit and customer interest.

At the core of the vision is the concept of Lifecycle Service Orchestration (LSO), which defines a methodology and practice for managing network connections at a service level. It looks at the whole service lifecycle from modelling the service to activating the network equipment to provisioning the whole support infrastructure needed to maintain the service.

However, as Figure 2 (Source: MEF Forum [0]) sets out, the vision of LSO extends to how you link the services that a typical MSO needs to provide into a coherent managed whole within an MSO. It also looks at how you manage the co-operation between MSOs to offer global scale offerings outside of their geographic region.

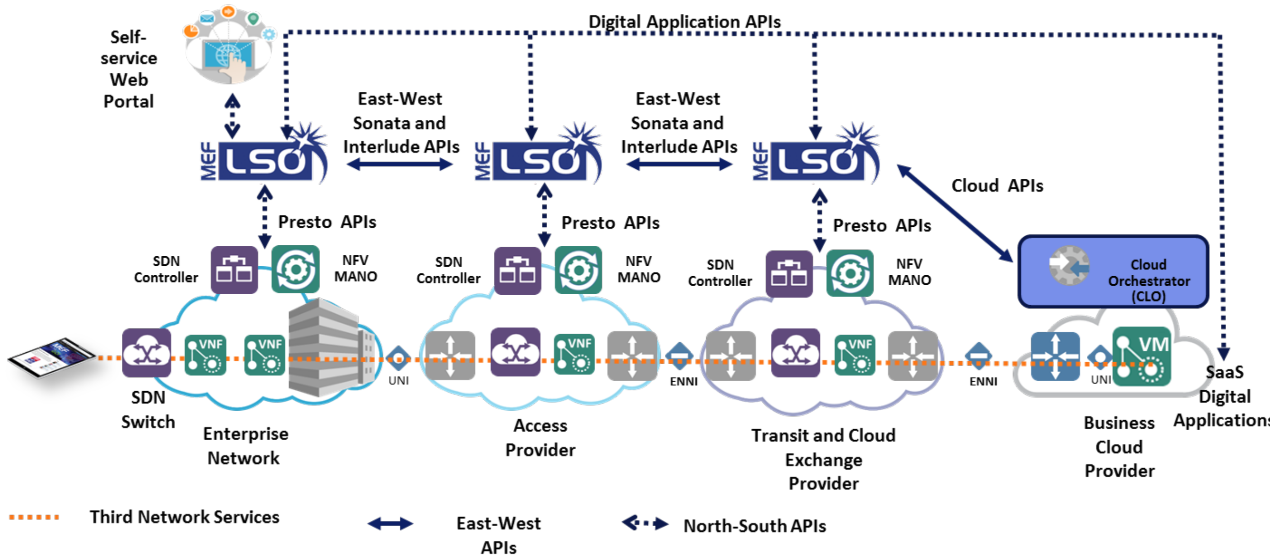

Figure 2 – MEF Third Network Architecture

In fact, in a recent survey [1] **over 90% of respondents said that dynamic, orchestrated Third-Network services will transform the Carrier Ethernet Market, having a significant impact in the time period of 2018+**.

1.2. Geo reach

Few things in life ever turn up out of the blue and just slot in to whatever plan or space you have handy. The same is true of enterprise customers asking an MSO to support a new service, especially when the enterprise is a sizeable one – the larger the customer, the more likely it is that their footprint will extend beyond the geographical boundary of their provider, and the more complicated the service turn-up process will be. A key measure for customers about whether the MSO will be able to support their request is how long it's going to take to get a new service up and running – an MSO that can overcome this first obstacle quickly can differentiate themselves from competitors.

In reality, it's likely that many enterprise customers, especially larger corporations, will require support that reaches beyond the limits of a single MSO (and a highly probable scenario in the US given the traditional geographical divide). But this is also the case in regions like Europe where many enterprises have established offices and data centers in cheaper locations – full end-to-end services can span multiple operating companies within the same provider, who will need to co-operate to provide services.

In the same survey, **over 91% of respondents cited orchestration over multiple provider networks as either a serious or major challenge** [0].

Recently, significant effort by industry bodies has been made to simplify the task of ordering between carriers. The Telemanagement Forum (TMF) OpenAPI work stream [0] has developed a number of inter-carrier ordering APIs: Service Qualification; Quote and Service Ordering. This work is now being picked up in MEF and used to generate industry standards for inter-carrier service ordering using normalized attribute lists and defining predictable outcomes. The first Release of the code (named LSO SONATA R1 after the inter-carrier ordering reference point) took place in July, with the open-source examples shared within the MEF community. While this first release was as much a proof of concept on the APIs themselves, work can now accelerate beyond the initial E-Access service use case to other inter-carrier services.

1.3. Reliable connections

Just because the network now makes it easier for the MSO to turn up and order, doesn't mean the enterprise customer will suddenly lower their expectations of performance and reliability. However, this equally doesn't prevent the MSO from differentiating itself at the point of sale either.

A MEF CE 2.0-certified service [0] will come with support for Operations and Maintenance (OAM) traffic, giving the MSO the ability to measure how the service is operating. This can be compared against the established KPIs for the service variant and allows for alarms and alerts to be generated when the service is non-conformant.

There are two points to consider here though:

The first is how, and when, those KPIs are established and linked to a service.

The traditional Fulfillment and Assurance model has treated them as separate operational stacks with no more than a "ships-in-the-night" passing acquaintance with each other. However, if full lifecycle orchestration is going to be achieved, that gap needs to disappear and the systems merge into a continuous whole. A better solution is that the service model that defines how to design the service instance also describes how to provision its measurement over time as well as the alarm levels it should have.

The second is what to do with the information that has been collected.

When a service has been ordered across operators and boundaries with a partner MSO, it raises the issue of operational information about congestion, packet loss and other OAM information – all of these are highly sensitive business intelligence. And while it might be laudable for an MSO to transparently share information with the purchasing partner about the operation of a leased connection, it is also an unlikely scenario. So setting the KPIs in the ordering process that will be reported against is critical to keeping the customer satisfied. The same is true of a simpler customer-MSO relationship for a more traditional Ethernet Service.

So regardless as to how the information is shared externally, the MSO must have a prepared action plan that is enacted when a critical KPI is hit. This is the final part of the activation process – to link the action plan to each OAM event that is of interest, (even if all that means is to generate an alert that goes into an operational dashboard).

Ensuring that the in-life response is ready when the service goes live ensures that the link between fulfilment and assurance is made.

1.4. Changing customer needs

A customer's needs regarding a connection often change over and the ability of the MSO to support this is becoming less of a "nice to have" feature and more of a service differentiator. This really boils down to one of two operations:

- Adding another service to an existing set
- Making a change to an existing service.

Adding a service will use the same sort of ordering process as the original connection was established but here a couple of alternatives are possible:

For example, when activating on the same UNI, it's important to establish that there's enough capacity to support the new service. The UNI service configuration helps with that, while still allowing the flexibility to support concepts such as overbooking. But this is a service design-level construct in the orchestration layer – endpoint activation is still enabling a VLAN and cross-connection to a connectivity layer such as MPLS.

The trick here is to be sure that the resource is available without creating a burden of reservation and management… Linking the service to a location at the point the order is taken and then linking that explicitly to a specific port often gets the right balance between knowing you can support the new service at order time and avoiding fallout during activation by picking the port too soon. (It also makes tidying up dormant orders simpler).

More interesting is how a customer makes a change to an existing service and the level of direct control an MSO wants to let a customer have over their network service. (And this will mainly depend upon how directly the options offered in the customer portal match against the resulting activation operation).

Figure 3 – Typical customer portal operations

For example, the portal might allow the customer to adjust the CIR by setting the exact rate. Here the command might take a fairly direct path to the activation layer to alter the port configuration. However, even to enable that, the change must be within policy – this means that the portal must only permit allowed values, and the change operation still needs to be represented in the service model and checked against the operational policy too. For example, if there is any overbooking at all on a physical port, the system needs to check that not all customers have turned up the CIR at once.

And what if the customer wants to make a change that's outside of the policy?

Well in this case, you're back to an ordering interaction since this becomes a billable interaction with the nature/level of the service changing. In this scenario, the portal should only offer the customer the options that can be supported at the location, based on the device capabilities. This needs a good integration between the activation layer and the service catalog. This will then be further refined by the BSS layer to constrain the options that can be offered to the customer.

So activation only occurs here once the change is agreed upon, (although for changes which *don't* require new physical resources, the order-to-activation interaction can be very rapid.)

Finally there's a question here regarding where the portal resides. A customer portal does not have to be in the customer domain – one common approach is for the MSO to host a portal that their customers log into to make service level changes. However it is also possible for the service provider to provide APIs on top of which the customer can build their own portals. (And as this allows direct access to MSO-owned devices, security of access is a key consideration).

1.5. Managed service

While giving a customer access to their service so they can make changes is definitely a step up from today's services, they don't enable the MSO to deliver extra value to their customers. If a customer has flexible connections, that customer still needs to run analytics and monitoring between their sites to understand how they are using their resources.

But here is how an MSO might be able to innovate, because rather than having just one connection service, they can offer customer choice of different connection levels:

- Standard Ethernet connection with only major outages being reported.
- Enhanced Ethernet connection where regular reports against KPIs are produced with a live alert stream in a customer portal
- Managed Ethernet connection where the MSO actively manages the service against KPIs and makes proactive suggestions about how better to manage the service. For example, if a customer has two connections and one is regularly hitting a capacity limit while the other remains under-utilized, the MSO could suggest adjusting the capacity limits on the two connections to even this out.

In all of the above models, the MSO could perform active service management, making changes to the network to sustain the service. But in the last case, **the MSO could start to offer proactive service management**.

For example, if a customer has a multi-site system but there is congestion to and from one site, they may choose to increase capacity to that location, while decreasing it elsewhere to provide a good customer experience. This takes the service towards a closed-loop style of operation where the network layer refers to a higher-layer operational policy for guidance and there is an automated re-activation of the network to match current conditions. Whether this looks like traditional activation (e.g. command line changes to a configuration file), or an operational policy change (to an SDN Controller) followed by monitoring, will depend on the MSO's operational environment.

1.6. Customer environment

An extension to managing the service for the customer inside-out, is to offer to manage the customer's edge devices. This Customer Premise Equipment (CPE) can take many forms, and it is this very lack of uniformity that makes CPE onboarding such a challenge for an MSO. While ordering is moving towards becoming a fully automated process, managing a new set of CPE equipment for a new customer can pose a different problem each time.

For MSOs seeking to offer a full managed service, the time it takes to model and be able to activate CPE devices is one of the biggest delays in a service going-live.

2. Towards a flexible activation solution

Wind the clock back ten years, and IP and Ethernet service activation was divided into a set of clear categories.

- Command Line Interfaces directly to device configuration files
- XML over HTTP interfaces to an EMS
- MTOSI variant of XML over SOAP
- RMI
- CORBA
- TL1

Come back to the present day and… well, in some instances, you still find these provisioning interfaces.

Most MSOs will run equipment that is stable and supporting thousands of services well beyond the end of extended warranty periods, rather than go through the disruption of managing the migration of those services onto new equipment. But obviously, you still need to touch those old devices occasionally.

Network mediation does move on though, and **in this section, we will examine new trends: from new mechanisms to the new operational patterns they enable**.

2.1. Ethernet to the edge

One of the more striking developments of the last 10 years has been the convergence in architecture to the last mile of a customer connection. Ethernet has not only become the de facto standard way to build out a network aggregation architecture but it has also normalized what such a network looks like to the extent that the difference between an MSO and Connectivity Service Provider (CSP) aggregate network diminishes and common toolkits and approaches can be shared.

As a result, MSOs can now benefit from the work in the CSP community to build Ethernet services. The certification process by the MEF (www.mef.net) has led to Ethernet providing a predictable and reliable substrate for a raft of services. It provides the sort of well-behaved underlay that makes operations more straightforward and lets CSPs and MSOs alike concentrate their efforts on delivering the overlay services that generate more money.

All that being said there are still a number of pieces that need to be marshalled to realize seamless operation.

2.2. NETCONF and YANG

NETCONF has been around for a number of years and was available on routers as a configuration method from the turn of the decade. Early data models were variants of CLI commands but started to harden with the introduction of YANG models a few years after NETCONF was finalized.

NETCONF provides a great standard way of managing configuration, with built-in functions around rolling back those activation commands that used to keep activation developers awake at night. These sort of high-level EMS functions for configuration management are now available on individual devices.

However, while the push and pull of commands has been standardized, there has been a huge diversification of the flavors of YANG that are on offer, described recently by Cisco CTO Dave Ward as "50 shades of YANG" [0]. For the activation developer, this means that the problem shifts from one of building out a robust API with the activation target, to one of flexibly modelling the differences in YANG and choosing how much of this "vendor variation" to hide from the service design layer, (and how much needs to be exposed as a feature). In this respect, not much has moved on from the days of Cisco IOS and its multitude of variations.

Most forward looking architectures mention YANG extensively due in part to the fact that is stratifies into two distinct layers. The lower layer looks like a traditional CLI and uses recognizable commands structures for turning up network functions. However, YANG also offers a service layer too, which, while this is not used for devices access, is used as an entry point to what used to be called the NMS layer. This introduces another decision in how to build activation into an overall system.

While YANG is relatively prevalent in IP and Ethernet core switches it is by no means ubiquitous and the activation layer needs to be adaptable to not only manage YANG and non-YANG equipment but to (attempt to) insulate the service design layers from the method used to provision the network.

2.3. Software Defined Networks and Controllers (and Orchestration)

Few terms in modern networking have evolved so dramatically in the last ten years than "**Software-Defined Networking"** (SDN). What started out as a means of decoupling the control plane of networking equipment from the data-forwarding plane now encompasses all means of real-time, near-real-time and offline network traffic engineering. An SDN-controller used to mean that it interacted with devices solely using OpenFlow – now the prevalent use cases are based on a YANG configuration.

For the purposes of this section, we are focusing on SDN Controllers – the modern, generally open-source, replacements for the NMS. The classic example here is the Controller from the OpenDaylight project [0].

To an extent, the SDN-C has changed what activation means in a modern network. There have always been two flavors of activation into a network: device direct connection and NMS-based, but the SDN-C now adds a whole extra level of capability, and builds a new strata layer too. While an NMS built out a local model of the devices that were under its control and could present a service-like model, it generally could only derive attribute values based on algorithms encoded into the system.

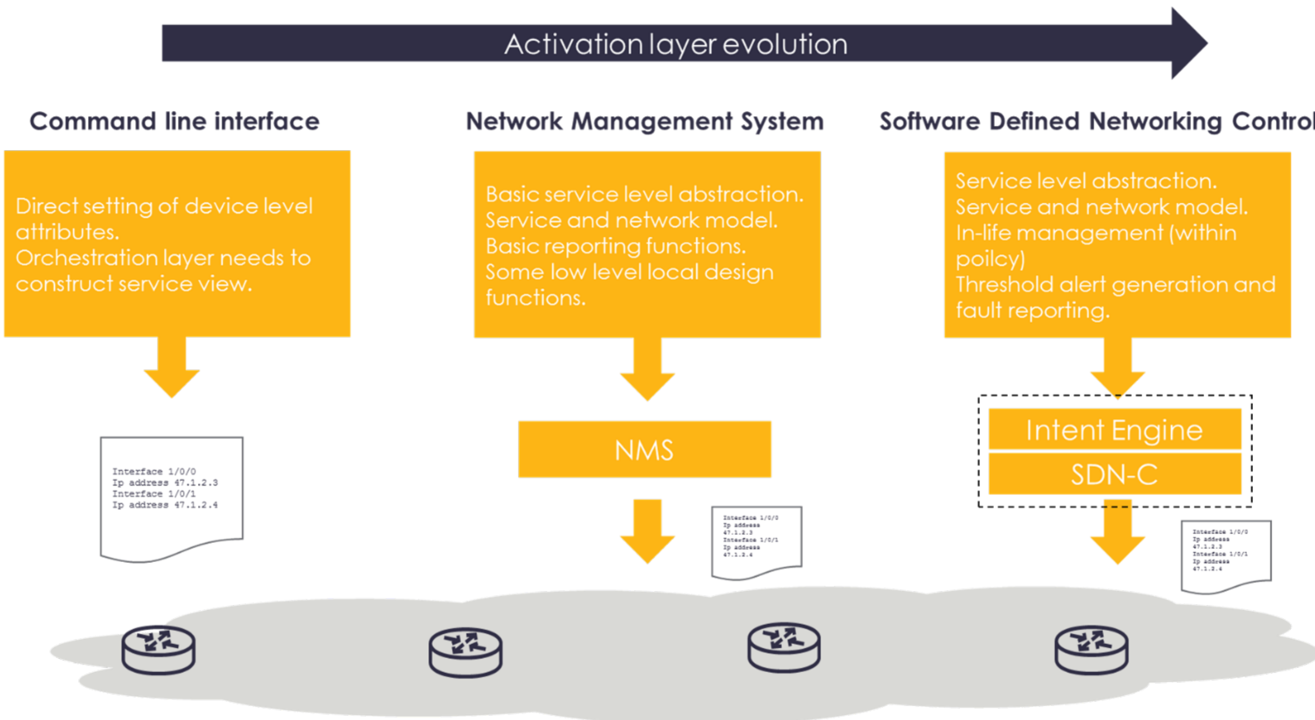

Figure 4 – Evolution of network activation

An SDN-C aims to abstract that further by using policies that indicate intent. In this model, the overlying OSS or orchestrator gives the SDN-C a fixed set of constraints for a new connection, but lets the SDN-C figure out the rest of the implementation detail. The aim here is to let the SDN-C, which has close proximity to the network and a local network model, figure out what's best for the connection over its lifetime and make adjustments as necessary.

However, in practice, the amount of freedom an SDN-C has will depend on:

- How the service is sold in the first place and what the UNI and NNI restrictions are.
- How many options there are in the network for moving the service between resources.

For an Access Ethernet Service on fiber-access network to a PE device on a fiber ring, there may be little practical flexibility.

An SDN-C tends to be able to exert more influence on the traffic streams that are supported by the Ethernet Services. As such, there is a growing market for deploying an SDN-C in the customer environment to make the best use of a leased set of Ethernet connections by adaptively tuning how IP traffic runs across this constrained infrastructure.

The ability to support some of these in-life operations means that most SDN-Cs will tend to also support some local orchestration capability. This is a decision point when deciding how to deploy an overall solution and where you want to build out the orchestration of a service.

So from one perspective an SDN-C is still technically activated, but in this case with a looser more intent-based description of a network connection, described as a YANG service. From another perspective, the SDN-C has now become the activation layer taking in requests to turn up a service and managing the whole process itself. Whatever the positioning, **the SDN-C is an important emerging architectural component**.

But it's not a universal panacea.

There are those who would position the SDN-C as a universal activation/orchestration layer with a single inbound API definition, the idea being here that with some extension modules to support legacy devices, all of your service activation needs can be delivered by a single solution.

In most activation systems, a key measure is how easy it is to add a new device or service type as this is a key dependency in time to market for a new offering. In many regards, the emerging SDN-C market is no more efficient than the existing mature activation market. In some cases, those mature solutions are significantly better having been invested with toolkits and configuration capabilities over the years. **In a hybrid network solution, an SDN-C might not always be the best answer. But there is a balance here between the number of platforms you need to support and the time to get to a working solution**.

2.3.1. Intent in focus

A recent development in the usage of SDN Controllers has been to entirely remove the Intent component of the SDN-C and make it a separate layer. Curiously this makes the rump of an OpenDaylight-based SDN-C resemble little more than a multi-vendor Open Source NMS. It also confuses the boundary of Intent somewhat.

Intent is an abstraction of design and implementation of a service where the specifics of an implementation is somewhat hidden from the way the service needs to operate. This is not really a new concept in Operational Support Systems, indeed often the only fixed point in a service design is the physical port that the customer connects to. All other aspects of the way data is transferred between these points is generally abstracted into a high level specification of the way the ports are connected (point to point, hub and spoke, …) and a QoS policy that is pre-configured in the network. Many Orchestration systems on the market today use a similar sort of approach too.

In the longer term it may be that Intent is fully re-homed into the Orchestration layer or it may evolve into a separate and separable module alongside path computation. Much of this will depend on the use cases it is asked to solve and whether the separation adds significant benefit.

2.3.2. Other Controllers

Not all network controllers come with an SDN-heritage, (or at least they are perhaps more honest in that regard). Most multi-vendor EMS solutions have nearly all the characteristics of a classic SDN-C but without the OpenFlow support – they are also generally not open source. However, some offer the same sort of service abstraction offered by an SDN-C and many have more developed toolkits for supporting new devices than are readily available in the OpenDaylight domain. Most seasoned activation vendors will have been challenged on the time it takes to add a new device and those that are successful will have overcome this problem.

2.4. Network inventory and configurations

It is a common mistake to assume that because you have a complete set of device configurations, you also have a network inventory. This isn't the case and can be proven by this simple thought experiment.

Suppose I (as the MSO) have a new card that I want to add to a device (which supports hot plug-in of the card). Once I have added the card, is it automatically configured with the way I want it to operate? No, you have to send the configuration down to run network services on this card.

In an ideal system, your network configuration and a discovered network inventory will sit side-by-side in the same Controller and will be able to be queried interchangeably. This is important as both are important data sources in decision-making prior to activation.

Let's say I'm trying to decide whether I can support a new service at a location. I have two master reference points for making a decision. The first is my top-down network service inventory, which tells me how many designed services I have terminating in that location, and also how many new planned services are waiting to go live – this is my demand. The second source is the network inventory from the activation layer, which tells me the current network situation of used ports – this is my supply. Though note that to get a true view of the supply, the network inventory needs to have looked at the Interfaces Management Information Base (MIB) which tracks all of the ports on a device, not just those that are active.

By subtracting the demand from the supply, you can make an accurate decision on whether you will choose to support a new service at that location. A qualifier here might be that at ordering time you might decide to allow overbooking of resources meaning that you will support a demand in excess of supply. And if you run this query immediately before provisioning, you can have a large impact on activation fallout.

There are other similar processes, but **the point is that having a local and accurate view of both network state in a data model and the device configuration gives the overlying orchestration layer invaluable insight that will make for more accurate overall processes**.

2.5. SD-WAN

Software Defined Wide Area Network (SD-WAN) is a classic example of a disruptive technology in the activation space. There are numerous whitepapers and web resources on what the technology does, but its main value for the MSO is:

- Rapid service enablement by piggy-backing on an existing Internet connection
- Capacity overflow management by handing off excess traffic to a backup connection
- Better customer management by placing priority traffic onto better links and handing off lower priority traffic onto wireless/Internet connections.

SD-WAN helps MSOs set up new connections fast and has quickly gained traction, but on the minus side, it complicates matters in the activation layer. Each SD-WAN offering on the market comes complete with its own controller and API layer. As SD-WAN has exploded in popularity, vendors in the space haven't had time to wait around for API standards or to build on open platforms.

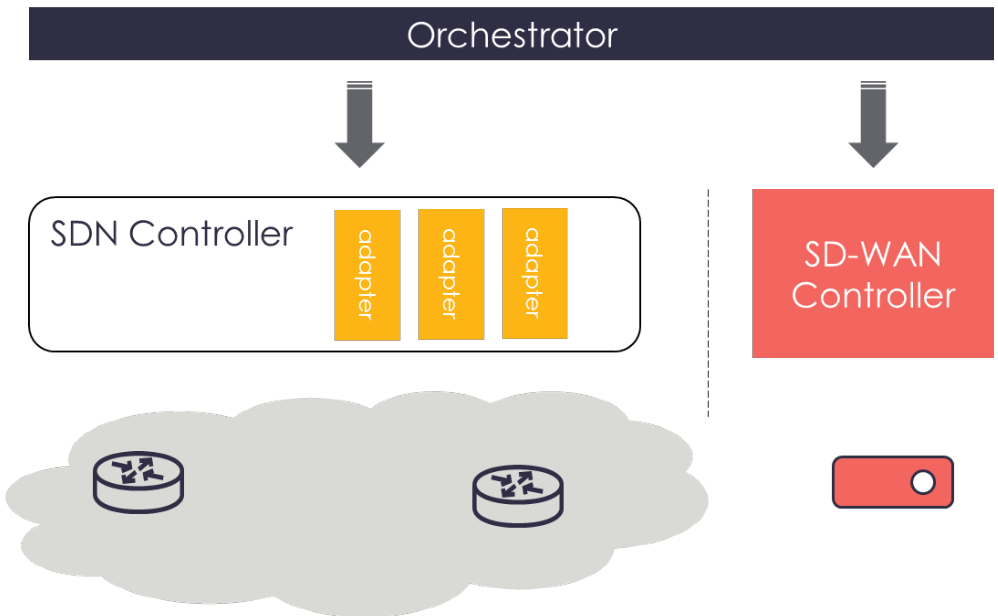

Figure 5 – SD-WAN new vertical

So how do you accommodate such a disruptive technology into an activation layer?

One approach is to position the SD-WAN controller under an overall activation system and build out a uniform northbound API that adds in the new SD-WAN solution.

Another approach is to go against accepted wisdom and add them as a new whole vertical under an orchestration umbrella. The orchestration layer now makes two calls to set up a service – one for the existing network in to the existing activation layer and another into the SD-WAN controller to activate that part of the service. (Note that for SD-WAN, there are likely to be many more interactions to make service policy changes then there are going to be changes to the underlying network connection).

On the surface, for a service like SD-WAN, the decision seems a little simpler since part of the point of SD-WAN is that it acts independently of the provisioned underlying network. In this case, it makes sense to run it in parallel and for the orchestration layer to consume the provided SD-WAN APIs and merge onto a consolidated platform as the need arises. (For example, when your activation provider releases a product upgrade or there is a standard interoperable API that you can converge upon).

In fact SD-WAN is a classic example of how new offerings are introduced and need to be incorporated into a solution space. When a new offering is identified, there is often a race to get a working implementation to market and the success can hinge on being able to provide a full working support infrastructure. In the world of software, this means providing enough of an API to activate/manage a service without worrying where in an existing support architecture this is accessed from. Most SD-WAN solutions ship with their own Controller which is purely responsible for configuring the SD-WAN technology, adding a new vertical that may or more likely may not migrate onto a common controller platform over time. Consider being an IT director making a case to migrate from a dedicated SD-WAN controller to a common infrastructure without there being some sort of monetizable benefit. The most likely push will be the retirement from support of a foundational software element, rather than any pro-active choice.

Architectural purity and alignment is often secondary to market success. And in truth this underlines the need for a flexible orchestration architecture that can flex the point where activation starts to accommodate more service-oriented APIs as a way into the network.

2.6. Virtual Network Functions (VNFs)

There is masses of information available about the way Virtual Network Functions (VNFs) are provisioned and the way the virtual infrastructure that they depend on needs to be managed**. This section will concentrate on the impact of VNFs on enterprise services and on the Ethernet connections needed to support them.**

Prior to the introduction of VNFs, the role of the Ethernet connection was relatively simple. It provided a single IP hop from an IP-enabled CPE device to an IP enabled Provider Edge (PE) device. In most cases this caused the activation system to need to model two device activation operations – one for each device type – to cross-connect the L2 interface at either end of the connection. There were only two target devices.

The separation of IP-forwarding components into VNFs instantly creates a larger number of activation targets, and each VNF is a new target. But there is also new infrastructure that needs to be managed, such as the vSwitch at the entry point of the Data Center (DC) that supports the VNFs. And there are also longer data paths with more IP hops too. The number of IP hops depends on the type of service being delivered and the location of data centers, but one thing is clear, there is not much more work to be done to both activate the VNF elements and to manage their configurations.

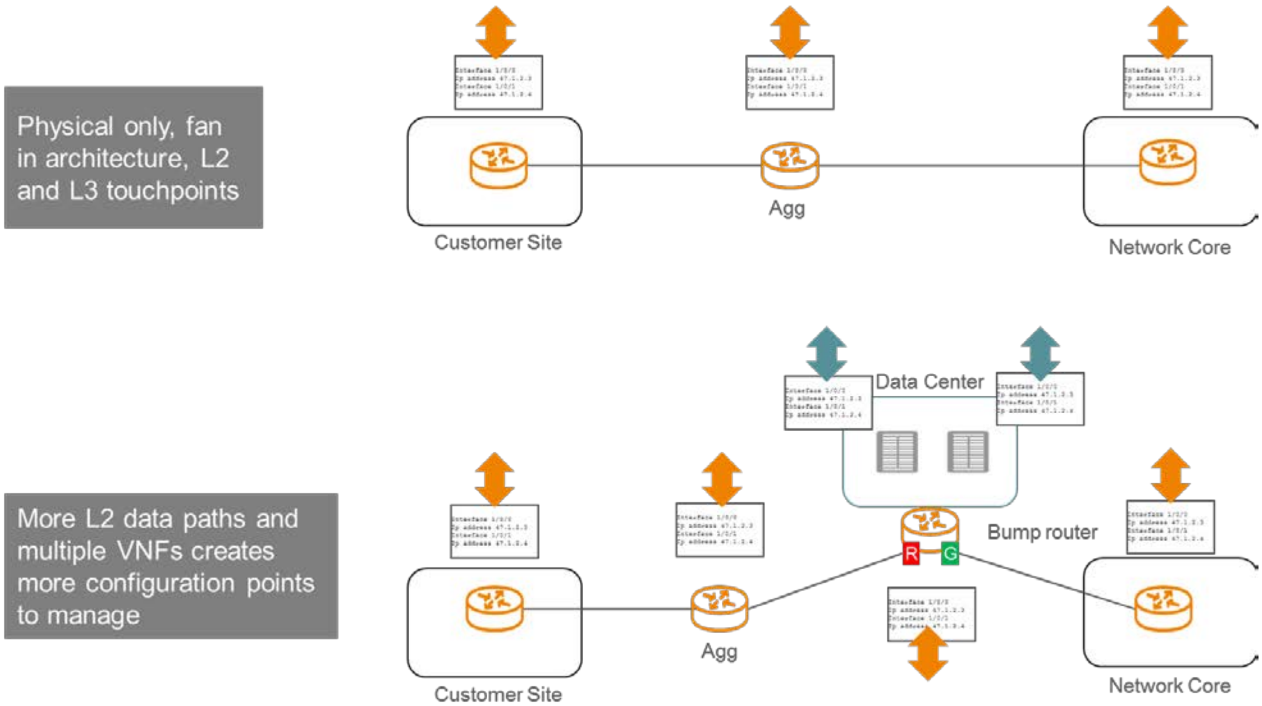

Figure 6 – The impact on network configuration of a simple VNF service

YANG has become the most common language for this configuration, but as noted above you still need a system that can flexibly model each different dialect that you encounter by VNF vendor.

This is also a common operational model where the controller of controllers model is going to be encountered. If there are a group of VNFs deployed in a single DC, then this service chain will be connected together in the DC by an SDN-Controller. The activation system will therefore not only have to describe the individual device configuration, but also the order in which to connect the VNFs so that the SDN-C can manage this.

2.6.1. Live forwarding plane design

As Timon Sloane noted in at MPLS SDN World 2017, deconstructing the IP components of routers into VNFs forces the forwarding plane to be re-constructed when the VNFs are provisioned. Activation now takes on another responsibility**: it's now not enough to accurately describe the configurations at activation time – you now have to get the service chain correct too**.

While in practice this chain will have been predesigned in the orchestration layer from components that have been loaded into the service catalog following lab trials, it still needs to be accurately passed into the activation layer, and the correct response to a failed activation operation or conditional success needs to be supported.

2.7. Assurance

What constitutes activation in current networks is already somewhat blurred and in some cases there might even be two or more levels of activation occurring. **What's important is that the fulfilment and**

assurance verticals are bridged to give a coherent network service-centric view that enables in-life measurement, adjustment and management.

Managing a network used to be the exclusive domain of the Network Operations team who acted as the gatekeeper to the smooth operation of the network and in some cases, regarded activation systems as just another potential source of disruption to a network. The requirement for activation systems to rigorously confirm "golden configs" for each network service being activated is still a benchmark for activation.

And yes, the Network Operation Centre (NOC) is still critical to a smoothly running service, but **an emerging new trend is that of the Service Operation Centre (SOC), which takes a service-level view and looks at performance and planning, rather than alarms and outage management**.

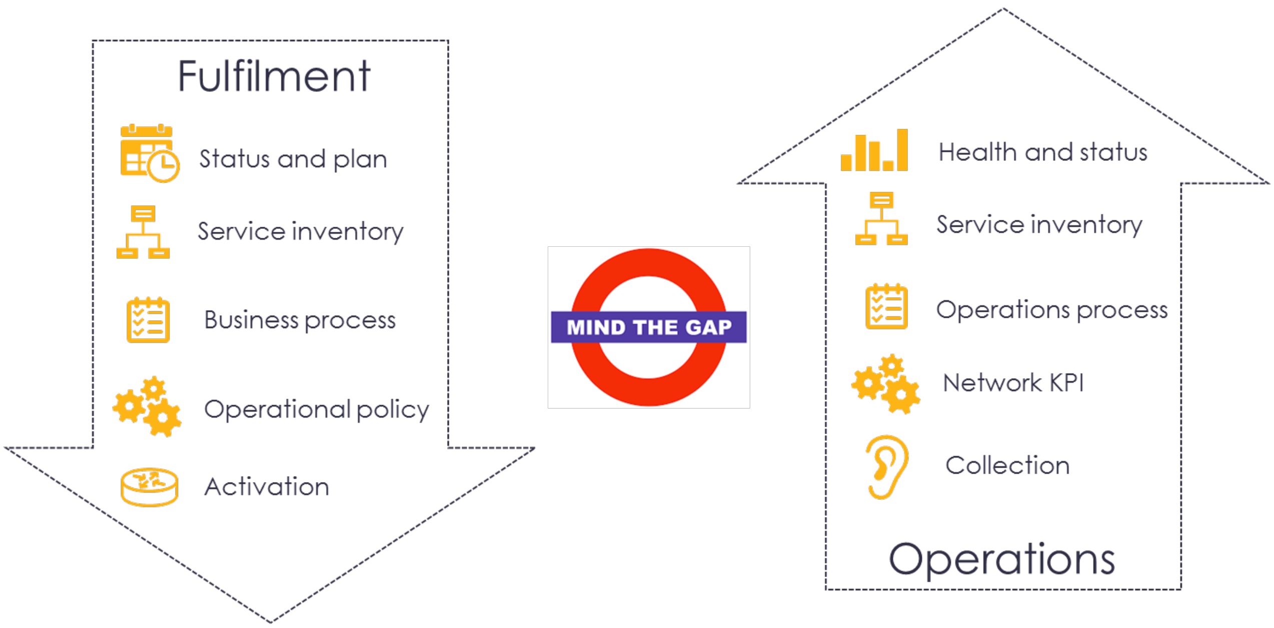

Figure 7 – Fulfilment and Operations gap

For the SOC to prosper, it's critical that there is an integrated approach to defining the KPIs for each service. This means that activation needs to extend beyond simply activating a service, it must also activate the measurement functions to be used during the service lifetime and link the output alarms, alerts and reports to the operations functions. Ethernet has done a great job of defining OAM tooling for EVCs, and an important part of service activation is to properly link an Ethernet service to its correct OAM function.

While this sort of measurement gives a good indication of overall health, a more specific probe often needs to be enabled to investigate a problem further. This results in more activation work, to both configure the probe and then to toggle it between active and passive states. Note that it's common for a probe to be deployed, but which is only turned on when accurate measurement is needed.

Though not strictly activation, setting up the SOC also extends to the configuration of the assurance and support infrastructure too, linking the KPIs for the service to the PM (Performance Management) and FM (Fault Management) infrastructure. While the collectors in a PM system will be set up so that they're liberal in the way they collect data – especially as most assurance systems deploy some sort of big-data warehousing structure to store incoming information – they are generally conservative in the way they generate alerts from that data. The reason is simple: large networks can generate millions of collected data points – however it's only those that relate to an actionable event that are of interest to the SOC and often what is actionable is that a number of smaller events have occurred on a specific connection.

Here's an example: for a given Ethernet connection, if there are five events of 80% capacity threshold crossing within a 24-hour period, this generates an alert for the MSO to investigate the connection since the customer has an SLA KPI that triggers an alert in their portal.

2.8. TOSCA Templates

Having looked at all the tools and levels available to an MSO to manage services, it makes sense to look at a prevalent means of linking it all together. What started as a template method for deploying cloud services, Topology and Orchestration Specification for Cloud Applications (TOSCA) is becoming widely adopted as a means of describing network and hybrid services.

For an IT application it is used to describe all of the artefacts needed to deploy an application, from the software image to the initial day zero configuration of the application. In a VNF context, this usage is extended to cover:

- A topology template that describes the options for deploying the VNFs in the service. This may be as simple as a single rigid example or may allow for context-based decisions. This template also provides links to all of the deployment artefacts needed to enable the service. This might include the initial configuration for standing up a VNF.
- The plan or process for deploying a service composed of multiple VNFs. For example, defining the dependencies and complex sequencing requirements of the VNFs over and above the topology template.

One of TOSCA's biggest strengths is that it is file-based and thus able to support any configuration file or operational template/method that is in common usage in an organization, though this also rather detracts from driving common standards. However, within a single organization it can harmonize operations effectively and provide a common means of service definition both into and out of an orchestration layer.

While it can support initial configuration, services that are enabled by TOSCA still need to be activated and have their configuration managed over their lifetime using mechanisms like NETCONF and YANG. But TOSCA can crucially group all of the service artefacts together meaning that it establishes a practice of binding the Assurance of a service to its initial deployment in a single definition. Here its flexibility can be a boon to bind disparate components together.

A note of caution however. TOSCA is still in the throes of settling down into a pattern of usage. While early deployments used XML, YAML is seen as the current preferred method of describing human readable templates. Even more confusingly, YANG service descriptions and TOSCA templates are often advanced for the same use cases within standards bodies. However, as most MSO deployments will show some variation based on existing deployed operational components, having a choice of tools at each layer allows for a choice to be made of the best fit.

3. How to deploy the tools you have

This whitepaper has set out a number of tools and related domains for activation and indicated a number of points of flexibility for Ethernet service activation.

This section will provide some guidance on how best to use these elements to build a successful solution (with the caveat that each environment will vary, and what's right in one place might not work as well elsewhere).

Nevertheless, there are a few important decisions to work through:

3.1. Where to orchestrate

Orchestration spans all MSO processes, from order capture and network build, to activation workflow. In general, it's good practice to cut the number of different orchestration systems back to a minimum in any single system so that the number of orchestration platforms supported by an IT organization is minimized. This allows greater re-use and establishment of best practices when developing. It also helps that the user community are familiar with the engine across long and complex processes.

In practice, this generally means centralizing design decisions into the orchestration layer and minimizing activation layer operations for service design, (though leaving in place the orchestration needed to interact with the network).

Moving the design into the LSO layer also makes supporting inter-MSO services simpler. The placement decisions needed to design a service are also needed when determining whether a new service can be supported. Think back to the port supply/demand example. That's the sort of re-usable process block that you want to code once and re-use often.

Which leads us to...

3.2. Unified activation layers or mulitple verticals

The choice here is related closely to the orchestration decision. Experience in this area has shown us that while unified activation layers sound great on paper, but they're very hard to build out in practice with long timelines. (And long timelines inevitably mean changing requirements, causing API changes which make the time to deliver a solution even longer).

That's not to say that each technology should have its own vertical. A typical activation layer in a current network would have 3 broad groupings:

- Fixed network activation for L2 and L3 devices. – this might be a "legacy" activation solution
- An SDN-C managing the VNFs and data center activation.
- An SD-WAN controller managing SD-WAN service configuration

Here the orchestration layer will be responsible for mastering the end-to-end service implementation according to the design in the service and network inventory, with each activation component dealing with the specifics of each technology domain. This lowest level of orchestration then effectively becomes the way into the activation layer.

3.3. Common models

It's far easier than it appears to end up with multiple databases in fulfilment, activation and assurance systems.

As a result it's important to draw sensible boundaries between the systems and get information via APIs when needed rather than duplicate locally. It's also relatively simple to spend your entire time merely synchronizing the data stored in overlapping locations rather than building out new solutions. As a rule of thumb you need:

- **A service and network inventory that models the network services that you have created and are managing and any service artefacts that are needed**. UNI port configuration is a good example of this, since it dictates the way that services can be multiplexed together at a single physical port.
- **A network inventory within the controller activation layer** – this will be segmented by the activation domain, and should be queryable by the upper orchestration layers during design-time operations. It should represent the "as-is" network and may include service models for use in discovery and background sync into the service and network inventory.
- **An assurance data warehouse that collects and collates the OAM data that the network generates**. Good systems will pre-process the inbound information and link it into the correct service and KPI, and this database will merge with the service and network inventory so that correlated operational events can be linked to specific service instances. (However, care should be taken to avoid too much duplication).

3.4. Catalog your processes

Definitely another important aspect, **the more you can make your process elements look like small functions in a catalog, the easier it will be to add new technologies or service variations**. And this is where having a common orchestration tool starts to pay even bigger dividends.

Consider adding support for a new VNF to an existing Ethernet service – for a catalog-based environment, once the VNF has been on-boarded into the catalog, you need to develop the new processes needed to support the activation of that component and define the rules for how it can be built into a service chain. Then you can build out a new service chain with the VNF included. Then it's ready to go, without having to re-build whole templates of service design.

4. The digital future

A recent whitepaper by Analysis Mason [0] looked at the key characteristics for a digitally-empowered enterprise in the future and found the following five big trends in the solutions that will enable those enterprises:

- Big Data Analytics, providing service-level information
- Robust inter-company interfaces for interconnection of ecosystem partners and services
- AI-driven software for design of customer networks and services
- Methodologies for structured specification of network and IT services
- Virtualized network slicing for highly customized, dedicated solutions.

There are two interesting sets of trends here:

The first is that many aspects of the solutions are based on making the process of delivering solutions in a simpler way and at a lower cost, which indicates that MSOs will still be looking to drive cost out of supporting an enterprise.

The other is that both the network and the infrastructure it's built upon will be more tailored to the needs of the enterprise, giving specific and managed behavior. What was also interesting in the report is that opinions strongly differ between whether a full managed service is desirable for the enterprise or whether a communication service provider (or MSO) wants to offer the customer full access via a portal.

This is a difference of opinion that has been repeated many times in the last few years. One MSO spent a number of months helping their customers get the best out of variable bandwidth Ethernet connections and avoiding bill shock. Another reckoned that variable bandwidth was just a differentiation exercise that was rarely used in practice.

Where there is more common ground is the need for an MSO to move up the value chain from simple connectivity provider towards all-in-one service and IT manager for a customer. The main demand from this tends to come from the smaller customer, for whom removing the need for an IT department to manage a bespoke service mix takes an overhead away from the business. This leaves an emerging business able to concentrate on building their offering and brand, rather than worrying about whether to use Microsoft Azure or Amazon Web Services to host their IT infrastructure.

While at the moment larger enterprises are still happy to manage over the top, a strong digital focused offer emerging from the management of smaller clients could persuade many to switch across over time.

Take a moment to look back at the vision laid out in Figure 2. This is the sort of offering that all MSOs are striving for and the good news is that many of them are pursuing this in bodies such as the MEF. This shared experience and ability to draw on some of the leading minds in the industry, coupled to the move towards building working prototypes that can be used and refined in practice means that there is a strong roadmap already being laid out. There is also a strong correlation with the work that is going on in the Open Network Automation Platform (www.onap.org), which is a linux-foundation hosted collaboration program born out of AT&T's ECOMP solution. MEF and ONAP have recently agreed to strong joint working practices and architectural alignment. As the code base evolves, ONAP is expected to provide solutions to the problems laid out in this whitepaper.

And new initiatives continue to emerge whose focus is on the operational practicalities. The first phase of a programme to explore the architecture needed to support Zero-Touch Orchestration of Network Service Management, which has developed from a workshop that Deutsche Telekom facilitated, has recently concluded and by the time of publication will have found a new home. This work looks at how to manage the lifecycle of a service automatically and starts to plot the balance between intent and orchestration.

Conclusion

What it means to activate a network has changed greatly over the last ten years. What started out as a device-direct mechanism that sent command line instructions to make changes to the config file of a device has evolved into a service level command set. What used to be a device-by-device operation has now become a service by service method. What used to stop at the point that a connection function had been activated now needs to extend into enabling the support operations for the service.

Likewise, the footprint of Ethernet has evolved. Ethernet used to just be a LAN frame type that could be transported over an MPLS L2 VPN. It has become the uniform means of building a connectivity service in the MSO and CSP space up to the edge of the network as a result of the hard work of the MEF. With the new focus on Orchestration in MEF, with certification of service orchestration planned and a direction that looks towards application management, Ethernet is positioned as the best technology to support the drive towards digital offerings.

Ethernet is, however, only a forwarding technology. For an MSO to evolve into a true Digital Service Provider, a full support infrastructure must be wrapped around the strong base that Ethernet provides. Services must consider from the bottom up how they are to be managed once they have been activated. Initial deployment cannot simply halt once the commands for the connection have been sent, activation must extend to include the measurement and assurance structure and then tie this back into the operational objectives of the service as a whole.

Many industry bodies are looking at parts of this solution space but only MEF is taking the holistic view of services needed to deliver this end-state. In the last 6 months ONAP has emerged as the pre-eminent place for discussing how the services that will be delivered over the top of this common foundation should be delivered. As MEF and ONAP strengthen their direction together, the joint outcome will provide a strong blueprint for the digital future.

Abbreviations

ISBE	International Society of Broadband Experts
SCTE	Society of Cable Telecommunications Engineers
API	Application Programming Interface
CIR	Committed Information Rate
CLI	Command Line Interface
CPE	Customer Premises Equipment
CSP	Communication Service Provider
DC	Data Center
DOCSIS	Data Over Cable Service Interface Specification (www.cablelabs.com)
ECOMP	Enhanced Control Orchestration Management & Policy
EMS	Element Management System
EVC	Ethernet Virtual Connection
FM	Fault Management
IT	Information Technology
KPI	Key Performance Indicator
L2VPN	Layer 2 Virtual Private Network
LSO	Lifecycle Service Orchestration
MEF	MEF (www.mef.net)
MIB	Management Information Base
MPLS	Multi-Protocol Label System
MSO	Multiple System Operators
MTOSI	Multi-Technology Operation System Interface
NETCONF	Network Configuration Protocol
NMS	Network Management System
NOC	Network Operation Center
OAM	Operations Administration and Maintenance
ONAP	Open Network Automation Platform
PM	Performance Management
RESTful	Representational State Transfer (web services)
RMI	Remote Method Invocation
SDN	Software Defined Network
SDN-C	Software Defined Network Controller
SD-WAN	Software Defined Wide Area Network
SLA	Service Level Agreement
SOAP	Simple Object Access Protocol
SOC	Service Operation Center
TL1	Transaction Language 1
TMF	Tele Management Forum
TOSCA	Topology and Orchestration Specification for Cloud Applications
UNI	User to Network Interface
VNF	Virtual Network Function
VoIP	Voice over Internet Protocol
XML	eXtensible Markup Language
YANG	[not an acronym]

Bibliography & References

Vertical Systems Group “Emerging Third Network Services Enabled By LSO, SDN, NFV & CE 2.0”, State of the Industry Research Report, Jan 2017.
https://www.tmforum.org/open-apis/
https://www.mef.net/Assets/Documents/Carrier_Ethernet_2.0_Certification_Blueprint_-_VERSION_1_1.pdf
https://www.youtube.com/watch?v=cnllFPRyBFs
https://wiki.opendaylight.org/view/OpenDaylight_Controller:Main
Analysis Mason “B2B User Experience Digitalisation” January 2017.
MEF Forum www.mef.net