inmanta

CSPs worldwide: automate end-to-end to lead the game

How end-to-end service orchestration enables zero-touch automation of network and service management

CSPs worldwide: automate end-to-end to lead the game

How end-to-end service orchestration enables zero-touch automation of network and service management

Today's end customers expect the instant delivery of value-added telecom services tailored to their specific needs. CSPs face difficulties with increasing the agility and productivity to successfully respond to changing expectations. Their current network and service management practices fall short in dealing with telecom networks that have become more heterogeneous and software-oriented. The amount of manual work gets out of proportion, and so do investments and delivery delays. End-to-end orchestration offers the opportunity to add the necessary agility to systematically align network infrastructure with business strategies. A fully automated process enables CSPs to reach the efficiency levels required to retain and grow their telecom business. True end-to-end orchestration results in zero-touch network and service automation over the entire service lifecycle. It enables CSPs to deliver services more flexibly and to release new features within days or weeks rather than months or years. That is a key asset for CSPs to strengthen their competitive edge and increase customer satisfaction.

1. Challenges in serving telecom end customers

CSPs compete fiercely, putting turnover and profit numbers under pressure. End customers today are more demanding and assertive. They expect an instant and seamless delivery of their specific services mix.

In the past decade a new breed of OTT players – Amazon, Google (Youtube), Netflix, Skype etc. – have successfully entered the market. End customers eagerly pay for their flexible offering of value-added services and content, while the actual communication bandwidth is seen as a commodity. More specifically, the end customers look out for the most favorably priced telecom packages, preferably covering bandwidth only. Then they go shopping the OTT players, where they are serviced instantly with on-demand series, voice and video calls, computing power, etc.

Today's industry setting offers interesting opportunities for CSPs. They can turn the challenges in their favor if they get their act together. The real challenges are flexibility, scale, and productivity.

2. CSPs face complexity that goes through the roof

CSPs find it difficult to take on a faster and more scalable approach to deliver and manage various new telecom services. One reason is that telecom networks exhibit increasing complexity in terms of distribution, scale, and heterogeneity. Currently, CSPs are operating in a landscape of highly distributed and heterogeneous communication infrastructures, including core network, cloud and edge physical as well as virtual network appliances. That makes it difficult to efficiently manage networks and services, and to ensure seamless operation.

The complexity also rises with CSPs providing many use cases, each with a dedicated set of service level requirements. The provisioning and servicing of new added-value telecom services require many changes and high flexibility.

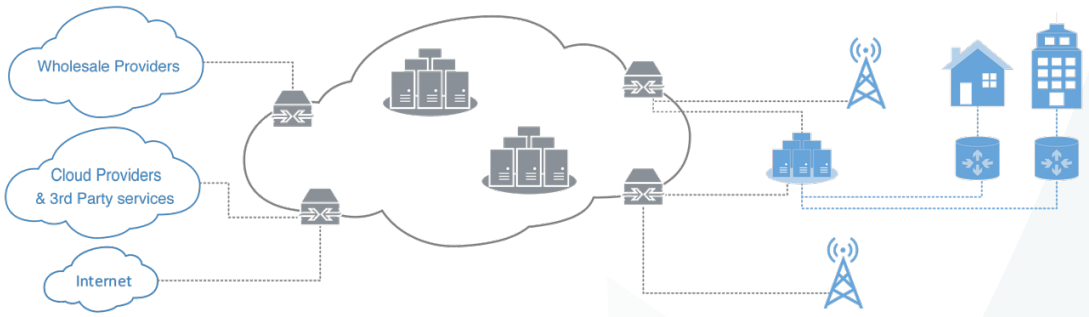

Figure 1 : CSPs operate in highly distributed and heterogeneous communication infrastructures.

In the current market setting, networks are managed with a strong focus on hardware. This approach is no longer in line with reality. Network hardware has become a commodity and network components are increasingly defined in software (e.g. SDN and NFV).

Network functions virtualization (NFV) allows traditional hardware appliances to be replaced with software. A virtualized network function consists of one or more virtual machines running on top of commodity hardware, typically in the form of cloud infrastructure. Multiple functions can be connected or chained together to create (new) communication services. Combined with software-defined networking (SDN), NFV is a key building block for setting

The ongoing digital transformation makes it even more complex for CSPs to respond to end customers' requirements. CSPs are trying to reduce delivery times, lower costs, and retain availability and reliability.

> *"When CSPs hang on to their traditional operations models, the complexity will be overwhelming and the benefits will fail to materialize."*

Overall, CSPs are facing higher complexity with respect to network scale and heterogeneity as well as the shift from hardware to software. It will be a challenge for CSPs to adapt to all the changes and market evolutions that are taking place. That is not all. CSPs are expected to implement 5G wireless communication by 2020. That means that complexity really goes through the roof.

3. End-to-end automation will save CSPs

Automation is the way forward. Truly end-to-end automation of network and service management makes the difference, across domains and layers and during the complete lifecycle. Automating bits and pieces is insufficient to eliminate human interaction as well as human errors that potentially cause network downtime. The perception that extensive manual work is required to manage and operate telecom networks no longer remains standing.

It is the complexity and the manual work that withholds CSPs to scale up and flexibly respond to changing customer needs. Typically, it takes months or even years to plan, implement and roll out strategic network-wide changes and new services. In addition the allocation of many full-time equivalents is needed to deliver, operate and maintain these services. 5G and other complex use cases simply will not be manageable without automation. Also more conventional use cases benefit considerably from end-to-end automation.

The limited agility of CSPs hampers their ability to execute strategic marketing decisions and extend market share. Slow technical network implementations and heavy maintenance load increase expenditure and reduce time to market. That means that incoming money streams are delayed by further stretching the time to bill. More importantly, churn rates go up by drawing end consumers to competitive CSPs or OTTs that do deliver innovative solutions & services.

Several years ago, a smaller European CSP decided to offer unlimited internet and communication by default. The subscriptions offered by the CSP only vary in bandwidth. With that offering, its customers are no longer bounded by periodic data limits. The ability to define and apply a different billing model enabled the CSP to gain a competitive edge in response to specific market evolutions. This CSP's adapted telecom offering was a hit and the commercial success has gradually increased the CSP's profit margin. By outperforming the profit margin of larger CSPs, the smaller player now matches larger players in terms of net profit.

This demonstrates how the agility to quickly introduce new business models and commercial services has a major impact on CSP's top-line revenues and profitability.

4. The power of digitization

Managing software is different from managing hardware. The ongoing transition from hardware to software in telecom demands a different operations model. A software-defined approach offers CSPs the potential to continuously upgrade, dynamically scale, and to automate every aspect of the network infrastructure. But it also imposes an enormous complexity to actually reap those benefits.

State-of-the-art software development and operations models, such as DevOps and site reliability engineering (SRE), are the key to success. These techniques are highly evolved, combining decades of experience, best practices, and trial and error. Through SDN, NFV and cloud, these models become available to the networking world.

Ideally, the automation of network and service management should be executed by software engineers as well as network operators. The software engineers master the software development practices needed to **automate in a structured and maintainable way**. The network operators have the domain-specific expertise, e.g. to manage large-scale WAN and voice networks.

Google is a perfect illustration of what automation driven by software engineering can bring. According to Ben Treynor, VP Engineering for Google: "The solution is asking a software engineer to design an operations function in software. The use of the software is a way to solve problems that had historically been solved by hand. That offers the ability to substitute automation for human labor." – Quote extracted from "What is site reliability engineering?", an "In Conversation" article published by Google.

5. End-to-end service orchestration

In the previous sections, we have touched upon 'end-to-end automation' and 'the power of digitization'. The combination of both is essential in reliably controlling the entire telecom network without massive manual work. Economically managing ever-more complex networks forces CSPs to go digital and embrace software engineering. The latter is a means to shield much of the complexity and to rapidly 'model' telecom services, enabling the fully automated delivery of the services.

A so-called **end-to-end service orchestrator** achieves this. It enables CSPs to automate service delivery and management by integrating all different elements of the telecom architecture, both virtual and physical. When evaluating service orchestrators, CSPs should take into account a number of key requirements, which are addressed separately below.

1. Openness to enable interoperability

To guarantee maximum flexibility, the orchestrator should allow CSPs to easily **select and integrate best-of-breed components from any vendor**. To avoid lock-in, interoperability must not be limited to a specific ecosystem. Overall, the ease of integration interrelates with both openness and interoperability.

A related requirement is that an orchestrator offers the technical capability to easily plug into any brownfield network environment, including OSS/BSS, CMDB, analytics and assurance. This means that the orchestrator should easily connect and interact with third-party southbound as well as northbound systems and interfaces. An orchestrator that flexibly integrates with the complete network is able to automate the entire service delivery.

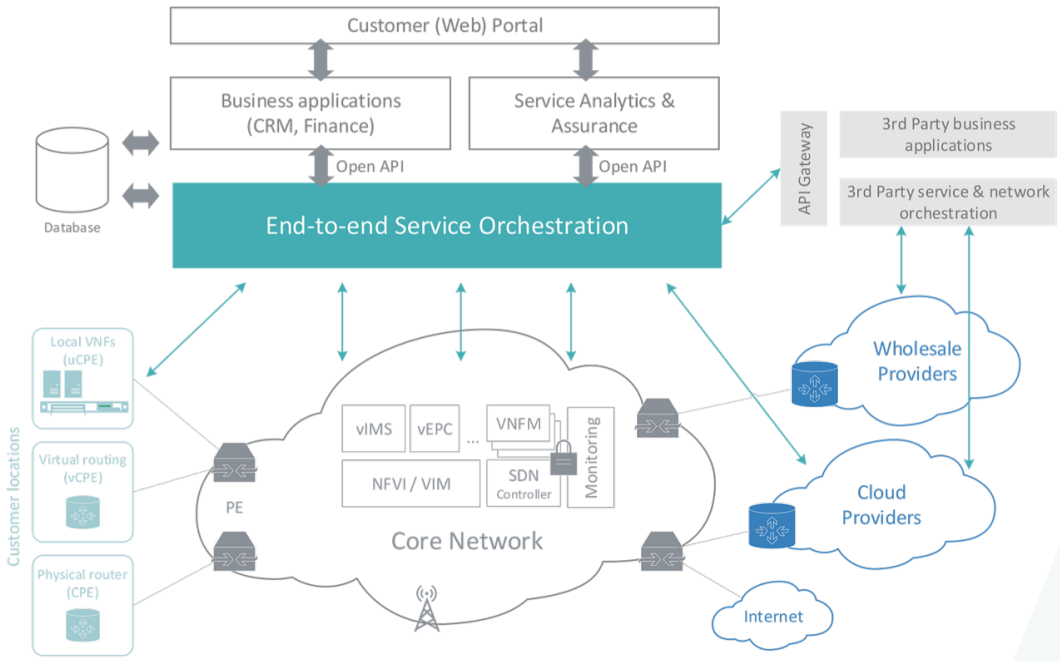

Figure 2 : An orchestrator that flexibly integrates with the complete network is able to automate the entire service delivery.

Buying from an independent orchestrator provider without its own VNF, VIM or accompanied equipment is a preference. Independent orchestrator providers put extensibility and multi-vendor compatibility first. They support the rapid development of new integrations with different vendor offerings. This is in contrast with vendors that sell a complete package (including orchestrator) but have a strong incentive to lock its customers into its own ecosystem.

2. Intent-based programmability

Programmability is key to agility, quality and rapid delivery of new services. It eliminates a tremendous operational workload. That responds to the CSPs' need to flexibly and rapidly define and deliver customized services, even for every individual end customer.

A software-defined infrastructure also enables efficient knowledge management. The model that describes the infrastructure results in documentation guaranteed to be up-to-date. As such, an orchestrator must incorporate a modeling language to describe the network in a way that is readable for people and executable by machines. Service engineers and operators benefit from efficient knowledge management as well as fully automated execution by the orchestrator.

Intent-based programmability goes one step further. It simplifies the programming of the network thanks to the use of a more powerful modeling language. It is possible to define the objective (the 'what') without having to program the detailed workflow or recipe including all individual actions (the 'how'). Orchestrators supporting intent-based programmability incorporate a higher degree of intelligence that enables them to automatically derive an appropriate chain of actions. The way a specific intent (the desired state and behavior) is realized, depends on the existing network configuration and the applicable SLAs and policies. Apart from the huge OPEX savings when delivering and maintaining services at a large scale, the approach offers increased long-term stability. The orchestrator is able to periodically verify whether all objectives are still met and take corrective actions if necessary. This self-healing requires no additional modeling effort, but is derived by the same mechanism as the initial setup.

3. DevOps makes delivery process straightforward and reliable

The modeling language used by an orchestrator should come with the necessary tools to support team collaboration and automate the deployment pipeline. This enables engineers and operators to develop, test, and operate new services in a continuous delivery mode, fully in line with the DevOps principles.

It should also be possible to plug the orchestrator into a CI/CD pipeline. That means that any changes to the model and the network can be validated automatically before they are actually deployed in production. The more powerful the modeling language, the sooner in the CI/CD pipeline that errors can be detected, even at design time.

4. Open source versus proprietary software

Telecom networks typically consist of hardware and software supplied in many versions by many vendors. The use of proprietary software means that all support and interoperability needs to be delivered and maintained by these vendors. Network configurations that do not map to a specific vendor ecosystem or a pre-configured setup, are likely not (entirely) supported.
Open-source software typically uses open standards that are accessible to everyone. The use of such software moves away from incompatible formats that may exist in proprietary software. The advantage of open source is its worldwide community that can contribute. Anyone is allowed to study, change, customize, and use the software without limitations.

The community is an asset, but it will not provide the support you need to run your telecom network operations.

Be careful with vendors offering solutions that are based on open-source software, but are not compatible with it (i.e. a fork). In many cases the offered solution is heavily customized, creating its own lock-in. Better is to opt for a vendor that offers an open-source product, including high-end support and perhaps some proprietary extensions and integrations, compatible with the open core.

6. How to concretely give CSPs end-to-end control

We have touched upon a number of important aspects related to gaining maximum network control through an orchestrator. But how do CSPs concretely gain more efficiency and flexibility in the context of complex network configurations and competitive environments?

By discussing real-life use cases, and evaluating the role and impact of the orchestrator, you get a good idea of how orchestrators enable CSPs to take end-to-end network control. Automation is critical in this regard, across domains and layers throughout the entire service lifetime.

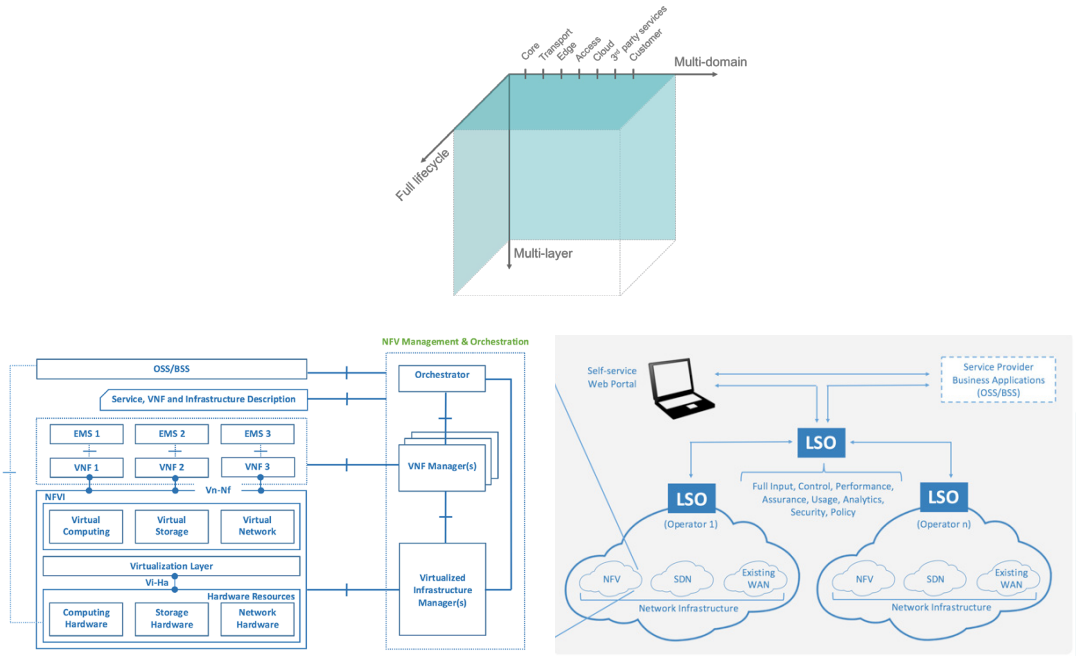

Figure 3 : An end-to-end orchestrator deals with all components of today's heterogeneous network architectures that span multiple domains. Based on ETSI NFV and MEF LSO.

1. Automate across all domains

Automating services end-to-end is only feasible with an orchestrator that is able to deal with all components of today's heterogeneous network architectures that span multiple domains. Examples of those domains are the core network, datacenter, access network, transport, edge, cloud, partner providers, etc.

In general, the use of a flexible and open orchestrator ensures straightforward integration with third-party systems and components. These may include virtual network functions (VNFs), SDN controllers, EMS, network management systems (NMS), virtual and physical network equipment, datacenter resources, cloud platforms, and even other orchestrators. Integration can occur through vendor-specific command-line interfaces (CLIs), APIs or standards such as NETCONF/YANG. Instead of interacting manually on individual network resources, a suitable orchestrator is the glue that allows the network to be managed as a network as a service (NaaS).

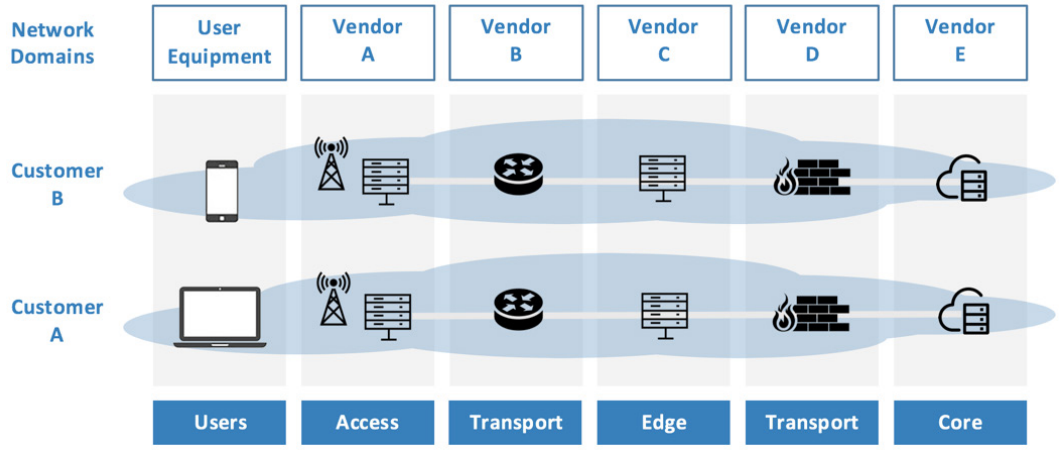

Figure 4 : An end-to-end orchestrator automatically ties together the different domains to flexibly and efficiently deliver the sliced network services to customers.

A typical example to demonstrate the need for multi-domain orchestration is network slicing in 5G networks, as shown on the graphic below. A network slice represents a cross-domain service dedicated to an individual business customer or market. As CSPs are likely to offer various types of network slices, they require true end-to-end automation to avoid tedious static and manual configuration for every single slice.

The end-to-end orchestrator automatically ties together the different domains so that CSPs can efficiently define, test and roll out the sliced network services commercially. Each domain can be an autonomous entity, managed by another orchestrator.

2. Incorporate multiple levels of abstraction

In any network configuration, there are many (implicit and explicit) abstraction layers between the OSS/BSS and the low-level network configuration details. This is nicely illustrated by the example of carrier ethernet services (E-Line, E-LAN). On the highest abstraction level an E-Line consists of a connection between point A and point B, with

an SLA associated to it. On the lower abstraction levels you find details related to IP addresses and network configuration specifications.

> The extreme agility and rapid delivery of software applications by software companies, e.g. the OTT players, is based on one thing: the proper use of abstractions. The ability to reuse, maintain, test, change, collaborate, etc. depends on modularity, which means making abstraction of how something works in favor of what it does. Applied to orchestration, modularity encapsulates the complexity to deploy and manage a specific component or service, or to use a third-party API. Each module exposes an interface toward other modules. CSPs must master this delicate art in order to achieve the speed and agility of the Amazons of this world.

The application, illustrated below, concerns the end-to-end orchestration of carrier ethernet services. Multiple layers of abstraction allow new services to be composed flexibly across vendors, technologies, and even providers. CSPs benefit from the fast and automated delivery of value-added services, including firewall (AV, web filtering), routing (e.g. BGP), SD-WAN, and VPN.

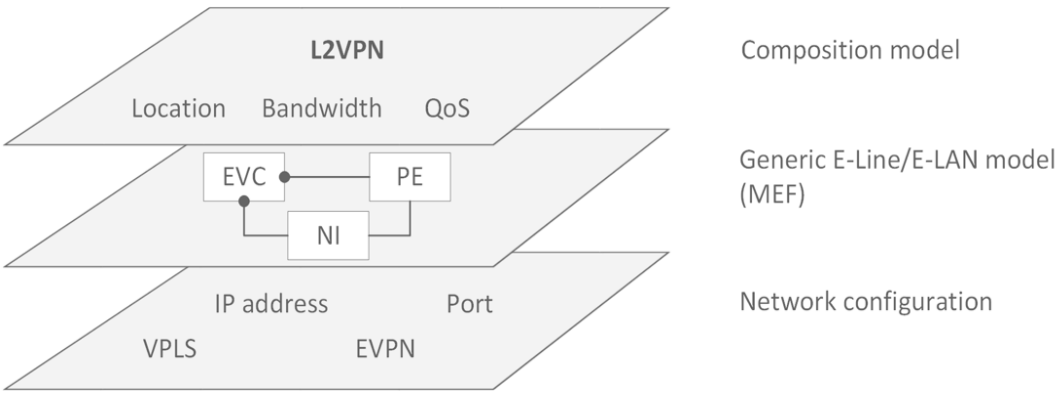

Figure 5 : Multiple layers of abstraction enable to flexibly compose new services across vendors, technologies, and even providers.

Typically, the abstraction gap is bridged by executing a lot of manual, ad-hoc scripts, using a bottom-up approach. In contrast, an end-to-end orchestrator coordinates network operation across domains and executes actions within each domain top-down. The levels of abstraction range from the high-level model of the full telecom service up to the low-level configuration details.

Product managers think on a high abstraction level when defining new products and services. They focus on strategic and commercial aspects and want to move fast with high-level building blocks that they can combine to create new products. Other stakeholders, such as operators and technicians, are involved in lower abstraction levels, delivering the reusable building blocks.

Abstraction levels play an essential role in architectural design of applications and networks (see graphic below). In architectural design, you gradually refine toward lower abstraction levels in order to unambiguously define the telecom service. The abstraction layers and views allow different stakeholders to discuss and work on their own telecom

service related levels, while abstracting away irrelevant details. Between the highest level of abstraction and the actual implementation, many iterations take place to gather requirements and fill in the details.

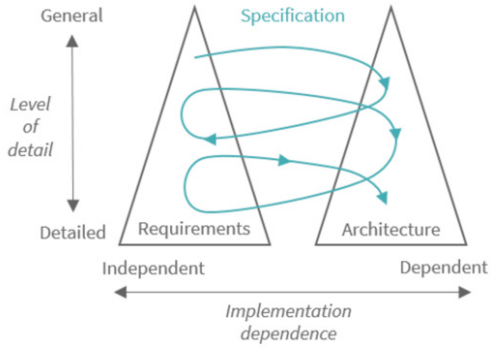

Figure 6 : The essential role of abstraction levels in architectural design of applications and networks – based on Nuseibeh's Twin Peaks model ("Weaving together requirements and architectures").

Similarly, an intent-based modeling language allows you to gradually define more details of the end-to-end service, until you get to the level where you can reuse existing building blocks. All the information structured in the abstraction levels is kept as part of the model, which offers overview and clarity.

The structured way to make abstraction of the underlying network complexity and heterogeneity, offers additional advantages. An end-to-end orchestrator enables CSPs to easily swap different implementations, and thus vendors, maximize reuse, improve manageability, and roll out changes in a reliable and structured way. Furthermore, by shielding most of the underpinning complexity, an end-to-end orchestrator heavily simplifies OSS development. Finally, an open-source orchestrator offers a CSP the freedom to develop custom implementations in support of its product offering.

The use of an orchestrator facilitates the tight point-to-point coupling between domains all the way down to individual network resources and IT systems (see chart below). The structured approach and the intent-based programmability supported by the orchestrator allow CSPs to define a specific higher-level intent (the desired state and behavior). The appropriate chain of actions is automatically derived and executed, depending on the existing network configuration and the applicable SLAs and policies.

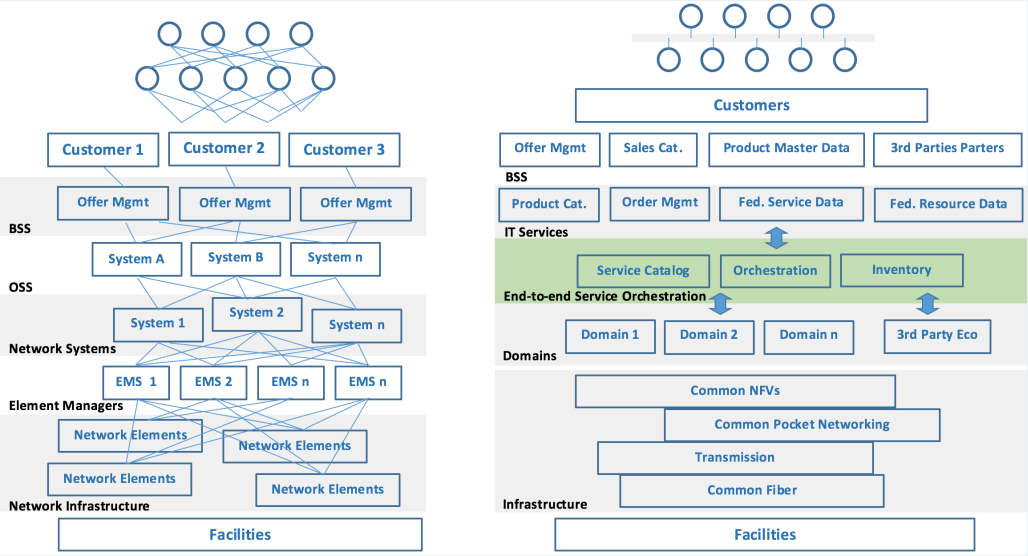

Figure :Currently the OSS/BSS of CSPs captures all detailed information up to the level of the network, which creates complex systems with scattered interrelations. By making abstraction of the complexity, an end-to-end orchestrator simplifies OSS/BSS and structurally manages all underlying network operations, even across multiple domains.

The use of an orchestrator allows for flexible and risk-free testing of any potential new or modified end-to-end services, both virtually and physically. That enables product managers to efficiently evaluate and tune services defined using intent-based programmability. The impact of this approach from a strategic viewpoint is huge.

3. Managing the full service lifecycle

Service orchestration is a continuous process, as new customers sign up and the services delivered to customers change over time. Not to mention any network extensions or changes that need to be controlled as well. Managing the full service lifecycle not only covers provisioning and deployment, but also service design, on-boarding, testing, monitoring, scaling, upgrading, and decommissioning.

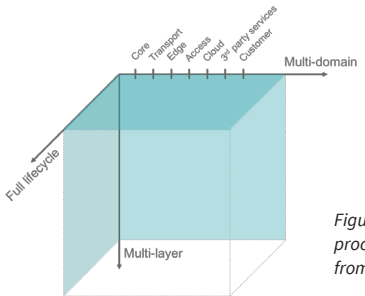

Figure 7 : Service orchestration is a continuous process that manages the full service lifecycle, from service design to decommissioning.

Traditionally managing and executing changes over multiple domains and across layers takes a lot of time and money to test and roll out, even for minimal changes. An end-to-end orchestrator addresses this problem in an automated and structured way.

More specifically, the orchestrator dynamically manages the state of each individual resource or entity that is defined in the service model. This way, the orchestrator succeeds in consistently updating many such states correctly and efficiently. Based on the defined intents, e.g. creating a new network slice or updating the bandwidth of the L2VPN service, the orchestrator ensures the transition across states of the independent components to ensure a seamless evolution of the end-to-end service.

Guidelines for establishing end-to-end service orchestration

Go truly intent-based

Currently, too much time and resources are spent on writing ad-hoc workflows and scripts with limited reuse and that are costly to maintain. These workflows and scripts can be derived automatically, creating more time to focus on rolling out (and automating) new services and delivering value.

Select an orchestrator supporting intent-based programmability that allows for defining a specific higher-level intent (the desired state and behavior). The appropriate chain of actions is automatically derived and executed, depending on the existing network configuration and the applicable SLAs and policies.

Do not start from scratch

Engineers and operators should have access to true infrastructure as code, enabling them to represent any concept or device. Use an orchestration framework that already incorporates all relevant software engineering methods, maximizes reuse and improves manageability, collaboration and knowledge sharing.

Reinventing the wheel is expensive. It is much easier to start from a generic orchestration framework that is easily extensible and customizable to your needs. It provides (and enforces) a uniform approach for automation, but is not limited to individual use cases.

Start small but end to end

The smart strategy is to start small and build up experience. However, the only way to have impact with orchestration is to work end to end, and thus by fully automating an end-to-end service. Therefore, start with a less-critical service with a limited number of users or a new offering, and build up experience.
Primarily focus on higher abstraction levels and automate truly end to end. At later stages, refine the automation by adding lower-level abstractions or plugging in third-party software.

After automating, marketing and servicing a first entire telecommunication service, CSPs can evaluate the impact on their network and service management. As soon as they see the proof of increased efficiency and agility, they will initiate the automation of another service.
Thanks to the reuse of integrations and even entire orchestration submodules, each subsequent project will have a lower cost relative to the previous ones. Additionally, reuse of orchestration submodules will naturally lead to a more uniform management across services and administrative domains, that would otherwise be unattainable. This will, in turn, lead to increased quality and reduced maintenance cost.

Avoid vendor lock-in

The digital transformation is an opportunity to reclaim control. From an economical perspective, CSPs remain too dependent on specific vendors from which they acquire specialized appliances over a span of five to ten years. The digitalization enables CSPs to become more flexible and to avoid vendor lock-in. The use of a truly open orchestration solution is advisable to enable this flexibility. It is important that the the orchestrator can interoperate with any software and connect with any type of network device.

In a digitalized world, it is also important to source the right knowledge and to select partners specialized in software engineering, from which CSPs can learn how to handle the challenges ahead and break lock-in through increased agility and control.

Act now – Do not wait for the golden standard

The telecom market has been focusing on standardization for too long. Today, standardization has the tendency to take ages to get the tiny details straight. This is justified for the design of hardware and protocols, otherwise it would not work. In software, however, the key is to agree on external interfaces or APIs. The internal design and implementation is a black box, thus offers a lot of opportunities for differentiation and innovation.

As the golden standard does not exist, it is advisable to follow existing standards that are useful and to develop independently whatever is needed in addition to that. When over time standard interfaces appear, you can easily adopt them by adapting and/or extending your external interfaces. But in the meantime, you are able to deliver real solutions to your customers, using the newest technologies available.

Waiting means losing as your backlog only increases. And a ‘big bang’ switch to a new architecture and approach is a large, all-or-nothing risk. While applying a software-driven approach actually calls for an iterative approach, where you gradually start automating more and more. Small steps with small risks, resulting in early feedback to learn and improve (i.e. fail-fast principle).

Network and service management agility in numbers

CSPs typically manage and operate heterogeneous telecom networks. That is in contrast with Amazon, Google, or Netflix. These giant players have the means to build their own compact homogeneous networks that are agile to manage and maintain. The billions of end customers using their popular services are accustomed to their speedy release cycles. That puts huge pressure on the many CSPs that operate in a single country or continent. They feel the need to continuously innovate their operations to maintain or increase their competitive edge.

Some mind-blowing numbers…

- Changes to production every 11.6 seconds (Amazon, May 2011)
- 50M+ deployments annually to development, testing and production hosts (Amazon, November 2014)
- On average 3 new features per day released on AWS (Amazon, December 2016)
- +100 new products per year (Google)
- 4,000 deployments per day (Netflix, March 2016)

inmanta

Inmanta enables telecom operators to deliver their services 10 times faster and with more flexibility. The award-winning service orchestrator of Inmanta automates and streamlines the entire service lifecycle. Instead of months and years to deliver services and release new features, Inmanta reduces the process to a matter of days and minutes.

Inmanta is a spin-off of KU Leuven University, and its automation and orchestration technology is based on 10 years of research. Inmanta has been awarded the international Call for Innovation on SDN/NFV 2.0, organized by Swisscom, Telia and Proximus.

© Inmanta. Reproduction and distribution of the white paper without written permission of Inmanta is prohibited.