# AT&T Vision Alignment Challenge Technology Survey

## AT&T Domain 2.0 Vision White Paper

*November 13, 2013*

**Copyright Notice**
© 2013-2014 AT&T Intellectual Property. All rights reserved.

# Contents

Introduction ..... 2
Domain 2.0 Summary ..... 3
Cloud Networking Architecture ..... 5
Network Function Virtualization Infrastructure ..... 7
Virtualized Network Functions ..... 8
Software Defined Networking ..... 10
SDN Summary ..... 10
Definition and Key Concepts ..... 11
SDN as Disruptive Technology ..... 13
AT&T & SDN ..... 13
Futures ..... 16
Orchestration Control and Policy Management ..... 16
Operations Transformation ..... 17
Key Operational Shifts ..... 18
The Emergent Operator Software Ecosystem ..... 19
Transition from system standards to component standards ..... 19
Transition to vertical integration and in-sourcing of core business competencies ..... 19
Transition to Information models and Software frameworks overshadowing boxes and protocols ..... 20
The opportunity to develop network software using open source ..... 21
Conclusion ..... 22

# Introduction[1]

AT&T regularly reviews and revises business and technology approaches. Significant emerging technologies and business drivers have encouraged the company to reassess desired future network technology, operations methods, and sourcing approaches in an effort called Domain 2.0. The work considered lessons learned from the current domain program as well as business and technical changes ongoing in our marketplace – especially those recently seen in data centers. The result includes a

[1] This white paper includes forward-looking statements. Because such statements deal with future events, they are subject to various risks and uncertainties and actual plans and actions taken by AT&T could differ materially from the Company's current expectations. A discussion of factors that may affect future results is contained in AT&T's filings with the Securities and Exchange Commission. AT&T disclaims any obligation to update and revise statements contained in this white paper.

revised architecture that borrows from cloud technologies and suggests adding domains to our program that will allow for rapid innovation, new business models, greater customer value, greater opportunities for third parties to participate in the customer value chain and an increased choice of suppliers.

While AT&T has managed risk and effectively served customer needs until now, we foresee changes in customer needs, changes in technologies, and changes in best practices for operating networks in the near future. This dynamic influenced the company to project what our business will look like in 2020 and beyond, and make adjustments to our architecture roadmap. These adjustments include the provision of new capabilities to meet customer needs, new technologies and architecture to increase operational efficiency, and establishment of new supplier domains that can nurture new types of technology and suppliers.

The rapid growth in IP endpoints requires greater scale and efficiency in handling the number and diversity of devices than we are getting from traditional network solutions. Movement of data to the cloud for use on any device and increasing use of virtual machine models redefines the endpoints and timeframes for provisioning network connections. These drivers require the business of networking to significantly improve the capital efficiency and human operations per unit of business. In conjunction, there are technologies, architecture, and operational approaches informing our choices for future network capabilities by a similar transformation going on in data centers and in cloud computing. In summary, the major aspects of AT&T's Domain 2.0 architecture are:

**Open** – Provide APIs, enable better participation of third parties, and improve visibility. Increase the number of suppliers and partners AT&T can do business with.

**Simplify** – Weed out complexity from services and operations; support more nimble business models.

**Scale** – Meet evolving customer requirements including traffic growth, diversity of traffic types, and diversity of performance and reliability expectations. Improve business efficiency in capital and operations.

## Domain 2.0 Summary

Domain 2.0 is a transformative initiative, both internal and external, to enable AT&T network services and infrastructure to be used, provisioned, and orchestrated as is typical of cloud services in data centers. It is characterized by a rich set of APIs that manage, manipulate, and consume services on-demand and in near real time. Moreover, these network services are to be instantiated, to the extent feasible, on common infrastructure. In a nutshell, Domain 2.0 seeks to transform AT&T's networking businesses from their current state to a future state where they are provided in a manner very similar to cloud computing services, and to transform our infrastructure from the current state to a future state where common infrastructure is purchased and provisioned in a manner similar to the PODs used to support cloud data center services.

Migrating AT&T businesses to a multi-service, multi-tenant platform implies replacing or augmenting existing network elements – which today are typically integrated to perform a single function. The

replacement technology consists of a substrate of networking capability, often called Network Function Virtualization Infrastructure (NFVI) or simply infrastructure that is capable of being directed with software and Software Defined Networking (SDN) protocols to perform a broad variety of network functions and services.

This infrastructure is expected to be comprised of several types of substrate. The most typical type of substrate being servers that support NFV, followed by packet forwarding capabilities based on merchant silicon, which we often call white boxes[2]. However it's envisioned that other specialized network technologies are also brought to bear when general purpose processors or merchant silicon are not appropriate.

AT&T services will increasingly become cloud-centric workloads. Starting in data centers (DC) and at the network edges – networking services, capabilities, and business policies will be instantiated as needed over the aforementioned common infrastructure. This will be embodied by orchestrating software instances that can be composed to perform similar tasks at various scale and reliability using techniques typical of cloud software architecture.

As an example, an edge router might be purchased as a monolithic device today – where the hardware, feature functions, and a specific applicable scale of use are pre-integrated into a single device. Often a variety of device sizes need to be purchased in order to support variances in workload from one location to another. In Domain 2.0, such a router is composed of NFV software modules, merchant silicon, and associated controllers. The software is written so that increasing workload consumes incremental resources from the common pool, and moreover so that it's elastic: so the resources are only consumed when needed. Different locations are provisioned with appropriate amounts of network substrate, and all the routers, switches, edge caches, and middle-boxes are instantiated from the common resource pool. Such sharing of infrastructure across a broad set of uses makes planning and growing that infrastructure easier to manage.

As was just shown, the key benefit of this transformation is that it will allow AT&T and our customers to share a common pool of resources (& CAPEX) and to use those resources in order to compose network capabilities and services on-demand, with elasticity, and driven with orchestration techniques similar to those seen managing the workloads in cloud data centers.

From an AT&T business perspective, this transformation is expected to give greater access to technologies and innovations from data centers, including rapid innovation in server hardware, virtualization, cloud computing, software defined network switches and controllers, competitive independent software and software development solutions, and well supported open source communities.

---

[2] For this white paper and in the other attachments, the term *White Box* means a network data plane forwarding/processing element that is based on readily available networking hardware, such as merchant silicon, network processors, or ASICs. It is available from many suppliers, is interchangeable with other white boxes; and has both limited integrated control plane functions as well as support for SDN Control (decoupled control) through a standard, open interface.

Additional benefits are foreseen, including more choice of components and suppliers, faster time to market for new products and services, better utilization of physical resources, and greater flexibility in the business models AT&T participates in with both customers and suppliers.

This change in infrastructure is likely to increase support for usage-based software models and diverse sources for hardware and software, and also forging partnerships and relationships with vendors and organizations who have not been traditional telecom vendors.

At this point, Domain 2.0 is not a completed architecture or technology plan; rather it sets direction. There remains much to do before this vision can be implemented, including pivots from networking craft to software engineering, and from carrier operations models to cloud "DevOps" models. We also see an important pivot to embrace agile development in preference to existing waterfall models.

## Cloud Networking Architecture

Domain 2.0 comprises more than simply a network or service architecture. It requires appropriate business practices, a supplier and software ecosystem, a software-savvy planning and operations organization, and management willing to try alternatives and fail fast. This section will set aside these aspects for the moment, and focus on the architecture of a system that is capable of supporting the vision described in the introduction. Since we intend to provide network services in a manner similar to the elastic compute and storage services provided in cloud computing and to use cloud computing technologies within the network infrastructure as a more flexible, cost effective approach to support both traditional network services and those delivered to application tenants in a cloud computing environment, we will call this architecture Cloud Networking. Figure 1 is a high level view of the architecture.

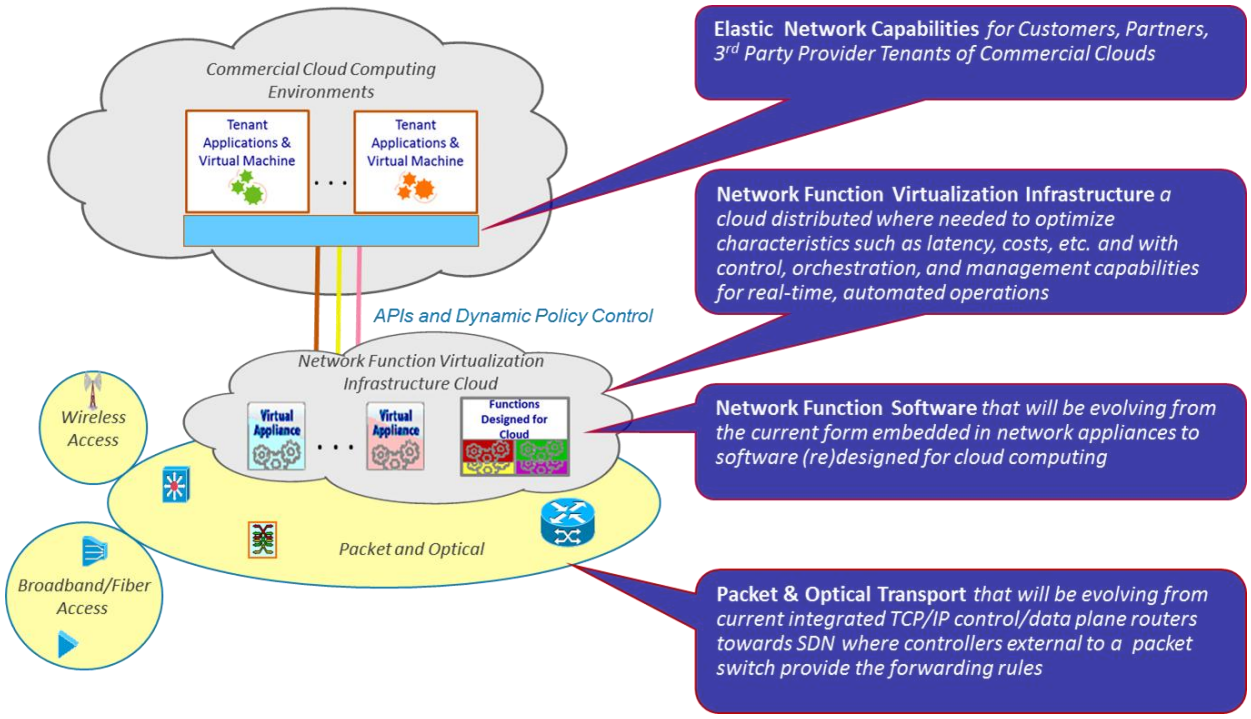


**Figure 1 – High Level Cloud Networking Architecture**

The yellow colored Packet and Optical as well as Wireless Access and Broadband/Fiber Access represent the current parts of AT&T's network that transition over time to common infrastructure and NFV software[3]. However, we do not foresee changing everything all at once. Sometimes the network function needs to be positioned at a specific place, and this allows for less sharing of common infrastructure. Other times, the network elements have specific physical layer adapters that cannot be abstracted or virtualized, and might require special DSP code and analog front-ends (AFEs) that do not lend themselves to significant repackaging. For many parts of these networks SDN techniques have the potential to add value and improve capability over the current way IP routers and optical transports are managed. High level SDN controller languages promise to simplify configuration, ease the introduction of nimble policy control, provide greater capabilities, reduce errors, and enable more real-time changes in a packet/optical network. Using controller-to-packet-switch-interface standards like OpenFlow and NETCONF, one can more easily manage upgrades, repair and maintenance, and enjoy hardware improvements impeded by fewer system dependencies.

The grey cloud labeled Network Function Virtualization Infrastructure Cloud represents the new infrastructure used to support NFV. This is where cloud operations, applications, and architectures are leveraged to support networking workloads.

The network function software represents either a one-for-one mapping of an existing appliance function or alternately some combination of network functions designed for cloud computing. For

[3] In NFV parlance, the applications that support networking are called Virtual Network Functions (VNF), the common supporting infrastructure is called NFV Infrastructure (NFVI), and the umbrella term for the overall approach is NFV.

example, functions designed for cloud computing might be combined, leverage distributed data services to eliminate a tier of appliances, and/or use software logic and characteristics of cloud computing to eliminate fault tolerant or one-for-one failover spare appliance hardware requirements. Many of the initial opportunities in developing cloud networking are centered around using NFVI – which closely resembles cloud computing infrastructure – and developing new applications that support many of the existing monolithic control plane elements, like route reflectors, DNS servers, and DHCP servers. Over time, more network edge functions and middle box functions are expected to migrate to this infrastructure. These functions include SAE gateways, Broadband Network gateways, IP edge routers for services like IP-VPN and Ethernet, and load balancers and distributers. Because these elements don't typically need to forward large aggregates of traffic, their workload can be distributed across a number of servers – each of which adds a portion of the capability, and overall which creates an elastic function with higher availability than its former monolithic version. An important aspect to this portion of the architecture is that the aforementioned functions are instantiated and managed using an orchestration approach similar to those used in cloud compute services. This means that there are catalogs of functions that can be instantiated any number of times and composed into various network topologies to dynamically serve AT&T and customers' interests.

Finally, in the Commercial Cloud Computing Environments, APIs expose functional capabilities of the VNFs, the flexibility and expanded capabilities of SDN controlled networks, and other components of the network infrastructure. Workloads in this environment may have applications distributed across the NFV cloud and in the commercial cloud, or might simply orchestrate workloads supported entirely in NFV infrastructure from these third party locations.

Developing the capabilities just described and being able to provide both those capabilities, as well as the services built upon these types of capabilities is what the architecture team believes to be the most important feature or *killer-app* of Domain 2.0.

## Network Function Virtualization Infrastructure[4]

NFV infrastructure (NFVI) might differ from a commercial cloud or an enterprise IT cloud. AT&T expects a high degree of reuse of the same software components and footprint with commercial cloud offers, but also expects that there will be functional or engineering tradeoffs that are different in the NFVI and want to think of these as separate. The top three likely differentiators identified by the team are: footprint/distribution, separation of data, control and management, and throughput intensity. Other differences included software and software architecture maturity, latency, support for some specialized hardware, and the overall range of workloads supported. Despite these and other potential differences, there remains a strong need to leverage common infrastructure across a broad variety of VNFs, and a desire to get to the point where this infrastructure is provisioned in pods rather than the traditional methods of racking and cabling equipment individually.

[4] Much of the information and terminology describing NFV in this section is taken from work AT&T participated in at the ETSI-NFV-ISG

## Virtualized Network Functions

AT&T's network is comprised of a large and increasing variety of proprietary hardware appliances. To launch a new network service often requires adding yet another variety, and finding the space and power to accommodate these boxes is becoming increasingly difficult. This difficulty is compounded by increasing costs of energy, capital investment, and rarity of skills necessary to design, integrate and operate increasingly complex hardware-based appliances. Moreover, hardware-based appliances rapidly reach end-of-life, requiring much of the procure-design-integrate-deploy cycle to be repeated with little or no revenue benefit. Additionally, hardware lifecycles are becoming shorter as technology and service innovation accelerates, and this can inhibit the expeditious roll out of new revenue earning network services and constrain innovation in an increasingly network-centric connected world.

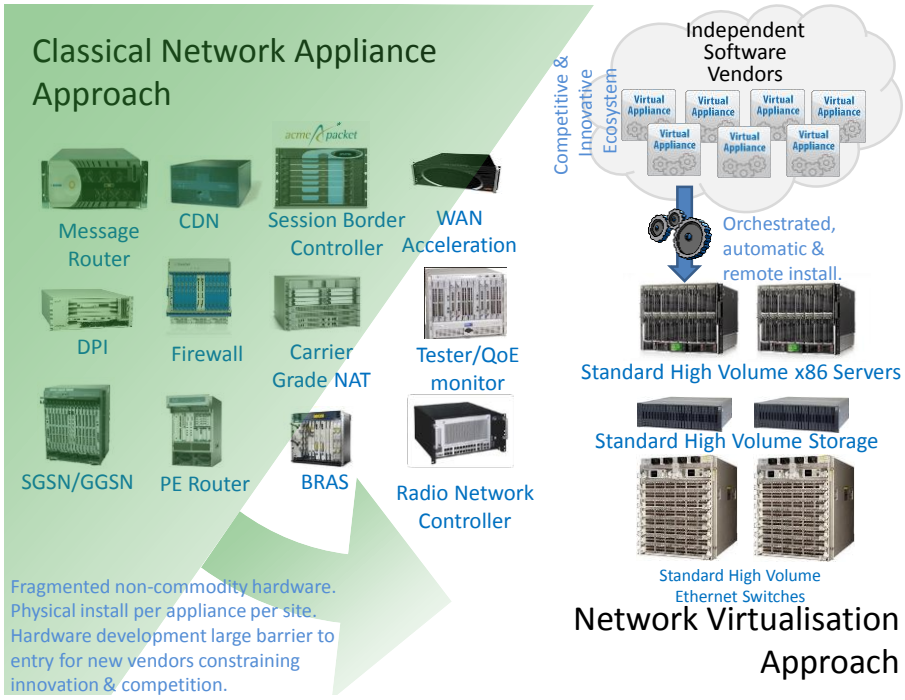


**Figure 2 – Network Virtualization**

NFV aims to address these problems by evolving standard IT virtualization technology to consolidate many network equipment types onto industry standard high volume servers, switches and storage that can be located in data centers, network PoPs or on customer premises. As shown in Figure 2, this involves the implementation of network functions in software, called VNFs, that can run on a range of general purpose hardware, and that can be moved to, or instantiated in, various locations in the network as required, without the need for installation of new equipment.

Virtualizing Network Functions can potentially offer many benefits including, but not limited to:

- Reduced equipment diversity and reduced power consumption through consolidating equipment and exploiting the economies of scale of the IT industry.

- Architecturally decoupling the network function, based in software, from the support infrastructure, based in hardware. This can provide independent scaling and innovation among both.
- Increased speed of Time to Market by minimizing the typical network operator cycle of innovation. Economies of scale required to cover investments in hardware-based functionalities are strongly mitigated through software-based deployment, making other modes of feature evolution feasible.
- Availability of network appliance multi-version and multi-tenancy, which allows use of a single platform for different applications, users and tenants. This allows network operators to share resources across services and across different customer bases.
- Targeted service introduction based on geography or customer sets is possible. Services can be rapidly scaled up/down as required.
- Ability to deploy systems that elastically support various network functional demands and which allow directing the capacity of a common resource pool against a current mix of demands in a flexible manner.
- Enable a wide variety of eco-systems and encourage openness. NFV opens the virtual appliance market to pure software entrants, small players and academia – thus encouraging more innovation to bringing more new services and new revenue streams quickly at much lower risk.
- Enable new types of network services. NFV can readily be applied to the control and management plane in addition to the data plane. This allows virtual networks to be created and managed by end users and third parties using the tools and capabilities heretofore reserved only for native network operators.

To leverage these benefits, there are a number of technical challenges which need to be addressed:

- Achieving high performance virtualized network appliances which are portable between different hardware vendors, and with different hypervisors.
- Achieving co-existence with custom hardware based network platforms while enabling an efficient migration path to fully virtualized network platforms. Similarly transitioning from existing BSS and OSS to more nimble DevOps and Orchestration approaches.
- Managing and orchestrating many virtual network appliances while ensuring security from attack and misconfiguration.
- Ensuring the appropriate level of resilience to hardware and software failures.
- Integrating multiple virtual appliances from different vendors. AT&T would like to “mix & match” hardware from different vendors, hypervisors from different vendors and virtual appliances from different vendors without incurring significant integration costs and avoiding lock-in.

SDN can act as an enabler for NFV, since the separation of control and data planes enables the virtualization of the separated control plane software. NFV can also act as an enabler for Software Defined Networks (SDN), since the separation between data plane and control plane implementations is simplified when one or both of them are implemented in software running on top of standard hardware.

# Software Defined Networking

## SDN Summary

SDN is an architectural framework for creating intelligent networks that are programmable, application aware, and more open. SDN allows the network to transform into a more effective business enabler. SDN enables applications to request and manipulate services provided by the network and allows the network to expose network state back to the applications. A key aspect to the architectural framework is the separation of forwarding from control plane, and establishment of standard protocols and abstractions between the two. However, the term SDN is also applied to a number of other approaches that espouse more open, software-centric methods of developing new abstractions for both the control plane as well as forwarding plane of networks.

Benefits of SDN include:

- Creating multiple, virtual network control planes on common hardware. SDN can help extend service virtualization and software control into many existing network elements.
- Enabling applications to request and manipulate services provided by the network and allow the network to expose network state back to the applications.
- Exposing network capabilities through APIs
- Making the control of network equipment remotely accessible and modifiable via third-party software clients, using open protocols such as OpenFlow, PCEP or even BGP-FlowSpec[5].
- Logically decoupling network intelligence into differentiated software-based controllers, as opposed to integrated routers and switches. Often this flexibility allows a more centralized layer of control with a more global network view, and that has some benefits in terms of improving control plane algorithms.

There is considerable interest in SDN from operators, vendors and researchers. There are several standards efforts by competing parties including new forums like the Open Network Forum (ONF) and traditional bodies like IETF.

[5] Recognizing that BGP-FlowSpec is not well aligned with the notion of control/forwarding separation, it is still seen as a potentially valuable transition technology in getting from today's network infrastructure to the one described in the vision.

## Definition and Key Concepts

AT&T believes that SDN embodies the separation of control from forwarding in network elements. This can be accomplished both with open standard interfaces, like OpenFlow or XMPP, or with proprietary interfaces[6]. We also include clever manipulation of existing interfaces in the SDN definition – like the type of route manipulations that can be accomplished with Route Control Platform (RCP) or BGP-FlowSpec. Finally, we also see the potential system benefits of using improved management interfaces like NETCONF.

Through the physical separation of forwarding and control, SDN can support relationships that differ from the typical one-to-one found in switches and routers today. For example, changes can include physical location, logical abstraction, many-to-many relationships, and a decoupling of the sources or developers of the forwarding and control elements used to create networks.

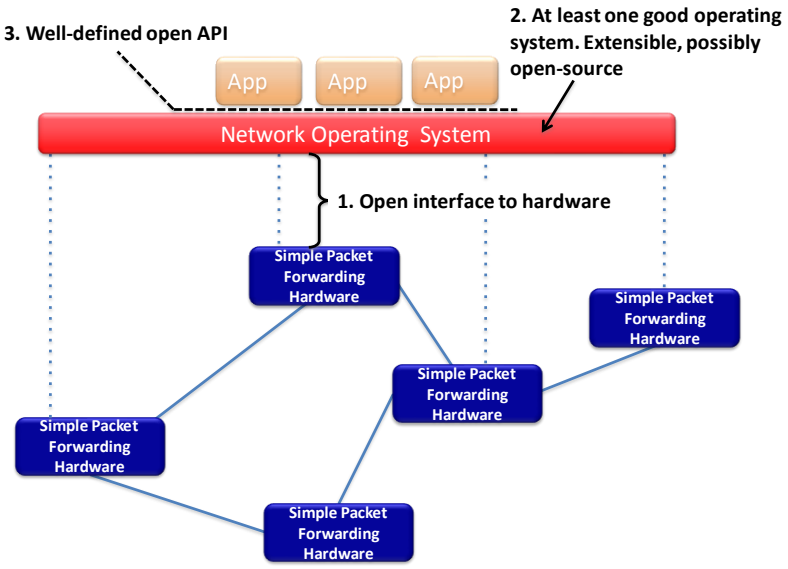


**Figure 3 – SDN Approach to Networking**

As numerically labeled in Figure 3[7], there are three salient aspects to the SDN architecture: 1) an open interface between simple packet [/optical] forwarding and a network operating system; 2) at least one usable, extensible operating system – preferably open source; and 3) a well-defined open API that supports development of applications.

In addition to this architectural approach, SDN offers the opportunity to develop software abstractions in the Network Operating Systems and Applications that were impractical in classic network element architectures.

[6] N.B. Although AT&T accepts that proprietary interfaces fit the architecture described by SDN, there is a strong aversion to being locked-in to a vendor-specific protocol as it's unlikely to allow us to reach our white box vision.
[7] Figure 3 and Figure 4 are taken from an SDN tutorial developed at Stanford and are used with permission.

Typical opportunities described for SDN abstractions include hiding the distribution of state within a network, providing a global view of salient network state, and developing a consistent network graph to applications. These abstractions allow extracting simplification of networking aspects and thereby allow solving more complex problems and increasing the pace of innovation as well as speed and accuracy of deployment.

A common concrete example of the simplification of network through abstraction is network slicing. Rather than exposing control plane applications to the details and complexity of multi-tenant networks, the slicing controller apportions the forwarding network elements among a number of independent controllers – each with a simple view of a single tenant's network or topology. The virtualization abstraction is so compelling, that it's typically represented as a basic part of SDN as shown in Figure 4. Note that this virtualization abstraction is virtualization of networks for users such as subscribers, as opposed to virtualization of network functions or elements as described in the NFV section. There are many other areas where new abstractions are expected to simplify development of network capabilities.

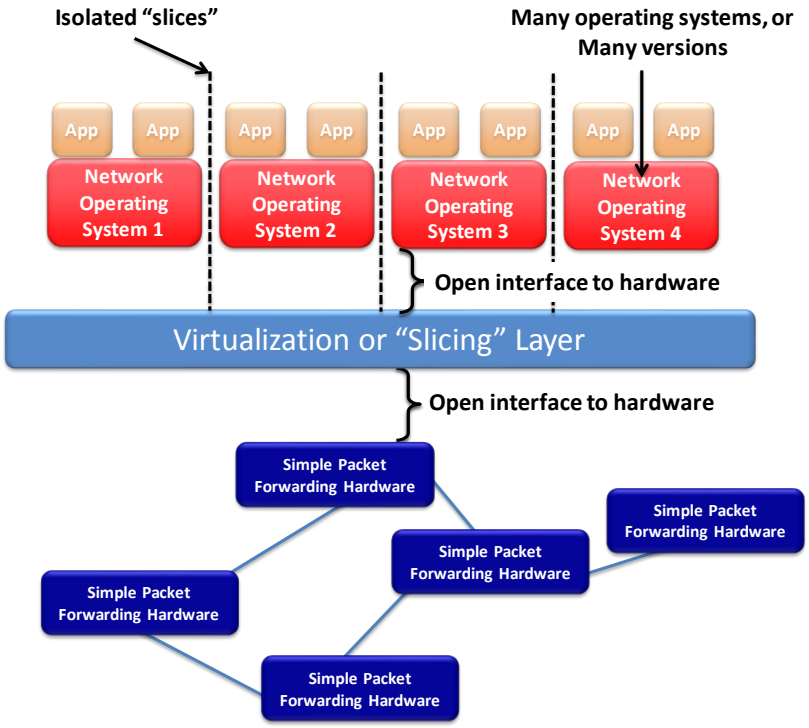


**Figure 4 – Virtualization as an SDN Abstraction**

SDN is already established in many IT and Cloud data centers – where it is often used to control virtual switches distributed among myriad hypervisors. However, it's applicability to carrier networks lags the capabilities for data centers. AT&T is interested in developing capabilities to apply SDN to carrier networks, and this document represents part of an industry engagement initiative intended to help mature SDN to better support carrier applications.

## SDN as Disruptive Technology

It was the intention of SDN to develop a fresh approach to networking, and it's no surprise that the SDN technology is often seen as disruptive. Some of these disruptive aspects include:

*Shift in Value-Add* – with the separation of packet forwarding and control, new opportunities emerge for new suppliers. Vendor lock-in is reduced, and value-add can be provided independently from providing the hardware.

*Enhancing Merchant Silicon* – With a sophisticated external controller, a simple network element based on merchant silicon can functionally compete with a much more sophisticated element where it could not before. Special-purpose network elements like firewalls, load distributors, and various types of gateways might become special-purpose software that can be applied to existing systems to augment their capabilities.

*Instant Overlays* – by controlling a distributed set of software packet switches (e.g. some type of vSwitch) and connecting them with overlay tunnels, sophisticated networks can be created and modified in near real time. Moreover, these networks often don't need much support from underlying physical networks – those have been abstracted into a common fabric. AT&T already uses this approach in our SilverLining™ product.

*Cloud Networks* – Leveraging the virtualization of network resources, new types of networks can be instantiated on shards of existing equipment. Using a single physical network element efficiently for disparate applications can allow de-fragmenting leftover capacity in existing deployments and reduce the number of physical devices in the network. Moreover, if this can be coupled with the development of a common SDN & NFV infrastructure that can be delivered in pods rather than discrete components, then the operational cost advantages can contribute toward the goal of providing networking services much more efficiently than is done today.

## AT&T & SDN

AT&T already supports many SDN capabilities and independently defined, proprietary mechanisms that fall under the SDN architectural framework. There are many use cases that AT&T envisions as opportunities for SDN. The rest of this section provides a sampling of potential ways to use SDN in different contexts, and suggests typical use cases for those contexts.

*Create a Flexible Fabric for Data Centers and NFV Infrastructure* – Use SDN to create virtual network capabilities within an infrastructure fabric, remove middle boxes, and provide customer control of private LAN capabilities. Also use SDN to stitch native, overlay and WAN networks together on demand.

> An example use case is to leverage SDN as a fabric control plane; using either Ethernet or MPLS labels to create a DC fabric from SDN enabled edge devices. Use SDN to create connectivity as it is orchestrated. Reap security benefits because the default fabric blocks.
>
> A second use case is the removal of various appliance middle boxes through the enhanced control of fabric with specialized SDN applications. Use SDN and Top of Rack [or other] switches

that make up data center fabric for load balancing, intrusion detection, simple firewall, network tap, and service chains.

Another use case example is virtual networks. Allow customers to instantiate both virtual and real switch slices and control them with their own SDN controller.

And a final use case is stitching. Use SDN to provision stitching of overlays, native LANs, and MPLS tails into DCs. Have SDN controller use BGP with the carrier network, and expose the entire DC as a virtual switch.

*Simplify CPE* - CPE gets simplified and OpenFlow-enabled. Service capabilities get instantiated in the network – typically in cloud resources. OpenFlow is used essentially as a policy interface, with both the ability to push static policies as well as the ability to react to conditions by referencing a controller. A consumer instance of this approach is called a Network-Enhanced Residential Gateway (NERG) and is being developed by the Broadband Forum. The architecture can be applied to residential as well as business access, and even to mobile handsets and VPN clients.

An example use case is the delivery of IPTV with a virtual set-top box (STB) placed in the cloud. In this case, a DLNA-enabled TV connects to a media server (the virtual STB) in the cloud. SDN is used to direct the Residential Gateway (RG) to "leak out" appropriate LAN protocols to the cloud. Another variant of this use case allows creation of a VXLAN overlay to link the RG LAN to DC LAN where an Ethernet access network does not exist.

*Simplify the IoT* - Customer Networks (both residential and business) start to accumulate devices that attach to various networks. These things have various and disparate networking needs and it's desirable to be able to instantiate networks to address those needs. Network Attachment[8] is facilitated with SDN in order to discover the *things* and to grant access or even instantiate an appropriate network for the purpose. Three classes of networking are foreseen, and a particular device may be in multiple classes at once: 1) LAN class – things have a need to communicate with other things using local networking only. 2) Network Specific Access – a thing needs to communicate with a server or service in a remote cloud. 3) General Access – a thing may need to consume or provide service to a broader set of peers - like the Internet, but with ACLs. General Internet access is a degenerate case of this last type.

An example use case is to use DHCPv6 or IPv6 RS to discover home automation devices, and then use SDN to instantiate an IPv6 prefix for them in the LAN - thereby creating a kind of VPN arrangement among them. The SDN controller can then identify the home automation controller for these devices and instantiate a WAN gateway flow entry to create a link to a cloud service associated with that home automation application. Finally, SDN creates dynamic ACLs (flow rules) for web clients to interact with the system to enable a secure web portal.

*Multi-service Access Networks* – An access network directs flows to service edges based on OpenFlow and/or BGP-FlowSpec direction. Access nodes contain the OpenFlow agent, and service edges are based

[8] E.g. 802.1X, DHCP, IPv6 Router Solicitation, etc.

on a variety of both traditional (e.g. Ethernet Switch, IP-VPN router) and NFV (e.g. TWAG or BNG VNF running in a VM)[9] implementations.

> An example use case is to use OpenFlow to divide and load balance user traffic into multiple service edge instances. This application can support basic load balancing as well as grooming, re-arrangement, and fault recovery: all functions that have been historically hard to perform in broadband access networks. Traffic flows can be delivered to multiple service edges from a single access session.
>
> An alternate variation of the use case could use BGP-FlowSpec installed in existing metro Ethernet service edge devices to discern and direct VoIP and Video service traffic to cloud instances of those services while allowing the balance of customer Ethernet traffic to be handled using default behavior.

*Virtual Access Networks* – The access network provides an API to expose Access as a Service to applications deployed in cloud data centers. The DC Apps can instantiate access network topologies, as well as configure and control them. This approach will support connecting customer premises device and application capabilities to DC-based applications without using the Internet to build overlays.

> An example use case starts with a typical hub and spoke geography for a business, and then adds an API to instantiate access network instances for that customer. The business can then control those spokes using their own instance of an SDN controller. They create a desired topology, and have the ability to direct access traffic from their locations into the topology or into other networks or services. All of this can be self-serve or a managed service.

*Customer-controlled Metro Networking* - The metro network gains the ability to steer traffic and provide load balancing, fault steering, and dynamic topology and capacity capabilities. This allows businesses to dynamically shape their network and traffic load in order to deal with demand, outages, and attacks.

> An example use case for metro networking is to selectively SDN-enable parts of the Metro Network – allowing steering of ingress traffic. Ingress from IXC PoPs can be load distributed, redirected at failures or overloads, and even intrusion detected and mitigated with a service chained scrubbing facility. Driven by an API, traffic bound for an Enterprise customer can be scanned for analytics, directed to the premises or to a cloud [burst] resource, and optionally scrubbed.

*Transport Control Plane* - Develop a vendor-independent control plane for WDM and transport layers. GMPLS is not delivering on its promise for better transport control and co-ordination with MPLS networks, so AT&T is very interested in developing SDN capabilities that would allow us to develop such capabilities. As an example, we would like to develop a potentially hierarchical controller architecture to install a dynamic control plane for WDM and transport infrastructures. The control plane protocol

[9] TWAG is a Trusted Wireless Access Gateway – used in 3GPP mobile networks – and BNG is Broadband Network Gateway - the typical IP edge device used for Internet access.

should be amenable to WDM and OTN abstractions, but with an eye toward interworking with other MPLS networks (probably using BGP).

*Augment existing core network capabilities* - Instrument MPLS core for analytics using SDN counters, and provide for flexible probes and taps. Develop SDN-centric policy controls to enhance peering relationships.

> Two use cases for this capability would include deploying capabilities at peering points of the core network in order to collect analytics and provide policy control. Policy controls can be based on analytics, like killing DDOS flows, or on business rules. Basic SDN abstraction is layered up to business policies. Demonstrate load balance capability driven through customer controller and API, as well as DC mobility co-ordination to support data center moves. Expose the core capabilities to customers using standard SDN abstractions.

*Multi-Network Control Plane* - Extend the aforementioned WDM / OTN control plane to interact with the MPLS packet core control plane. Show coordinated reaction to faults and ability to optimize networks routes for performance, availability, and economics. Support re-optimization across layers[10], on demand. Develop calendar scheduling capability and expose to DC and Enterprise customers. Determine how and when a WDM-based universal core provides economic advantage.

These represent a variety of SDN use cases, and AT&T is interested in learning about other ways SDN can be used both to simplify networking and to provide new, innovative network capabilities.

### Futures

Research continues in SDN, and there is evidence that merchant silicon providers can provide efficient, scalable SDN fabrics. AT&T also looks forward to advances through research that would make networking more directly applicable to developer needs with improved abstractions. Similarly, there's a growing school of thought that networking needs to become more sophisticated in the higher layers[11], and there is corresponding SDN work to allow development of new data forwarding abstractions to compliment that work. AT&T eagerly awaits solutions that simplify networking through this application of scientific methods to extract basic principles.

## Orchestration Control and Policy Management

The Domain 2.0 architecture requires a sophisticated operational framework capable of supporting the new ecosystem of software components, APIs and virtualized hardware capabilities. This framework is embodied in the exposure, combination and manipulation of VNF components and service resources into operable entities using a model-driven methodology for service creation, composition, and deployment. The software framework represents this collection of all functional components and their

[10] In this case layer does not refer to ISO OSI communications layers. Instead it refers to the several important networks that are built-up within AT&T. Layer 0 is photonic/WDM. Layer 1 is OTN. Layer 2 is Ethernet. Layer 2.5 is MPLS. Layer 3 is IPv4 and IPv6. Layer 4 is TCP, Layer 7 includes SIP, IMS, and other application constructs.
[11] E.g. http://rina.tss.org

relationships. The instantiation of the model components provides all of the run/real-time virtual network/service topology/inventory, state and management capabilities.

Orchestration capabilities provide real-time, "plug and play", "instant" availability of service at scale. Policy-driven controls span operator, customer, service and resource dimensions. Analytics (both real-time and predictive) and feedback loops to policy controls to efficiently distribute and manage elastic workloads. Dimensions of analytics include temporal (real time, recent, past, near future, future), locational, functional, quantitative (statistics), trends, polices, external influences, forecasting/engineering, traffic management, etc. Analytics will utilize Big Data technologies, including extraction, stream processing, real-time and off-line analysis.

Examples of specific orchestration functions include:

- Service catalog
- Tenancy, affinity & assignment of customers to pools & resources
- Service composition & dependencies
- Service instantiation & activation
- Service execution & lifecycle management
- Service usage capture & forwarding
- Service & resource topologies
- Service & resource abstraction
- Dynamic creation, modification, customization & release of virtualized resources
- Run-time management, monitoring, & scheduling of computing, storage, & network resources into consumable services
- Actions from threshold crossing events
- Monitoring of resources, transactions, flows and workloads
- Security enablement
- Allocation of available resource capabilities within a pool to optimize utilization, performance, reliability
- Analytics, trending, & prediction feedback into optimization
- License management
- Business & operations rules, workflows & processes
- Audits & diagnostics

## Operations Transformation

As shown in Figure 5, the Domain 2.0 architecture includes an operational shift from physical/hardware to a virtual/software-centric system which significantly compresses timescales and enhances flexibility. This software-centric architecture provides the opportunity to implement an evolved operational model derived from the union of best practices from telecom and data center operational tools and approaches.

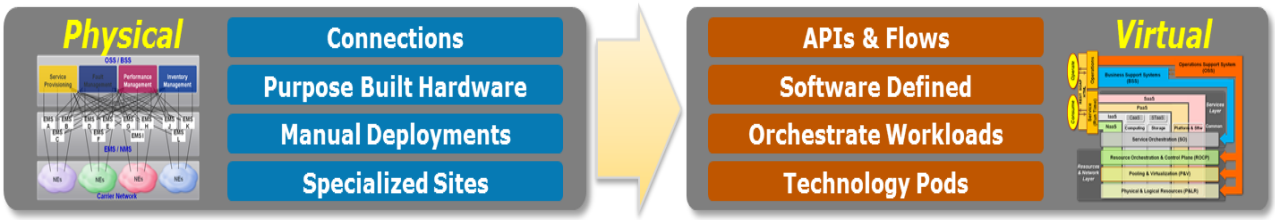


**Figure 5 – Physical to Virtual Operations Transformation**

## Key Operational Shifts

Software-centric architecture allows changing how and where many FCAPS (Fault, Configuration, Accounting, Provisioning, Security) functions are performed, and enables the transformation of many OSS (Operations Support Systems) and some BSS (Business Support Systems) platforms. Current OSS/BSS functions do not "go away" but will likely be refactored, simplified, and in some cases functionally expanded to reflect new operational needs and opportunities for operational excellence. Key to this will be the adoption of various shifts in operations.

Expected operational shifts will include:

| From | To |
|---|---|
| Hardware Centric | Software Centric |
| Separate IT/data center & Network/CO | Common technology & technical plant |
| Quarterly software releases | Continuous software process - "sandbox." |
| Geographically fixed, single purpose equipment | Highly dynamic & configurable topology & roles. |
| Tight coupling of NE, generic, EMS & NMS/OSS | Separation of physical & logical components. |
| Separation of service elements & support systems | Integrated orchestration, automation & virtualization. |
| Faults as service failures | Faults as capacity reduction events. |
| Hardware monitoring appliances | Software based monitoring. |
| Service specific resource combinations | Profiles, templates & reusable resource combinations. |
| Special design and provisioning processes | Configurable catalog/rule-driven delivery frameworks. |
| Optimized provider network & ops process | Optimized customer experience. |
| Highly constrained, independent & disaggregated control planes | Highly integrated & automated control planes driven by customer & operator policies. |
| Limited service dimensions | Multifaceted service dimensioning. |
| Highly constrained data translation & synchronization solutions for shared management knowledge between network & systems | Shared management "Data Bus" technology between network & systems. |
| Slow tooling changes requiring coding | Rapid tooling changes using polices/rules |
| Network management | Customer experience management |
| Long lead provisioning times – often hardware and process constrained | Real-time provisioning |
| Static billing and charging | Granular and dynamic usage-based charging, billing, financial management, subscription |

# The Emergent Operator Software Ecosystem

## Transition from system standards to component standards

Traditionally, carriers interested in a new architecture would gather with their suppliers and start a new standardization activity in one or several SDOs, often calling the work Next Generation Network or NGN. The standards body would gather requirements from interested parties, work out backwards compatibility, and negotiate an outcome that was mutually acceptable and described the end-to-end system as an optimized and tightly coupled whole. This process was [is] lengthy & expensive, diminished a carrier's ability to navigate their own technology transitions, and often created entities that fail to serve the interests of the companies that fund them. A thought provoking article on this topic was published by Dr. John Waclawsky in Business Communications Review (Oct, 2004 p57) entitled: *Closed Architectures, Closed Systems and Closed Minds*.

This is not to say that standardization is no longer valuable, but rather that the goals of standards activities are better targeted toward smaller, re-usable components that can be composed and recomposed into various systems and architectures. The tenants Waclawsky puts forward are: 1) *You can't think of everything*, so once you pick network elements and the protocols for an architecture you have effectively limited your flexibility and created a system that is brittle; and 2) *Dependencies up the ante*, meaning the technology bets grow larger as you progress from architecture to the design phase and then an eventual deployment – this requires you to see the future very clearly.

Domain 2.0 seeks to follow agile development processes, and will avoid locking-in to a specific system architecture. The good news is that with SDN and NFV technologies it becomes possible to envision changes in basic system architecture without having to dump and re-deploy infrastructure and its associated capital investment. Since the essence of networking is provided through software with dynamic capabilities of instantiation, it becomes possible to use the same supporting infrastructure to deploy largely different network architectures – even at the same time.

## Transition to vertical integration and in-sourcing of core business competencies

Large successful Internet scale companies are demonstrating a pattern toward in-sourced or jointly open-sourced development of the key technologies and systems needed to support their core business. In some cases this comes from an imperative to develop new technologies that did not exist, but more times than not, these companies take the same approach even when that technology already existed and had a thriving ecosystem.

In many of AT&T's businesses the company sources critical technology from one or two key vendors who work closely to help configure and optimally deploy that technology. In this model, core technologies, which are often proprietary to the suppliers, become critical to our business. To mitigate business risk, the company has developed business rules for second suppliers and evaluates the risk of doing business with suppliers should they go out of business.

This practice is different from that of successful web-scale businesses. AT&T expects to increase the depth of understanding of our core technologies held by our staff to the point that they can integrate, and even design the systems from scratch. AT&T expects to develop key software resources in a way that they can be openly used, and cannot be lost through the acquisition or insolvency of a vendor partner. This pivot will enable AT&T to do business with startups and small businesses that we might have deemed too risky in the past. While they may not always endure, small businesses demonstrate the large fraction of innovation and agile development in the marketplace and enabling the company to do business better with these small companies is a key element of Domain 2.0.

## Transition to Information models and Software frameworks overshadowing boxes and protocols

NFV components (VNFs) plug into a framework which enables an emergent operator software ecosystem. The essence of this framework is in a set of API and event standards that provide the plug-and-play linkage between the VNFs, which must conform to these standards, and the software layers that support them.

In addition to providing FCAPS functionality, support software provides the ability to view and manage the software ecosystem including the:

- VNF and other software component hierarchies
- Network topology(s) – including a way for components to request/reserve bandwidth as needed.

The software framework also enables the linkage between VNFs and product implementations that need to discover and use the unique capabilities of novel VNFs created by the vendor community. These NFV capabilities are made available and accessed through APIs and event types presented through catalogs and repositories.

All of the APIs and events, catalogs and repositories noted above collectively make up the framework which enables the emergent operator software ecosystem to flourish. Vendors are free to implement novel VNFs as they desire, confident in the knowledge that their creations will plug into and be instantly usable by an operator ecosystem. Indeed, this concept is also intended to allow 3rd parties to deploy functions and build businesses on that self-same infrastructure.

Framework standards can start small and evolve over time. Ultimately it should be possible to power-on physical equipment and install VNFs that leverage such equipment and have both of those things be automatically recognized by a component manager which organizes equipment and VNFs into a hierarchy of functionality that can be browsed and managed using APIs and GUIs.

As they are instantiated, components declare themselves to be in a standard part of the functional hierarchy (e.g., physical connectivity – optical switch) or in a new branch of the hierarchy (e.g., some new abstraction or unforeseen innovation). Each component declares and exposes properties that can be set and managed. They also expose standard APIs that support management and data gathering by

the framework. Finally, they emit standard events (e.g., usage and fault events) that complete their plug-and-play integration with the framework FCAPS environment.

Some components in the functional hierarchy make up the network topology(s), which automatically morph and change as they are plugged in and come on and off line. A bandwidth allocation or user experience manager makes it easy for other components and management layers within the ecosystem to request and reserve resources through a set of APIs. The whole topology is brought to life through a GUI which depicts the network and enables drill down to see detailed interconnections. Links are right clickable and provide information on the bearer, capacity, assignments, tickets/alarms, etc…

The framework that enables the above vision is built using open source technologies and an open information model that confers no special advantage to any supplier or participant. It is constructed using an agile approach that doesn't seek to boil the ocean and get everything right the first time but rather starts small and evolves over time as the community collectively enhances and improves the ecosystem.

## The opportunity to develop network software using open source

Domain 2.0 seeks to deploy a software framework suitable for large carriers based on open source which we can use freely and to which we can contribute and encourage others to contribute as well. As described in the transition to vertical integration, this will allow AT&T to draw from a much broader set of software sources, and attract top-tier software architects and developers. This software framework will be the scaffolding for AT&T services, and also serve as the foundation for 3rd parties to build on – either in their networks, or in ours as a NFVI customer or business partner. The APIs that AT&T would expose to others are the same that we use to develop services ourselves.

Needless to say, in the development of this software ecosystem, we will not accept a proprietary information model, API, or system that would bind us tightly to a single supplier or approach. Moreover, the software framework should allow for the same system flexibility obtained through the loose coupling of small components that was described under the *Transition from system standards* section.

There's evidence that AT&T is not the only carrier thinking in these terms. It makes business sense to work with others who have the same needs and desires on common technologies and components that are not key differentiators in our competitiveness. Just as we can cooperate with other carriers in the development of standards, there is an opportunity to cooperate with others in the development of open source software used to provide networking services. Cooperation in this space accelerates the ecosystem maturity. It seems like an important aspect to NFV and SDN to use open source and open sharing of concepts to foster strong and steady maturation of the technologies and to engage broader participation.

Cloud and DevOps approaches appear to be suitable and desirable – and there can be efforts to develop wrappers for legacy systems that would allow them to be integrated into the new framework.

AT&T does not expect to get everything right the first time. We will espouse an agile approach, and look for a level of modularity and decoupling that allows for easy trials of alternatives and migrations.

## Conclusion

AT&T seeks to engage in a transformation of network technology, operations, infrastructure, software, and APIs in a way that provides greater value to customers and application development partners. The result will be a cloud-centric networked capability that can support varied workloads along with flexible business models. Comments and engagement are welcomed.