A Software Defined Networking Approach To Cable Wi-Fi

Alon Bernstein, Vince Pandolfi, Charles Duffy, Sangeeta Ramakrishnan
Cisco Systems

Abstract

Software Defined Networking (SDN) is a framework that lends itself to coordinating different networks. Because of that it has the potential to tie Wi-Fi, the HFC and the core/access in a consistent and manageable way. The paper will present several options for this coordination – from evolutionary to game changing.

SDN definition?

Software Defined Networking (SDN) is a term that started being used in the networking industry around 2009. SDN can trace it's lineage to the Clean Slate Research Program at Stanford University [1] whose mission was to explore what kind of Internet we would design if we were to start with a clean slate and 20-30 years of hindsight. SDN is defined as the separation of Control and Data planes using an open standard protocol to communicate between them as depicted in Figure 1. This differs from a traditional network device (such as a Router, Switch, or CMTS) in which the data and control planes are vertically integrated. SDN promises flexibility and rapid innovation by virtue of the fact that the control software would be removed from the relatively constrained network device to a generic server that can be easily scaled to have more processing and memory capabilities.

In the SDN architecture, key functions such as routing, topology discovery, and policy are removed from the individual devices and located centrally in a controller and the applications that reside on top of it. The controller, with its " bird's eye" view of the network uses a simple protocol such as Openflow [2] to populate the forwarding tables of the networking devices. Figure 2 illustrates the Classic network architecture vs. SDN.

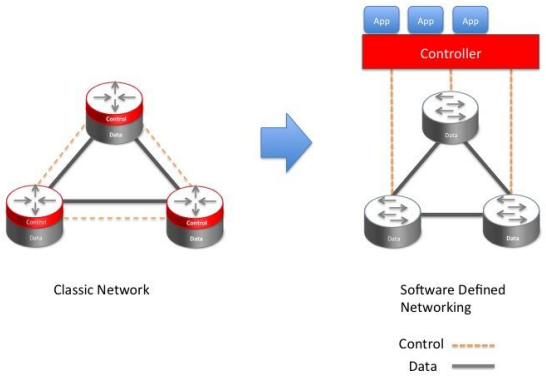

Figure 1 centralized vs. distributed control

In the center of the diagram is a Controller that handles communications to network devices including physical and virtual switches via a southbound protocol like OpenFlow. Northbound, the controller presents a level of abstraction of the underlying network. Applications communicate with the controller using "controller APIs" and the controller in turn interacts with the network. The applications could be written by the vendor of the controller, by any third party or by a service provider. In other words the controller acts as middleware that helps in providing a higher lever of abstraction to the application developers.

Another key attribute of SDN is the two-way communication with the network devices. The network can be thought of as a large distributed database of flows and states. Current configuration tools tend to either uni-directional (CLI which does not have a good error reporting) or localized to only one device (PCMM which is limited to the CMTS) limiting the ability of the configuration tools to rollback when an error occurs or verify the end-to-end correctness of a configuration.

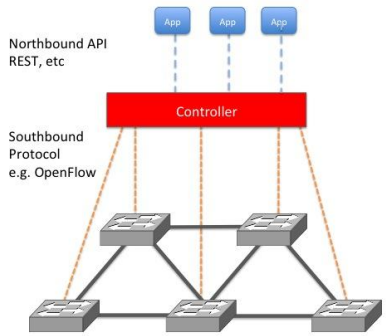

Figure 2 SDN architecture

It is worth calling attention to three emerging SDN related technologies. These are Programmatic APIs, Overlay Networks, and Network Function Virtualization (NFV). The first technology is where open published APIs are provided for existing and new network devices. These APIs allow an operator to control many if not all of the functions of the device, beyond the basic packet forwarding control that is provided by openflow. For example, an APIs could allow an application to change routing or implement QOS policy or simply retrieve the network topology. The second technology is Overlay Networks. The concept here is similar to a Virtual Machine where software creates an "overlay" network that rides on top of an "underlay" or traditional network. The Overlay replicates the functionality provided by a traditional network but since it is completely defined in software it is potentially more flexible. The overlay relies on the underlay network to provide stable transport and connectivity using all of the methods and protocols found in traditional networks. The third technology, NFV is being developed by the European Telecommunications Standards Institute (ETSI) in the NFV Industry Specification Group. NFVs goal is virtualize network functions such as Network Address Translation (NAT), Session Border Controllers (SBC) Deep Packet Inspection (DPI), etc. by using industry standard server virtualization techniques. In contrast to SDN, which moves the control plane to the cloud, NfV moves the data plane as well.

Matching SDN framework capabilities to Wi-Fi requirements

SDN is a framework that can apply to a large range of application. Part of the challenge of applying SDN is a good "divide and conquer" strategy. To decide what to focus on we will list the requirements for supporting a Wi-Fi network and then match them up with the SDN framework capabilities.

To support Wi-Fi networks the following requirements need to be met:

Mobility: Including micro-mobility, hostspots mobility and mobile offload (which will be explained in more detail later in the document).
Authentication: Validate subscriber identity
Subscriber Services: QoS, BW accounting, legal intercept, charging, parental control etc.
Access point (AP) management: SW upgrade, configuration
Mobile Node (MN) management: capability discovery etc.
RF resource management: Frequency planning
Seamless handoff: minimize packet drops when moving between access points.

Though SDN is a framework that may be defined in several ways, the following list is a high level set of capabilities that should fit all SDN flavors along with the benefit of each capability:

Separation of Control from Data plane: Feature velocity, easier debug, and fault tolerance
Logically centralized control plane: better control over the network, feature velocity, easier Debug/test/simulation.
Control plane can run off-box (aka "cloud"): developing SW in a non-embedded

environment, hope for better scale, and access to modern SW development tools
Use openflow or higher layer formal API: solve vendor interoperability issues, "end of protocols" (since a protocol does not have to be defined a-priority for two endpoints to interoperate), Fast release of new features; ISP can do its own SW customization
Create dedicated paths through the network: Assure services, simplify the network by flattening
Direct application control of the network: simplified configuration (auto-configuration/automation).

This paper focuses on the intersection of mobility requirements and SDN capabilities and outlines where SDN can help with improving mobility solutions.

PMIPv6, SoftGRE and anchor points

Since the concept of an anchor point is essential to our discussion we will start by examining how it applies to two common mobile arch1itectures before diving into the SDN discussion:

1. PMIPv6
2. SoftGRE

PMIPv6 is an IETF standard (RFC5213[4] and RFC5844), and provides mobility to endpoints, without requiring client modifications. PMIPv6 involves Mobility Access Gateway (MAG) and Local Mobility Anchor (LMA). LMA is defined to be the topological anchor point i.e. home agent for the Mobile Node's (e.g. Wi-Fi user device's) IP prefix (es) and manages MN's binding state via MAG. The MAG manages mobility-related signaling for the MN that is attached to its access link and is responsible for tracking the MN's movements to and from the access link and for signaling to the LMA.

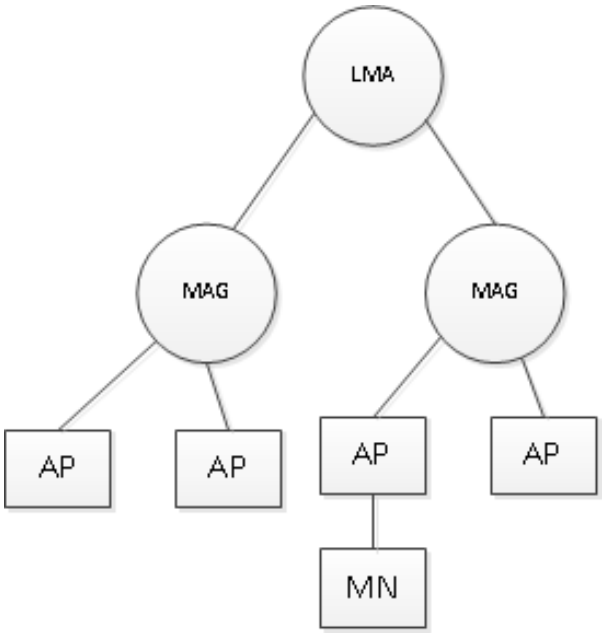

Figure 3 PMIPv6 Components

In PMIPv6, the MAG informs the LMA about the Wi-Fi users during user authentication/authorization. This allows the MAG and LMA to send/receive Wi-Fi user traffic (i.e. Ethernet frames) over the PMIPv6 tunnel. The MAG functionality could be embedded in a Wi-Fi Access Point, or on the CMTS. While the Wi-Fi user is connected to AP/MAG at layer2, its IP address is anchored the LMA. This allows IP mobility, when the Wi-Fi user roams and changes AP/MAG attachments.

SoftGRE

The architectural approach with softGRE is to build an over-the-top IP tunnel to deliver the Wi-Fi user device's Ethernet traffic between AP and a remotely located anchor point (i.e. tunnel termination entity), using GRE. This approach requires IP connectivity between AP and the centralized entity. The data plane comprises users' "Ethernet over GRE over IPv4|v6 over Ethernet [over DOCSIS (or PON)]" in the last-mile access and "Ethernet over GRE over IPv4|v6" (over MPLS, if existed) in rest of the network (up to that centralized entity). While Ethernet over GRE

over IP usage is not well known or used, it is standardized at the IETF [RFC1771].

With softGRE an AP establishes a L2TP tunnel with the remote L2TP tunnel concentrator (e.g. centralized entity) and sends/receives Wi-Fi user device's Ethernet frames, over GRE (over IP) tunnel. It is important to note that GRE doesn't require a control channel and can be set up in a stateless manner (aka soft GRE) without requiring any tunnel configuration.

Although both approaches described above use an anchor point, there are differences between the two. For further details on the trade-offs of these architectures refer to [3].

For the reminder of the paper we will refer to an "anchor" without detailing the protocol framework around it.

How can SDN help?

There are several benefits for looking at SDN to solve challenges in SP Wi-Fi deployments. The first is that Wi-Fi mobility is in its infancy. The L1/L2 micro roaming has been efficient and enhanced by several standards like 802.11i/r/u. The greenfield exists in the area of macro roaming. The CAPWAP RFC5415 (see ref [7])_standard provides some limited geographic mobility and the concept of a separated control and data plane but the current implementation by vendors generally co-locates processing of both the control and data plane. The CAPWAP protocol provides a layer 2 transport and requires handoff of data traffic to other services like Internet services gateways and network address translation.

This leaves the service providers with having to stitch the network paths together for these different services. The use of the overlay tunnels allows for over the top deployments that may or may not be traffic engineered correctly into the core network. This is a particular challenge with larger providers that have silos of responsibility in their organizations.

There are numerous moving parts in a successful SP Wi-Fi implementation. The challenge becomes connecting these moving parts within and in many cases between services provides. The number of these touch points increases when considering the numerous use cases for SP Wi-Fi.
The use of SDN provides a dynamic network fabric to connect these different elements. The benefit increases exponentially when these components are provided by multi-vendor solutions.

There are two types of solutions SDN can offer:

1. **Evolutionary:** current Wi-Fi solutions rely on an anchor point in one way or another, i.e. Even though the MN moves between AP's the anchor remains a static point of connection to the rest of the network. An example of an anchor point of PMIPv6 would be the LMA. For softGRE it would be the wireless LAN controller. For anchored architectures we can identify current issues and outline how SDN may help address them.
2. **Game changing:** with SDN we create a Wi-Fi architecture without an anchor point. Instead of anchoring a MN to a fixed point so that the rest of the network can treat it as a fixed asset SDN allows the network (or more precisely a network overlay) to move along with the subscriber.

Application of SDN to anchored architectures

Mobility solutions range in complexity depending on how far from the home network a subscriber can get, but all the Wi-Fi anchored architecture share the same base concept: they maintain IP address persistency by tunneling traffic from the MN, so even though the tunnel end points may move as the

MN moves, the MN still "feels" as if its connected to its home network. Different MSO network implement (or considering implementing) architectures from basic IP persistence to mobile offload. This section will go into the details of each one of these solutions and how SDN may help with the mobility aspects:

1. **Basic IP address persistence within a domain**: The simplest mobility solution is to have a single anchor point – a direct tunneling of SSID traffic to a fixed anchor point. This means that subscribers can maintain IP persistency only within a "domain' (which in reality can be fairly large). One simple example of this approach is an Intra-CMTS mobility mechanism whereby an MN keeps the same IP address as long as it moves between Access Points that are connected to the same CMTS. The current issues with the this solution are (a) when a handoff is needed a new tunnel setup needs to be established with a new anchor point which slows down the handoff process, (b) a single anchor point creates a large failure domain. On the positive side these tunneling/anchoring solution to IP persistence are widely deployed, relatively vendor neutral and simple. The way SDN may help in this architecture is to add the flexibility to attach to multiple anchor points by dynamically moving the tunnels without the need for the AP to establish a new one.
2. **Intra MSO mobility**: a step up from case (1) is IP address persistence anywhere within the MSO network (i.e. across domains and anchor points). Current solutions such as softGRE and PMIPv6 allow for this mobility and solve many of the problems with case (1) by having the ability to deal with MN movement across anchor points. The main help SDN can offer in this architecture is dynamic configuration of the anchor points and helping to decide on the optimal anchor point to connect to.
3. **Inter MSO mobility (branded services):** A subscriber may roam between MSO A and MSO B networks. From a technical implementation point of view the solution to this type of roaming is the same as case (2) described above and the main challenge is coordinating authentication and identity information between two different providers, which is outside the scope of this paper. From an SDN point of view this may require an even wider "bird's view" of the network since the connection point to an external network have to be taken into account.
4. **Mobile offload**: mobile operators are interested in offloading their cellular data network to Wi-Fi hotspots, including MSO Wi-Fi hotspot. This would be the ultimate mobility play where the only common equipment is the MN itself. However, from a mobility point of view its still the same basic tunnel architecture as in the inter-MSO case (3) which in turn has the same data plane architecture as (2) and just like the previous case the challenges of exchanging authentication and identity across providers is out of scope for this paper. Note that with mobile offload the handoffs between the networks may be more frequent then in other use-cases because of the greater chance of overlap between the networks (mobile and WiFi).
5. **Local Breakout (application based)**: A mobile service can be finer-grained then MN mobility. Each application running on an MN might have its own mobility domain, therefor a single MN may require multiple anchor points to

connect to. For example, a user may have a dedicated connection to a home service provider for watching premium video content and at the same time, on the same MN, another connection for non premium content that runs in the background (messaging, e-mail etc.) and can be sent through the hosting MSO as a native service.

The use of tunnel technologies provides the mechanism to allow mobility in a geographic Wi-Fi deployment but the tunnels generally require termination on the anchor or tunnel endpoint. This behavior allows for seamless mobility but requires more extensive anchors due to higher amounts of signaling traffic and needing to service all user traffic – a local breakout based on applications puts further scaling requirements on an anchored system.

There are several popular methods to break traffic out of the tunnels. The first is Selected IP Traffic Offload (SIPTO see ref[6]). The SIPTO method supports offload of IP traffic directly to the Internet or other locally hosted services. This provides the best user network performance and prevents excessive traffic from needing tunnel transport back the anchor. One of the drawbacks would be loss of visibility of the user traffic flow for applications such as lawful intercept for the home provider and mobility for those applications that are offloaded. One of the other methods would be a concept of LMA (local mobility anchor) chaining. This allows Mobile IP traffic to be relayed between LMA anchor points based on the home relationship of the end user. This method does not necessarily provide for local breakout but larger provides could use this method for regional breakout of user traffic. This method does allow for home provider visibility and control, mobility and distrubted signaling load. SDN can help with dynamically selecting an LMA, which would be a simpler solution then LMA chaining. Note that with SDN intermediate nodes can do their own classification of the traffic and do not rely on the MN to tag application traffic.

For all the anchored architectures it becomes clear that although the control part might be unique the data plane part similar and that's not surprising given that the are all based on the premise of a tunnel to an anchor point. The SDN contribution can be summarized for all these technologies as:

1. **Optimized path selection**: when there are multiple anchor points to choose from an SDN solution has the advantage of seeing the network as a whole and dynamically choosing the best anchor based on various criteria (geographical proximity, load, cost and more)
2. **Dynamic anchor point selection**: in addition to selecting the optimal anchor point an SDN solution can assist in dynamically moving a tunnel to a new anchor point as network conditions or MN location change.
3. **SDN may address vendor interoperability issues**: different providers use different equipment and different tunneling technologies. However, for SDN enabled equipment we can configure a consistent tunnel across equipment and provider sites. Note that for the cases of inter-MSO mobility and mobile offload this will require controller-to-controller coordination, which is outside the scope of this paper.

It is worth mentioning that there are other mobile technologies, most notably LISP-MN that requires changing to the MN itself. LISP-

MN requires tunnel termination that is not strictly an "anchor". However, the discussion of LISP is outside the scope of this paper.

Un-anchored Architectures

With an un-anchored architecture there is no need to tunnel user traffic. The network can follow the subscriber and present a virtual port that appears constant to the MN even though it's physically moving. For example, the response to ARP/DHCP and other L2/L3 network protocol can mimic that of the original home port. It can be view as a connection to an MN that moves as the MN moves.

In the data center world a solution such as mentioned above already exist and is generally referred to as "network virtualization", though one should note that for the Wi-Fi case we discuss the primary use-case is not to create a whole network overlay, but rather to create a dedicated overlay to a particular user.

In its pure form the un-anchored architecture can become a scaling challenge for the network because every subscriber, and in fact, every application within an MN, will need its own dedicated "flow" or connection through the network. While network devices of the future might support such scale we propose an interim solution; all flows that share a common destination, for example a path outside the MSO and into the global internet (at some exit point which is coordinated by SDN as well) can share a tag and do not need to be managed as individual flows, thereby reducing the need to have per-flow scaling in the core of the network. Note that the aggregation point does not create a new anchor – it has nothing to do with mobility. Its role is only to aggregate flows in order to improve network scaling.

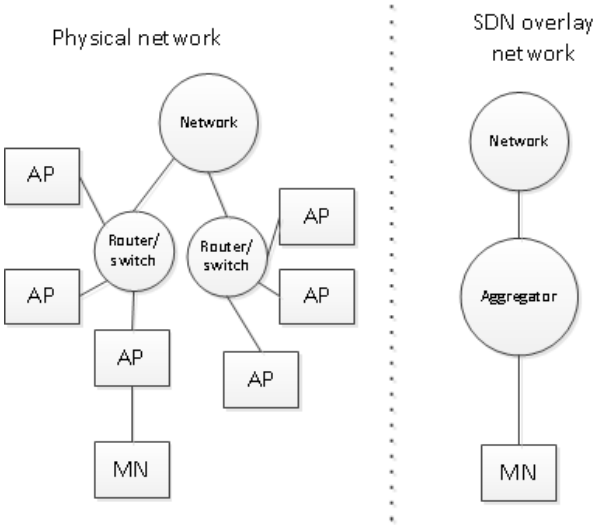

Figure 4 Overlay network with SDN

Figure 4 illustrates the differences between the overlay network and the physical network. The MN is logically connected to the virtual network (since it's the virtual network that responses to ARP/DHCP etc from the MN) and that network is a simple static network that does not change even as the MN moves between AP.

While one may argue that a tunnel is also a way to implement an overlay network, however, the SDN approach has several advantages; there is no need for a tunnel and therefor no MTU issues with a tunnel encapsulation or scaling issues with tunnel control. In addition the centralized control of the overlay allows for better optimization of network paths, and faster response to changing network conditions then a traditional tunnel.

Conclusion

SP Wi-Fi deployments are accelerating throughout the industry. Furthermore, there are elements in existing Wi-Fi protocols that can be considered "SDN" such as the separation of data and control in CAPWAP. Current deployments are able to provide a number of capabilities including nomadic access, mobility, and roaming capability.. However the typical SP Wi-Fi solution is fairly complex. Most implementations are

based on some type of tunnel architecture from the access point to a controller/BNG. There may also be tunnels from these controllers to roaming partners and or managed services customers. These various over the top tunnels are generally implemented by several groups: a Wi-Fi group, network engineering, server operations, security and partner providers, all operating as independent silos. While existing architectures are functional and deployable, the use of SDN may simplify the architecture significantly. The use of SDN may allow a standard interoperable pipe between these service points, and more specifically, as covered in this paper, a more optimized and dynamic selection of anchor points with current architectures as well as complete elimination of anchor points.

References

1. http://cleanslate.stanford.edu
2. https://www.opennetworking.org/sdn-resources/onf-specifications/openflow
3. "Architectural approaches for integrating SP Wi-Fi in Cable MSO networks", Rajiv Asati, Sangeeta Ramakrishnan, Rajesh Pazhyannur, SCTE 2012.
4. "Proxy Mobile IPv6", http://www.ietf.org/rfc/rfc5213.txt
5. "Challenges of Unlicensed Wi-Fi Deployments: A Practical Guide for Cable Operators" SCTE 2012
6. Local IP Access and Selected IP Traffic Offload (LIPA-SIPTO) 3GPP TR 23.829
7. Control And Provisioning of Wireless Access Points (CAPWAP) Protocol Specification IETF RFC 5415