LEVERAGING OPENFLOW IN DOCSIS® NETWORKS

Chris Donley
CableLabs

Abstract

Networks are becoming virtualized. With the launch of new services and new demands on the network, operators are demanding greater flexibility, configuration consistency and control. One set of tools for meeting these demands is Software Defined Networking, specifically OpenFlow.

CableLabs, in partnership with MSOs and technology suppliers, has begun a technical exploration of how MSOs could leverage OpenFlow in a DOCSIS® environment. Our research is considering which subscriber services would see the greatest benefit from OpenFlow and how to architect OpenFlow into the DOCSIS network, specifically at the CMTS.

This paper will present findings from our research, specifically:

- *Targeted subscriber services enhanced through OpenFlow*
- *Key MSO benefits from the introduction of OpenFlow*
- *An architecture for hybrid Open Flow/DOCSIS networks*

INTRODUCTION

MSOs are expressing a growing interest in Software Defined Networking (SDN) and Network function Virtualization (NfV) technologies that have begun to emerge. These technologies promise a platform for rapid innovation and service deployment. They also promise a holistic view of the network - to be able to monitor and manage the network from a service perspective, rather than a device perspective. These trends, when fully realized, have the potential to improve operational efficiencies and accelerate the introduction of new services.

At CableLabs, we are working to enable the cable industry to capitalize on this paradigm shift towards software-controlled networks through:

- Knowledge sharing
- Architecture development & standards contributions
- Supplier readiness

This effort will contribute to a software-controlled network architecture allowing MSO resources to be configured, monitored, and optimized to improve subscriber experiences and business results.

Recently, we have analyzed MSO use cases to explore how OpenFlow, one SDN technology, could be applied in cable access networks, and the value to MSOs.

As we will discuss in this paper, our analysis found incremental enhancements to CMTS routing and forwarding for existing services such as L2/L3VPN configuration and deployment. OpenFlow also provides a step towards virtualization with traffic control features and management tools for managed firewalls and security, Carrier Grade NAT, and caching services.

This paper will also present an architecture for adding SDN/OpenFlow to cable access networks.

OPENFLOW

OpenFlow is one of the best-known SDN technologies. Developed by the Open Networking Foundation (ONF), OpenFlow specifies a two-way communication protocol by which a centralized OpenFlow controller (OFC) can add and remove “flow entries”

(forwarding instructions) on network elements such as Ethernet switches and routers. This allows software programs interacting with the OpenFlow controller to programmatically control network forwarding.

OpenFlow 'flows' are logical constructs applied to traffic matching certain (and dynamically changing) classifiers. For instance, a flow could refer to a TCP connection to a particular website, all packets from a certain MAC or IP address, all packets tagged with a particular VLAN ID, or packets arriving on the same physical interface.

Each entry in the Flow Table contains three fields:

1. A classifier (packet header) that defines the flow;
2. An action, which directs the switch processing of the flow; and
3. Flow statistics, such as the number of packets or bytes for each flow.

OpenFlow defines three actions for each flow entry:

1. Forward the flow to a given port. This may also include instructions for manipulating the header or adding encapsulation headers;
2. Encapsulate the flow and send it to the OpenFlow controller. This is typically used for the first packet in a new flow; or
3. Drop this flow's packets (e.g. to combat Denial of Service attacks).

While OpenFlow is an important technique for programmatic manipulation of traffic, it does not fully deliver the platform for rapid innovation and the holistic view of the network discussed above. In particular, it does not currently manage device provisioning; nor does it have a mechanism for directly controlling DOCSIS Quality of Service (QoS) techniques, bonding groups, or other DOCSIS-specific constructs.

OPENFLOW USE CASES

In order to identify the value provided by OpenFlow, it is helpful to analyze the protocol in the context of use cases based on services offered to cable subscribers. The use cases presented below assume all traffic forwarding is controlled by OpenFlow. As we will discuss in Section 0, it is likely that OpenFlow will first be added alongside traditional forwarding methods, rather than completely replacing traditional forwarding. However, analyzing the use cases below as pure OpenFlow use cases allows us to identify the value provided directly by OpenFlow.

In this paper, we will consider the following use cases:

- Basic Traffic Forwarding
- Traffic Optimization
- Security
- Virtual Private Networks
- Carrier Grade Network Address Translation
- Quality of Service (QoS)

Basic Traffic Forwarding

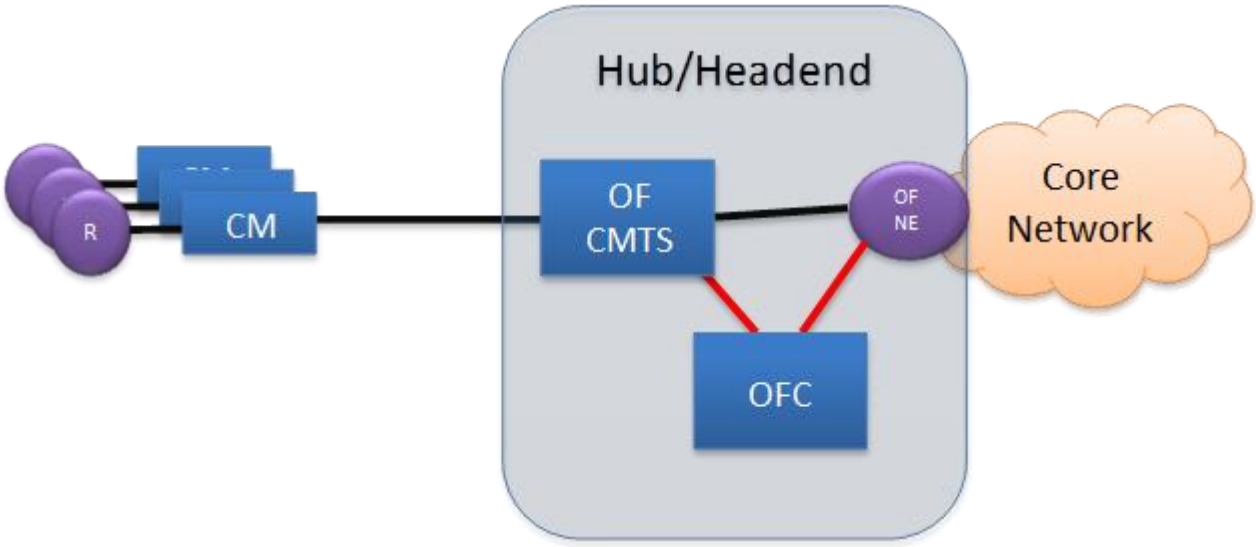


Figure 1: Basic Access Network with OpenFlow

In today's DOCSIS networks, the CMTS serves as a router. It uses routing protocols such as OSPF, ISIS, and BGP to learn the topology of the access network, serves as subscribers' default gateway, and routes traffic from subscriber devices to aggregation routers, which in turn route traffic across the MSO core network to the Internet or MSO servers.

In an OpenFlow environment, as shown in

As was described above, the initial http request for web content would be forwarded by the CMTS to the OpenFlow Controller. The OpenFlow Controller would invoke the content cache application to see if a cached copy is available. If the cache server has a copy of the requested content, it sets up a connection with the subscriber and serves the content locally. However, if the content cache does not have a copy available, or the content is too old, the OpenFlow Controller will direct the request to the content provider webserver. The response from the webserver can be sent directly to the client and also mirrored to the cache server for storage. Subsequent requests for this content would then be answered by the cache server. In the event of a cache server failure, the OpenFlow Controller would simply direct the request to the Internet.

Local caching offers several benefits to MSOs and their subscribers:

- Faster content retrieval for subscribers
- Reduced traffic on the MSO core network and transit links
- Increased content availability in the case of a webserver error or peak usage
- It facilitates the IPv4/IPv6 transition by serving as a proxy between the two protocols

OpenFlow facilitates the deployment of local cache servers by allowing MSOs to deploy the servers out-of-line, and direct specific flows to the caching servers. This would increase the overall scalability and reliability of the solution. It also allows more specific targeting of which content sites are cached, and which are not.

Lawful Intercept

Lawful interception (LI) is a telecommunications function of collecting communications network data for a Law Enforcement Agency (LEA) for the purpose of analysis or evidence. Such data generally consists of signaling or network management information and in some instances, the content of the communications. These use cases explore how OpenFlow could facilitate lawful intercept functions.

Cable Broadband Intercept Specification (CBIS) Overview

The CableLabs CBIS specification identifies the specific interface points between the MSO and the LEA that has served the Broadband Intercept Order and enumerates the specific requirements for these interface points.

CBIS Outline

The following specific interfaces and logical functions have been identified and defined (as shown in Figure 4 and Figure 5) in order to meet the Law Enforcement's (LE) objectives and high-level requirements for Broadband Intercepts related to Transparency, Confidentiality, Authentication, Validation, Non-Repudiation, Correlation, Isolation, Completeness, Compression, and Encryption.

Access Function

The access function is a site-specific means of directing data to an out-of-band interface or to a packet stream interface. The access function may be implemented as an optical tap, a UDP data stream, a port mirror, or something else that is reliable and fast enough to manage multiple streams with no packet loss.

Mediation Function

The mediation function creates hashes and formats all events, headers and packet data depending on the type of intercept. The intercepts can be of two types: full packets or packet headers only. In either case, out of band data (e.g., DHCP) packets are captured. The mediation function has interfaces to collect raw data from the access function, and store formatted data at the broadband intercept function.

Broadband Intercept Function (BIF)

The broadband intercept function includes a buffer area that is used to store 24 hours of formatted data. The BIF is an optional function for MSOs; operators may choose to implement the BIF or request that the LEA provide the BIF. The operator needs to ensure that the buffer space is sufficiently sized on an LEA-by-LEA basis.

Collection Function

The collection function provides a secure means to deliver data to LEA. It is possible for more than one LEA to have simultaneous access to such data. Some LEAs will want to set up a VPN connection, while others will use SSH and/or portable storage. For each intercept, the operator and LEA must negotiate a single common solution for the topology and protocol (e.g., IPv4 or IPv6).

CBIS Operation

When a Lawful Intercept Order is received by the operator, the CBIS data collection begins. Access to CBIS equipment is strictly controlled by law and limited to prevent disclosure of the presence of an active intercept. CBIS data collection involves the following steps:

The operator identifies the cable modem associated with the subject facility.

CPE devices are identified via the cable modem MIB tables.

CBIS equipment is provisioned to capture the traffic, either by directly using CLI or using SNMP Tables such as extractions from the dot1dTpFdbTable.

The intercepted data is forwarded via the 'Access Function' to the 'Mediation Function'.

CBIS has enough information to create identification tags for expected data streams.

Data matching the IPv4 five-tuple or IPv6 six-tuple filters are formatted at the Mediation function and then passed to 'Broadband Intercept Function'. See Figure 5; the data includes both packet data and out of band messages.

At the end of the intercept, the data is forwarded to the 'Collection Function' for collection by the Law Enforcement Agency (LEA).

OpenFlow-Facilitated Lawful Intercept

As shown in Figure 5OpenFlow can facilitate Lawful Intercept, as all new traffic flows pass through the OpenFlow Controller. This would allow MSOs to initiate Lawful Intercept from the OpenFlow Controller, rather than the CMTS.

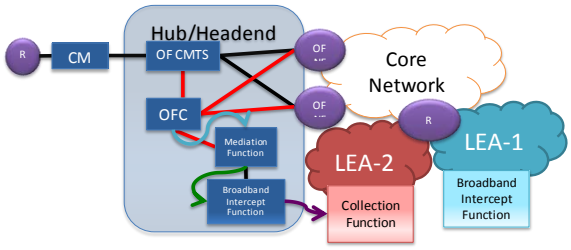


Figure 5: OpenFlow-facilitated Lawful Intercept

When a subscriber initiates a new flow, the CMTS forwards it to the OpenFlow Controller. The OpenFlow Controller will then invoke the Mediation Function to determine whether the flow is subject to an intercept order. If the Mediation Function finds that the flow subject to such an order, it directs the OpenFlow Controller to add a flow entry on the CMTS to mirrored the flow to one or more BIFs. The BIF then forwards data to the Collection Function, as today.

Using OpenFlow for Lawful Intercept provides several benefits to MSOs. First, it offers increased granularity in identifying flows, as it can look at additional fields beyond what is specified in CBIS. Second, it reduces the need to configure the CMTS for intercepts, preventing typographical errors from impacting CMTS operations. Third, it provides more control over the forwarding of "intercept flows" to one or more parties. Finally, it decouples CMTS code from Lawful Intercept updates. Should Lawful Intercept requirements change in the future, this approach allows them to be developed for a standalone application, and reduces the need for feature interaction testing on the CMTS.

Packet Inspection

OpenFlow can also be used to facilitate packet inspection. Since the OpenFlow controller only sees the first packet in every flow, it is still necessary to insert an inline Intrusion Detection System(IDS)/Intrusion Prevention System (IPS) to monitor traffic. When the IDS/IPS detects an intrusion, it notifies the OpenFlow Controller to help mitigate the attack. Depending on the nature of the attack, the OpenFlow Controller can

instruct network elements to drop new and existing flows matching the attack signature or by setting per-flow bandwidth constraints. Similarly, connection-oriented attacks such as syn-floods can be mitigated by the OpenFlow Controller itself, without requiring an IPS. Using this approach, all security controls are added on an ad-hoc per-flow basis. This means that there is no configuration change on network elements to update after the attack abates.

This approach is more flexible than the systems available today, which directly update device configuration. Since the OpenFlow approach only temporarily updates flow tables, rather than issuing configuration commands, it is less likely to cause service problems on false-positives. Also, OpenFlow can augment IDS/IPS systems by directly mitigating connection-based Denial of Service attacks.

Managed firewall

With its ability to control per-flow forwarding behavior, OpenFlow brings MSOs the opportunity to offer a managed firewall service. The subscriber or MSO would first develop a security policy, defining which traffic should be forwarded and which should be dropped. Using OpenFlow, this policy is pushed out to both sides of the network via the OpenFlow Controller – customer edge (CMTS or CM) and aggregation/peering routers. When traffic from either direction enters the network, network elements forward the first packet to the OpenFlow Controller, which can check against the firewall policy and allow or deny traffic at the edge. The OpenFlow Controller can also maintain connection state, maintaining stateful firewall capabilities.

This approach provides a new managed service opportunity for MSOs without requiring dedicated equipment on the customer site. Also, services can be configured on the fly via a web portal, enabling self-provisioning, and reducing time and effort required for service changes. Also, the distributed nature of this firewall service allows for filtering close to both edges of the network, reducing transit bandwidth for malicious traffic within the MSO network.

Virtual Private Networks

Today, Layer 2 Virtual Private Networks (L2VPNs) are delivered over cable networks according to the CableLabs L2VPN specification. Provisioning is performed using a per-VPN CM config file that sets up L2VPN service flows and classifiers, and instructs the CMTS which encapsulation type to apply. The CM classifies upstream traffic flows onto L2VPN service flows, and the CMTS encapsulates the traffic using one of a number of encapsulation headers, including 802.1Q, 802.1ad, MPLS, L2TPv3, etc. While the specification allows for multipoint support, only point-to-point is implemented. L3VPN services are not specified for cable networks, although proprietary solutions are available.

As shown in **Error! Reference source not found.**, OpenFlow provides an alternative mechanism for delivering L2VPNs and L3VPNs, and can be implemented on either the CM or CMTS. If implemented on the CM, the CM receives upstream traffic and talks to the OpenFlow Controller about the new flow. The OpenFlow Controller checks with the VPN provisioning application and responds with encapsulation parameters. The CM then applies the encapsulation directly, in a manner similar to the model used by DPoE (DOCSIS provisioning of EPON). When the traffic reaches the CMTS, it checks with the OpenFlow Controller, and forwards the encapsulated traffic based on OpenFlow

Controller directions. When the CMTS receives encapsulated traffic to be sent downstream, it checks with the OpenFlow Controller and forwards the traffic to the CM. The CM then removes the encapsulation header based on OpenFlow flow entries and forwards the traffic to the destination. This approach does not require a per-CM config file, and requires minimal CMTS involvement with the VPN; however, as the encapsulation is applied at the CM, it could cause issues with large packets, as it would add headers that could cause the packet to exceed its MTU.

If OpenFlow is not enabled on the CM, a similar approach could be used for OpenFlow VPNs at the CMTS. In this case, the CMTS would talk to the OpenFlow Controller about a new upstream flow and receive encapsulation parameters, itself. It would then encapsulate and forward the upstream traffic as directed. Likewise, the CMTS would talk to the OpenFlow Controller about encapsulated downstream traffic, and then remove the encapsulation headers and forward the traffic to the RF interface, as directed.

This OpenFlow approach offers several advantages to MSOs. First, it allows dynamic provisioning with no per-CM config files. Second, it reduces technician touches of the CMTSs, as configuration is performed on an OpenFlow application. Third, it enables selective on-demand VPNs, e.g., for a home-office or road warrior. Finally, it enables multipoint support on the CMTS with no MAC address learning required at the CMTS, and no CMTS development to support the feature.

Performance Monitoring

Today's Metro Ethernet and L3VPN customers are demanding Service Level Agreements that define delay, loss, delay variation, and availability metrics. MSOs can monitor service performance against these metrics using ITU-T Y.1731 and MEF 35 performance monitoring standards. These standards define layer 2 messages to measure 1- and 2-way delay, loss, and delay variation. For true end-to-end measurements, they require support at each UNI (User-Network-Interface); however, current CMs do not implement this functionality. Using OpenFlow, it is possible to implement Y.1731 and MEF 35 without CM implementations.

When an MSO wants to measure service performance, a technician initiates the Performance Monitoring function via a network monitoring application. This application directs the OpenFlow Controller to send a special OpenFlow pkt_out message to the CM. This message tells the CM to generate a special message and forward it out its RF interface to the remote UNI. The remote UNI then responds; when the message reaches the CM, it sends it to the OpenFlow Controller, which sends data back to the Performance Monitoring application.

This approach offers a new feature that MSOs are requesting, but is not currently available. It also does not require Y.1731 support directly in the CM; only OpenFlow support, which also offers the benefits of the other use cases described here. This approach is also extensible, as new related features and new messages such as for Service Activation Testing would not require CM or CMTS development.

Carrier Grade Network Address Translation

Today, Carrier Grade NAT (CGN) is typically implemented in a dedicated hardware appliance or blade that serves 50,000 subscribers. OpenFlow can offer

improvements to IPv4-IPv4 and IPv4-IPv6 Carrier Grade NAT by distributing the feature across multiple devices and bringing it closer to the subscriber.

As shown in Figure 8, CGN can be implemented as an OpenFlow application, rather than a standalone device. In this case, OpenFlow decouples upstream and downstream translations, and brings the feature closer to the subscriber.

In the upstream direction, the subscriber is provisioned with an IPv4 address in the 100.64.0.0/10 range and initiates a traffic flow. When the traffic reaches the CMTS, it forwards it to the OpenFlow Controller. The OpenFlow Controller communicates with the CGN application. For most flows, it then instructs the CMTS to rewrite the IPv4 source address to a predetermined CGN IPv4 address and the port to one selected using a deterministic algorithm. For flows requiring payload re-writes (e.g., VoIP), it instructs the CMTS to forward untranslated traffic to a dedicated network element with Application Layer Gateway (ALG) support to provide a better customer experience.

In the downstream direction, the process is reversed. An OpenFlow-enabled network element such as an aggregation router sends an incoming flow to the OpenFlow Controller, which then consults the CGN application, reverses the translation, and forwards the traffic to the CMTS.

This approach offers several benefits to MSOs. First, it allows deployment of the CGN feature closer to the subscriber. This lessens the latency and traffic engineering characteristic of a centralized approach. It also improves geolocation accuracy. Second, this approach does not require dedicated CGN hardware (except, potentially, for ALGs); however CGN software would be required. Third, this approach distributes translation duties across multiple devices, improving scalability. Finally, it provides an easier way than present methods for separating CGN subscribers from non-CGN subscribers.

In a similar manner, OpenFlow could also be used for IPv4-to-IPv6 NAT, a feature not available with current CGNs. This would allow IPv4-only clients (e.g. smart TVs) to talk to IPv6-only content servers without a proxy server, facilitating the transition to IPv6. This could also support inbound IPv4 services such as gaming and web hosting, if desired.

Quality of Service (QoS)

OpenFlow can be used to facilitate QoS, as shown in Figure 9. When the OpenFlow-enabled CMTS receives new subscriber-initiated traffic flows, it sends them to the OpenFlow Controller. For MSO services such as VoIP, the OpenFlow Controller directs the CMTS to set the Differentiated Services Code Point (DSCP) and 802.1p bits of the traffic flow. As traffic progresses through the network, network elements apply QoS policies as determined by the DSCP and 802.1p bits.

OpenFlow does not have a way to directly influence DOCSIS QoS on the RF network. However, when DOCSIS QoS is needed, the OpenFlow Controller can send the traffic flow to the PacketCable MultiMedia (PCMM) Policy Server (PS),

which then communicates with the CMTS and initiates DOCSIS QoS. This helps with PCMM deployment by only exposing the PS to traffic flows for which DOCSIS QoS is required.

For third-party services, OpenFlow can also help direct traffic to the nearest server using Global Server Load Balancing (GSLB). Usually, content providers use GSLB to monitor the path from their servers to subscribers and direct subscribers to the closest server; in this scenario, the MSO uses GSLB to direct subscriber traffic according to its criteria, such as the lowest latency path, lowest cost path, etc. As before, the CMTS sends new traffic flows to the OpenFlow Controller. In this case, the OpenFlow Controller invokes the GSLB application, which measures connectivity to the content provider's data centers and identifies the best path. The OpenFlow Controller then directs the CMTS to forward the subscriber's new flow to the appropriate data center according to its criteria. If necessary, the OpenFlow Controller could instruct the CMTS to rewrite the packet headers to reach the appropriate server.

This approach offers MSOs increased flexibility in the deployment of QoS. For MSO-provided services, OpenFlow sets QoS bits on traffic in the access and core network. While it cannot directly enable DOCSIS QoS, OpenFlow can interface with PCMM. This allows for more dynamic control of QOS, compared to today's deployments. OpenFlow also offers the ability to optimize the path between the subscriber and content server(s).

HYBRID OPENFLOW CMTS ARCHITECTURE

At CableLabs, we expect that MSOs will not move directly from traditional networking approaches to SDN, but rather will phase it in over time. As such, the architecture presented herein allows MSOs to leverage SDN in the access network for service agility, while preserving existing operational models for some services.

It is important to note that the architecture described below is the view of the author, and is not incorporated into any CableLabs specification, requirements document, or technical report.

The Big Picture

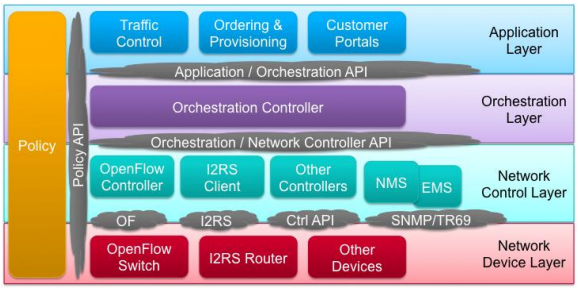


Figure 10: Emerging Network Architecture, the CMTS serves as a Layer 2 device (although, as we will discuss below, it can appear to exhibit Layer 3 behavior). When a new upstream traffic flow is initiated, the CMTS sends the first one or two packets to the OpenFlow controller, which determines where and how to forward the packet. The OpenFlow controller directs the CMTS to add a flow entry into its Flow Table, and the CMTS makes future forwarding decisions based on the flow entry. The OpenFlow Controller could also install a wildcard flow entry, matching multiple traffic types and/or destinations, to speed up the processing of aggregated flows.

This change to the CMTS' forwarding behavior would be transparent to subscribers. Subscriber devices would still use DHCP for provisioning. Devices supporting IPv4 would use DHCPv4 to acquire their default gateway router address, while devices supporting IPv6 would receive Router Advertisement (RA) messages informing them of their default

router. Such devices would send traffic to the CMTS as they do today, with no knowledge that the CMTS uses OpenFlow. The only difference is that instead of using the CMTS' IP address as the default gateway, subscriber devices would use the IP address of the aggregation router, instead.

Traffic Optimization

DNS Caching

The Domain Name System (DNS) allows for caching as a means to attain higher scalability by storing DNS information locally, instead of sending all queries to a central server. Indeed, MSOs have deployed caching name servers for years. Adding OpenFlow in the access network helps to push the DNS infrastructure closer to the subscribers, reducing demands on the core network while providing more responsive service to subscribers.

would receive the queries and direct them to the OpenFlow Controller. The OpenFlow Controller would then invoke a DNS caching application to see if it can respond to the query. If the caching application can respond, it will send a DNS response back to the subscriber. If the DNS caching application cannot respond (e.g., because it is unavailable) or if it does not have current information, the OpenFlow Controller forwards the query to the MSO's centralized DNS server. When the centralized DNS server responds, the OpenFlow Controller sends one copy to the subscriber and a second copy to the caching application.

Local DNS caching provides several benefits:

- It reduces DNS traffic on the MSO core network
- It offers faster response times for subscribers

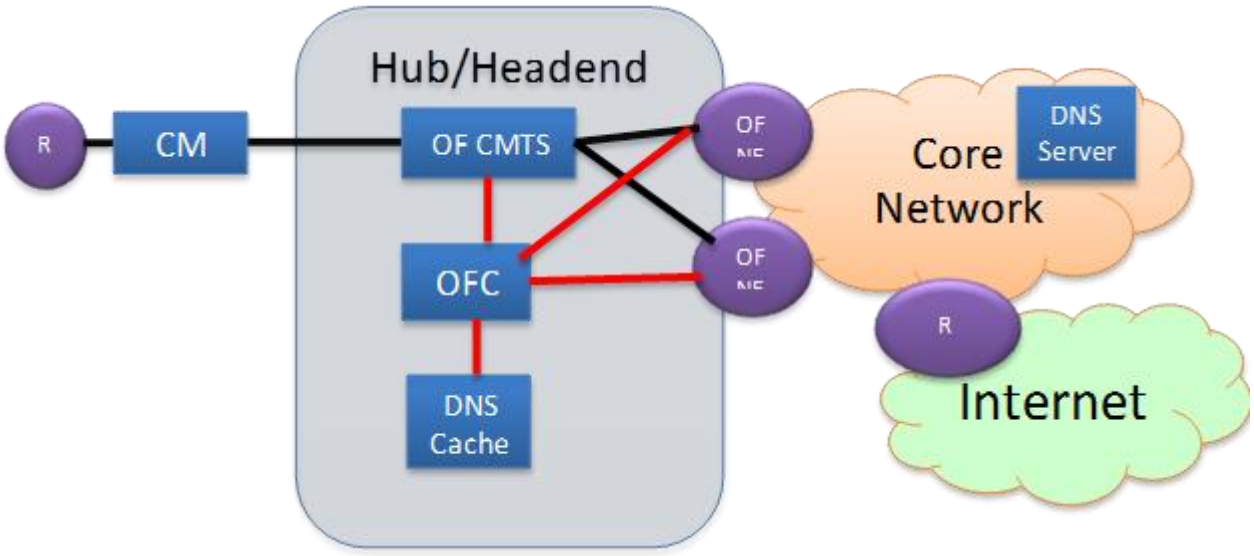


Figure 2: OpenFlow-facilitated DNS caching

OpenFlow provides additional control for serving DNS requests, as shown in Figure 2. Subscribers would generate DNS queries for services they use (e.g., www.google.com) and send them to the MSO's anycast DNS server address, as they do today. The CMTS

- It opens the possibility of tailoring the DNS response for local needs. For instance, MSOs could implement Global Server Load Balancing as part of the caching application, where the DNS

server measures availability and latency of Internet content servers and directs local subscribers to the server with the least latency for them.

OpenFlow facilitates local DNS caching by allowing MSOs to direct traffic to local servers. Without OpenFlow, MSOs use anycast addressing to direct traffic to the nearest server. This requires more complicated routing and can cause localized outages during server failures before the anycast route is withdrawn. Using OpenFlow simplifies the deployment of DNS caching servers in local networks, and can reduce the impact of local failures by forwarding a copy of the request to core servers.

Content Caching

As with DNS, content caching has been available to MSOs for some time. However, content cache servers are generally required to be deployed in-line to be able to capture and respond to http requests. They typically have to examine all traffic, whether it can be served by the cache servers or not. Also, MSOs deploying content caches need to plan for high-availability in the event of a server failure. OpenFlow facilitates the deployment of caching servers, as shown in Figure 3.

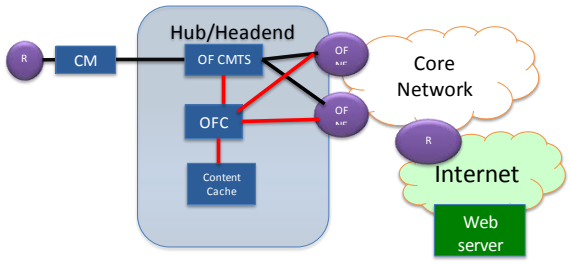


Figure 3: OpenFlow-facilitated Content Caching

As was described above, the initial http request for web content would be forwarded by the CMTS to the OpenFlow Controller. The OpenFlow Controller would invoke the content cache application to see if a cached copy is available. If the cache server has a copy of the requested content, it sets up a connection with the subscriber and serves the content locally. However, if the content cache does not have a copy available, or the content is too old, the OpenFlow Controller will direct the request to the content provider webserver. The response from the webserver can be sent directly to the client and also mirrored to the cache server for storage. Subsequent requests for this content would then be answered by the cache server. In the event of a cache server failure, the OpenFlow Controller would simply direct the request to the Internet.

Local caching offers several benefits to MSOs and their subscribers:

- Faster content retrieval for subscribers
- Reduced traffic on the MSO core network and transit links
- Increased content availability in the case of a webserver error or peak usage
- It facilitates the IPv4/IPv6 transition by serving as a proxy between the two protocols

OpenFlow facilitates the deployment of local cache servers by allowing MSOs to deploy the servers out-of-line, and direct specific flows to the caching servers. This would increase the overall scalability and reliability of the solution. It also allows more specific targeting of which content sites are cached, and which are not.

Security

Lawful Intercept

Lawful interception (LI) is a telecommunications function of collecting

communications network data for a Law Enforcement Agency (LEA) for the purpose of analysis or evidence. Such data generally consists of signaling or network management information and in some instances, the content of the communications. These use cases explore how OpenFlow could facilitate lawful intercept functions.

Cable Broadband Intercept Specification (CBIS) Overview

The CableLabs CBIS specification identifies the specific interface points between the MSO and the LEA that has served the Broadband Intercept Order and enumerates the specific requirements for these interface points.

CBIS Outline

The following specific interfaces and logical functions have been identified and defined (as shown in Figure 4 and Figure 5) in order to meet the Law Enforcement's (LE) objectives and high-level requirements for Broadband Intercepts related to Transparency, Confidentiality, Authentication, Validation, Non-Repudiation, Correlation, Isolation, Completeness, Compression, and Encryption.

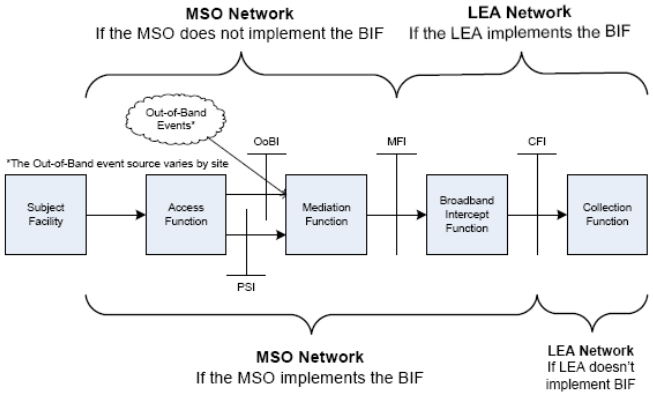


Figure 4: CBIS Broadband Intercept Interfaces

Access Function

The access function is a site-specific means of directing data to an out-of-band interface or to a packet stream interface. The access function may be implemented as an optical tap, a UDP data stream, a port mirror, or something else that is reliable and fast enough to manage multiple streams with no packet loss.

Mediation Function

The mediation function creates hashes and formats all events, headers and packet data depending on the type of intercept. The intercepts can be of two types: full packets or packet headers only. In either case, out of band data (e.g., DHCP) packets are captured. The mediation function has interfaces to collect raw data from the access function, and store formatted data at the broadband intercept function.

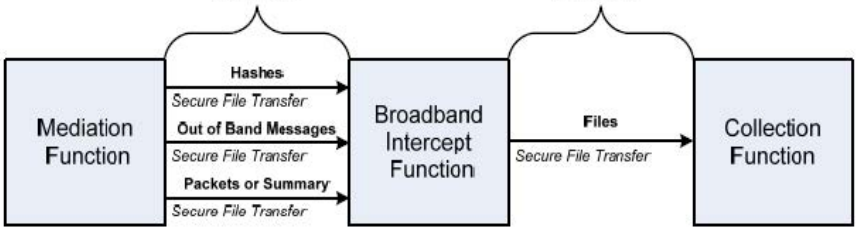
Broadband Intercept Function (BIF)

The broadband intercept function includes a buffer area that is used to store 24 hours of formatted data. The BIF is an optional function for MSOs; operators may choose to implement the BIF or request that the LEA provide the BIF. The operator needs to ensure that the buffer space is sufficiently sized on an LEA-by-LEA basis.

Collection Function

The collection function provides a secure means to deliver data to LEA. It is possible for more than one LEA to have simultaneous access to such data. Some LEAs will want to set up a VPN connection, while others will use SSH and/or portable storage. For each intercept, the operator and LEA must negotiate a single common solution for the topology and protocol (e.g., IPv4 or IPv6).

Figure 6: CBIS Logical Network



CBIS Operation

When a Lawful Intercept Order is received by the operator, the CBIS data collection begins. Access to CBIS equipment is strictly controlled by law and limited to prevent disclosure of the presence of an active intercept. CBIS data collection involves the following steps:

The operator identifies the cable modem associated with the subject facility.

CPE devices are identified via the cable modem MIB tables.

CBIS equipment is provisioned to capture the traffic, either by directly using CLI or using SNMP Tables such as extractions from the dot1dTpFdbTable.

The intercepted data is forwarded via the 'Access Function' to the 'Mediation Function'.

CBIS has enough information to create identification tags for expected data streams.

Data matching the IPv4 five-tuple or IPv6 six-tuple filters are formatted at the Mediation function and then passed to 'Broadband Intercept Function'. See Figure 5; the data includes both packet data and out of band messages.

At the end of the intercept, the data is forwarded to the 'Collection Function' for collection by the Law Enforcement Agency (LEA).

OpenFlow-Facilitated Lawful Intercept

As shown in Figure 5OpenFlow can facilitate Lawful Intercept, as all new traffic flows pass through the OpenFlow Controller. This would allow MSOs to initiate Lawful Intercept from the OpenFlow Controller, rather than the CMTS.

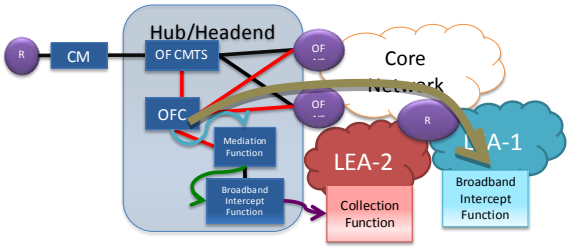


Figure 5: OpenFlow-facilitated Lawful Intercept

When a subscriber initiates a new flow, the CMTS forwards it to the OpenFlow Controller. The OpenFlow Controller will then invoke the Mediation Function to determine whether the flow is subject to an intercept order. If the Mediation Function finds that the flow subject to such an order, it directs the OpenFlow Controller to add a flow entry on the CMTS to mirrored the flow to one or more BIFs. The BIF then forwards data to the Collection Function, as today.

Using OpenFlow for Lawful Intercept provides several benefits to MSOs. First, it offers increased granularity in identifying flows, as it can look at additional fields beyond what is specified in CBIS. Second, it reduces the need to configure the CMTS for intercepts, preventing typographical errors from impacting CMTS operations. Third, it provides more control over the forwarding of “intercept flows” to one or more parties. Finally, it decouples CMTS code from Lawful Intercept updates. Should Lawful Intercept requirements change in the future, this approach allows them to be developed for a standalone application, and reduces

the need for feature interaction testing on the CMTS.

Packet Inspection

OpenFlow can also be used to facilitate packet inspection. Since the OpenFlow controller only sees the first packet in every flow, it is still necessary to insert an inline Intrusion Detection System(IDS)/Intrusion Prevention System (IPS) to monitor traffic. When the IDS/IPS detects an intrusion, it notifies the OpenFlow Controller to help mitigate the attack. Depending on the nature of the attack, the OpenFlow Controller can instruct network elements to drop new and existing flows matching the attack signature or by setting per-flow bandwidth constraints. Similarly, connection-oriented attacks such as syn-floods can be mitigated by the OpenFlow Controller itself, without requiring an IPS. Using this approach, all security controls are added on an ad-hoc per-flow basis. This means that there is no configuration change on network elements to update after the attack abates.

This approach is more flexible than the systems available today, which directly update device configuration. Since the OpenFlow approach only temporarily updates flow tables, rather than issuing configuration commands, it is less likely to cause service problems on false-positives. Also, OpenFlow can augment IDS/IPS systems by directly mitigating connection-based Denial of Service attacks.

Managed firewall

With its ability to control per-flow forwarding behavior, OpenFlow brings MSOs the opportunity to offer a managed firewall service. The subscriber or MSO would first develop a security policy, defining which traffic should be forwarded and which should be dropped. Using OpenFlow, this policy is pushed out to both sides of the network via the OpenFlow Controller – customer edge (CMTS or CM) and aggregation/peering routers. When traffic from either direction enters the network, network elements forward the first packet to the OpenFlow Controller, which can check against the firewall policy and allow or deny traffic at the edge. The OpenFlow Controller can also maintain connection state, maintaining stateful firewall capabilities.

This approach provides a new managed service opportunity for MSOs without requiring dedicated equipment on the customer site. Also, services can be configured on the fly via a web portal, enabling self-provisioning, and reducing time and effort required for service changes. Also, the distributed nature of this firewall service allows for filtering close to both edges of the network, reducing transit bandwidth for malicious traffic within the MSO network.

Virtual Private Networks

Today, Layer 2 Virtual Private Networks (L2VPNs) are delivered over cable networks according to the CableLabs L2VPN specification. Provisioning is performed using a per-VPN CM config file that sets up L2VPN service flows and classifiers, and instructs the CMTS which encapsulation type to apply. The CM classifies upstream traffic flows onto L2VPN service flows, and the CMTS encapsulates the traffic using one of a number of encapsulation headers, including 802.1Q, 802.1ad, MPLS, L2TPv3, etc. While the specification allows for multipoint support, only point-to-point is implemented. L3VPN services are not specified for cable networks, although proprietary solutions are available.

As shown in **Error! Reference source not found.**, OpenFlow provides an alternative mechanism for delivering L2VPNs and L3VPNs, and can be implemented on either the CM or CMTS. If implemented on the CM, the CM receives upstream traffic and talks to the OpenFlow Controller about the new flow. The OpenFlow Controller checks with the VPN provisioning application and responds with encapsulation parameters. The CM then applies the encapsulation directly, in a manner similar to the model used by DPoE (DOCSIS provisioning of EPON). When the traffic reaches the CMTS, it checks with the OpenFlow Controller, and forwards the encapsulated traffic based on OpenFlow Controller directions. When the CMTS receives encapsulated traffic to be sent downstream, it checks with the OpenFlow Controller and forwards the traffic to the CM. The CM then removes the encapsulation header based on OpenFlow flow entries and forwards the traffic to the destination. This approach does not require a per-CM config file, and requires minimal CMTS involvement with the VPN; however, as the encapsulation is applied at the CM, it could cause issues with large packets, as it would add headers that could cause the packet to exceed its MTU.

If OpenFlow is not enabled on the CM, a similar approach could be used for OpenFlow VPNs at the CMTS. In this case, the CMTS would talk to the OpenFlow Controller about a new upstream flow and receive encapsulation parameters, itself. It would then encapsulate and forward the upstream traffic as directed. Likewise, the CMTS would talk to the OpenFlow Controller about encapsulated downstream traffic, and then remove the encapsulation headers and forward the traffic to the RF interface, as directed.

This OpenFlow approach offers several advantages to MSOs. First, it allows dynamic provisioning with no per-CM config files. Second, it reduces technician touches of the CMTSs, as configuration is performed on an OpenFlow application. Third, it enables selective on-demand VPNs, e.g., for a home-office or road warrior. Finally, it enables multipoint support on the CMTS with no MAC address learning required at the CMTS, and no CMTS development to support the feature.

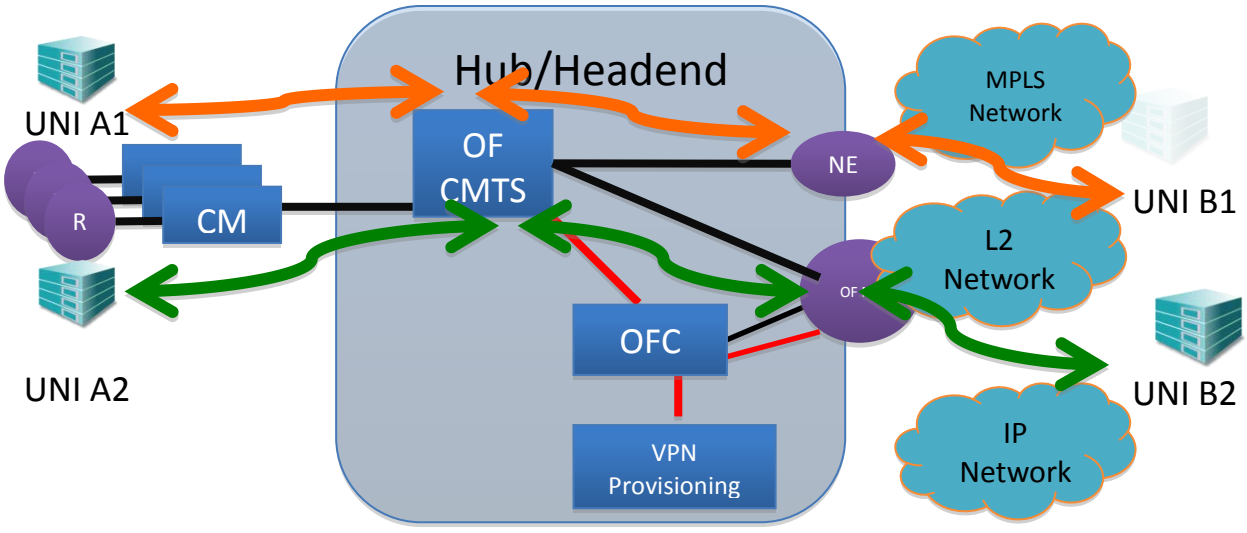


Figure 7: OpenFlow-facilitated VPNs

Performance Monitoring

Today's Metro Ethernet and L3VPN customers are demanding Service Level Agreements that define delay, loss, delay variation, and availability metrics. MSOs can monitor service performance against these metrics using ITU-T Y.1731 and MEF 35 performance monitoring standards. These standards define layer 2 messages to measure 1- and 2-way delay, loss, and delay variation. For true end-to-end measurements, they require support at each UNI (User-Network-Interface); however, current CMs do not implement this functionality. Using OpenFlow, it is possible to implement Y.1731 and MEF 35 without CM implementations.

When an MSO wants to measure service performance, a technician initiates the Performance Monitoring function via a network monitoring application. This application directs the OpenFlow Controller to send a special OpenFlow pkt_out message to the CM. This message tells the CM to generate a special message and forward it out its RF interface to the remote UNI. The remote UNI then responds; when the message reaches the CM, it sends it to the OpenFlow Controller, which sends data back to the Performance Monitoring application.

This approach offers a new feature that MSOs are requesting, but is not currently available. It also does not require Y.1731 support directly in the CM; only OpenFlow support, which also offers the benefits of the other use cases described here. This approach is also extensible, as new related features and new messages such as for Service Activation Testing would not require CM or CMTS development.

Carrier Grade Network Address Translation

Today, Carrier Grade NAT (CGN) is typically implemented in a dedicated hardware appliance or blade that serves 50,000 subscribers. OpenFlow can offer improvements to IPv4-IPv4 and IPv4-IPv6 Carrier Grade NAT by distributing the feature across multiple devices and bringing it closer to the subscriber.

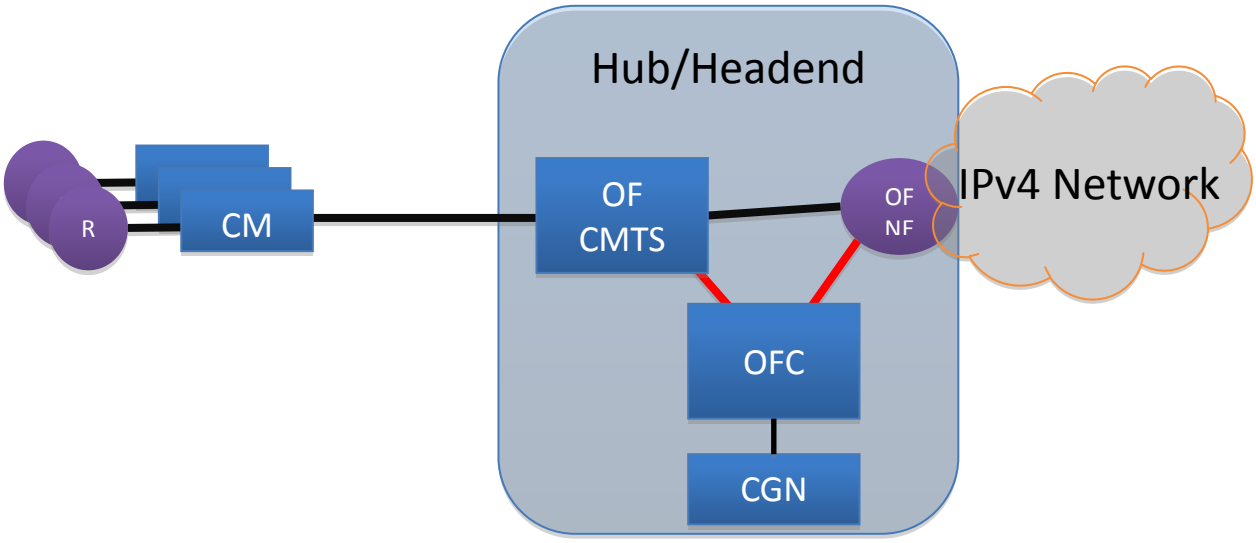


Figure 8: Carrier Grade NAT using OpenFlow

As shown in Figure 8, CGN can be implemented as an OpenFlow application, rather than a standalone device. In this case,

OpenFlow decouples upstream and downstream translations, and brings the feature closer to the subscriber.

In the upstream direction, the subscriber is provisioned with an IPv4 address in the 100.64.0.0/10 range and initiates a traffic flow. When the traffic reaches the CMTS, it forwards it to the OpenFlow Controller. The OpenFlow Controller communicates with the CGN application. For most flows, it then instructs the CMTS to rewrite the IPv4 source address to a predetermined CGN IPv4 address and the port to one selected using a deterministic algorithm. For flows requiring payload re-writes (e.g., VoIP), it instructs the CMTS to forward untranslated traffic to a dedicated network element with Application Layer Gateway (ALG) support to provide a better customer experience.

In the downstream direction, the process is reversed. An OpenFlow-enabled network element such as an aggregation router sends an incoming flow to the OpenFlow Controller, which then consults the CGN application, reverses the translation, and forwards the traffic to the CMTS.

This approach offers several benefits to MSOs. First, it allows deployment of the CGN feature closer to the subscriber. This lessens the latency and traffic engineering characteristic of a centralized approach. It also improves geolocation accuracy. Second, this approach does not require dedicated CGN hardware (except, potentially, for ALGs); however CGN software would be required. Third, this approach distributes translation duties across multiple devices, improving scalability. Finally, it provides an easier way than present methods for separating CGN subscribers from non-CGN subscribers.

In a similar manner, OpenFlow could also be used for IPv4-to-IPv6 NAT, a feature not available with current CGNs. This would allow IPv4-only clients (e.g. smart TVs) to talk to IPv6-only content servers without a proxy server, facilitating the transition to IPv6. This could also support inbound IPv4 services such as gaming and web hosting, if desired.

Quality of Service (QoS)

OpenFlow can be used to facilitate QoS, as shown in Figure 9. When the OpenFlow-enabled CMTS receives new subscriber-initiated traffic flows, it sends them to the OpenFlow Controller. For MSO services such as VoIP, the OpenFlow Controller directs the CMTS to set the Differentiated Services Code Point (DSCP) and 802.1p bits of the traffic flow. As traffic progresses through the network, network elements apply QoS policies as determined by the DSCP and 802.1p bits.

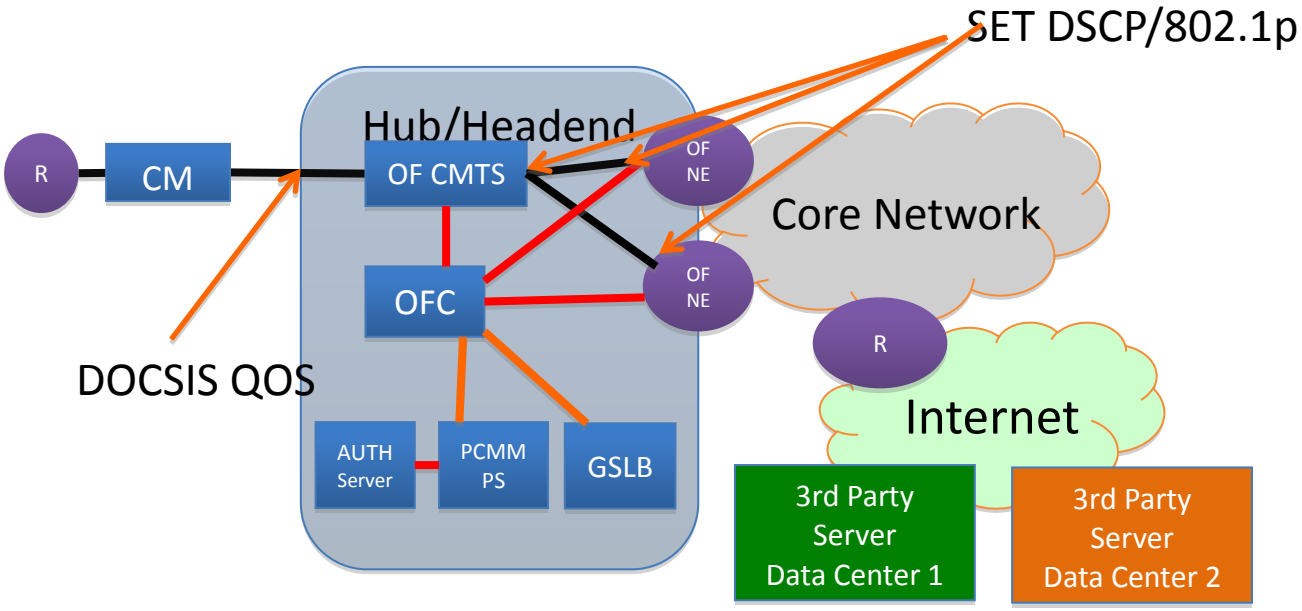


Figure 9: OpenFlow-facilitated QoS

OpenFlow does not have a way to directly influence DOCSIS QoS on the RF network. However, when DOCSIS QoS is needed, the OpenFlow Controller can send the traffic flow to the PacketCable MultiMedia (PCMM) Policy Server (PS), which then communicates with the CMTS and initiates DOCSIS QoS. This helps with PCMM deployment by only exposing the PS to traffic flows for which DOCSIS QoS is required.

For third-party services, OpenFlow can also help direct traffic to the nearest server using Global Server Load Balancing (GSLB). Usually, content providers use GSLB to monitor the path from their servers to subscribers and direct subscribers to the closest server; in this scenario, the MSO uses GSLB to direct subscriber traffic according to its criteria, such as the lowest latency path, lowest cost path, etc. As before, the CMTS sends new traffic flows to the OpenFlow Controller. In this case, the OpenFlow Controller invokes the GSLB application, which measures connectivity to the content provider's data centers and identifies the best path. The OpenFlow Controller then directs the CMTS to forward the subscriber's new flow to the appropriate data center according to its criteria. If necessary, the OpenFlow Controller could instruct the CMTS to rewrite the packet headers to reach the appropriate server.

This approach offers MSOs increased flexibility in the deployment of QoS. For MSO-provided services, OpenFlow sets QoS bits on traffic in the access and core network. While it cannot directly enable DOCSIS QoS, OpenFlow can interface with

PCMM. This allows for more dynamic control of QOS, compared to today's deployments. OpenFlow also offers the ability to optimize the path between the subscriber and content server(s).

HYBRID OPENFLOW CMTS ARCHITECTURE

At CableLabs, we expect that MSOs will not move directly from traditional networking approaches to SDN, but rather will phase it in over time. As such, the architecture presented herein allows MSOs to leverage SDN in the access network for service agility, while preserving existing operational models for some services.

It is important to note that the architecture described below is the view of the author, and is not incorporated into any CableLabs specification, requirements document, or technical report.

The Big Picture

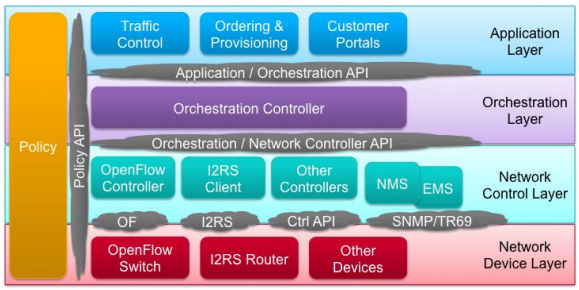


Figure 10: Emerging Network Architecture

While this paper has discussed the applicability of OpenFlow up to this point, it is important to note that OpenFlow by itself does not fulfill the promise of SDN. As shown in Figure 10, OpenFlow is one component of a larger architecture that will provide MSOs with the service agility and holistic control they will need in coming years. Other technologies are being developed to fill out the capabilities of this new architecture and to interface between new applications and different legacy and emerging network technologies.

OpenFlow is perhaps the most developed and widely researched SDN technology, however. Therefore, the next few sections describe an architecture for adding OpenFlow to the CMTS in a manner that also allows for addition of additional SDN technologies as they become available and relevant to MSOs. It deals specifically with traffic differentiation between OpenFlow and non-OpenFlow traffic and the OpenFlow forwarding model to use for the CMTS. This paper does not address topics such as redundancy or feature migration.

Traffic Differentiation

The first questions to answer is how to differentiate upstream traffic forwarding to be directed by OpenFlow from traffic to be forwarded using conventional means. There are five possible approaches to traffic segmentation at the CMTS:

1. Separate DOCSIS channels – establish separate pools of bonded RF channels for OF and non-OF traffic. Use the CM config file to direct traffic to a particular channel.
2. Separate DOCSIS Service Flows – establish a separate Service Flow for OF traffic, and use DOCSIS classifiers to direct traffic into the OF Service Flow. The CMTS processes any traffic received on an OF Service Flow using OpenFlow, and all other traffic using traditional forwarding methods.
3. Per traffic type – configure an access list in the CMTS that segments traffic by destination port, with some well-known ports processed using OpenFlow and others using traditional methods.
4. Per Source IP or MAC address – configure an access list in the CMTS

that segments traffic by source IP or MAC address, with traffic from predetermined source addresses processed using OpenFlow, and all other traffic processed using traditional methods.

5. Sequentially – the CMTS checks the OpenFlow Flow Table first, then the CMTS FIB second. This means that all traffic flows are first sent to the OpenFlow controller; if the OpenFlow Controller sets up a Flow Table entry, additional traffic from that flow is processed

We believe that reserving separate DOCSIS RF channels for OpenFlow traffic is economically infeasible. Also, sequential processing increases latency, as every traditionally-managed flow would need to be processed by OpenFlow first. Therefore, we concentrate our analysis on applying the use cases described above to the remaining three approaches, as shown in

. Approaches well-matched to the use cases receive a (✓); approaches partially matched to the use case, or that can only work in limited circumstances receive a (~); and approaches unsuitable for the use case receive a (✗).

Table 1: Viability of traffic classification methods

Use Case	*Traffic Type*	*Per-SF*	*Per-Port*	*Per-IP/MAC*
DNS Caching	DNS	✓	✓	✗
Content Caching	Varies; primarily http, https	✓	✓	✗
Lawful Intercept	All	✗	✗	✓
IDS/IPS	All	✓	~	✓
Managed Firewall	All	✓	~	~
L2VPN/L3VPN	All	✓	✗	~
Carrier Grade NAT	All Primarily http/https and DNS	✓	~	✓
QoS	VoIP: SIP and RTP Video: http Gaming and other OTT services: varies	✓	~	~

As shown above, per-service flow separation of OpenFlow and traditionally managed traffic appears to fit most use cases, except Lawful Intercept. As service flow establishment would send a message to the CM that could be observed by a subject under an intercept order, alternative approaches (particularly per-source MAC or IP address) would be required to initiate Lawful Intercept without notifying the subject.

One way to take advantage of service flow-based classification, while preserving flexibility for additional use cases such as Lawful Intercept, is a hybrid approach shown in **Error! Reference source not found.**. Traffic arriving on service flows identified as OpenFlow service flows are processed directly by the OpenFlow flow table. Traffic arriving on other service flows can be sent to an Access Control List (ACL), which directs specific flows to the OpenFlow flow table and all remaining traffic to the CMTS Forwarding Information Base (FIB). This approach satisfies all of the use cases described above, and offers MSOs significant flexibility to experiment with OpenFlow in the lab and field trials.

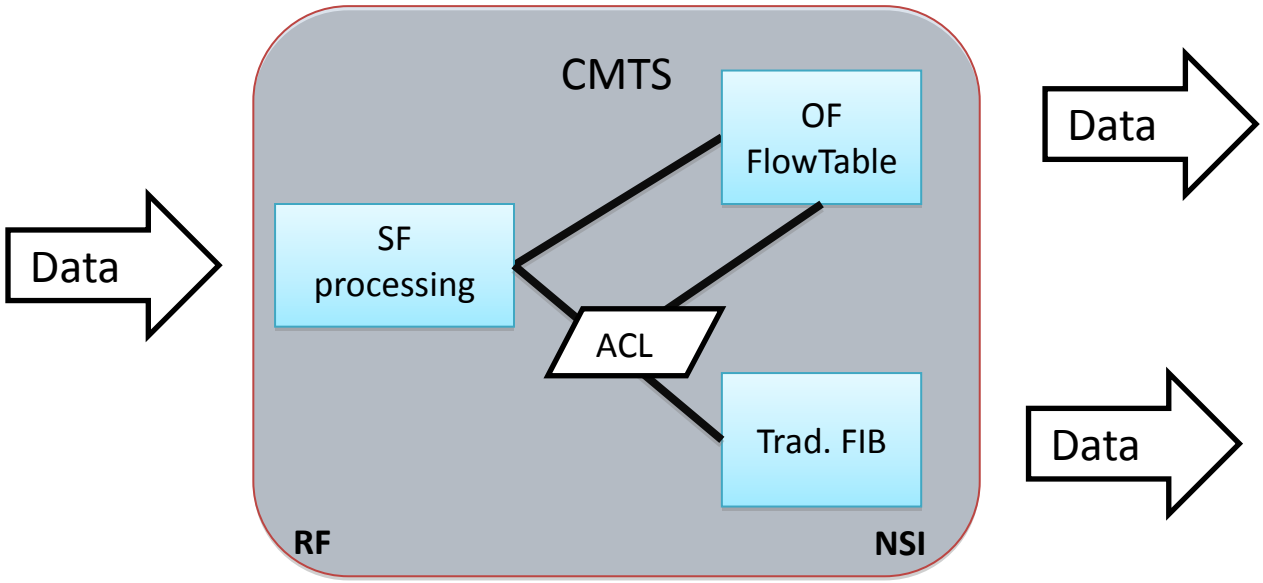


Figure 11: Hybrid SF-based Traffic Separation

Traffic Forwarding

CableLabs identified two hybrid OpenFlow CMTS models, one referred to as the "L2 Model", and the second as the "L2/L3 Model". Both are described below.

L2 Model

In the L2 model, the CMTS behaves solely as a Layer 2 device. Within a headend or hub site, there is a common Layer 2 domain for all DOCSIS interfaces on all CMTSs. Subscriber devices are provisioned from the same IPv4 subnet/IPv6 prefix, and they receive the address of the aggregation router (not the CMTS) as their default gateway.

When subscriber traffic arrives at the CMTS, the CMTS talks to the OpenFlow Controller and installs flow entries based on load balancing, failover, traffic control, premium services, and other factors that tell it where to forward traffic. In order to reduce MAC learning on the routers, the CMTS transforms the Ethernet header on upstream flows to use its source MAC address, rather than the subscriber device. In

order to keep broadcast and multicast traffic rates low in the access network, the OpenFlow Controller can either direct IPv6 Neighbor Discovery (ND) and IPv4 ARP messages directly to their target nodes, without flooding the network, or suppress them on the access network and respond with pkt_out messages directing the CMTS to generate the messages locally.

In the downstream direction, routers add ARP and ND entries mapping the subscriber IP address to the CMTS MAC address. Routes to subscriber IPv6 prefixes would be mapped to the appropriate customer router, and traffic directed to the respective CMTS. As in the upstream direction, the CMTS would remap downstream flows to point to the subscriber MAC address.

This approach offers MSOs several benefits. First, it allows MSOs more granular control over traffic forwarding in the access network for load balancing, failover, and traffic control. It also offers the possibility of separate paths for premium services. Second, OpenFlow provides operational benefits such as eliminating the need for IP address renumbering during node splits and reducing the need for routing protocols in the CMTS.

L2/L3 Model

In the L2/L3 model, the CMTS behaves like a Layer 3 device, as it does today. During provisioning, each CMTS is assigned a different subnet for each RF interface. Subscriber devices are then provisioned to use the CMTS as the default gateway router. As it does today, the CMTS would be responsible for sending IPv6 Router Advertisement and IPv4 ARP messages. However, the CMTS makes traffic forwarding decisions based on OpenFlow, rather than traditional methods.

When the CMTS receives a new subscriber flow, it talks to the OpenFlow Controller to learn where to direct the flow and how to transform Ethernet headers. In this case, the transformation looks like the routing process. The CMTS changes the Ethernet source and destination addresses from Subscriber MAC:CMTS MAC to CMTS MAC:Router MAC, where the router MAC is the MAC address of the router selected by the OpenFlow Controller as the next hop. As discussed above, router selection could be based on load balancing, failover, traffic control, and premium services.

In the downstream direction, routers can either use traditional routing or OpenFlow to direct traffic to the correct CMTS for forwarding to subscribers. When the flow reaches the CMTS, it again changes the Ethernet headers from Router MAC:CMTS MAC to CMTS MAC:Subscriber MAC .

This model shares many of the same values as the L2 Model. In addition, this model reduces the size of the router's ARP and ND tables compared to the L2 model, enhances the scalability due to the use of subnets, and provides an easier transition path for traditional CMTSs. A comparison of the two models is included in

Table 2 .

Table 2: Comparison of L2 and L2/L3 Models

Values	L2	L2/L3
Load balancing, failover, traffic control, premium services	X	X
Reduces need for routing protocols on each CMTS	X	X
No network renumbering during node splits	X	X
Reduced MAC learning at Routers (Hybrid OF/non-OF routers)		X
Enhanced scalability due to subnets		X
Easier transition for existing CMTSs		X

Implications for Cable

Both the L2 and L2/L3 models are viable for the CMTS. Regardless of the model, OpenFlow CMTSs offer benefits such as load balancing, failover, traffic management, and premium service support. Also, both models are capable of addressing the use cases described above. However, because of the simpler transition path for existing CMTSs, I recommend use of the L2/L3 model for initially phasing in OpenFlow support.

CONCLUSION

As we have discussed, OpenFlow is one piece of the overall SDN puzzle. As the most fully-developed SDN technology, and as the one with the most exposure, it is important to identify its place in cable networks.

The use case analysis described above identified several benefits OpenFlow brings to cable. First, it provides incremental enhancements to existing services such as L2VPN and lawful intercept. Second, as a step towards virtualization, it provides traffic control features and management tools. These enhancements were evident in the managed firewall, Carrier Grade NAT, IDS/IPS, and caching use cases.

Adding OpenFlow to an existing CMTS requires a mechanism to differentiate OpenFlow from non-OpenFlow traffic and a forwarding model. The hybrid service flow approach for traffic differentiation described above best fits the use cases discussed in this paper. Traffic on a specially marked service flow, or from a defined source MAC/IP address is processed by OpenFlow, while other traffic is forwarded using traditional methods. Also, the L2/L3 forwarding model

provides a way to introduce OpenFlow to existing networks without disrupting established services.

REFERENCES

CableLabs BSoD L2VPN Specification, CM-SP-L2VPN-I10-121004, October 4, 2012, Cable Television Laboratories, Inc.

DOCSIS Cable Broadband Intercept Specification CM-SP-CBI2.0-I04-110224, February 24, 2011, Cable Television Laboratories, Inc.

ITU-T Recommendation Y.1731 (07/11), OAM functions and mechanisms for Ethernet based

networks.

MEF Technical Specification 35: Service OAM Performance Monitoring Implementation

Agreement, April 2012.

OpenFlow: Enabling Innovation in Campus Networks. Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, Jonathan Turner. March, 2008