AN ARCHITECTURE FOR SOFTWARE DEFINED WIRELESS NETWORKING

CARLOS J. BERNARDOS, ANTONIO DE LA OLIVA, PABLO SERRANO, ALBERT BANCHS, LUIS M. CONTRERAS, HAO JIN, AND JUAN CARLOS ZÚÑIGA

Carlos J. Bernardos, Antonio de la Oliva, and Pablo Serrano are with Universidad Carlos III de Madrid.

Albert Banchs is with Universidad Carlos III de Madrid and IMDEA Networks Institute.

Luis M. Contreras is with Telefónica I+D.

Hao Jin and Juan Carlos Zúñiga are with InterDigital Communications, Inc.

ABSTRACT

Software defined networking, characterized by a clear separation of the control and data planes, is being adopted as a novel paradigm for wired networking. With SDN, network operators can run their infrastructure more efficiently, supporting faster deployment of new services while enabling key features such as virtualization. In this article, we adopt an SDN-like approach applied to wireless mobile networks that will not only benefit from the same features as in the wired case, but will also leverage on the distinct features of mobile deployments to push improvements even further. We illustrate with a number of representative use cases the benefits of the adoption of the proposed architecture, which is detailed in terms of modules, interfaces, and high-level signaling. We also review the ongoing standardization efforts, and discuss the potential advantages and weaknesses, and the need for a coordinated approach.

INTRODUCTION

The telecommunications sector is experiencing a major revolution that will shape the way networks and services are designed and deployed for the next decade. We are witnessing an explosion in the number of applications and services demanded by users, which are now really capable of accessing them on the move. In order to cope with such demands, some network operators are now following a cloud computing paradigm, enabling the reduction of overall costs by outsourcing communication services from specific hardware in the operators' core to server farms scattered in data centers. These services have different characteristics from conventional IT services that have to be taken into account in this cloudification process [1].

Virtualization of functions also provides operators with tools to deploy new services much faster when compared to the traditional use of monolithic and tightly integrated dedicated machinery [2]. As a natural next step, mobile network operators need to rethink how to evolve their existing network infrastructures and how to deploy new ones to address the challenges posed by the increasing customer demands, as well as by the huge competition among operators. All these changes are triggering the need for modification in the way operators and infrastructure providers operate their networks, as they need to significantly reduce the costs incurred in deploying a new service and operating it.

Some of the mechanisms that are being considered and already adopted by operators include sharing of network infrastructure to reduce costs, virtualization of core servers running in data centers as a way of supporting their load-aware elastic dimensioning, and dynamic energy policies to reduce monthly electricity bills. However, this has proven to be tough to put into practice, and not enough. Indeed, it is not easy to deploy new mechanisms in a running operational network due to the high dependence on proprietary (and sometimes obscure) protocols and interfaces, which are complex to manage and often require configuring multiple devices in a decentralized way.

Building on the revolutionary forward thinking in computer networking, software defined networking (SDN) is currently being considered as an alternative to classic distributed approaches based on highly specialized hardware executing standardized protocols. Until now, most of the key use cases used to present the benefits of the SDN paradigm have been limited to wired environments (e.g., Google uses SDN in its data centers [3]).

In this article we analyze the potential of applying the SDN paradigm to mobile wireless networks. First, we identify use cases where a wireless SDN approach could bring additional benefits. Then we derive the main characteristics of a wireless SDN mobile operator's architecture, paying special attention to the main functions and interfaces. In order to illustrate the operation of the proposed mobile wireless SDN framework, we introduce the high-level interactions required between the defined functions to support an example use case of interest to operators. We finish this article with a review of current standardization efforts and trends in this arena, and then elaborate on the need for specific actions toward

 1536-1284/14/$25.00 © 2014 IEEE

the standardization of what we call software defined wireless networking (SDWN).

Background: SDN, OpenFlow, CAPWAP, and Reconfigurable Wireless Devices

Software defined networking is a networking paradigm [4] that separates the control and data forwarding planes. Such separation allows for quicker provisioning and configuration of network connections. With SDN, network administrators can program the behavior of both the traffic and the network in a centralized way, without requiring independently accessing and configuring each of the network's hardware devices. This approach decouples the system that makes decisions about where traffic is sent (i.e., control plane) from the underlying system that forwards traffic to the selected destination (i.e., data plane). Among other advantages, this simplifies networking as well as the deployment of new protocols and applications. In addition, by enabling programmability on the traffic and the devices, an SDN network might be much more flexible and efficient than a traditional one.

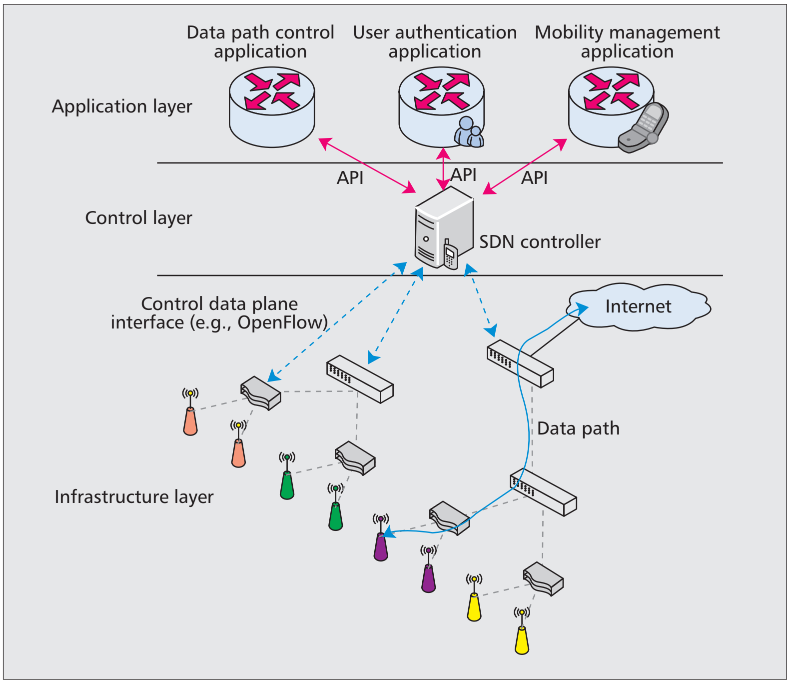

Figure 1. SDN reference architecture.

Figure 1 shows a logical view of the commonly accepted SDN reference architecture [4]. In this architecture, the intelligence is centralized in software-based SDN controllers, which have a global view of the network and are capable of controlling, in a vendor-independent way, the network devices. These network devices are no longer required to implement and understand many different network protocol standards; instead, they can provide such functionality by accepting instructions from SDN controllers. This saves a lot of time and resources, as the network behavior can easily be controlled by programming it in the centralized controllers rather than using custom configurations in many different devices scattered across the network.

A key requirement to deploy an SDN architecture, such as the one defined above, is to standardize the interface to control the mobile devices. This can be done with OpenFlow [5], which is a standardized interface between the control and forwarding layers of the SDN architecture. The vendor-agnostic nature of OpenFlow facilitates the integration of heterogeneous devices in a common way, simplifying the operation of multi-vendor infrastructures, which are typically found in commercial telecom networks. It allows the forwarding plane of network devices such as switches and routers to be accessed and modified by the definition of specific rules for matching packet flows against a selection of layer-2 to layer-4 packet header's field values and the flows' ingress port number. These rules take the form of entries in forwarding flow tables residing in the network devices.

It should be noted that this separation of the control and data planes for the switching fabric also exists, to some extent, in the wireless domain. Indeed, the Internet Engineering Task Force (IETF) standardized several years ago the Control and Provisioning of Wireless Access Points (CAPWAP) protocol [6], which centralizes the control in wireless networks. In principle, CAPWAP is technology-agnostic and requires specific bindings for each considered access standard, although so far only the binding for 802.11 has been defined. Radio configuration is expressed in terms of management information base elements included in the standard, such as the operating channel or the transmission power, but also the beacon interval or contention parameters used by the medium access scheme. With CAPWAP, control frames are delivered to a central controller, which is responsible for medium access control (MAC) layer control, in a way that can easily be related to the way OpenFlow delivers to the controller information about new incoming flows.

Along the same lines, but in a more visionary approach, a novel paradigm for the reprogramming of wireless interfaces has been proposed in [7]. In this vision, wireless nodes execute a wireless MAC processor in charge of running "MAClets" (i.e., programs specifying the MAC protocol). In this way, the central controller can dynamically upload the protocol to use at a given point in time, for example, changing from carrier sense multiple access with collision avoidance (CSMA/CA) to time-division multiple access (TDMA)-based access when the traffic load increases.

There have also been some efforts looking at the use of SDN in mobile networks, such as MobileFlow [8], which proposes an SDN approach for the core network. However, the proposed solution does not provide an integrated vision including wireless access.

While efforts on SDN so far have mostly focused on wired and core networks, we believe that the adoption of a similar concept for wireless access and backhaul environments can be even more beneficial. Indeed, the control plane of wireless networks is more complex than that of wired networks, and therefore higher gains

The QoE of a service is determined by roughly three factors: the service architecture (e.g., server capabilities, caches, and their location), the core network performance, and the service provided at the "wireless last mile," that is, the combination of the wireless link(s) — including the backhaul — and the capabilities of the terminal.

can be achieved from the increased flexibility provided by an SDN approach.

USE CASES

Before addressing the design of an SDWN architecture, in this section we describe some use cases in which the adoption of an SDWN approach brings significant advantages. Rather than identifying an exhaustive set of use cases, the purpose of these use cases is simply to illustrate some of the potential advantages of adopting an SDN approach in a mobile wireless environment.

VIRTUALIZATION

Current deployments support virtualization to some extent, but existing network devices and mechanisms are not designed to support the dynamic reconfiguration required for timely and efficient sharing of resources. More specifically, although servers can be virtualized and allocated to different physical resources in almost real time, the paths communicating these virtual appliances with the rest of the world still require manual installation or interaction with protocols that are not designed for dynamic and fast responses. Hence, current adaptation mechanisms in network technologies are seen as one of the bottlenecks for the general deployment of virtualized infrastructures. Previous approaches for core network virtualization (e.g., PlanetLab[1] or GENI[2]) have implemented different overlay networks so that researchers can run their experiments by time-sharing access to resources. However, these approaches operate on a coarse timescale, need manual planning and dimensioning, and lack the required timeliness to operate in production networks. These approaches differ from actual trends in virtualization for the access network, more focused on sharing and enforcing the radio and transport network resources among different operators.

Following this trend, the adoption of SDN (with relatively mature technologies such as OpenFlow or ForCES[3]) should improve support for timely and efficient virtualization of a wireless network, but there are some challenges that need to be successfully tackled. First, in order to provide the required flexibility in terms of network topology and architecture demanded by virtualization applications, an SDN network must be able to implement a wide set of control logics that are simultaneously applied to the same set of physical resources. The support of several different control logics on the same network raises scalability and compatibility issues. For example, each of the control logics may be working on top of a different realization of the network, each of them requiring fast reaction upon changes in the underlying physical infrastructure. This challenge calls for a scalable network orchestration mechanism that coordinates SDN control and data plane operations, and resolves any contentions between different control logics.

In addition, the different control logics must be able to work in an isolated way. This must be implemented in two different planes. On one hand, traffic from a certain instantiation of a virtual operator must be isolated from traffic belonging to a different virtual operator for security and privacy reasons. On the other hand, changes performed in the virtual infrastructure of an operator must be isolated from the rest of the virtual instances sharing the network; for example, a change in the configuration of the virtual infrastructure must not affect the rest of the instances running on top of the real deployment.

Another key issue is the allocation and sharing of network resources, considering both time resolution and isolation. This not only prevents resource wastage due to the coordination and sharing, but also allows gaining from statistical multiplexing, so the use of this solution is substantially more efficient than having independent deployments.

Note that SDN is independent from and complementary to another notable virtualization initiative, network functions virtualization (NFV) [2]. While SDN focuses on the virtualization of network devices, NFV aims to enable the virtualization of network services and functions, such as NAT, firewall, and cellular core functions so that the time to deploy services can be shortened, and operator capital/operational expenses (CAPEX/OPEX) can be reduced. One example of the mutual benefit between NFV and SDN is that on one hand, NFV may improve the efficiency and flexibility of SDN's control plane services; on the other hand, SDN may ensure the delivery and quality of the network traffic between NFV's virtualized functions.

QOE-AWARE NETWORK OPERATION

Current networks are provisioned and operated toward providing a certain level of quality of service (QoS). However, this does not always ensure a minimum quality of experience (QoE) to the user. The QoE of a service is determined by roughly three factors: the service architecture (e.g., server capabilities, caches, and their location), the core network performance, and the service provided at the "wireless last mile," that is, the combination of the wireless link(s) — including the backhaul — and the capabilities of the terminal. With current architectures, a service provider has to anticipate its needs in terms of infrastructure, negotiate a service level agreement (SLA) based on these estimations, and at most try to adapt to users' experiences on a coarse timescale, for example, changing the encoding of the video being served (as YouTube does). It is clear that such a scenario precludes efficient use of resources, as the service provider has no mechanism to react in a timely manner to the changing conditions because the service provider has limited indicators of the user's performance in real time and no ability to quickly deploy more architectural elements or improve the SLA with the network providers.

Furthermore, mobile networks intrinsically present a need to integrate QoS objectives in the radio part (i.e., service layer) and backhaul network (i.e., transport layer) [9]. This drives the necessity of dynamically orchestrating resources in both layers to provide a uniform and efficient QoE.

The use of an SDWN architecture would

[1] https://www.planet-lab.org/

[2] http://www.geni.net/

[3] http://datatracker.ietf.org/wg/forces/charter/

allow the network to offer the service provider an application programming interface (API) to control how the networks behave to serve traffic that matches a certain set of rules (of course, the degree of control would depend on the agreements between network operators and service providers, and the kind of requested control). Furthermore, through this API the provider is also able to dynamically change the forwarding paths of the flows (in both directions), so traffic traverses opportunistically deployed middleware, which can serve, for instance, as data caches or video transcoders. Finally, the provider, now acting as a true service composer, can use the API to change the behavior of the wireless last mile in three ways: first, by dynamically prioritizing traffic at the last hop, so in case of poor wireless conditions, some packets (e.g., I frames of a video stream) are provided with better service than others (e.g., B frames), because they are marked as more important; second, by being aware of the service experienced by the user, thus timely adapting following his/her preferences; and third, by supporting traffic on- and off-loading based on these preferences and the availability of different communication links (each with a different performance vs. cost trade-off). In this way, the provider personalizes the operation of the network after the user's behavior and preferences.

The above requires the design of customer-oriented traffic management services, which are able to connect different applications to different technologies, and adapt content to network conditions and available resources. By considering the specifics of each application, a differentiated service can be provided to different flows. For example, quality and/or timely delivery of a given flow can be balanced against for extra capacity for a different flow with more stringent delay requirements. This requires the ability to design and coordinate offloading/seamless mobility mechanisms across the heterogeneous (technology and operator) networks.

Network Access Selection and Mobility Control

Existing mobile terminals are generally equipped with multiple network interfaces, typically WiFi and cellular. This, together with the proliferation of femtocells, WiFi hotspots, and WiFi access to fixed residential home gateways, has complicated the process of selecting the best access technology at each moment. A mobile node may decide to use the available attachment options sequentially (i.e., move all traffic from one technology to another) or simultaneously (i.e., move selected flows from one access to another [10]). Despite the fact that telecom operators can offer both residential fixed and mobile services, WiFi hotspot accesses are usually not directly managed by the operator, and their characteristics make it challenging to ensure a given QoS (as opposed to cellular accesses). Because of these factors, the decision of how to select and opportunistically use the heterogeneous available access is not a trivial one.

Additionally, the network might want to keep control over how the traffic (on an application granularity level) is delivered to the mobile terminal. This involves, for example, selective and opportunistic traffic offloading. By using an SDWN solution, an API could be offered to external parties (e.g., service providers) so they can influence the decision of which access technology is used to deliver a certain type of traffic to a specific mobile terminal or group of users. This particular scenario could also benefit from enabling the programmability of the mobile node as well, for example, to enable easy control from the network side on how the different available network accesses are used by the traffic generated by applications running on the mobile.

SDWN Architecture

We next describe a generic SDWN architecture. Such an architecture aims to bring the benefits of logical orchestration by providing well defined interfaces for control plane functions and enabling richer flexibility in user plane traffic handling.

System View

Figure 2 shows an SDWN-based architecture of a mobile network operator, where a solid line in the figure denotes a user plane connection, and a dashed one is used for the control plane. We take the 3GPP Evolved Packet System as a reference architecture to link the proposed concepts with a well established and understood system architecture.

A mobile network typically exhibits multiple heterogeneous radio access networks (RANs) connected to a common transport core network. Note that the connection between the last network entity providing radio access and the core transport network might involve a wired or wireless backhaul network (shown as part of the RAN in the figure) by using a combination of technologies (e.g., fiber optic, microwave) and topologies (e.g., ring structure, daisy chain) in the backhaul segment. Three well-known examples of RANs are shown in Fig. 2: the UTRAN (for UMTS), E-UTRAN (LTE), and a WiFi hotspot. However, it should be noted that the proposed architecture is generic enough to support other RAN technologies as well, both already existing ones (e.g., WiMAX) or future ones.

In the SDWN architecture, RANs are enhanced with programmability (as introduced in more detail below), supporting multiple functionality levels to allow for incremental deployments. The core transport is composed of programmable L2 switches and L3 routers, allowing setup of unicast and multicast forwarding at the flow level (as supported, e.g., by OpenFlow). Multiple (virtual) operators might share part of the radio, backhaul, and transport core network, which requires the interconnection of the core control plane entities — in charge of functions such as authentication, authorization, charging, subscriber management, mobility management, QoS provisioning, and connection to external services/networks — with the programmable network.

Two different models can be adopted to implement an SDWN architecture: "evolutionary" and "clean slate." The evolutionary model

Existing mobile terminals are generally equipped with multiple network interfaces, typically WiFi and cellular. This, together with the proliferation of femtocells, WiFi hotspots, and WiFi access to fixed residential home gateways, has complicated the process of selecting the best access technology at each moment.

The API offered by the SDN controller supports different access levels to the external parties, so personalization can vary on different dimensions: per application, per user, per (virtual) operator, per access network, or a combination of them.

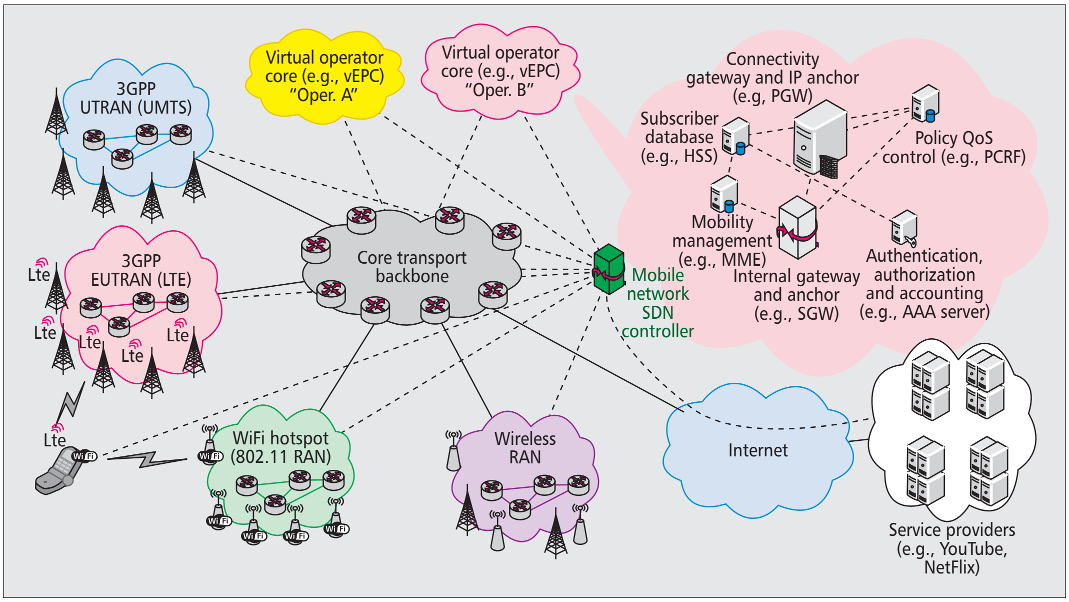

Figure 2. SDN-based mobile network architecture.

allows for incremental deployment in existing networks: legacy control plane entities from operators can connect to the transport core network without modifying the existing interfaces. In this model, the SDN controller implements standardized interfaces to support the internetworking with existing legacy entities even if they run on a virtualized environment (what is known as virtual EPC, vEPC).

In the clean slate model, the control plane functions are directly programmed on the SDN controller or on top of it as applications, using a software API between the virtual operators and the SDN controller. While this approach does not allow for easy incremental deployment, it brings from day one all the advantages of programmable network architectures. For example, the deployment of new network functions and services is much easier and faster, as it can be directly implemented on the controller and does not have an impact on multiple interfaces and equipment from different vendors. We can just take the simple, but very representative, example of IPv6 support on a mobile network. With the clean slate approach, adding IPv6 support would just require additional code on the SDN controller, compared to defining new interfaces and procedures on the different control and user plane entities, which require software/firmware updates (if not even replacing some hardware).

The brain of the architecture, the SDN controller, is connected to each programmable entity. Note that the SDN controller is a logical entity, which might also be decentralized into different physical boxes to improve scalability and performance, although this is currently the subject of extensive research [11].

In order to allow third parties (e.g., service and application providers) to influence/control the behavior of the network, an API is enabled. This API effectively enables external players to get access to the network resources, similar to what an operating system (OS) does with the access of applications to computational resources and peripherals. The API offered by the SDN controller supports different access levels to the external parties, so personalization can vary on different dimensions: per application, per user, per (virtual) operator, per access network, or a combination of them.

Key Interfaces

We now focus on the description of the different interfaces (Fig. 3):

- A northbound interface to the (virtual) operators sharing the same physical set of network resources allowing them to dynamically change the share of resources, for example, to adapt to network load or to the number and profile of users attached to the physical shared network at any given moment of time. This interface should be able to implement richer SLAs than the ones available nowadays, as a more dynamic and almost-real-time reconfiguration of the network would be possible. Each (virtual) operator should have access to an abstracted view of its assigned resources so they can program that "virtual" network as a physical one.
- A northbound interface to the external parties (service and application providers) authorized to influence the network behavior. This interface should be properly secured, granting access with different granularities and permissions. The interface should be powerful enough to allow an application provider to influence how its traffic is handled, even taking into consideration the virtual operator from which its users are getting access. Note that this is possible because of the centralization achieved by the use of the SDN approach, although this may introduce scalability issues (up to per flow signaling, need for frequent network monitoring, etc.) that need to be taken into account.
- A southbound interface to the physical user plane network entities in the core transport backbone. This interface is used by the SDN controller to implement the different behavior policies according to requests from external par-

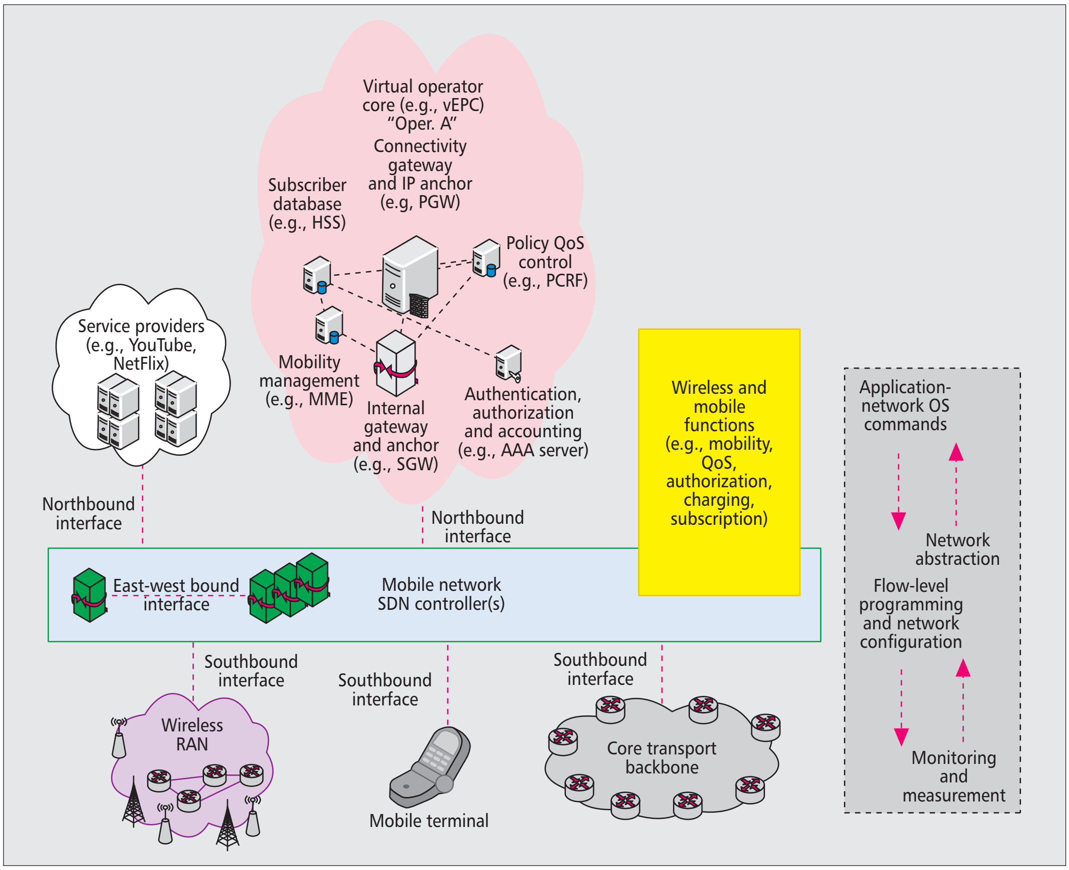

Figure 3. SDN-based mobile network interface architecture.

The logical centralization of SDWN offers the ability to change the forwarding of user data traffic in the access (radio and backhaul) and core transport networks, which can be used to provide functionalities such as mobility management and QoS provisioning.

ties, the virtual operators associated to the different users attached to the network, and the network conditions. Given the logical centralization provided by SDN, close to maximum utilization of the capacity of the network links can be achieved. This interface also allows for effective sharing of a common backbone and backhaul network by different operators, which may even connect to the Internet via different gateways.

- A southbound interface to the physical user plane entities in the RAN. This interface allows for effective virtualization of the access network, therefore sharing the same physical resources among different operators. Besides, this interface should allow programming the wireless access technologies to provide the expected behavior, depending on the specific needs and characteristics of the mobile terminal, the requests from the external providers, and the different SLAs the virtual operators may have in place with their users.
- A southbound interface with the mobile node. This interface provides the network with certain programmability capabilities on the mobile node. This can be used, for example, to improve the mobility experience by better exploiting the simultaneous use of available wireless access networks (e.g., helping in access network and interface selection).

A proper implementation of the above interfaces, together with the required intelligence on the SDN controller, will provide new functions and flexibility not available on today's architectures. We next describe some examples.

The logical centralization of SDWN offers the ability to change the forwarding of user data traffic in the access (radio and backhaul) and core transport networks, which can be used to provide functionalities such as mobility management and QoS provisioning.

The programmable configuration of the RAN (including the backhaul), such as MAC and radio access parameters, allows the best dynamic use of available resources, considering current load, users' distribution, and network sharing among virtual operators. Examples of this are configuration of the wireless multimedia extensions for WiFi, configuration of radio bearers for 3GPP accesses, configuration of IEEE 802.11aa behavior for multicast transmission over WiFi, management of optical and microwave parameters in backhaul links, and more. Overall, this programmability can be used to help meet both global network-wide goals (in terms of efficiency) as well as particular targets on a per virtual operator and service provider basis (e.g., differentiation on a per application/mobile node).

The joint dynamic configuration of the RAN and transport core network allows global network updates to be performed in order to better adapt to current conditions. Examples of these updates are using different unicast/multicast distribution schema within the network in order to better meet specific QoS requirements. But this can also be used to offer dynamic adaptation of the traffic at the transport and application levels; for example, NAT or IPv6/IPv4 transition functionality can be dynamically enabled and moved

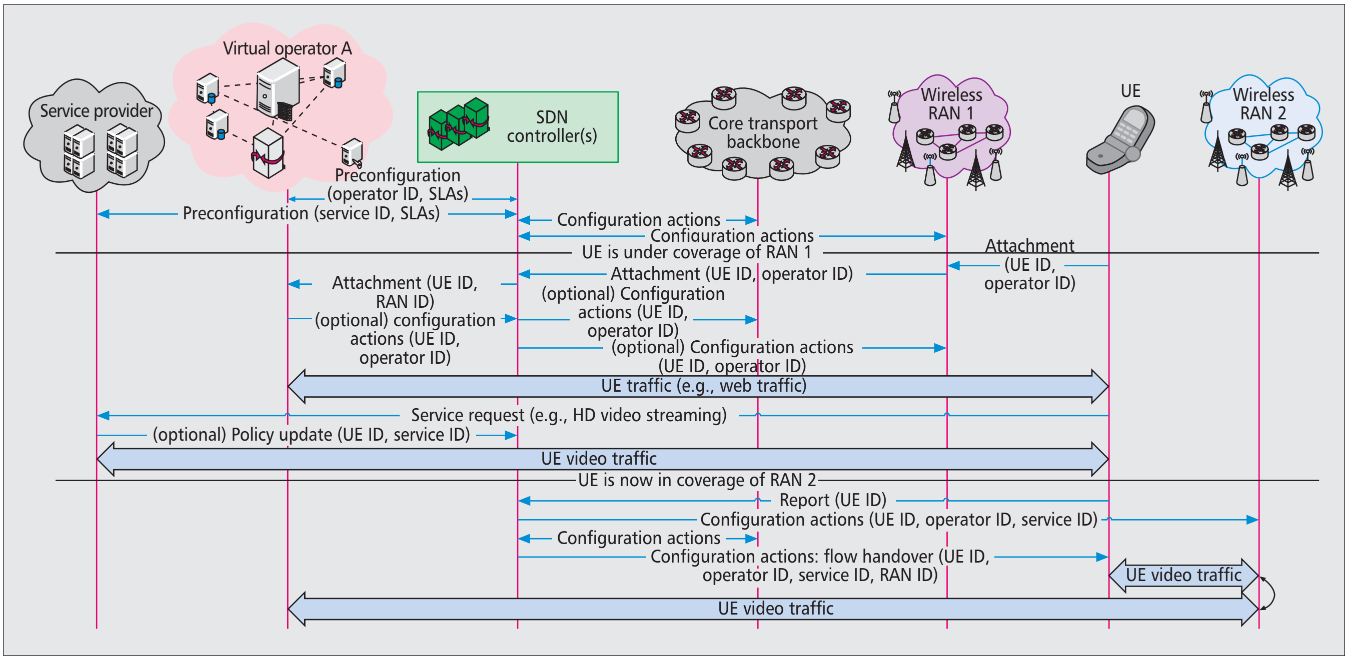

Figure 4. Case study: SDWN high-level interactions.

within the network. Even more, the forwarding path of selected multimedia flows can be updated, so they traverse a middleware box capable of adapting the content to the network and client conditions (e.g., by re-encoding the multimedia stream or selectively dropping some packets).

Extending the programmability to the mobile nodes enables very interesting enhancements of the users' experience. Handover management can be made much more efficient, as the network and client sides would be in tight coordination, which eases tasks such as cellular offloading and faster handovers. Network discovery can also be simplified, as the network can enforce its policies more easily and almost in real time.

Case Study

In order to show how the proposed SDWN framework works, we next make use of an example case study, describing the high-level interactions between the main SDWN components (Fig. 4). We not only explain how to make use of the defined interfaces — highlighting the opportunities enabled by the use of SDN in a wireless mobile network — but also identify some of the challenges posed by the adoption of such an architecture.

In our case study, which is meant to serve just as an example and therefore does not capture all the possible interactions, virtual operator A operates using the resources of a core transport backbone and different RANs. Using the defined northbound interface with the SDN controller(s), virtual operator A is capable of preconfiguring the physical network resources, so it appears as a valid operator in the area covered by wireless access networks. Similarly, a service provider can also preconfigure the network in order to provide a certain default treatment to its traffic (e.g., provide low end-to-end latency). Note that in both cases, the use of the interfaces provided to configure/influence the behavior of the network requires proper SLAs to be in place.

In this scenario, if a UE falls within the coverage of wireless RAN 1 and attaches to the network, this event can be reported to the SDN controller(s) using the existing southbound interface. This attachment event is also reported to virtual operator A, which can optionally trigger some specific configuration actions on the network for that particular UE. These additional configurations can affect both the transport core and the wireless access networks, for example, by offering a prioritized wireless access by configuring the MAC service provided to the UE.

If the UE requests a service for which the network has an agreement in place (video in this example), its traffic is provided with a differentiated service. The network configuration required to do so can already be in place, or can be triggered on the service provider by the SDN controller(s). Both approaches are possible (preconfiguration of the traffic forwarding and on-demand dynamic configuration), allowing for different service models (e.g., gold users are provisioned on demand, while regular users are provided with a default service). Note that scalability might be a concern if preconfigured policies are not used, and the traffic from all the users is treated on demand. This is one of the main challenges to be addressed in SDN architectures.

If the UE gets into coverage of another wireless access network, also managed by the same SDN controller(s) and with the required SLAs in place, this event is received by the network, which can then evaluate whether to selectively move some flows to the new access in order to improve this particular UE's QoE, the overall utilization of the network, or both. In this example, video traffic is moved to wireless RAN 2 by exploiting the existing SDN interfaces with both the infrastructure network entities and the UE.

Standardization: Status and Future Needs

Current Efforts

Regarding standardization efforts in the area of SDN, the most relevant standards developing organization (SDO) is the Open Networking Foundation (ONF),[4] a member-driven standards organization aiming to promote and adopt SDN through open standards development. The ONF is the home of the well-known OpenFlow standard and OpenFlow Switch Specification [5], defining the protocol used for the communication between the OpenFlow controller and switches. In addition to these core standards, the ONF also publishes a testing specification to guide the conformance of OpenFlow switches [12]. The ONF is structured into several working groups (WGs), which address topics ranging from extensibility (adding new features) to migration (for existing networks to adapt to the OpenFlow standard). Recently, a Wireless and Mobile Working Group has been established to address the specific requirements of mobile networks. The charter of the group[5] lists a number of identified use cases ranging from mobile backhaul to mobile core issues. OpenFlow extensions can be expected in the near future to cover the gap in existing specifications.

At the IETF,[6] the SDN trend is impacting several WGs. The most representative SDN-related WGs are the following: Forwarding and Control Element Separation (FORCES), which is tackling the separation between the control and data planes; Network Virtualization Overlays (NVO3), focusing on the data center overlay problem space and architecture; and finally, the Interface to the Routing System (I2RS), which focuses on providing a real-time interface into the IP routing system.

In addition, the Internet Research Task Force (IRTF) has created the Software-Defined Networking Research Group (SDNRG),[7] which is analyzing the approaches that can be used both in the near term and in the future. Finally, there are some other WGs at the IETF that can bring useful knowledge to SDN standardization, such as the ALTO, PCE, NETCONF, NETMOD, and DMM WGs.

The IEEE 802 LAN/MAN Standards Committee has also started SDN-related activities. Although there is currently no WG focused specifically on SDN in IEEE 802, there are ongoing discussions on how to introduce SDN capabilities into wireless and wired technologies. The most relevant activities are the Open Mobile Network Interface for Omni-Range Area Networks (OmniRAN) Study Group, defining an access network specification based on IEEE 802 technologies (802.11, 802.1X, etc.) that includes a network architecture, recommended communication protocols, and link-specific parameters usage, which has specifically discussed the SDN capabilities that are needed in the specific IEEE 802 technologies [13]; ongoing discussion within the IEEE 802.16 WG on introducing the bridging capability directly within the wireless stack in such a way that SDN functionalities can be applied to the wireless connections [14]; and the IEEE 802.11ak[8] and 802.1Qbz[9] specifications, which are very relevant as they specify how 802.11 access points (APs) interconnect with wired bridged technologies in a simpler (and more SDN-friendly) manner.

Key benefits	Key challenges
Easier deployment of new services	Specification of the interfaces
Reduced management and operational costs of heterogeneous technologies	Need to integrate scheduled-based and contention-based systems
Efficient operation of multi-vendor infrastructures	Harmonization of the standardization efforts
Increased accountability and service differentiation	Verifiable security and privacy architecture
Continuous and transparent enhancement of network operation	Operation and management of wireless networks is more complex

Table 1. The case for SDN in mobile networks.

Currently, the 3GPP is not directly addressing how to introduce SDN concepts on their specifications, although SDN-based approaches are being investigated as a possible solution to some current challenges. For example, the concept of reconfigurable backhaul is used for RAN sharing. The sharing of the RAN by different operators requires the traffic to be routed to the correct network, depending on the operator policies. In addition to this functionality, SDN-related concepts are being discussed as a technology that can be applied to self-organizing networks (SONs) [15]. Self-organizing networks provide mechanisms for self-optimization, self-healing, and self-configuration, applying concepts, such as programmable control of the network, that are shared by SDN-based approaches.

Other SDOs such as the European Telecommunications Standards Institute (ETSI) and International Telecommunication Union Telecommunication Standards Sector (ITU-T) are also considering approaches similar to SDN to define their architectures for future networks. This is the case of ITU-T SG 13[10] (future networks including cloud computing, mobile, and next-generation networks), which has included in its work program several topics related to SDN networks as mechanisms for realizing network intelligence capability enhancement (NICE), developed at the ITU-T. In the case of ETSI, the Autonomic Network Engineering for the Self-Managing Future Internet (AFI) Industry Specification Group (ISG) is working on solutions for autonomic (self-) management and control of the network resources in mobile networks.

Additionally, ETSI has recently (October 2013) delivered several specifications on NFV[11]. Although the NFV and SDN concepts are mutually beneficial but not dependent on each other, both proposals share the idea of providing a set of functions enabling the programming of network functions.

---

[4] *https://www.opennetworking.org/*

[5] *https://www.opennetworking.org/images/stories/downloads/working-groups/charter-wireless-mobile.pdf*

[6] *http://www.ietf.org/*

[7] *http://irtf.org/sdnrg*

[8] *http://www.ieee802.org/11/Reports/tgak_update.htm*

[9] *http://www.ieee802.org/1/pages/802.1bz.html*

[10] *http://www.itu.int/en/ITU-T/studygroups/2013-2016/13/Pages/default.aspx*

[11] *http://www.etsi.org/technologies-clusters/technologies/nfv*

To organize and focus the work performed by the different bodies addressing the important labor of standardizing SDN related technologies, it is desirable to create a common view of use cases, network services and functionality among all SDOs, which can be further used to create a real momentum and push forward the next generation of SDN concepts.

FUTURE NEEDS

Standardization efforts represent one of the key building blocks of the telecommunication industry. A standard answers the need for communication among devices of different manufacturers. Hence, the focus of a standard is placed on protocol definition, message formats, and corresponding behaviors, leaving specific implementations to the choice of the manufacturer. For example, OpenFlow, one of the most relevant SDN tools available, uses a central controller and a set of interfaces: a northbound interface to communicate with control applications and a southbound interface to communicate with the actual network hardware. Through these two interfaces, applications can be developed to use network functions and interact with the deployed hardware. Hence, it is of vital importance to standardize these two interfaces in order to be able to communicate and interact with hardware provided by different vendors.

Currently, different SDOs are working on closely related and entangled issues, although each one is focused on a different point of view. As such, the current panorama of standardization activities is formed by a number of non-collaborative activities, several of them trying to solve the same issues in a slightly different way. SDN relies on APIs and standardized service access points to be able to control the behavior of physical network elements. As such, standardization plays a crucial role for this purpose, since without a clear definition of these interfaces, SDN cannot become a reality. In order to organize and focus the work performed by the different bodies addressing the important work of standardizing SDN related technologies, it is desirable to create a common view of use cases, network services, and functionality among all SDOs, which can be further used to create real momentum and push forward the next generation of SDN concepts.

CONCLUSIONS

In this article we have identified the opportunities that software defined networking can bring to wireless and mobile networks. We propose a high-level architecture leveraging on the advantages of the logical centralization provided by SDN. We have first defined the main functions that should be supported by a mobile SDN architecture, and then specified the required interfaces and described some of the interactions that would be needed to enable new and/or richer use cases. A summary of the key benefits and challenges for SDN in mobile networks is provided in Table 1.

Last but not least, we have reviewed ongoing standardization efforts around SDN topics, identifying the future needs to ensure a successful practical deployment of SDN mechanisms in the wireless arena.

ACKNOWLEDGMENTS

The research leading to these results has been partly funded by the European Community's Seventh Framework Programme FP7/2007–2013 under grant agreement no. 317941 — project iJOIN, http://www.ict-ijoin.eu/.

REFERENCES

[1] Y.-J. Chang *et al.*, "Scalable and Elastic Telecommunication Services in the Cloud," *Bell Labs Tech. J.*, vol. 17, no. 2, 2012, pp. 81–96.
[2] "Network Functions Virtualisation — An Introduction, , Benefits, Enablers, Challenges, and Call for Action," white paper, http://portal.etsi.org/NFV/NFV_White_Paper.pdf, 2012.
[3] S. Jain *et al.*, "B4: Experience with a Globally Deployed Software Defined WAN," *Proc. ACM SIGCOMM '13*, 2013, pp. 3–14.
[4] ONF, "Software-Defined Networking: The New Norm for Networks," 2012.
[5] ONF, OpenFlow Switch Specification, v. 1.4.0. Oct. 14, 2013.
[6] P. Calhoun, M. Montemurro, and D. Stanley "Control and Provisioning of Wireless Access Points (CAPWAP) Protocol Specification," IETF RFC 5415, Mar. 2009.
[7] G. Bianchi *et al.*, "MAClets: Active MAC Protocols over Hard-Coded Devices," *Proc. 8th Int'l. Conf. Emerging Networking Experiments and Technologies*, 2012, pp. 229–40.
[8] K. Pentikousis *et al.*, "Mobileflow: Toward Software-Defined Mobile Networks," *IEEE Commun. Mag.*, vol. 51, no. 7, July 2013, pp. 44–53.
[9] NGMN, "Backhaul Evolution — Integrated QoS Management."
[10] A. de la Oliva *et al.*, "IP Flow Mobility: Smart Traffic Offload for Future Wireless Networks," *IEEE Commun. Mag.*, vol. 49, no. 10, Oct. 2011, pp. 124–32.
[11] A. Dixit *et al.* "Towards an Elastic Distributed SDN Controller," *Proc. 2nd ACM SIGCOMM Wksp. Hot Topics in Software Defined Networking*, 2013, pp. 7–12.
[12] ONF, "Conformance Test Specification for OpenFlow Switch Specification," v. 1.0.1, June 13, 2013.
[13] R. Marks, A. de la Oliva, and J.C. Zuñiga, "Proposed OmniRAN SDN Use Case for External Communication," Aug. 2013.
[14] IEEE P802.16r, "View of Connection-Oriented Software-Defined Networking for Wireless Backhaul of Small Cells," July 2013.
[15] 3GPP, TS 32.500, Telecommunication Management, "Self-Organizing Networks (SON); Concepts and Requirements," Dec. 2011.

BIOGRAPHIES

CARLOS J. BERNARDOS (cjbc@it.uc3m.es) received his degree in telecommunications engineering in 2003 and a Ph.D. in telematics in 2006, both from Universidad Carlos III de Madrid (UC3M), where he worked as a research and teaching assistant from 2003 to 2008, and since then has worked as an associate professor. His current work focuses on mobility in heterogeneous wireless networks. He has published over 50 scientific papers in international journals and conferences, and he is an active contributor to the IETF. He has served as a Guest Editor of *IEEE Network*.

ANTONIO DE LA OLIVA (aoliva@it.uc3m.es) received his degree in telecommunications engineering in 2004 and a Ph.D. in telematics in 2008 (receiving a national award ex-aequo for the best Ph.D. thesis in IPTV services) , both from UC3M, where he worked as a research and teaching assistant from 2005 to 2008, and since then has worked as a visiting professor. His research is focused on mobility in heterogeneous networks and wireless systems. He has published over 30 scientific papers in prestigious international journals and conferences, and he is also an active contributor and voting member of IEEE 802.21, where he has served as Vice-Chair of IEEE 802.21b and Technical Editor of IEEE 802.21d. Currently he is involved in the FP7 CROWD project.

PABLO SERRANO (pablo@it.uc3m.es) got his degree in telecommunications engineering and Ph.D. from UC3M in 2002 and 2006, respectively. He has been with the Telematics Department of UC3M since 2002, where he currently holds the position of associate professor. He was a visiting researcher at the Computer Network Research Group at the University of Massachusetts Amherst in 2007, and at Telefonica Research Center in Barcelona in 2013. He has over 50 scientific papers in peer-reviewed international journal and conferences. He serves on the Editorial Board of *IEEE Communications Letters*, and has been a Guest Editor for *Computer Networks*.

ALBERT BANCHS (banchs@it.uc3m.es) received his degree in telecommunications engineering from the Polytechnic University of Catalonia in 1997 and his Ph.D. degree from the same university in 2002. He received a national award for the best Ph.D. thesis on broadband networks. He was a visiting researcher at ICSI, Berkeley, California, in 1997, worked for Telefonica I+D, Spain, in 1998, and for NEC Europe Ltd., Germany, from 1998 to 2003. He has been with UC3M since 2003. Since 2009, he has had a double affiliation as deputy director of the IMDEA Networks Institute.

LUIS M. CONTRERAS (lmcm@tid.es) completed a six-year telecommunications engineer degree at the Universidad Politécnica of Madrid (1997), and holds an M.Sc. in telematics from UC3M (2010). In 1997 he joined Alcatel in Spain, taking several positions in both the wireless and fixed network fields. In 2006 he joined the Network Planning Department of Orange in Spain (France Télécom group) taking responsibilities for the IP backbone and mobile packet core planning. Between 2002 and 2010 he was also an adjunct lecturer at the Telematics Department of UC3M, where he is currently a Ph.D. student. Since August 2011 he is part of Telefónica I+D/Telefónica Global CTO, working on SDN, transport networks, and their interaction with cloud and distributed services. He is an active contributor to IETF and ONF.

HAO JIN (hao.jin@interdigital.com) is a senior engineer at InterDigital Communications, Inc., where he is currently working on research and development of SDN and network virtualization for wireless and mobile networks. He earned his B.S. degree in electrical engineering from Nanjing University, China in 2006, and his Ph.D. degree in electrical engineering from Florida International University in 2012. His research interests include SDN, network virtualization, next generation wireless networks, and data center networking.

JUAN CARLOS ZÚÑIGA (juancarlos.zuniga@interdigital.com) received his engineering degree from the UNAM, Mexico, and his M.Sc. and DIC from Imperial College London, United Kingdom. He is a principal engineer at InterDigital, currently leading the standardization activities in the areas of heterogeneous networks, software defined networking, distributed content management, and Internet privacy. He has actively contributed to and held leadership roles in different standards fora, such as IEEE 802, IETF, and 3GPP. He has been with InterDigital since 2001. Previously he worked with Harris Communications in Canada, Nortel Networks in the United Kingdom, and Kb/Tel in Mexico. He is an inventor of more than 30 granted U.S. patents and 70 published applications.