# Demo: Orion – A Radio Access Network Slicing System

Xenofon Foukas
The University of Edinburgh
x.foukas@ed.ac.uk

Mahesh K. Marina
The University of Edinburgh
mahesh@ed.ac.uk

Kimon Kontovasilis
"NCSR" Demokritos
kkont@iit.demokritos.gr

## ABSTRACT

Emerging 5G mobile networks are envisioned to support the dynamic deployment of services with diverse performance requirements, accommodating the needs of mobile network operators and verticals. Virtualizing the mobile network components in a flexible and cost-effective way is therefore of paramount importance. In this work, we highlight the capabilities of Orion, a novel RAN slicing architecture that enables the dynamic virtualization of base stations and flexible customization of slices to meet their respective service needs. Our demonstration of Orion's capabilities is based on a prototype implementation employing a modified version of the OpenAirInterface software LTE platform. Using this prototype, we demonstrate the functional and performance isolation, and the efficient sharing of radio hardware and spectrum that can be achieved among Orion RAN slices. Moreover, we show how Orion can be used in an end-to-end network slicing setting and demonstrate the effects of the slices' configuration and placement of virtual functions in the overall quality of the deployed services.

## KEYWORDS

5G mobile networks; network architecture; RAN slicing; RAN virtualization; abstractions

## 1 INTRODUCTION

The rapid increase in the usage of mobile devices in recent years has placed a big strain on the current mobile network architecture is paving the way for next generation (5G) mobile networks. Apart from the performance and efficiency improvements that 5G is expected to bring, another equally significant aspect of the 5G vision is the support of a *service-oriented* mobile network architecture [4]. This service-oriented approach envisions network support for a wide range of services, differing significantly in their performance requirements and supported device types. A one-size fits all architecture is unlikely to be suitable for such diverse use cases. Therefore, it is of paramount importance to find ways for increasing the flexibility of the architecture, so that the underlying physical infrastructure may be turned into multiple logical networks or *slices* that are tailored in terms of resources (computing, network, storage, radio, access hardware and virtual network functions (VNFs)) to meet the requirements of the service in question [5]. In this context, network softwarization via Software Defined Networking (SDN) and Network Functions Virtualization (NFV), and virtualization are seen as key enablers that can be used to create end-to-end network instances spanning both the core and the radio access network (RAN).

For mobile core slicing, research prototypes and operational systems using virtualization technologies and SDN/NFV principles have already started to appear. On the other hand, RAN slicing is at a more premature stage, with the main challenge being that apart from computing, storage and network resources, the limited radio resources need to also be virtualized and efficiently assigned to slices. This must be achieved while guaranteeing the functional and performance isolation between tenants and the infrastructure provider, and among tenants themselves, so that these tenants can maintain full control and independence of their slices to tailor them to the respective service requirements. State-of-the-art RAN slicing solutions solve this problem only partially, by focusing on some aspects of RAN slicing while sacrificing others. Specifically, approaches with origins in RAN sharing focus mainly on the efficient sharing of radio resources with no support for functional isolation, giving the infrastructure provider full visibility and control over all slices (e.g. FlexRAN [1]). On the other end of the spectrum, the isolation is put at the center stage without considering the efficient and adaptive use of resources (e.g. FLARE [3]).

We attempt to balance these objectives through Orion [2], which in our knowledge is the first RAN slicing system that provides full functional and performance isolation while also facilitating efficient sharing of radio and spectrum resources. The focus of this demo (elaborated in Section 3) is on showcasing the unique capabilities of Orion, as well as on demonstrating how Orion can fit in the emerging 5G paradigm as a key component of an end-to-end mobile network slicing architecture. The next section gives an overview of Orion's design and implementation.

## 2 ORION

### 2.1 Design

Orion's design (Fig. 1) explicitly distinguishes the infrastructure provider from the service providers, who own the slices. The infrastructure provider is the owner of physical base stations, comprising of hardware resources (i.e., radio equipment, memory, CPU and network) and a chunk of spectrum. Regardless of its realization, either via dedicated specialized hardware or in a cloud environment using re-programmable hardware (e.g., C-RAN baseband processing unit and remote radio heads), each physical base station supports a single Radio Access Technology (RAT), meaning that all radio and spectrum resources available at the base station can be exploited through a *shared* physical layer.

The `Base Station Hypervisor` that sits over the physical layer is the heart of Orion's design. It is the component used for managing RAN slices, enforcing resource segregation among slices (for performance isolation) and segregation in terms of control

*MobiCom '17, October 16–20, 2017, Snowbird, UT, USA.*
© 2017 Copyright held by the owner/author(s). ISBN 978-1-4503-4916-1/17/10.
DOI: https://doi.org/10.1145/3117811.3119860

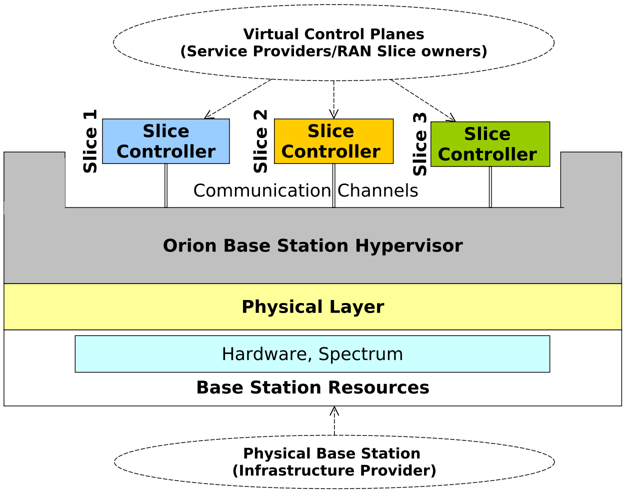

**Figure 1: High-level architecture of Orion.**

logic segregation (for functional isolation) and for facilitating efficient sharing of underlying physical resources. Essentially, the `Hypervisor` is the liaison that binds the individual (and mutually isolated) slices to the physical infrastructure, providing them with a virtual view of the underlying radio resources and data plane state, and maintaining a mapping between virtual and physical resources so that slice state changes may be applied over the physical data plane. The `Hypervisor` is part of the infrastructure provider's software infrastructure to support RAN slicing. The infrastructure provider is also responsible for admission control.

Service providers (e.g., MVNOs and verticals) in Orion realize their RAN slices through the creation of virtual base stations over the `Hypervisor`. Each virtual base station is a composition of a virtual control plane, responsible for managing the data plane state revealed to it by the `Hypervisor`. Following SDN principles, Orion's design assumes control-data plane separation, where the virtual control plane of a slice is effectively a local RAN-level slice controller running as a *separate process*, responsible to tailor the functionality and manage the allocation of resources to the mobile devices associated with the slice as if the slice was operating using its own dedicated infrastructure. The virtual control plane is also responsible for implementing the control protocols required for the communication and coordination of the virtual base station with the rest of the mobile infrastructure (e.g., S1 and X2 interfaces in LTE). This means that all operations defined for a given mobile network architecture can be supported by slices so long as the appropriate interfaces and messages are implemented as part of the respective virtual control planes.

## 2.2 Key Properties of Orion

The design of Orion provides strong isolation guarantees while allowing efficient resource sharing.

**First,** since each of the slice controllers runs as a separate process, isolation among controllers in terms of memory and CPU can leverage well known OS and process virtualization techniques, like virtual machines (e.g., KVM) or containers (e.g., Docker).

**Second,** the `Hypervisor` is the sole entity responsible for handling actual radio resources which it virtualizes and distributes among slices in an abstract form, ensuring isolation from a radio resource perspective. The isolation properties of Orion are illustrated in the example of Fig. 2 for 2 slices, where the throughput of the UEs in one slice remains unaffected from the changes in the state of the other slice. More specifically, the addition of more UEs in one slice at $t1$-3 only affects the throughput of the UEs co-existing in the same slice, leaving the other slice unaffected (indicating performance isolation). Similarly, a change in the scheduling policy of slice 2 at $t4$, from proportional fair to class-based only affects the performance of the UEs in that slice, having no effect for slice 1 (reflecting functional isolation).

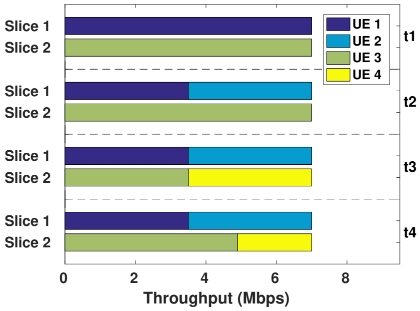

**Figure 2: Isolation of RAN slices in terms of radio resources and control functions (scheduling).**

**Third,** it can internally facilitate efficient use of the spectrum pool via a suitable allocation algorithm that considers the slices' SLAs and current state as well as the underlying physical conditions. This flexible allocation of radio resources to slices can lead to an improved performance compared to a static allocation mechanism, as illustrated in the example of Fig. 3 (slice 2 borrowing the unused resources of slice 1).

**Finally,** from a UE perspective, the whole slicing operation is transparent with each slice appearing as a different MVNO like in RAN sharing.

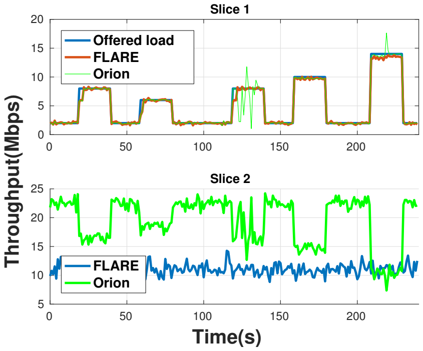

**Figure 3: Instantaneous throughput of two slices in static (FLARE) vs dynamic (Orion) resource allocation.**

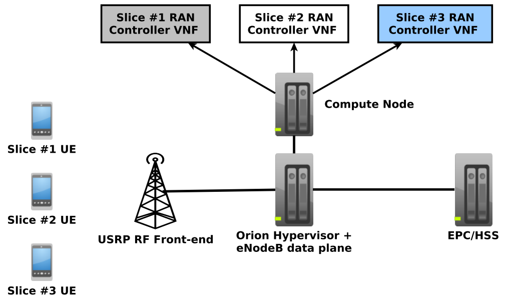
Slice #1 RAN Controller VNF
Slice #2 RAN Controller VNF
Slice #3 RAN Controller VNF
Compute Node
Slice #1 UE
Slice #2 UE
Slice #3 UE
USRP RF Front-end
Orion Hypervisor + eNodeB data plane
EPC/HSS

**Figure 4: Basic Orion demonstration setup with shared core network among slices**

## 2.3 Implementation

Following the aforementioned design, we developed a prototype implementation of Orion considering LTE as the supported RAT, providing full support for the virtualization of downlink radio resources. The Orion Hypervisor was implemented completely from scratch in C. For the separation of the control and the data plane required by Orion, we leveraged the API and controller of the publicly available FlexRAN platform [1], which is based on the well known open-source LTE software implementation of OpenAirInterface (OAI).

# 3 DEMONSTRATION

The demonstration of Orion has three main goals:

- To show how the Orion architecture accommodates the dynamic creation of RAN slices sharing the underlying common radio hardware and spectrum, while providing functional and performance isolation among the co-located tenants (MVNOs and verticals).
- To highlight the efficiency and flexibility of Orion's mechanisms in distributing the available radio resources among slices based on the service requirements and corresponding SLAs of the latter.
- To show how Orion can be integrated into an end-to-end network slicing architecture composed of virtualized RAN and core network functions and how the configuration of the slices and the placement of the virtual functions composing the network can impact the characteristics of the provided service.

For the first and the second part of this demonstration, we will use the testbed setup illustrated in Fig. 4, including three commercial LTE smartphones (LG and Samsung), 3 Intel-based mini PCs (Intel i7 at 3.4GHz and 8GB RAM) and 1 Ettus USRP B210 RF front-end. The USRP is connected to one of the PCs, over which the eNodeB data plane and the Orion Hypervisor is deployed. The Hypervisor communicates through an Ethernet connection with a compute node that is used for the deployment of the virtual control planes (RAN controllers) of three RAN slices. The controllers are deployed in the form of virtual network functions, running in isolation by employing Docker containers. The Hypervisor also communicates with the third PC, which acts as an EPC (MME and SP-GW) and HSS shared among all three RAN slices.

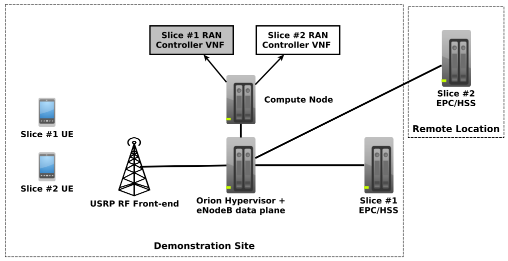

**Figure 5: End-to-end Orion demonstration setup with local and remote core network**

Through this setup, we demonstrate the capability of Orion to create RAN slices on-the-fly through a simple set of tools and configuration files. We show how each UE can be mapped dynamically to a slice through Orion during its attachment process, based on the configuration provided by the slice owner. Moreover, we show the functional and performance isolation that Orion enables, by using different downlink scheduling functions with different radio resource allocation policies on each of the slices' RAN controllers, streaming video flows and observing how the introduction of new slices leaves the performance of other co-located slices completely unaffected. Finally, we demonstrate the flexibility of sharing the available spectrum among slices in an efficient way, by creating slices with services differing in their requirements for radio resources (some slices asking a fixed number of radio resource blocks and others specifying throughput guarantees in ideal conditions) and by showing how the unused radio resources of one slice can be dynamically assigned to other slices running over the same physical base station to improve their performance.

For the final goal of this demonstration, the testbed is slightly modified (Fig. 5) so that, instead of having a shared core for all slices, two EPCs are connected with the eNodeB, each associated with a different slice. One EPC is co-located with the base station at the demonstration site (direct Ethernet connection), while the other is deployed in a remote location. Based on this setup, we show how the various EPC components can be deployed as virtual network functions in different physical locations (at the network edge or in a central cloud) and how they interact with the RAN slices of Orion to create an end-to-end network slicing setting. Additionally, we show how the placement of these virtual functions and the radio resource allocation policies used by the Orion Hypervisor can affect the service provided by the slices, using latency of the slices' traffic as a concrete example.

# REFERENCES

[1] Xenofon Foukas et al. 2016. FlexRAN: A Flexible and Programmable Platform for Software-Defined Radio Access Networks. In *Proceedings of ACM CoNEXT*. ACM, 427–441.

[2] Xenofon Foukas et al. 2017. Orion: RAN Slicing for a Flexible and Cost-Effective Multi-Service Mobile Network Architecture. In *Proceedings of ACM MobiCom*. ACM.

[3] Akihiro Nakao et al. 2017. End-to-end Network Slicing for 5G Mobile Networks. *Journal of Information Processing* 25 (2017), 153–163.

[4] NGMN-Alliance. 2015. 5G White Paper. (Feb 2015).

[5] Peter Rost et al. 2017. Network Slicing to Enable Scalability and Flexibility in 5G Mobile Networks. *IEEE Communications magazine* (2017).