TEJAS NETWORKS

Developing a Truly Open Virtualized RAN

Published Oct 2020

Table of Contents

1. Introduction 1

2. Current DU Landscape 1

3. Innovative Approach to Designing Virtualized O-RAN Solutions 3

   3.1 Software Framework 3

      3.1.1 RAN Abstraction Layer 4

      3.1.2 RAN Hypervisor 5

   3.2 Open DU (ODU) Hardware platform 6

4. Benefits to the MNOs 7

5. Conclusion 8

Executive Summary

In the past, Radio Access Network (RAN) deployments for 4G, 3G, and 2G relied on proprietary aggregated Radio Frequency (RF) and Baseband equipment from a few dominant suppliers, with very few other vendors getting an opportunity to enter the fray. Closed, proprietary designs and interoperability challenges made it tough to deploy multi-vendor RAN equipment. Network operators were reliant on a single vendor for all aspects of RAN implementation and optimization. One of the main problems with this is that these hardware-based appliances rapidly reach the end of life and require the long design cycles to be repeated often, resulting in little or no revenue benefit for operators. Additionally, with the acceleration of technology and service innovations, hardware life cycles are becoming shorter. This ultimately inhibits the rollout of new revenue-earning network services and puts constraints on innovation in today's dynamic, network-centric connected world.

Several initiatives emerged from this tightly held landscape to create open, dis-aggregated, scalable RAN equipment with interoperable interfaces. These include the Open RAN initiative from TIP and ORAN Alliance.

vRAN introduced the use of Commercial off-the-shelf (COTS) servers and network virtualization, while ORAN helped disaggregate the RAN hardware and bring diversity and some level of openness to the ecosystem. However, hardware and software frameworks remain limited to very few vendors and have not resulted in a truly open RAN architecture. Apart from Openness, the current ORAN-based solutions have certain limitations including lack of elasticity, portability, and higher Total Cost of Ownership (TCO).

Saankhya Labs' new approach to creating a truly open, more optimized, elastic RAN solution will help address these issues and develop a more extensive ecosystem. This differentiated solution aims to optimize the use of resources and the reduction of TCO. By enabling Independent Hardware Vendors (IHV) to bring in cost-effectiveness, Saankhya's open solutions will also speed up more innovation in an otherwise closed RAN architecture.

The primary purpose of this paper is to introduce Saankhya's new approach to make a genuinely Open and ORAN compliant DU solution and to define an open software framework that provides a uniform model of the hardware underneath and lowers the barrier to build portable RAN software

1. Introduction

With 5G networks becoming increasingly software-driven, the move towards a virtualized, cloud-optimized RAN architecture is gaining traction, with several new challengers trying to bring diversity into the ecosystem. Variations including C-RAN (Centralized RAN), vRAN (virtualized RAN), and O-RAN (Open RAN) architectures are being explored by the different players.

Use of Commercial off-the-shelf (COTS) servers and network virtualization began with the vRAN, optimized for cloud deployment and operation. However, despite all efforts, hardware and software frameworks remain limited to very few vendors and have not resulted in a truly open RAN architecture.

The ORAN Alliance, built on the core principles of intelligence and openness, disaggregated the RAN into sections – such as Radio Unit [RU], Distributed Unit [DU] and Central Unit [CU]) and defined standard interfaces between these sections. These promote network intelligence through open and standardized interfaces in a multi-vendor network.

Despite this, there are several challenges in today’s ORAN based DU solutions that need to be addressed, before an entirely ORAN based solution can see large scale adoption by operators around the world. For example, while Commercial off-the-shelf (COTS) hardware (with Intel/ARM CPU) is economical, this alone is not suitable for high capacity networks. It needs to be populated with additional accelerator cards like FPGA/GPU/DSP to increase capacity and connectivity.

This paper intends to discuss a new innovative approach for building a DU platform architecture. It attempts to:

1. Introduce Saankhya's new approach to make a genuinely Open and ORAN compliant DU solution for both vRAN and non vRAN deployments
2. Define an open software framework based on ‘Domain Specific Language’ (DSL) to virtualize custom hardware solutions to provides a uniform model of the RAN hardware underneath

Saankhya’s new approach will lead to greater flexibility, scalability, lower TCO and optimized network resources when the DU is used, either at a cell site or in a vRAN styled data center.

2. Current DU Landscape

ORAN has set an ambitious but achievable goal of a new diverse and open ecosystem for RAN deployments. From a DU perspective, the disaggregation promoted by ORAN has resulted in the decoupling of Independent Software Vendors (ISV) and Independent Hardware Vendors (IHV). Today, DU RAN chipset/hardware vendors can be classified into two broad categories, based on the style of deployments

1. **Quasi Open but COTS solutions for vRAN**: These solutions are based on server-centric chipsets with ISA extensions to support RAN workloads. While the Instruction Set Architecture (ISA) and the Software Development Kit (AVX and CUDA) are available to all, the frameworks built to support the vRAN solution such as FlexRAN and Aerial remain closed and are not freely available. This goes against the philosophy of openness and is therefore considered Quasi Open. These include vRAN solutions from both Intel and Nvidia.

2. **Traditional custom solutions based on ORAN**: These are solutions from legacy vendors within the 4G aggregated base station markets. These solutions are proprietary and are incremental revisions over existing 4G accelerator-based chipsets. They are O-RAN compliant, specific to tower-mounted cell site deployments and are not easily amenable to a pooled vRAN deployment. Technically, these chip vendors promote their local ecosystem of software providers and thereby, are considered Quasi Open.

These DU hardware solutions will form the bulk of the initial deployments of ORAN compliant hardware in fiber deficient cell sites and vRAN based ‘pooled’ data center topologies. As with any initial technology deployments, they suffer from flaws that must be resolved.

The significant issues with these implementations are:

1. **Limited Openness**: Very often, openness gets confused with Commercial off-the-shelf (COTS). However, not all COTS are open source technologies. At present, there is no quantitative measure for Openness, and a qualitative measure would be compared to the Linux model.
2. **Lack of Elasticity**: Inspired by its usage in data center hardware deployment, the term ‘Elasticity’ was coined at Saankhya Labs. It refers to the ability to scale up and down, compute, and adopt other resources dynamically. From a vRAN perspective, it means that a standard solution should be optimal for both cell site and data center deployments. However, current vRAN solutions are optimized for only either one of these deployments. Elasticity is closely related to its cousin ‘Scalability' and refers to the ability to adapt to the dynamic nature of the network.
3. **Non-Portability**: This is limited to the vRAN based RAN Cloud Native Function (CNF) or Virtual Network Function (VNF). Traditional chipsets have customized RAN software, and so, by definition, they are not portable. vRAN software solutions are currently implemented in specific Instruction Set Architecture (ISA) and are tied down to hardware architecture. As portability is a very desirable feature, programmers have resorted to newer abstractions and languages such as Java and Python, which help to build this feature in the cloud application world. Portability allows operators to mix and match ISV with IHV and thereby further reducing dependence on hardware vendors.
4. **RAN Compute Over-Provisioning and ‘Un-lit’ Silicon**: Compute provisioning at cell sites has always been static and does not consider the dynamic nature of the radio environment or traffic conditions. As a result, provisioning based on standards has

always been static, and systems are built for worst-case scenarios. This leads to over-provisioning computing resources, leading to the under-utilization of silicon resources, and thereby resulting in ‘Un-lit’ Silicon.

5. **Higher Total Cost of Ownership (TCO)**: Most COTS solutions are a ‘square peg in a round hole’ designs. The extensions to support RAN workloads have been afterthoughts, resulting in very high PPA (Performance, Power, and Area) metrics. This has a direct impact on the capex and op-ex of operators deploying such solutions.

3. Innovative Approach to Designing Virtualized O-RAN Solutions

The new approach to address some of the issues of current RAN architectures (as mentioned in the previous sections of this paper) involves opening up a new hardware platform and a software framework to create a truly open and portable solution.

The combined hardware and software solution's primary purpose is to introduce all the benefits of a powerful RAN specific hardware, combined with a flexible software framework that will make the hardware portable. The hardware platform and the software framework described in this paper can work independently of each other and offer their own benefits. The main benefits of the proposed combined hardware/software solutions are:

1. **Flexibility**: Ability to implement Virtual Network Functions (VNF) on a merchant silicon and permit 'change' features in the software.
2. **Scalability**: Ability to provision and group together compute elements dynamically to provide a high aggregate compute throughput to provide more elasticity to the hardware platform.
3. **Portability:** Ability to seamlessly ‘port’ DU software on different hardware platforms, with almost no changes to the DU software.
4. **Interchange COTS and custom hardware solutions:** Ability to accrue all benefits of a custom solution like lower TCO while retaining all the benefits of a COTS solution like containerization.

3.1 Software Framework

The key feature of this new design philosophy is a **standardized software framework for RAN workloads on the lines of TensorFlow & OpenCL**. This framework integrated into a Telco cloud infrastructure software provides a uniform model for the hardware underneath and lowers the barrier to build portable RAN software. Two key elements of the RAN software framework include the RAN Abstraction Layer that enables modem developers to specify waveforms in a Domain Specific Language (DSL) and a RAN Hypervisor that executes the modem waveform on a particular hardware platform.

RAN functionality, in the past, when implemented on programmable architectures, was designed using procedural languages like C/C++ and assembly, which are very specific to a hardware implementation. This was necessary, as the overheads and latencies could affect real-time performance. With domain-specific architectures and cheap compute power in lower geometries (i.e. smaller silicon nodes), it is now possible to build a domain-specific language that captures the essence of the problem at hand and provides the right level of abstraction for programmers writing code for such solutions.

In an open ISA and a software framework that is hardware agnostic, the overall entry barrier to building RAN silicon comes down. This allows operators and ISVs the ability to build hardware agnostic, portable DU software.

The two key elements of the RAN software framework mentioned earlier include the RAN Abstraction Layer that enables modem developers to specify waveforms in a Domain Specific Language (DSL) and a RAN Hypervisor that executes the modem waveform on a particular hardware platform.

3.1.1 RAN Abstraction Layer

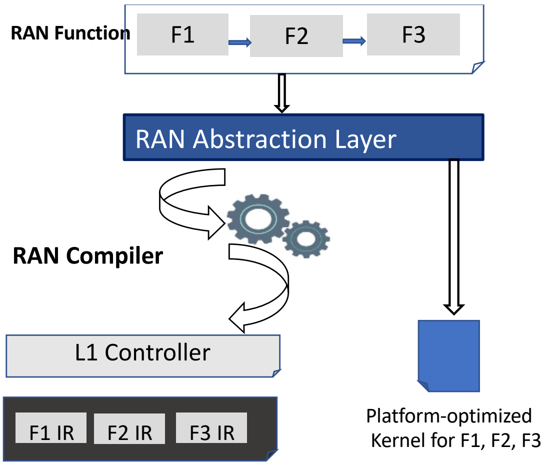

Figure 1: RAN abstraction Layer

RAN Abstraction Layer is a high-performance library that provides the basic building blocks for RAN function development. The key feature of this layer is the ability to specify signal processing elements as stream processors. The stream processors can be mapped to an optimized kernel implementation for a target platform. Modem developers can use the stream processor to create RAN functions, connect them using “connection operators”, specify the required control logic and thus create a complete waveform. This can be done in a DSL or a language of their choice (Python, C++, Golang etc.). RAN Abstraction Layer will also provide bindings to several programming languages. Modem developers can thereafter add their own custom stream processors to the RAN Abstraction Layer.

RAN Abstraction Layer is intended to be a standardized framework for portable RAN function development. The current RAN development methodologies result in vendor lock-in. For example, a RAN function developed for a hardware platform such as Intel FlexRAN or Nvidia cannot be easily ported to another hardware platform. RAN Abstraction Layer, along with the RAN compiler and the RAN Hypervisor, will enable the development of a portable RAN function, thus preventing vendor lock-in, and allowing users to move from one hardware platform to another. The RAN compiler generates an L1 controller that acts as an interface between the L2 layer and the Hypervisor. The Hypervisor schedules execution of the RAN workloads in the IR.

3.1.2 RAN Hypervisor

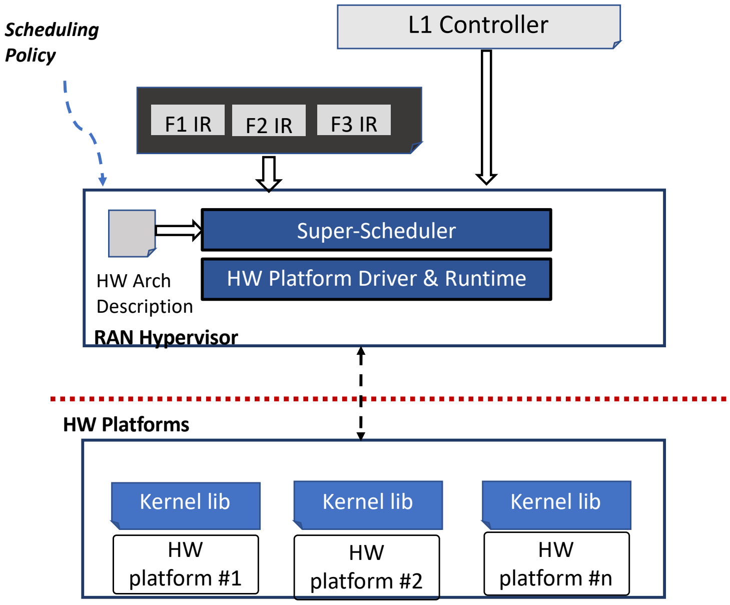

Figure 2: RAN Hypervisor

The two main elements of the software execution architecture of the RAN Hypervisor are 1) Super Scheduler and 2) The Hardware Architecture Description

- **The Super Scheduler** within the hypervisor takes the IR, scheduling policies and constraints as input and schedules the RAN functions (modem waveform) for execution on one or more compute elements of the underlying HW platform.

- **The Hardware Architecture Description** captures the compute architecture of the hardware platform. It comprises details of the different compute elements, their type such as CPU, GPU, DSP, look-aside accelerator, and an execution cost function associated with each of the compute elements. It also captures the memory hierarchy starting from the global memory of the platform and the memory for each of the compute elements. The Super Scheduler analyses the Hardware Architecture

Description and identifies the best schedule for a given RAN IR (of the RAN function) on the target hardware platform.

The primary functionality of a typical Hypervisor is to virtualize the underlying computing resources to enable the execution of multiple guest Virtual Machines (multiple RAN functions). The RAN Hypervisor achieves the same effect utilizing the Hardware Architecture Description and the Super Scheduler. The Hardware Architecture Description is designed to capture the underlying compute architecture of any hardware platform. The Super Scheduler ensures the execution of one or many RAN functions on the target hardware platform by performing effective scheduling of RAN workloads belonging to a particular RAN function.

It must be noted that unlike traditional Hypervisors, the RAN Hypervisor is not tightly coupled to the underlying hardware platform. Since the Super Scheduler relies only on the Hardware Architecture Description of the target platform, it can easily create a schedule for another target hardware platform, as long as the Hardware Architecture Description for that platform is available.

The RAN Abstraction Layer and the RAN Hypervisor thus provide a framework for portable RAN software development. With such a framework available in a telco cloud, operators are no longer tied to a single hardware vendor and can move across multiple hardware vendors.

3.2 Open DU (ODU) Hardware platform

In addition to the Saankhya Labs virtualized software framework described in the previous section, another component of the proposed truly Open DU solution includes an Open Hardware platform. The proposed ODU hardware platform solution is based on Saankhya's 'elastic RAN' (e-RAN) processor – which is a heterogeneous multi-core SoC with DSP tiles as compute elements. Undeniably, Saankhya's 'e-RAN' processor delivers cutting edge performance for running RAN workloads. Compared to an FPGA, which only accelerates specific tasks such as FEC, this e-RAN processor can accelerate the entire HIGH PHY and portions of the L2 accelerator, freeing up the CPU for other tasks. Alternately, a less powerful CPU can be selected. This offers benefits such as **cost-saving, lowest capacity, and power reduction**.

At Saankhya Labs, we have been working on SDRs, or Software Defined Radio Architectures, for over a decade. We have designed and developed three generations of silicon-based SDRs on heterogeneous multi-core DSP architectures. Unlike current vRAN chipsets for DU, Saankhya's elastic RAN (e-RAN) chipsets are ground-up designs, based on a new paradigm called 'Domain-Specific Architectures'. This is a new approach to designing silicon for the post Moore's Law, based on a standardized software framework.

With this solution, a single chip can process data from a channel with a bandwidth of up to 100MHz. Multiple chips can be connected on-board via high bandwidth expansion bus to process data from channels as wide as 800MHz. The chip can implement all the ORAN splits and be deployed in either a pooled topology in the data center or a cell site deployment.

Key features of the chipset:-

- **Flexible domain-specific architecture**
    - Ground-up native ISA optimized for modems
    - Specified top-down by 'software' and 'compiler' engineers
    - Heterogeneous multi-core DSP optimized for RAN workload
    - Thin on hardware accelerators.
    - Maximum functionality in software
- **Scalable**
    - Dynamic provision of computing through the expansion interface allows ‘ganging‘ of chipsets
    - Supports traditional aggregated (split 8) and disaggregated ORAN (split 7.2)
    - RAN hypervisor allows a single virtual compute platform to software
    - Hardware support for Virtualization and RAN Hypervisor
- **Lowest power and best silicon die utilization factor**
- **Hypervisor driven virtualization of computing resources**

4. Benefits to the MNOs

The foremost advantage of such an implementation is the overall lower Total Cost Ownership (TCO). The order of magnitude lower power consumption and smaller die sizes make this TCO possible. Additional benefits of our solution to MNOs are listed below:

Elasticity: It is the ability to scale up/down in a data-center deployment or cell site deployment. Capacity can be added, by adding additional chips, either within the same card or adding additional PCI-e cards. Elasticity allows operators to provision rightly, thereby increasing the utilization factor of the silicon die. The ability to scale up compute farms for RAN deployments gives operators the flexibility to 'provision as you go'.

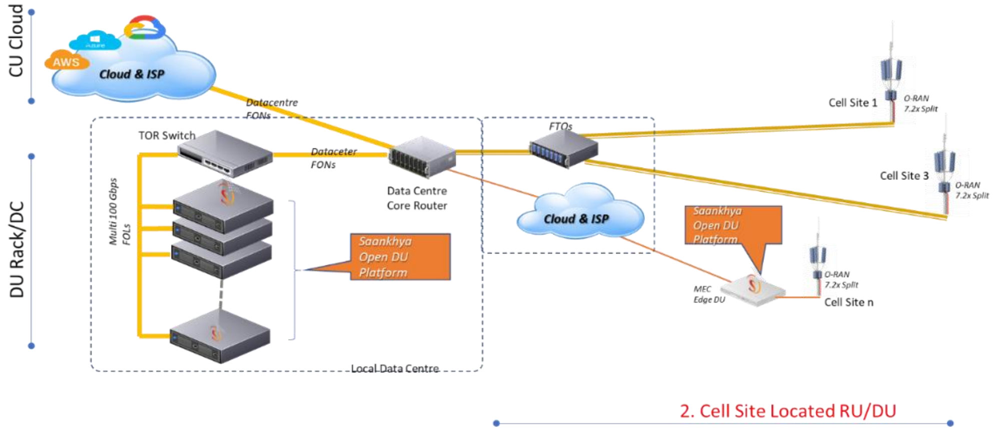

Figure 3: Elasticity (Deployment in Cell-site or in Data Center)

- **Portability:** The RAN Abstraction Layer and the RAN Hypervisor provide a framework for portable RAN software development. With such a framework available in a telco cloud, the operators are no longer tied to a single hardware vendor and can move across multiple hardware vendors

- **Economies of Scale**: Ability to deploy the solution across urban, suburban, and rural markets. Depending on the need of the specific market, the flexible Saankhya DU can be co-sited with the RU or pooled at the remote data center

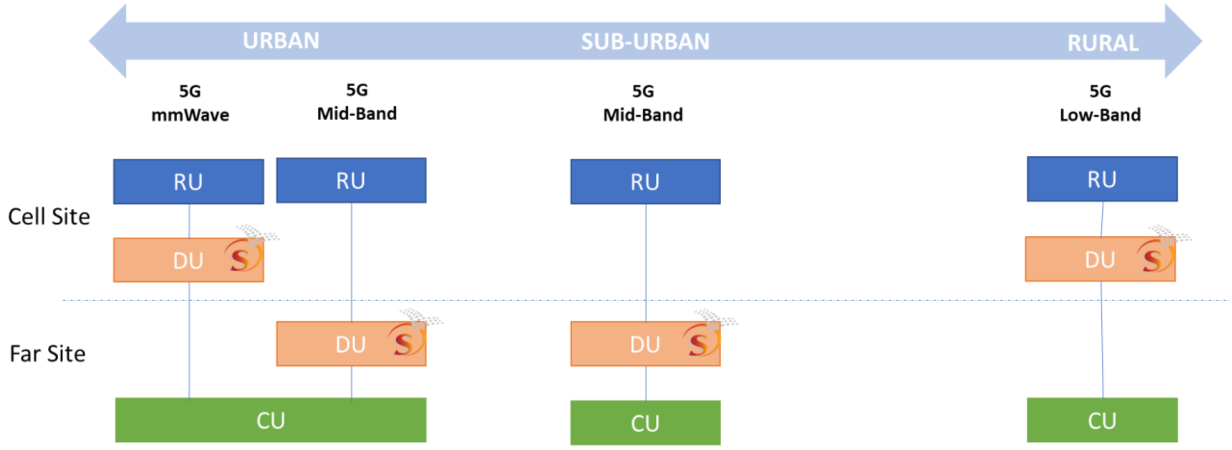

Figure 4: Economies of Scale

5. Conclusion

It has been established that current ORAN based solutions are not open in the true sense of the word. Moreover, the paper presents the fact that the entire edifice of an ORAN based vRAN deployment stands on a COTS solution that results in a high total cost of ownership and lack of "elasticity" to support various deployment topologies based on fiber availability. It also shows a way forward for operators to work with custom RAN hardware without sacrificing a COTS solution's benefits. If these fundamental challenges are not addressed, the large-scale adoption of Open RAN solutions will face severe headwinds.

Saankhya Labs' new approach to creating a truly open, more optimized, elastic RAN solution will help address these issues and develop a more extensive ecosystem. This differentiated solution assures optimization of resource use and reduction of TCO. By enabling Independent Hardware Vendors (IHV) to bring in cost-effectiveness, Saankhya's open solutions will also speed up more innovation in an otherwise closed RAN architecture. Having both - advanced analytics capabilities and RAN Intelligent Analytics platform, the new approach will lead to greater flexibility, scalability, and optimized network resources, as is the intent and the urgent need of the hour.