REFERENCE DESIGN

Converged Multi-Access and Core (COMAC)

ONF TS-102

Version 1.0 | March 2020

About ONF Reference Designs

Reference Designs (RDs) represent a particular assembly of components that are required to build a deployable platform. They are "blueprints" developed by ONF's Operator members to address specific use cases for the emerging edge cloud.

RDs are the vehicles to describe how a collection of projects can be assembled into a platform to address specific needs of operators. By defining RDs, ONF's operator members are showing the industry the path forward to solutions they plan to procure and deploy.

Each RD is backed by specific Operator partner(s) who plan to deploy these designs into their production networks and will include participation from invited supply chain partners sharing the vision and demonstrating active investment in building open source solutions. The RD thus enables a set of committed partners to work on the specification and a related open source platform.

Assembling the set of selected components defined by the RDs into a platform enables a proof-of-concept to allow the test and trial of the design. These platforms are called Exemplar Platforms and each of them will be based on a Reference Design and will serve as reference implementations. These platforms are designed to make it easy to download, modify, trial and deploy an operational instantiation and thereby speed up adoption and deployment.

About the Open Networking Foundation

The Open Networking Foundation (ONF) is an operator led consortium spearheading disruptive network transformation. Now the recognized leader for open source solutions for operators, the ONF first launched in 2011 as the standard bearer for Software Defined Networking (SDN). Led by its operator partners AT&T, China Unicom, Comcast, Deutsche Telekom, Google, NTT Group and Turk Telekom, the ONF is driving vast transformation across the operator space. For further information visit http://www.opennetworking.org

Disclaimer

THIS DOCUMENT HAS BEEN DESIGNATED BY OPEN NETWORKING FOUNDATION ("ONF") AS A **FINAL SPECIFICATION** AS SUCH TERM IS USED IN THE ONF INTELLECTUAL PROPERTY RIGHTS POLICY.

THIS SPECIFICATION IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE. WITHOUT LIMITATION, ONF DISCLAIMS ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF ANY PROPRIETARY RIGHTS, RELATING TO USE OF INFORMATION IN THIS SPECIFICATION AND TO THE IMPLEMENTATION OF THIS SPECIFICATION, AND ONF DISCLAIMS ALL LIABILITY FOR COST OF PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, LOST PROFITS, LOSS OF USE, LOSS OF DATA OR ANY INCIDENTAL, CONSEQUENTIAL, DIRECT, INDIRECT, OR SPECIAL DAMAGES, WHETHER UNDER CONTRACT, TORT, WARRANTY OR OTHERWISE, ARISING IN ANY WAY OUT OF USE OR RELIANCE UPON THIS SPECIFICATION OR ANY INFORMATION HEREIN.

No license is granted herein, express or implied, by estoppel or otherwise, to any intellectual property rights of the Open Networking Foundation, any ONF member or any affiliate of any ONF member.

 A license is hereby granted by ONF to copy and reproduce this specification for internal use only. Contact the ONF at http://www.opennetworking.org for information on specification licensing through membership agreements.

WITHOUT LIMITING THE DISCLAIMER ABOVE, THIS SPECIFICATION OF ONF IS SUBJECT TO THE ROYALTY FREE, REASONABLE AND NONDISCRIMINATORY ("RANDZ") LICENSING COMMITMENTS OF THE MEMBERS OF ONF PURSUANT TO THE ONF INTELLECTUAL PROPERTY RIGHTS POLICY. ONF DOES NOT WARRANT THAT ALL NECESSARY CLAIMS OF PATENT WHICH MAY BE IMPLICATED BY THE IMPLEMENTATION OF THIS SPECIFICATION ARE OWNED OR LICENSABLE BY ONF'S MEMBERS AND THEREFORE SUBJECT TO THE RANDZ COMMITMENT OF THE MEMBERS. A COPY OF THE ONF INTELLECTUAL PROPERTY RIGHTS POLICY CAN BE FOUND AT: https://www.opennetworking.org/organizational-documents/

Converged Multi-Access and Core (COMAC)

Reference Design v1.0

Document Revision Date: March 31, 2020

Document Revision Number: v1.0

Document Release Status: Final Specification

This reference design specification was authored by an operator-led Reference Design Team (RDT) composed of experts from:

Operator Group:

AT&T, China Unicom, Deutsche Telekom, Google, Turk Telecom / Netsia

ONF Supply-Chain Partners:

Adtran, Ciena, Dell EMC, Edgecore, Infosys, Intel

Write to rdspec@opennetworking.org with comments or questions.

<u>Contributors</u>

AT&T: Doug Eng (RDT Co-Chair), Tom Anschutz

Deutsche Telekom: Manuel Paul (RDT Co-Chair), Michal Sewera, Stefanus Roemer

Turk Telekom / Netsia: Mahir Gunyel

Dell EMC: Tim Epkes

Intel: Christian Maciocco

ONF Liaison: Oguz Sunay

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF ABBREVIATIONS

BNG	Broadband Network Gateway
COMAC	Converged Multi-Access and Core
CUPS	Control-User Plane Separation
DNS	Domain Name System
DOCSIS	Data Over Cable Service Interface Specification
DU	Distributed Unit
FMS	Fixed Mobile Substitution
FWA	Fixed Wireless Access
OMEC	Open Mobile Evolved Core
ONF	Open Networking Foundation
PCO	Protocol Configuration Options
PON	Passive Optical Network
PNF	Physical Network Function
RAN	Radio Access Network
RD	Reference Design
SEBA	SDN-Enabled Broadband Access
VNF	Virtual Network Function

DEFINITION OF TERMINOLOGY

FWA	Access technology supporting fixed broadband service offers, through either mobile or fixed broadband portfolio, or both.
FMS	Solution for fixed broadband services, offered through mobile broadband portfolio, in areas where wireline access is not or not yet available.

1 INTRODUCTION

This Open Networking Foundation (ONF) Reference Design (RD) describes the first release for Converged Multi Access and Core (COMAC) implementations. COMAC intends to support use cases and feature needs of multiple operators with a common architecture.

Pursuing a phased approach allows addressing short-term use cases and architecture in this first document release, expecting subsequent document releases to follow that will define additional solution elements and requirements to fulfill the target architecture.

While the RD considers and expects that ONF will develop an exemplar platform guided by this RD, it allows multiple implementation streams to meet the requirements in whole or in parts as a set of modules and compositions that allow for a mix of SDN, NFV and also PNF components to be used as compositional elements in a deployment.

1.1 PURPOSE & SCOPE

COMAC leverages SDN and Cloud principles to create both converged access and converged core capabilities as part of a holistic platform. Built on disaggregated RAN, fixed access, and CUPS-compliant core network components, COMAC's target architecture is a highly optimized stack that aggregates a programmable access layer with a converged Control Plane and a converged User Plane, each containing elements of the disaggregated mobile RAN Central Unit (CU), fixed access network, and core (mobile core and BNG). COMAC facilitates different implementation choices for LTE and 5G as well as PON, WiFi, DOCSIS, mobility, IoT and fixed wireless broadband connectivity. COMAC addresses the realization of the converged User Plane and converged Control Plane using SDN principles. A P4-based realization of the converged User Plane is in scope.

The first step towards realizing the overall COMAC vision is the realization of disaggregated, virtualized components of the User and Control Planes of both mobile and fixed broadband access. In the case of fixed broadband access, this translates to developing a disaggregated, User Plane - Control Plane separated BNG, as defined in the ONF SEBA RD and pursued within the ONF SEBA project. A parallel task that pursues the development of the disaggregated Control Plane and User Plane for 3GPP core is necessary. This constitutes the first phase of the COMAC project which is detailed in this document.

The version 1.0 of the COMAC RD focuses on the development of the disaggregated, CUPS-compliant EPC, with a specific focus towards the enablement of ONF member operators' use cases that are detailed in this document.

Subsequent versions of COMAC RD releases will focus on:

- Support of multiple User Planes, including SEBA, and running standard CUPS-centric User Plane functions on a converged platform with a converged Control Plane;
- The COMAC edge cloud platform as part of a multi-cloud connectivity service realization;
- Disaggregated, O-RAN compliant mobile RAN Central Unit (CU) and Distributed Unit (DU), as well as the Radio Intelligent Controller (RIC);
- The programmable converged User Plane;
- The programmable converged Control Plane;
- Potentially incorporating additional access technologies in this convergence.

RDs represent an abstract architecture and assembly of components that are required to build a deployable platform. They are "blueprints" developed by ONF's Operator members to address specific use cases for the emerging edge cloud.

Assembling the set of selected components defined by the RDs into a platform enables a proof-of-concept to allow the test and trial of the design. These platforms are called Exemplar Platforms and each of them will be based on a Reference Design (RD) and will serve as reference implementations. These platforms are designed to make it easy to download, modify, trial and deploy an operational instantiation and thereby speed up adoption and deployment.

1.2 ASSUMPTIONS, DEPENDENCIES, PROCESS VARIANCES, OUT-OF-SCOPE SUMMARY

This document assumes the following relationships among Reference Design, target, and solutions.

The Reference Design is described in terms of solution elements, expecting implementation stream document(s) will define how to deliver the elements and how to produce control, data and management plane. The most

important of these steps is the target or end state. Target describes the most desired Reference Design in the context of the preferred future environment.

Elements that might be needed to go to market sooner or with near term constraints are defined as Time-to-Market solutions.

Given these relationships, the target Reference Design describes both a set of assumptions and dependencies in addition to the body of the design itself. Because the design may evolve over time, it is also expected that the assumptions and dependencies may also change with the Time-to-Market solutions.

The assumptions for this first version of the COMAC Reference Design are:

- Operators are interested in initial field trials and commercial deployments in 2019;
- The existing carrier automation platforms include both legacy OSS as well as new orchestration systems, e.g. ONAP;
- There are no strictly-greenfield deployment expectations. This means that COMAC will need to be able to comply to standard interfaces and co-exist with existing networks, services, and operational models;
- The 3GPP architecture should be built on CUPS (Control User Plane Separation) the basic architectural principle;
- There are requirements for multiple options to support User Plane functions and these include User Plane on host, offloaded / embedded VNF placement, and merchant silicon;
- Deployment environment: COMAC could be deployed in virtualized infrastructure aggregates, constructed using containers running Kubernetes as the cloud underlay;
- Build toward multiple User Planes, including SEBA in future. COMAC will build toward a standard CUPS-centric User Plane function being developed in SEBA so that SEBA can transition into an eventual converged Control Plane;
- Provide a migration path toward standard CUPS architecture with control and User Plane components that support existing OMEC use cases.

It is recommended that implementations of the COMAC Reference Design are built using:

- Working in a containerized (e.g. Kubernetes) environment;
- CI/CD Tools (e.g. Jenkins, Github, etc.) for development as well as deployment instances.

COMAC is related to other reference designs and mature exemplar implementations and intends to build on/make use of them, wherever appropriate.

1.3 AUDIENCE

The Reference Design team is the current audience of this document (work in progress), per the ONF Reference Design Process - at this stage this document may not be shared outside of the ONF Partners defined at the top of the ONF membership page.

Upon reaching the criteria for an Alpha stage RD, the ONF TLT at its discretion will send drafts to the full ONF membership list. Following the ONF RD process for the timeframe for members to review and comment, and following review of comments by the TLT, the TLT will provide decisions about revision of the document and when to release the RD as a Final Specification.

1.4 DOCUMENT RELATIONSHIP

The COMAC RD is a standalone document.

1.5 SOFTWARE RELEASES

The COMAC project should define software releases as a solution set for the software components. The software release documentation should provide the solution set information for these software releases.

The COMAC software release documentation should also provide the lifecycle management of the compatible releases between these components, to define flexibility and dependencies for coordinated upgrades of the components.

The hardware from vendors may also include embedded software for controlling, monitoring and abstracting low level functions of the hardware, including BIOS, firmware, board support drivers and board management controllers (BMCs). The vendors shall identify the required versions of these embedded software components, and how to upgrade these embedded software components using open software lifecycle management procedures.

1.6 HARDWARE RELEASES

The operators define the hardware solution, including vendors, models, and releases. ONF suppliers do provide value to identify hardware for an ONF Reference Design, and to update the carriers with roadmaps and new product information for enhancements and improved cost.

2 REFERENCE DESIGN TARGET (REL.1 TIMEFRAME: 6-8 MONTHS)

2.1 USE CASES

For COMAC RD Release 1, two use cases will be included: Lightweight EPC for Fixed Mobile Substitution; and Simplified, Disaggregated EPC Allowing Local Breakout.

2.1.1 Lightweight EPC for Fixed Mobile Substitution and Fixed Wireless Access

Goal is a lightweight EPC (lwEPC) implementation to provide a Broadband Fixed Mobile Substitution (FMS) service as illustrated in **Figure 1**. While the location of the terminal in the customer's premises is fixed, mobility support is still required (at lower scale). A relatively small number of sessions are envisioned per instance with significant bandwidth per session. The existing gateway components (PGW and SGW) are centralized and handle the traffic that takes a normal path to the internet. The SPGW components perform the same functions but they are distributed to provide a shorter local path to the internet. Ideally the distributed SPGW software will be based on the same software components that run in the centralized SGW and PGW, except that they are packaged and configured for distributed execution.
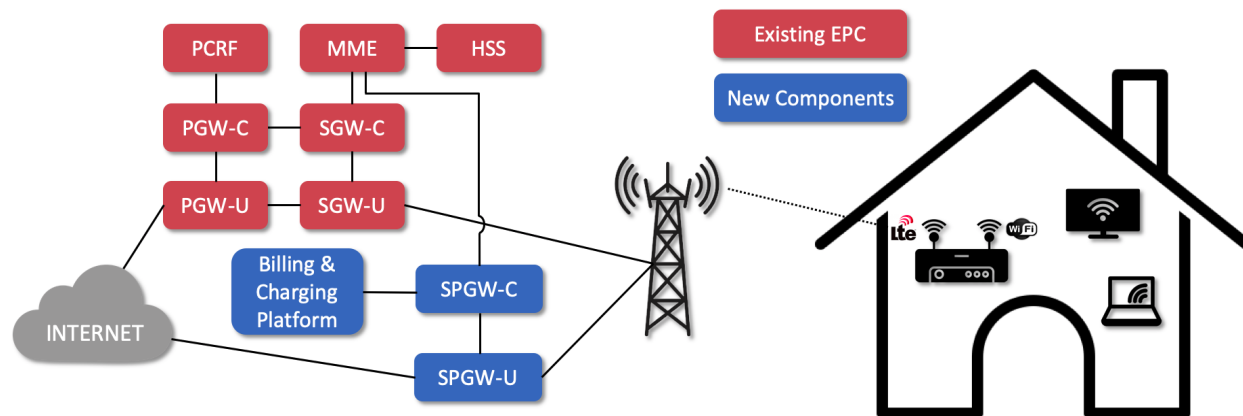
Figure 1: Use Case - Simplified, Disaggregated EPC Allowing Fixed Mobile Substitution

2.1.1.1 Required Features

- Support Fixed Mobile Substitution traffic to be routed to relevant EPC Gateway (based on APN and subscriber charging characteristics profile – using APN resolution extension mechanism)
- Bandwidth cut to 20/60 Mbps per user (derived from subscription);
- Lawful interception (required by law)
- Simple billing (for data retention, required by law)
- 4G support (initially only 4G for MVP POC, 5G as next step)
- Necessary interfaces: S1-U, S11, S5/S8, SGi, X1/X2/X3 (Lawful interface), Ga (Offline billing)
- Generic functions
  - GTP Tunnel en-/decapsulation (TS29.060, Rel.13)
  - GTP-C Echo sequence number
  - CUPS support (TS23.214, Rel.14)
  - Multiple MME support – ability to define MME IP range (CP config)
  - CG-NAT (Optional)
  - DNS in PCO
  - DHCP function (IP pool for APN configuration)
  - Offline billing & LI

2.1.1.2 Key Assumptions

- "Start-small" approach - focus on Fixed Mobile Substitution service only ("Lightweight EPC");
- Limited to 3GPP Gateway with distributed architecture;

- Minimum Viable Product approach (only mandatory features required to go for production).

2.1.1.3   Technical Considerations

- Support both virtualization as well as bare metal approaches for User Plane handling;
- Support 10k users with total aggregate throughput of 10GBit/s per 1U server at minimum;
- Support of standard Linux OS networking mechanisms (routing/switching/monitoring) as part of the solution;
- Acceleration of the User Plane traffic (i.e. use of DPDK, VPP, SR-IOV, etc.);
- Streamlined, simple and cost-efficient implementation of 3GPP-defined core network;
- Support use of simple, open, programmable hardware enabled by disaggregation and use of SDN for both forwarding and management control for the platform;
- Allow customization with NFV, SDN and even legacy network elements;
- Ensure the system is modular with well-defined interfaces to support component flexibility – both SW and HW;
- Support multiple instantiations of subscriber management functions;
- At least, initially no requirement for integration with MANO framework;
- Troubleshooting capabilities (per subscriber) with extensive use of Linux generic toolset (i.e. capturing Control Plane and using plane packets).

2.1.2 Simplified, Disaggregated EPC Allowing Local Breakout

Goal is to provide an EPC-based CUPS-compliant SAE-GW that selects the closest data termination (at the edge site) for a user as illustrated in **Figure 2** below. Minimal impact to existing production systems is essential. As such, the solution should not require any upgrades to eNodeBs nor any changes to Tracking Area configurations.
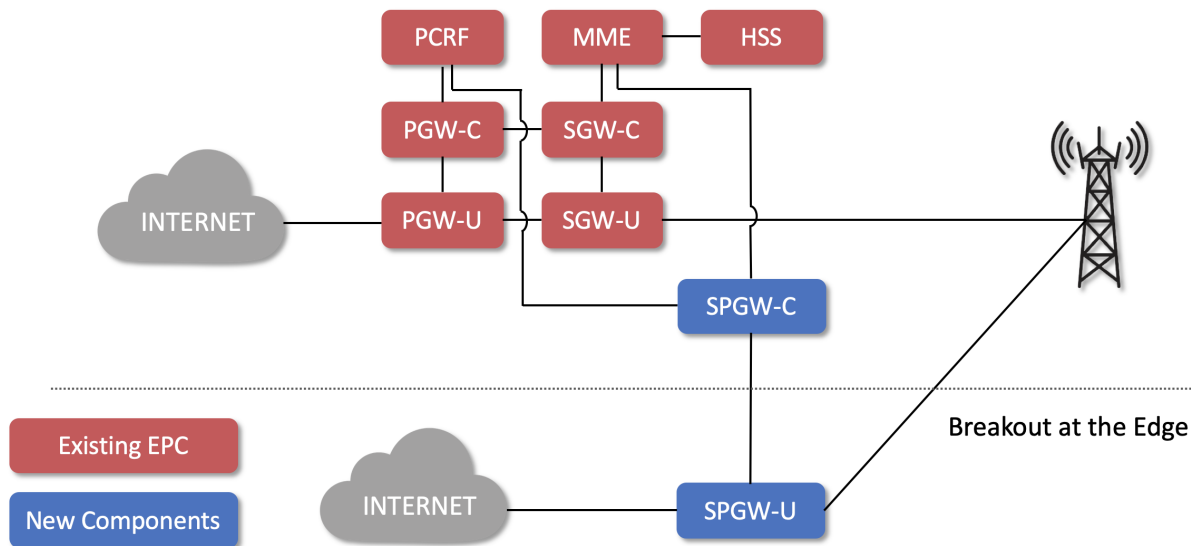
Figure 2: Use Case - Simplified, Disaggregated EPC Allowing Local Breakout

2.1.2.1 Required Features

- SAE-GW is to select the closest data termination for a user
- Support dynamic and operator assigned edges
- TAI DNS server
- eNodeB-ID DNS server
- topon - topology based gateway selection
- GTP-C and GTP-U support
- PFCP support
- Diameter support
- All Stage 2 (TS 23.401) S11 GTP with GTP mobility procedures to be supported
- OAM support
- Restoration and Recovery (TS 23.007)
- Lawful interception (required by law)
- Simple billing (for data retention, required by law)
- 4G support (initially only 4G for MVP POC, 5G as next step)
- Necessary interfaces: S1-U, S11, S5/S8, SGi, X1/X2/X3 (Lawful interface), Ga (Offline billing), Sx, Gx

2.1.2.2 Key Assumptions

- Solution maintains 3GPP compliance while supporting all scenarios;
- Solution is limited to 3GPP Gateway with distributed architecture;

- Solution is accomplished with:
  - TAI DNS Server
  - eNodeB-ID DNS Server
  - topon - Colocation determination

2.1.2.3 Technical considerations

- Support virtualization approaches for the CUPS compliant gateway architecture;
- Support geographically separate User Plane - Control Plane operation;
- Solution to follow the following 3GPP specs: TS 29.244 and TS 29.303;
- Troubleshooting capabilities (per subscriber);
- Modular platform with well-defined interfaces.

2.2 RD TARGET ARCHITECTURE

COMAC is designed as a set of container elements running in a Kubernetes environment. The system is modularized per typical microservice system architectures, and there is a hierarchy of modularity used to allow flexible compositions at different scales.

2.2.1 Release 1.0 Architecture

COMAC v1.0 is a set of components designed and assembled together to support the two usage models mentioned above. The implementation style of these components is left to the operator's choice, e.g. bare metal services, virtual machines, or container, installed with scripts or orchestrated. At this stage there is no specific COMAC platform, the COMAC platform will be introduced with the next version of the architecture.
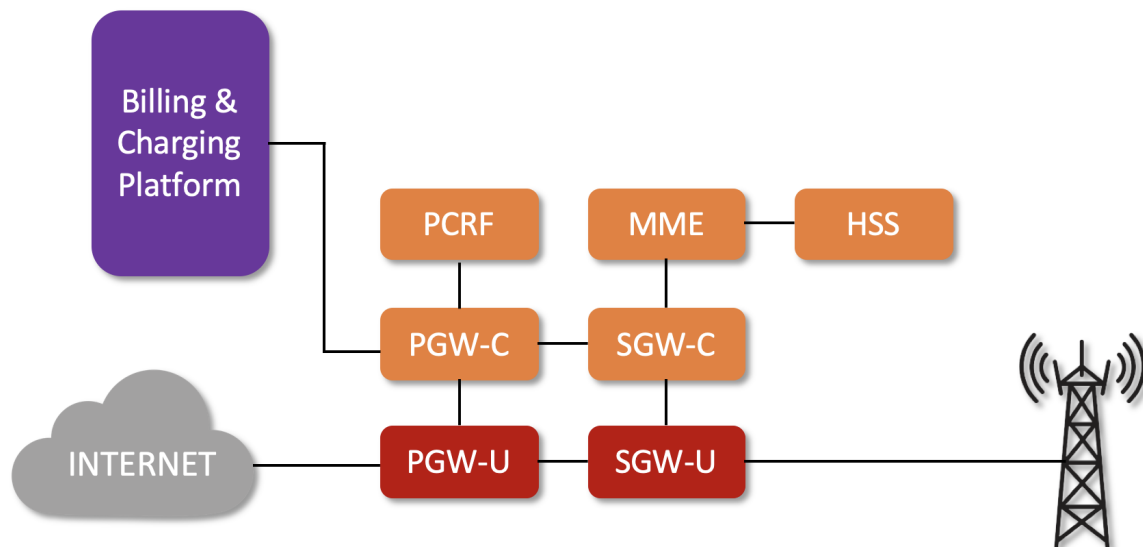
Figure 3: Target Architecture for COMAC Release 1

The high-level hardware and software modules, as shown **in Figure 3** for COMAC Rel.1, include:

- SGW-C, PGW-C, SGW-U, PGW-U (includes embedded PCEF), MME, HSS, HSS Database, Diameter Capability, PCRF, Forwarding Policy Control SDN Controller, Billing and Charging Platform, including CTF and CDF;
- CLI, Logging and Statistics Interface APIs to VNFs;
- Deployment automation for bringing up core network VNFs, associated networking, package installation, provisioning, and configuration.

The focus of COMAC Release 1 use cases are on CUPS-based gateway implementations. The RD expects downstream implementation stream documents to develop and maintain specific implementation details, both from choices of components and from instances or releases of those components.
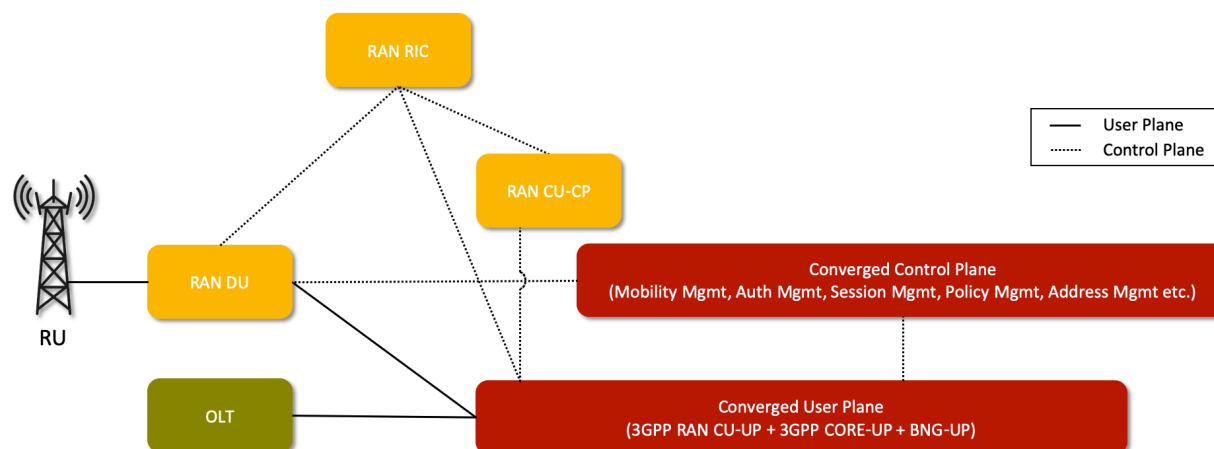
2.2.2 Long-term Target Architecture

Figure 4: Long-Term Target Architecture

COMAC takes a fresh approach to building a unified access and edge cloud platform. Its architecture realizes this vision by first disaggregating all mobile and broadband components, and then recombining and creating common elements, such as User Plane processing, authentication, charging, policy management, etc. that can serve both. Leveraging disaggregation and SDN principles, COMAC's goal is to drive implementations of a cloud-native platform for the operator edge on which operators can modularly deploy all necessary components, including the converged Control and User Plane, to satisfy their desired use-cases.

The current high-level COMAC target architecture is described in **Figure 4**. This architecture might evolve as COMAC work progresses.

Major technical elements the COMAC target architecture will potentially include:

- Converged User Plane: COMAC manages traffic regardless of user's access link. Built on disaggregated, programmable access network (RAN and fixed broadband access, as defined in SEBA) and core (mobile core and BNG) components, COMAC's architecture is a highly optimized stack that aggregates an access layer with an SDN Control Plane and SDN User Plane, which could be implemented using P4 or other mechanisms;

- Converged Control Plane: COMAC enables the user base to be managed as a whole across different access technologies. This includes unified subscriber management, blending authentication and billing functions into a common platform;
- Radio Intelligent Control of RAN User Plane: Compliant with the O-RAN architecture, COMAC provides the necessary interfaces and functionality for its converged User Plane to be software-defined controllable by the RIC controller as well as the CU Control Plane while it is processing RAN CU packets;
- COMAC Platform: a cloud-native, multi-access edge cloud platform that modularly hosts all or some of the following components: converged User Plane, converged Control Plane, RAN CU-C, RAN RIC, and edge applications while plumbing through monitoring data of both the infrastructure as well as the connectivity service, defining services and runtime workflows.

Further details on all of these components will be developed as the COMAC work progresses.

2.3 KEY DESIGN CRITERIA

Each of the key characteristics below represent key industry drivers for highly performant, secure, flexible common solutions that aim to represent a lowest TCO infrastructure for operators.

- Cost-efficient and transparent – startup cost & operations;
- Architecture of HW and SW that enables small start and scalability;
- Open Systems First, Disaggregation by Design, Modularity by Design, Loose Coupling;
- Highly modular HW and SW, including peripherals and acceleration;
- Easy to integrate with automation and management systems and approaches;
- Secure and reliable.

2.4 COMPLIANCE WITH END STATE

This section defines guidelines that the Reference Design (RD) should follow to achieve the desired end state. Include Key Performance Indicators (KPIs) to achieve the technical and business goals.

Guidelines for the end state of the Reference Design include:

- Implementers can use this RD to build the solution;
- Users (i.e. operators) can verify the compliance of implementations to the RD;
- Reliability above a defined number of "9s" for availability of the solution.

3 SOLUTION ELEMENTS AND SOLUTION REQUIREMENTS

There will be multiple options, some trivial and some substantial. The motivation is for an operator to realize a substitutional model for hardware and software selection to enhance features and reduce costs. There may also be variants amongst the operators in the operator group. It is highly desired that operator variants be minimized to the functional components rather than the interfaces between the components.

3.1 PRINCIPLES AND STRATEGIES

Service provider networks are evolving from vendor proprietary PNF solution to VNF hosted on network clouds using open and common, software and hardware. Along with this transition, cloud-native network functions have new software and distribution architectures and need to take on more automation as cooperating components that are loosely coupled to their environment. From this and other concerns the following principles are put forward both for the COMAC architecture and also exemplar implementation streams.

- As much as practical, adopt a Las Vegas principle to the COMAC system. Keep as much complexity inside the system as possible and provide abstract external interfaces. Think of providing services rather than exposing PNFs to manage - i.e. COMAC as a Service;
- Implied by the Las Vegas principle is that COMAC is a locality, not a globally distributed system like the PSTN. There are other systems that manage distributed services, like ONAP and classic OSS. Don't poach on their space;
- Use common hardware and software platforms as much as possible across tenants, businesses, and across the industry. Maintain loose coupling to infrastructure and discover capacity and capabilities that can enable drivers and optimized components;

- Develop key application and workload automation driven by common cloud native tools and being developed and matured in a broad ecosystem beyond just ONF. For example, first look to use the tools and architectures espoused by the Cloud Native Computing Foundation;
- Instrument workload applications with gNMI and gNOI – look to retain and reuse models at higher layers and augment them for streaming management and operations data. Look for opportunities to adopt and make use of data center models and approaches, as they will have the support and innovation from a much broader ecosystem than just telcos;
- Build and use workloads based on frameworks and architecture patterns that are prevalent in cloud systems, as these are going to be easier to re-use, mash-up, and find competent staff to operate;
- Develop architectures, systems, and tools that embrace automation, but before automating an existing system or management process, challenge it to be re-architected to embrace automation and modern frameworks internally and from the ground up rather than only automating the management tools;
- Develop applications tooling (composition, orchestration, operations, billing, etc.) to be flexible across a broad set of runtime environments and tenants. Avoid getting locked into an environment and expect innovation and change in needed runtime environments over time;
- Develop network abstractions that support connection to and among many runtime environments with common APIs and constructs;
- Ensure that applications are not hard coded to message buses, data stores, and protocols. This enables changing, innovation, and consolidation among applications within a locality. For example, an application that can be adapted to a (e.g. Kafka) bus in order to provide its performance and management data is much more likely to be able to re-use a set of performance and management tools that were developed to listen to a bus and collect data for a number of applications. That same application could be modified to use gNMI from OpenConfig, or even NETCONF if that were suitable;
- Don't over-rotate on creating or setting Infrastructure "standards." They are likely to slow us down until technology is quite mature. The foundational infrastructure is likely to evolve and improve over time. We need to be ready to replace or augment an Open Stack or Kubernetes with a version 2 or with an emergent toolset that works better for us;
- Seek to collapse service architectures into a single (or as few as possible) location. This changes the burden on OSS/BSS or top-level orchestrators for managing workloads distributed across locations.

However, leave entitlements and other data lakes centralized so that these distributed architectures can pull from a single source;

- Seek to make data stores common across workloads that share a common abstraction layer. Be sure to layer data in workloads from the deeply technical to a level understandable by customers. Remember that customers really want to help themselves regarding both service selection and management;
- Store data in common data stores at the appropriate abstraction layer. Loosely couple to common data store technologies and tools;
- Store data with sufficient context (metadata) to allow its use by other applications;
- Store data holistically. It's important to understand data dependencies across disparate data islands, and this requires a sufficiently holistic data model to make those connections;
- Federate the centralized data, including both operations data for performance and security as well as customer entitlements. This will allow re-use across services and facilitate more sophisticated services.

3.2 INFRASTRUCTURE PLATFORM

The Infrastructure layer includes the hardware in the solution, including the devices, racks and shelves, powering equipment and connections, and external fibers or electrical cables, and other passive devices. The deployment of COMAC includes the definition of the virtualization approach of the software to the hardware resources.

The basic assumption is that an operator's desire is to standardize on infrastructure environments and leverage large-scale orchestration and lifecycle management in an operator's cloud environment.

3.3 USER PLANE

The following packet routing and forwarding functions need to be supported:

- *MTU Management (Limitation)*
- *Support for IP Packets Fragmentation*
- *Static IP Routing*
- *Dynamic IP Routing*
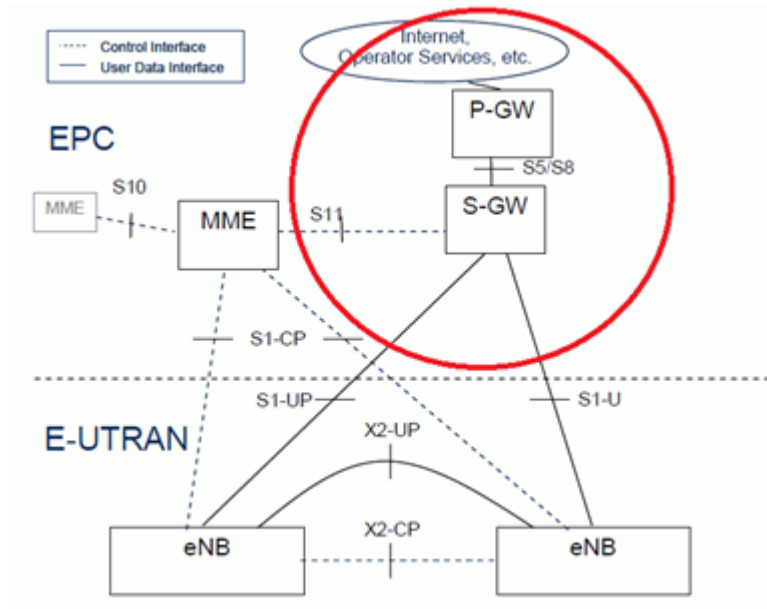- *User Plane IPv6 Support*

3.3.1. Gateway

Figure 5: Gateway Implementation and Supported 3GPP Interfaces

Note: The Gateway may be implemented as combined S/P-GW. In this case the S5/S8 interface would not be required.
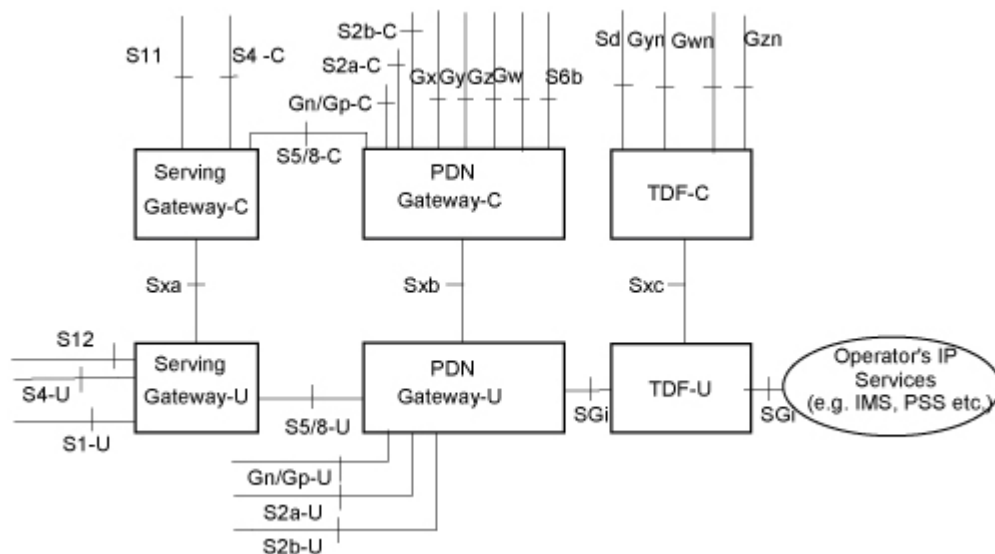

Figure 6: Evolution of 3GPP interfaces with CUPS

3.4 CONTROL PLANE

3.4.1 EPC Procedures

- Attach / Detach
- Tracking Area Update / Routing Area Update
- Service Request
- S1 Release
- Subscribed QoS Modification

3.4.2 Mobility Management Functions

- S/PGW Overload Control (Signalling Storm, Restoration Procedures)
- UL and DL Rate Enforcement Based on APN-AMBR
- ARP/APN-AMBR/QCI for Default Bearer

3.5 MANAGEMENT PLANE

For COMAC, the management plane includes several aspects. This section defines the aspects, as well as high level principles that are desired in the COMAC architecture and also in follow-on implementation streams. Management plane aspects include: OSS/BSS concerns, Cloud Infrastructure concerns, and Internal Concerns.

3.5.1 Principles and Strategies of the Management Plane

Of all the components and subsystems of COMAC, the management plane is the one most likely to require additional flexibility and ease of use. It's envisioned that carriers may use existing OSS/BSS or that they may have completely new systems - as 5G brings significant changes to the system capabilities it becomes a likely trigger to make major changes to OSS. It is already the case that carriers have and use different infrastructure, and we are in a transition from first generation SDN/NFV that typically deployed VNFs that were managed similarly to PNFs to a next generation SDN/NFV that seeks to deploy cloud-native VNFs, and to espouse cloud-centric operations and tooling in lifecycle management.

3.5.2 OSS/BSS Concerns

The first aspect of the management plane is that the system will need to interact with existing and also with future OSS/BSS for carrier deployments. The expectation is that specific linkages to these types of systems will be different for every user. Thus the OSS/BSS functional linkages need to be expressed in an information model that can be easily adapted to various management, orchestration, and protocol requirements.

3.5.3 Cloud Infrastructure Concerns

COMAC is likely to be hosted in a variety of infrastructures with a variety of capabilities and interfaces to allow for workload lifecycle automation. At the top level, COMAC should bootstrap or nestle into a deployment environment by resolving and instantiating capabilities depending on the existing capabilities. For example, COMAC should determine existing message bus, data store, and runtime environment capabilities; then use or adapt existing capabilities; and if not present then instantiate a capability not in the environment. This may be handled through templates and blueprints of various types.

3.5.4 Internal Concerns

To remain loosely coupled with infrastructure, orchestration, and OSS/BSS, COMAC will hide much of its internal complexity from outside systems. However, COMAC also seeks to provide a flexible and modular internal structure - allowing for modules to be open or closed, commercial or custom. This means there needs to be a binding process for the COMAC components to learn about their sources and sinks, providers and consumers. Because it will not be a static system, the binding should not be limited to initial installation, but rather should support re-binding at upgrades and potentially at times when service configurations are changed.

3.6 NON-FUNCTIONAL REQUIREMENTS APPLICABLE TO ALL SOLUTION ELEMENTS

3.6.1 Operations and Management

3.6.1.1 Fault Management

The system must support logging, alarming and diagnostics.

External probes are expected to be used to monitor the liveliness of the system.

3.6.1.2 Configuration Management

COMAC shall provide abstract configuration interfaces. Each implementation should provide simple easy to use APIs providing the minimum required parameters.

Zero outage configuration management:
Configuration should be changed w/o stopping the S-PGW service e.g. MME IP range, APN Pool.

3.6.1.3 Software Lifecycle Management

COMAC shall provide APIs capable of managing software lifecycles for all components. During an upgrade APIs will be provided to track the progress. The APIs shall be able to determine the success of the upgrade activities.

Activities impacting customer service shall be performed in maintenance periods, usually 2 to 4 hours.

3.6.1.4 Accounting

For data retention, as required by law in covered markets/geographies, at least offline billing must be supported.

3.6.1.5 Performance Management

System performance measurements in the internal processing functions are important to measure and understand with respect to system response for control and management transactions, and for scalability of the system to provide a determinate number of operations of a certain type, while the system is operating under a defined load profile (profile of a variety of defined operations at defined frequencies over a defined period of time).

The implementation of monitoring tools to monitor system performance must be considered and selected to minimize their own impact on system performance and resources.

3.6.2 Performance

The system should support 10k users with a total aggregate throughput of 10GBit/s per 1U server at minimum. The User Plane capacity should be extensible by adding servers and/or racks. In order to increase performance, acceleration techniques (i.e. DPDK, VPP, SR-IOV, etc.) or embedded User Plane implementations are required.

3.6.3 Regulatory

To comply with regulatory requirements in the markets where the system will be deployed, Lawful Interception must be supported.

3.6.4 Reliability, Resiliency, Redundancy

Reliability is the definition of the probability of a system or component to function under stated conditions for a period of time. The reliability is also defined in terms of availability of a system or component in terms of number of "9s", such as "five 9s" indicating 99.999% availability.

Resiliency is the ability of a server, network component, or POD to recover from a failure (such as a power failure, or equipment failure) and quickly resume operations.

Redundancy or clustering is employed to help improve both reliability of a POD, and resiliency of operations in a POD.

3.6.5 Scalability and Dimensioning

The system is expected to support scaling out by adding physical servers and logically independent P-GW instances. Each server represents an independent P-GW instance.

3.6.6 Security

COMAC must provide secure communication channels internally between its components, and externally to the operator's management network. The data path channel must not be compromised and provide unlawful access to the management and control channels into the COMAC infrastructure and provider's management network.

3.6.7 Ecosystem

3.6.7.1   *Open Source Software*

Open source software should follow the guidelines of ONF as to the open software licenses that ONF projects can use to incorporate open source. Contributors are responsible to conform to the guidelines to contribute code to ONF.

3.6.7.2   *Open Hardware*

The classification of open hardware will follow the definitions within the OCP, such as the "OCP Accepted" and "OCP Inspired" trademark definitions.

End of

Converged Multi-Access and Core (COMAC)

Reference Design

Write to rdspec@opennetworking.org with comments or questions.