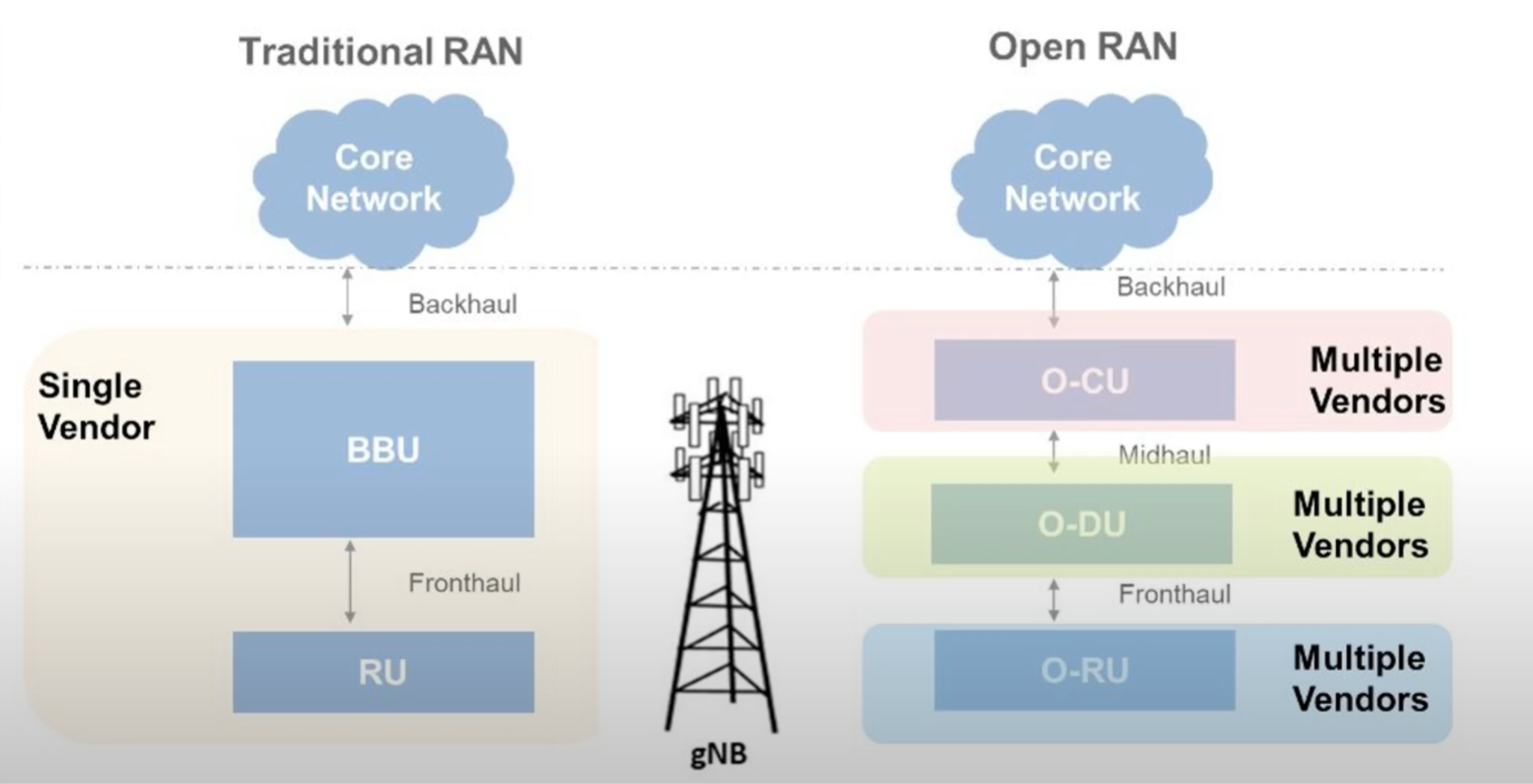
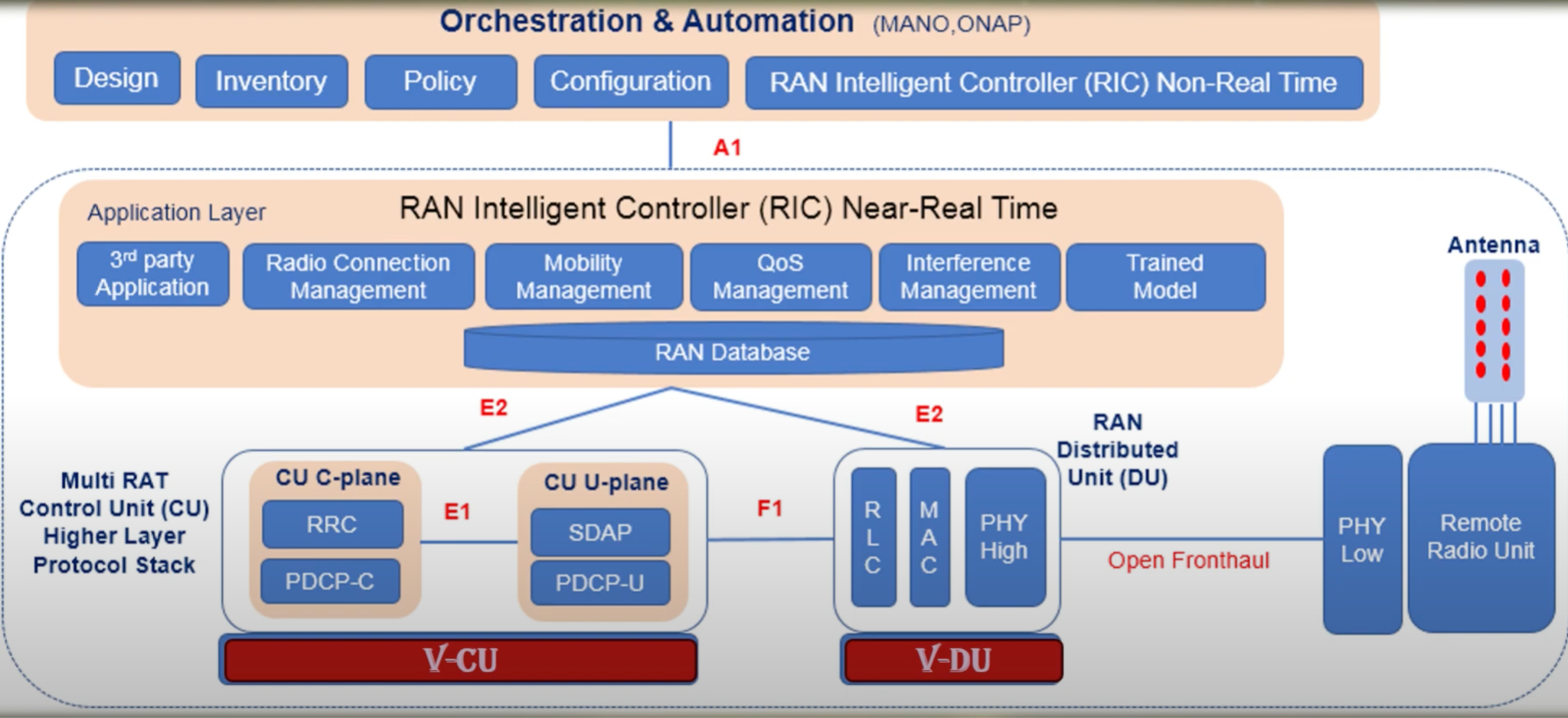
**How did ORAN came to be?**

To understand why ORAN is created, we need to understand the problems with the current RAN (Radio Access Network). Back in the day, due to the complexity of the RAN technology, every component of hardware and software is provided by a single vendor. Which means that telecommunication operators are limited by the specific ecosystem that their vendor provides. In addition to that, since the technology behind RAN is proprietary, operators have no way of escaping that limitation.

This fundamental problem is why ORAN nor Open Radio Access Network is a set of network architectures that enables service providers the use of non-proprietary subcomponents from a variety of vendors.



**What defines ORAN?**

Most of the ORAN framework is still being research today. In addition to the nature of ORAN being open sourced, the specific details of ORAN is loosely defined. However, from an architecture point of view, ORAN is divided into 4 main components: Open- Radio Unit (O-RU), Open-Distribution Unit (O-DU), Open-Central Unit (O-CU), and RAN Inteligent Controler (RIC)  
  


These four components of ORAN, have some key differentiator compared to ORAN which are:

* Non-proprietary hardware
  + Allows operators to use of the shelf hardware, reducing cost, and increasing flexibility
* AI and ML models
  + Enable intelligent optimization of network performance and resource allocation
* Disaggregation
  + Seperates hardware and software components, allowing for more flexibility in network design
* Software and API support
  + Provide access to wide range of software and APIs for easier integration and customization
* Open Source
  + Encourages collaboration and innovation by making the source code freely available for modifications and distribution
* Cloud
  + Enable deployment of network function in a cloud environment, increasing scalability and efficiency

**O-RAN Architecture**

Near-RT-RIC: Acts as the brain of the near-real-time control plane, positioned at the network edge close to O-DUs to ensure rapid responses within 10ms to 1s. It dynamically optimizes RAN elements according to network conditions and user needs, with a focus on managing radio resources, beamforming, and power control.

O-RU: The O-RAN Radio Unit serves as the physical link between mobile devices and the network. It handles the transmission and reception of radio signals to and from devices like smartphones and tablets, converting these signals into digital data and vice versa for communication with the broader Open RAN system. O-RUs can be deployed across various frequency bands to meet specific use cases and coverage requirements.

O-DU: The O-RAN Distributed Unit is a logical node within the O-RAN architecture, hosting protocols such as the radio link control (RLC), medium access control (MAC), and physical interface (PHY). It’s responsible for real-time L1 and L2 scheduling and modulation functions.

O-CU: The O-RAN Central Unit is a logical node within the O-RAN architecture that hosts the RRC, SDAP, and PDCP protocols, handling non-real-time, higher L2, and L3 tasks.

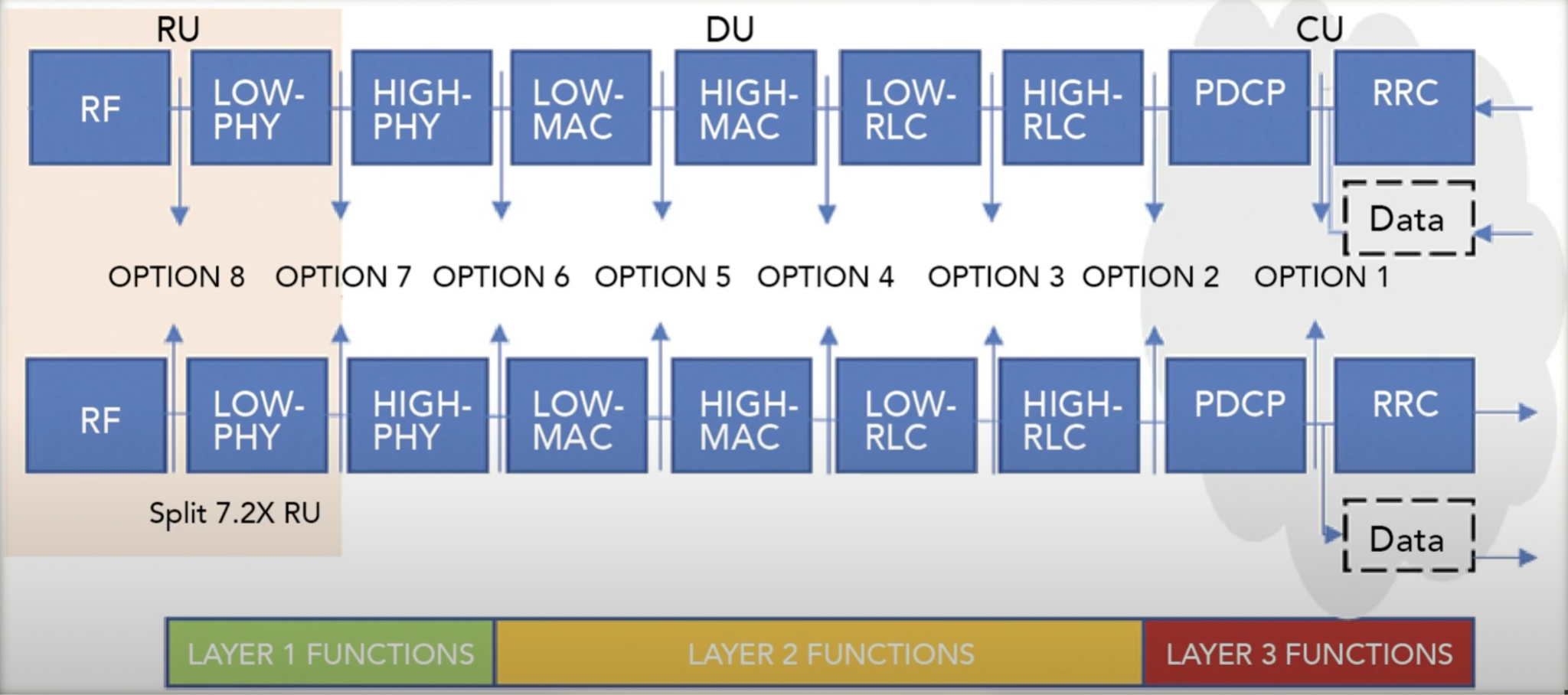
O-RAN Fronthaul: Defines the interface and protocols connecting the O-DU and O-RU components within an O-RAN network.

**O-RAN Fronthall**O-RAN Fronthaul refers to the interface and communication protocols that connect the Distributed Unit (O-DU) and the Radio Unit (O-RU) within an Open Radio Access Network (O-RAN) architecture. This interface is crucial for the transport of data, control, and management information between the O-DU and O-RU, enabling the coordinated operation of the RAN components. The O-RAN Fronthaul supports the transmission of various types of signals, including user data and control signals, and is designed to be flexible and interoperable across different vendors' equipment, which is a key characteristic of the O-RAN architecture. This flexibility allows network operators to mix and match components from different manufacturers, promoting innovation and reducing costs.

ORAN Fronthall can be divided into 4 main components which are:

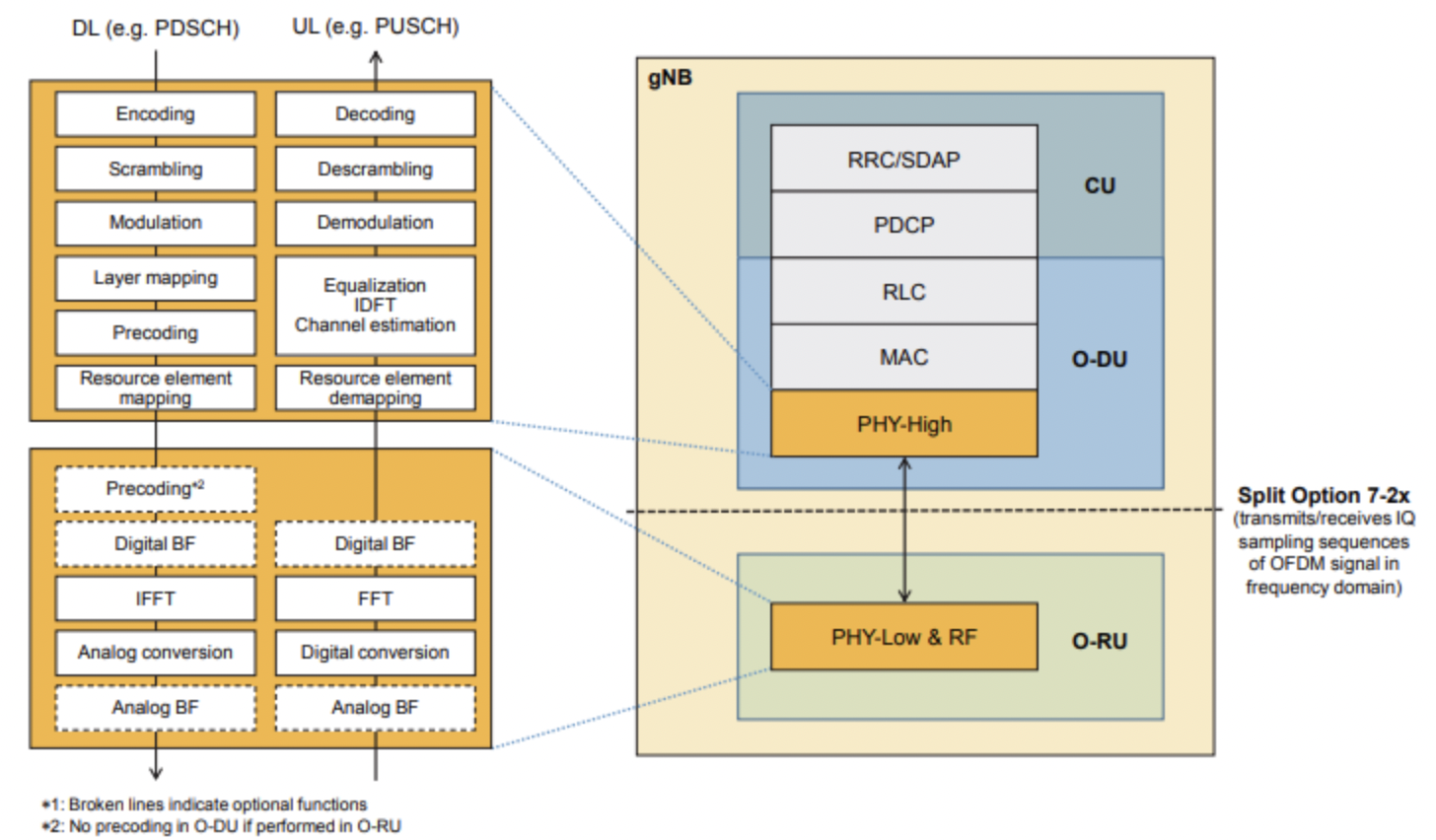
* Control Plane (C-plane): Carries configuration and management messages for the radio link.
* User Plane (U-plane): Transmits actual user data (like voice and video) between the DU and RU.
* Synchronization Plane (S-plane): Ensures precise timing alignment between the DU and RU, crucial for accurate signal transmission.
* Management Plane (M-plane): Handles the management of the O-RU from the O-DU, including fault management, performance management, and security management.

**O-RAN Split Option**

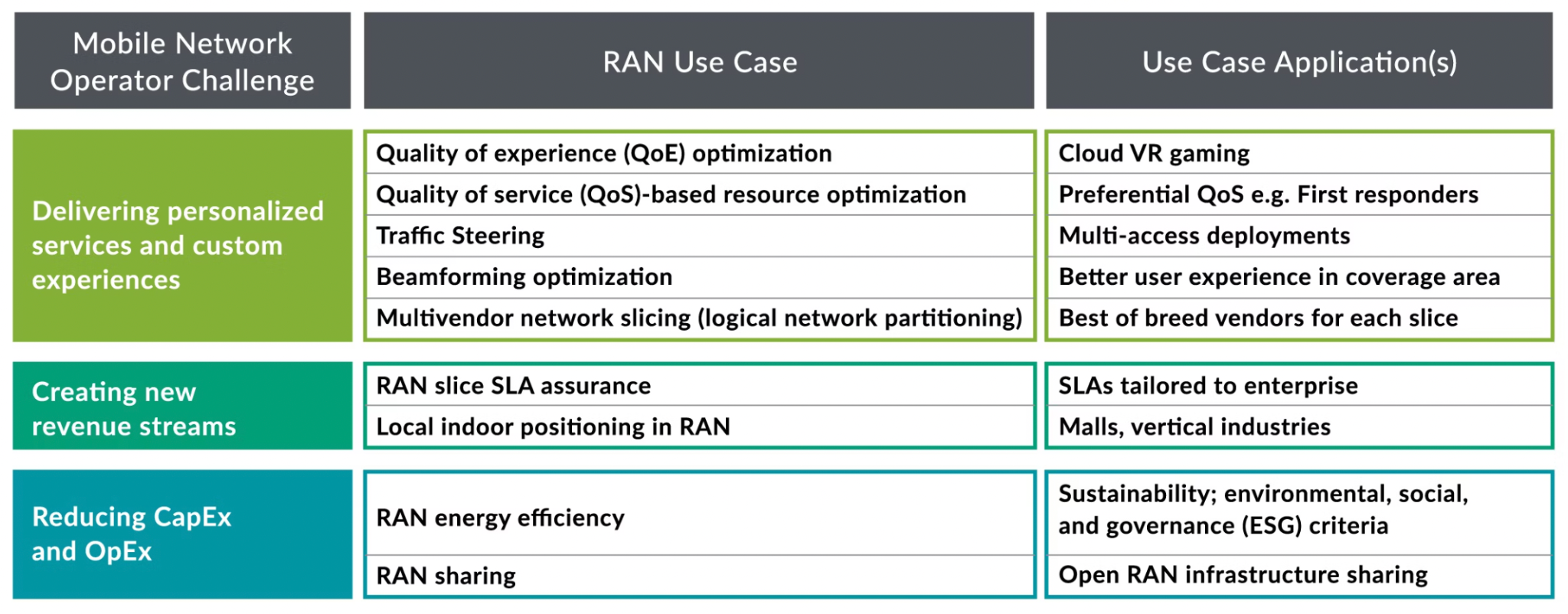
The ORAN split option refers to the way functions and tasks in the Radio Access Network (RAN) are divided, or "split," between different network components, primarily the Radio Unit (RU), Distributed Unit (DU), and Central Unit (CU).  ****

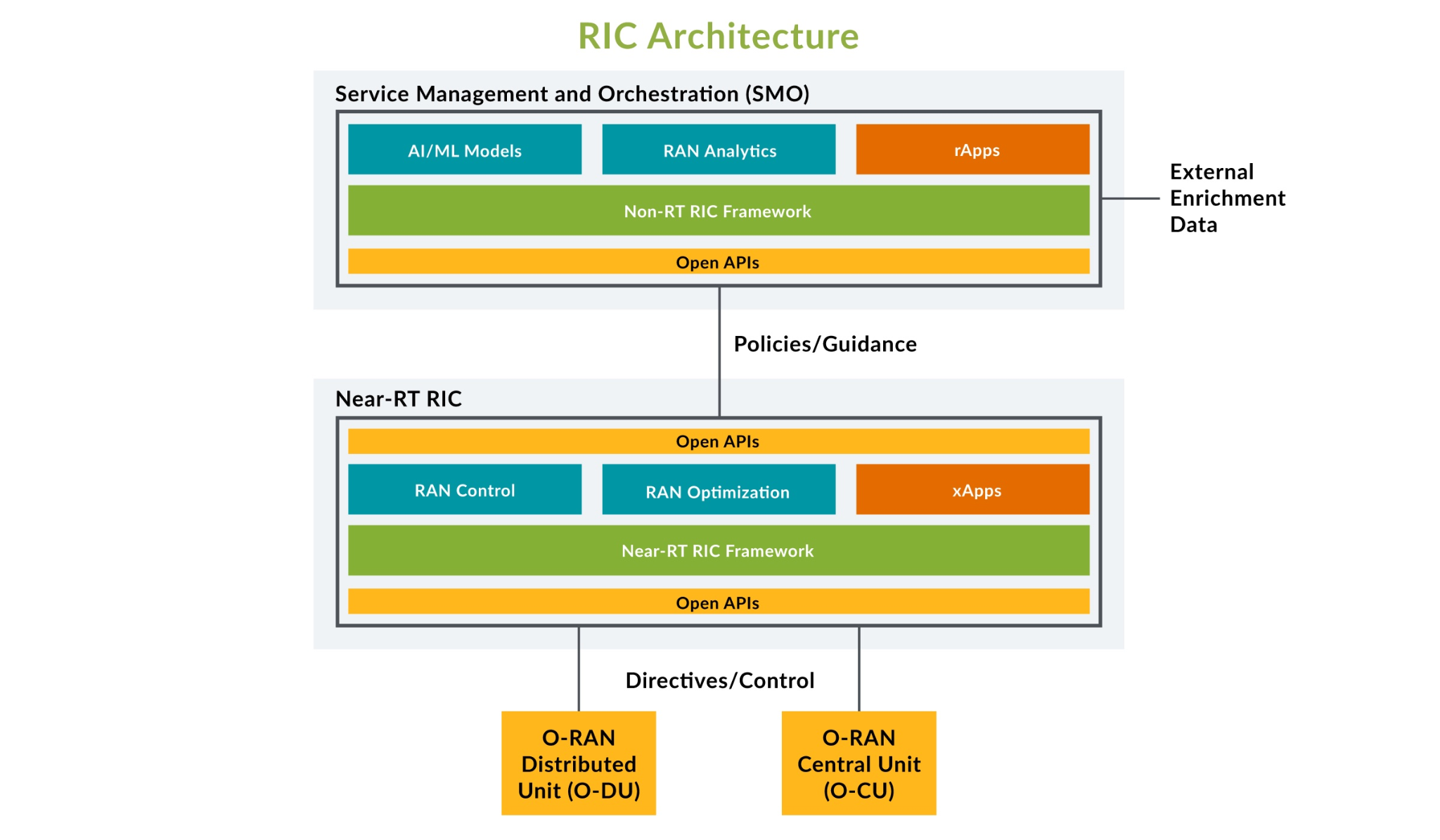
One of the popular ways to spilt oran is through Split 7.2x. In Split 7-2x, the physical layer (PHY) is split between the RU and DU. The lower PHY layer is handled by the RU, while the higher PHY functions, including MAC and RLC, are managed by the DU.

This approach offers a good balance between performance and transport requirements, making it a popular choice for many operators.Which is why it is often used for centralized and distributed RAN deployments, where there is a need for a compromise between the complexity of the fronthaul and the performance of the network.



**RIC**

A RAN Intelligent Controller (RIC) is a software-defined component of the Open Radio Access Network (Open RAN) architecture that’s responsible for controlling and optimizing RAN functions. The RIC enables the onboarding of third-party applications that automate and optimize RAN operations at scale while supporting innovative use cases that lower mobile operators’ total cost of ownership (TCO) and enhance customers’ quality of experience (QoE)  
  
The RIC is divided into non-real-time and near-real-time components. The non-RT RIC is an element of the operator’s centralized Service Management and Orchestration (SMO) Framework, as defined by the O-RAN Alliance. Using specialized applications called rApps, the non-RT RIC enables > 1-second control of RAN elements and their resources. It also uses network data, performance metrics, and subscriber data to provide AI-based recommendations for network optimization and policy guidance to xApps running on the near-RT RIC, which in turn provides policy feedback to the non-RT RIC. The near-RT RIC resides within a telco edge or regional cloud and typically enables network optimization actions that take between 10 milliseconds to one second to complete



**Security Challenges in RIC**

The RIC centralizes control over radio functions and enables network programmability, adopting SDN controller principles. However, this centralization could make networks vulnerable to intrusions, such as DoS attacks, and unauthorized access to critical APIs, potentially leading to network collapse. Although RIC supports integrating security algorithms via xApps, there isn’t yet a standardized framework for securing Open RAN architecture. This new technology introduces significant security challenges, as novel infrastructure and systems bring fresh threats.

**RIC Vulnerabilities**

* Uncertified access to the radio resource

The deployment of Open RAN requires diverse radio resources from multiple suppliers, increasing the demand for third-party access. The RIC, crucial for managing xApps and APIs, faces challenges in integrating reliable applications from various sources. While open-source software offers flexibility and cost savings, it also introduces security risks like insecure xApps, API exploitation, and software vulnerabilities.

* Insecure xApps: xApps running on the RIC, often developed by third parties, have privileged access to network resources and can modify RAN functions. If compromised, they can disrupt services or access sensitive data, posing significant security risks.
* API Exploitation: The RIC provides various APIs for functions like mobility, handover, and RAN optimization. However, these APIs are vulnerable to exploitation, potentially granting unauthorized access to sensitive network data and functions.
* Open-source Software Vulnerabilities: Open-source components in RIC and xApps, while beneficial, can introduce risks like known vulnerabilities, malicious backdoors, and compliance issues, which attackers can exploit, compromising system security.
* Unauthorised access to UE identifications

The UE identity is crucial in the RIC for UE-specific control, making its exposure a major security risk that could compromise user privacy and data confidentiality. For instance, UE identifiers like SUPI are frequently exchanged over the E2 and A1 interfaces. If an attacker gains access to these interfaces, they could use a xApp to identify specific UEs. Additionally, vulnerabilities like SDL API exploitation could allow attackers to access sensitive UE information in the UE-NIB database, including real-time user locations, exposing personal data to potential threats.

* Conflict and inconsistency in radio access policies

RIC allows for easy programming of mobile and wireless networks via xApps, but this can introduce security vulnerabilities. Configuration changes or decisions from multiple xApps can cause inconsistencies, and the RIC lacks mechanisms to prevent conflicts that could bypass mandatory network policies.

* + RIC and gNB: The unclear division of responsibilities between RIC and gNB, depending on xApps and gNB functionalities, can create discrepancies. This ambiguity could be exploited, for example, by a rogue xApp triggering radio resource management actions that conflict with the gNB, potentially leading to a DoS attack.
  + Multiple xApps: Different xApps developed independently might conflict, especially if they attempt to alter the same parameter for the same user simultaneously. These conflicts can lead to network inconsistencies, creating vulnerabilities that attackers could exploit.
* Network misconfiguration  
  The RIC manages and controls the configuration of network elements in the data plane, receiving information from the RAN and core network to optimize performance. However, the dynamic and complex nature of the network increases the risk of misconfiguration and faults. Reducing network complexity is essential to avoid these risks.
  + Integrating Multiple Suppliers: Incorporating components from various suppliers in O-RAN’s data plane adds complexity, which RIC must manage. This complexity can lead to errors, increasing the risk of misconfiguration, network outages, and security vulnerabilities.
  + Decoupling Hardware and Software: Decoupling hardware and software, especially when using components from different suppliers, increases the risk of misconfiguration. Security features may depend on specific hardware or software combinations, and improper integration can lead to vulnerabilities. Careful selection and integration are crucial to ensuring security features are effectively utilized.
* Poisoned AI/ML RAN functions

AI can significantly boost network efficiency and security, but poorly designed AI frameworks can create new security risks. In O-RAN, the RIC uses AI/ML to manage and optimize RAN functions but may also become a target for attacks, such as injecting poisoned models that could disrupt network operations. Open interfaces in O-RAN, which allow data exchange and integration with multiple vendors, increase the risk of AI/ML model poisoning. The lack of standardization for these interfaces makes it easier for malicious actors to introduce harmful data or exploit vulnerabilities. Therefore, securing the RIC and its AI/ML models is crucial to prevent these risks.

* Incomplete or inadequate hardening of the RIC

default settings and configurations are secure. Without adequate hardening, RIC components may be exposed to attacks, such as malware injection, unauthorized access, or operational disruption. Key vulnerabilities include:

* OS Vulnerabilities: Default passwords, back-door accounts, and insecure services in the OS can be exploited by attackers.
* Software Vulnerabilities: Flaws in RIC software can be used to disrupt operations or reconfigure the network for further attacks.
* Cryptographic Key Management: Weak or poorly managed cryptographic keys can compromise the security of RIC functions and information.
* Patch Management: Missing security patches can leave RIC components vulnerable to exploitation.
* Log and Audit Mechanisms: Inadequate protection of security logs can delay response to incidents and lead to undetected attacks.