The Operational Impacts of Supporting a Disaggregated, Distributed, Cloud-based Network Architecture

A Technical Paper prepared for SCTE by

Aliraza Bhimani
Principal Network Engineer/Team Lead
Comcast Cable
New Jersey
609 929-3346
aliraza_bhimani@comcast.com

Idris Jafarov
Delivery Team Leader
DriveNets
New Jersey
302-559-5952
ijafarov@drivenets.com

Table of Contents

Title **Page Number**

List of Figures

Title **Page Number**

List of Tables

1. Introduction

Managing and maintaining highly scalable networks has historically been a challenging task. A plethora of ISPs/CSPs have been trying to simplify processes and procedures; yet this task gets more complicated as they are faced with the growing cost pressure of supporting today's IP network traffic demands (driven by video, gaming, and remote working) and future 5G/6G cellular traffic volumes. Based on several reports, the internet usage has increased by 1,355% over the last 22 years [1]! With the coming deployment of the Full Duplex DOCSIS 4.0 system in the access networks [2] and the proliferation of the 400/800Gb/s Ethernet technology [3] in the aggregation/core layers, traffic utilization could continue to accelerate in the upcoming decade.

As Mannan Venkatesan, a Distinguished Engineer from Comcast Cable, notes in his NANOG N81 2021 presentation:

> "There is a significant amount of port growth we need to support. We all know the internet traffic has exploded within the last decade. The ports you would need to support the traffic are also proportional to the traffic volume, so we must make sure we stay on top of the port capacity we support on core and aggregation routers" [4].

All these trends and market forces are impelling cable operators to rethink and rearchitect their existing IP networks and operations to maximize performance and efficiency.

Network operators must also consider the total cost of ownership of various hardware and software options, not just that of an individual component or software option. Network components don't operate in a vacuum, so choosing feature-light hardware may necessitate greater investment in hardware and vice versa. There is always a tradeoff between feature-rich software and bare bones hardware. It's critical for network operators to understand what's needed for end-to-end delivery, rather than just individual component costs and specs, which may not paint the whole picture.

Due to the above-described forces, a new innovative system is desired to solve these problems. The primary requirements for such a system are that the cost per bit and time to market of new services be reduced, and yet be able to permit the simplicity of the network operations while minimizing the blast radius or failure zone of the network. Losing one router has the potential to impact millions of customers and puts the network in a hazardous condition. Another failure can isolate the whole network, either the one managed by the operator or a peer's network. The main goal is to reduce the blast radius and impact of one router or component and increase the availability of the network.

Disaggregated Distributed Chassis and Disaggregated Distributed Backbone Router (DDC and DDBR) are powerful solutions based on open-source specifications that solve current operational challenges when building and scaling IP backbone/aggregation networks [5, 6]. These two options addresses and mitigates many of the issues described above. These models arm operators with the flexibility of selecting the best breed of IP products in the market.

In this paper we dive into how cable operators can leverage open transport building blocks across different segments of their transport networks (access, aggregation, backbone) and implement concepts of real DDC/DDBR architecture, while utilizing orchestration, automation, and analytics. We will also touch upon how they can overcome operational challenges and pitfall considerations while proactively positioning themselves for future disaggregated network solutions.

2. Disaggregation as a success criteria for ISPs/CSPs

2.1. DDC and DDBR

Disaggregated Distributed Chassis and Disaggregated Distributed Backbone Router are open-source specifications for carrier grade routing systems put forth by Open Compute Project (OCP) and Telecom Infra Project (TIP), respectively, which are global collaborative communities focused on innovation and development of open, disaggregated, and standards-based technology solutions [7, 8]. One of the main motives for the inception of these organizations was the huge influx of new registered users and data which resulted in an exponential growth of services and platforms developed and deployed by hyperscalers. These developments incurred unforeseen control costs and energy consumption which can only be optimized by the benefits of open source and open collaboration to hardware. This collaboration model is now being applied to advance the telecommunications industry as ISPs/CSPs experience similar pressures of unprecedented traffic growth.
Telcos and Cablecos are rigorously researching, testing, and deploying disaggregated networking solutions across their footprint. To cope with the tremendous demand, they have been envisaging an evolution path to their core/aggregation networks to introduce innovation, efficiency and mostly openness. Ideas have been entertained to initiate a shift in the IP backbone architecture from the traditional single chassis-based routing systems to more disaggregated distributed ones.

ISPs/CSPs not only desire to meet current needs when deploying core/aggregation transport networks but also staying ahead of the evolving trends in terms of resiliency, capacity scaling & End-To-End (E2E) network automation.

Before going into the nitty gritty details of the challenges that operators face, let’s review the key points of the DDC/DDBR solution [5, 6]:

- **Disaggregation is driving competition**: creating opportunities for new players in the market driving costs down.
- **Pay as you grow**: model that allows operators to purchase capacity incrementally as it is needed.
- **Innovation**: open software and hardware to improve flexibility and innovation while reducing time to market.
- **Operational Efficiency**: Taking advantage of centralized control and monitoring tools.
- **Reliability**: Always targeting higher availability & multi-level redundancy while minimizing blast radius impact to decrease customer impact and outages.

This system can be placed not only in the IP/MPLS backbone but also in the access layer and act as an Internet Gateway Router (IGW), as depicted in Figure 1. The same hardware can be utilized for these routing functions regardless of Network Operating System features or implementation.

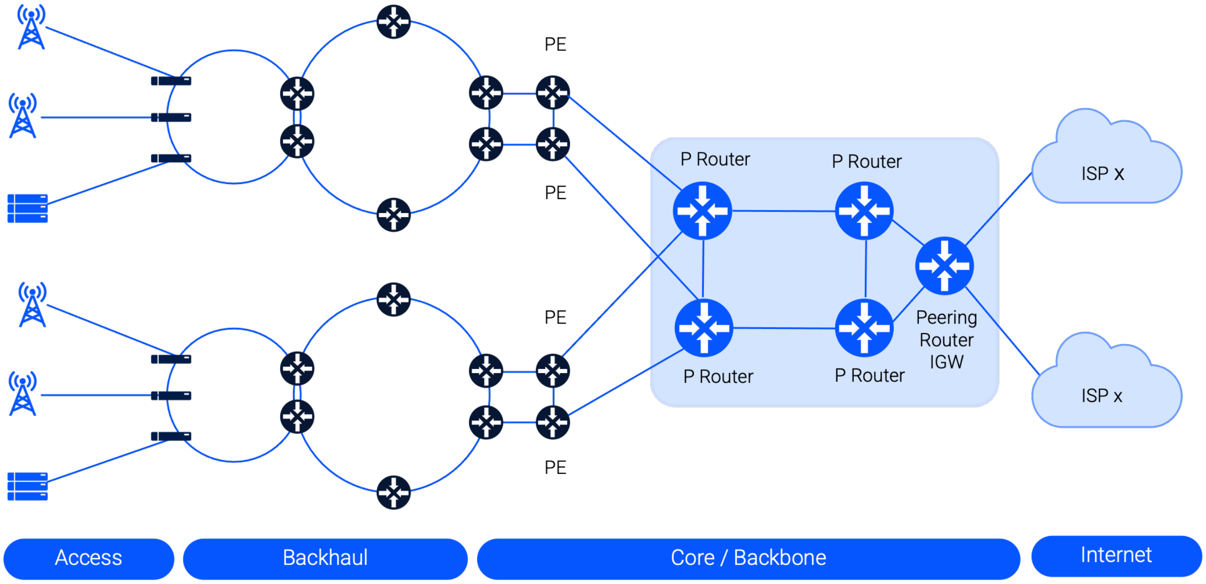

Figure 1 - DDC/DDBR placements in the network

2.2. Difference between DDC/DDBR and traditional routers

2.2.1. Modular Chassis

To understand why DDC/DDBR is an evolutionary architecture, a reasonable convenient starting point would be to analyze the traditional routing platform architecture first.

Figure 2 shows a front and back view of a finished product view of a typical modular chassis routing system from one of the vendors in the ecosystem.

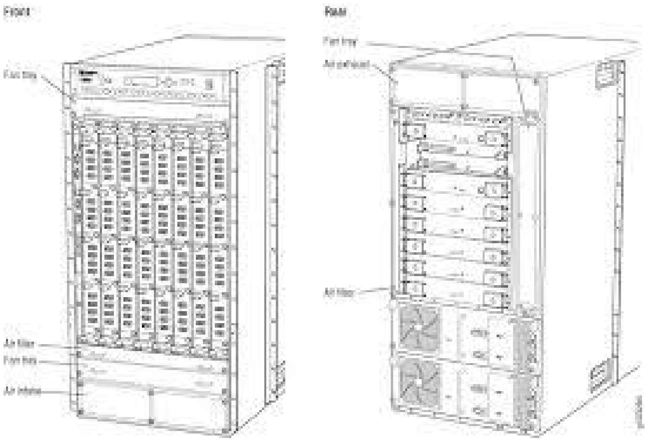

Figure 2 - Front and Back View of a Modular Chassis [5]

There are multiple components that comprise this system which reside inside the chassis and are not visible: line cards, fabric modules, Route Processor modules, PSUs, and FAN FRUs to name a few. Modular chassis routers use pluggable line cards to scale up capacity and service features and form the foundation of ISPs/CSPs' backbone and aggregation networks today. If a service provider needs more ports, more linecards can be bought and inserted into the system. The growth ceiling for this type of system is dependent on how many available slots there are on a modular chassis. One more option to scale up is to purchase a totally new chassis with higher port density and replace the old one due to the scarcity of slots for line cards. An example would be replacing a ten-slot chassis with a twenty-slot chassis.

Edson Erwin invented the highly scalable Clos architecture/Clos Network in 1938 and which then was formalized by Charles Clos in 1952. Later in 1953, Charles Clos published a paper titled "A Study of Non-blocking Switching Networks" in the Bell System Technical Journal [21] where he describes a method of designing arrays of cross points for use in telephone switching systems. In recent years, the variation of a Clos network had been widely deployed by hyperscalers and is now being adopted by the largest telcos and cablecos.

Another possible course of action is the scale-out (horizontal disaggregation) model. This model decomposes the chassis system into a spine and leaf Clos architecture which has been pioneered and successfully adopted in data center designs by hyperscalers. ISPs/CSPs are now making a significant effort to take these concepts and apply them to the routed Wide Area Network (WAN).

As can be seen, these traditional routers mainly scale up hardware, with some scale out options existing such as multi-chassis racks or back-to-back multi-router options. It is viable to architect incremental growth, but then there is a need to account for the same incremental up-front cost for cooling, power, space along with rack, stack, and cabling installation costs. The power per rack may be limited at the facilities and new next-generation single chassis systems have very high-power demands, more cooling is needed and there may be space limitations at the headend or datacenter. This traditional option is not portable.

At one time, this routing system (RS) architecture was an effective way to build backbone/aggregation networks. It was a one-stop shop to have all your port, fabric, and software needs in one fixed chassis. This made troubleshooting easier as well since the networking vendor would be responsible for any software or hardware issues encompassing the whole chassis. The limitations of this system lie in its mechanical design to solve thermal and spatial challenges and live Online Insertion and Removal (OIR) of modules within the system which sometimes can cause traffic impact and may require multiple maintenance windows to ensure business continuity. On top of that, the mentioned hardware design and the software controlling the system are solely proprietary to the specific vendor thus preventing potential ecosystem players to participate and compete in either HW or SW market.

However, with the evolution of networks and traffic patterns, ISPs/CSPs had to adapt and adopt disruptive and innovative technology solutions. DDC/DDBR RS architecture that resembles Clos Spine-Leaf (S/L) topology in terms of interconnections between elements but operates as a single routing system is the optimal way for operators to accelerate business agility and drive revenue growth. This solution also allows per port or speed step function which we will discuss later.

2.2.2. *Disaggregated Routing System*

So, you may wonder, what is a Distributed Disaggregated Routing System? All the elements that were shown in the Figure 2 traditional refrigerator single chassis router; namely the line cards, fabric modules and route processors are all disaggregated onto separate physical "pizza" whitebox and commercial off-the-shelf (COTS) x86 server components that run virtual machines and software containers. This variation of Spine/Leaf Clos design allows various scalability options. Clos has been already deployed previously in datacenters, backbone/core networks for many decades as we will learn more about later. Now, we have the ability of taking it one step further to the regional area networks or closer to the edge.

In Figure 3 we can see a conceptual diagram that demonstrates such a system from a birds-eye View [5]:

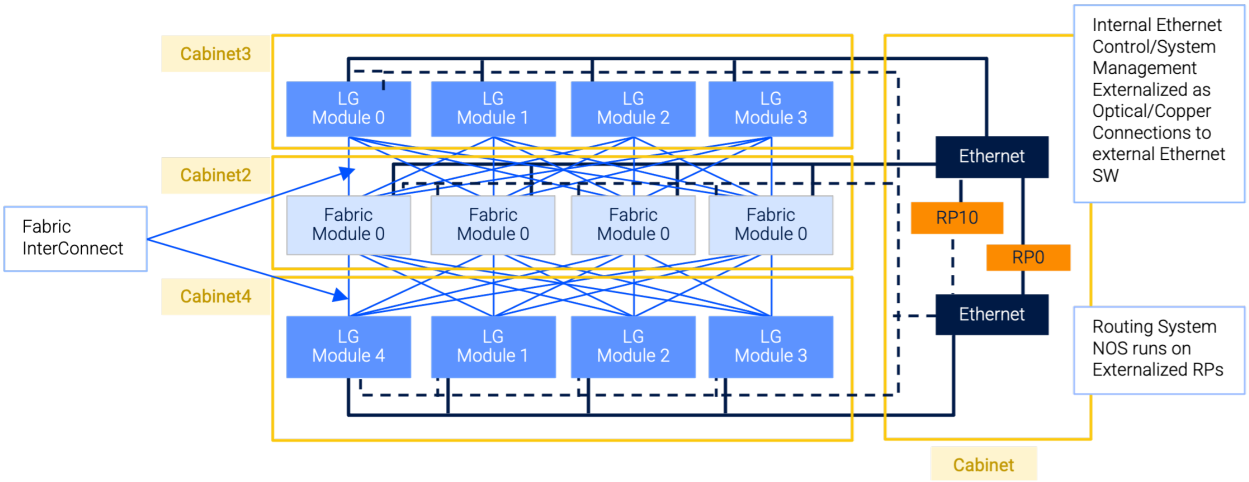

Figure 3 - High-Level View of a DDC Routing System

In this graphic, the disaggregated distributed RS is comprised of a certain number of fixed-RU Packet Forwarders (PF) which represent the leaf of the cluster and act as the line cards, and several Fabric Forwarders (FF) which represent the spine of the cluster and act as the backplane. The "brains" of the system are running on redundant on-premises COTS x86 servers or as containers in a cloud-native fashion which execute all the sophisticated route computation algorithms and performs management of the whole cluster. The external ethernet switches provide distributed communications channels between all 3 described components. This routing system can be implemented across multiple physical racks with proximity at a headend or data center, thus making it portable and having the flexibility to overcome any potential space or rack issues.

Packet forwarders perform strictly forwarding functions and do not store protocol state information which allow them to support line-rate forwarding across all ports without any limitation. Fabric forwarders are responsible for interconnecting all PFs in a full mesh and transporting cells between them. The interconnections between the PFs and FFs are not IP based but rather leverage proprietary cell-based technology from multiple merchant silicon manufacturers to achieve optimal redistribution of traffic across fabric links.

Figure 4 demonstrates an example of how PFs can be interconnected to FFs in 3-stage non-blocking Clos topology. The S/L based architecture has helped the web-scale companies to efficiently grow their infrastructure to a massive scale that can deal with many technical challenges they have faced before. There is an ability to scale vertically by adding more FFs or spines to increase the fabric throughput and redundancy, while also allowing to scale horizontally by adding more PFs or linecards to accommodate port demands and future growth.

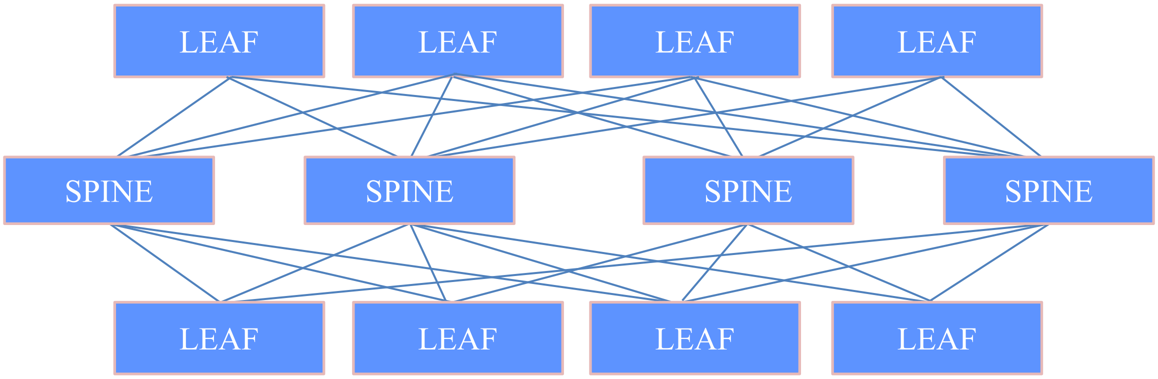

Figure 4 - 3-stage non-blocking Clos topology

However, the Spine & Leaf architecture by itself has multiple drawbacks when trying to apply it to the carrier grade networks. We will provide a detailed comparison of these 2 architectural approaches in an upcoming section.

The combination of the Spine & Leaf based architecture and disaggregation can lead to phenomenal advantages such as disjointed innovation paths between data plane and control plane. As control plane is completely decoupled from the data plane, the innovation of these 2 components can happen independently. For example, with the evolution of the ecosystem ISPs/CSPs can easily change Routing Network Operating System vendors according to their current needs while using the same hardware or even combining hardware from different Original Design Manufacturers (ODM). In recent years, merchant silicon manufacturers have developed products that are on par with custom silicon developed by Original Equipment Manufacturers (OEM). Disaggregation enables operators to purchase simplified hardware based on merchant silicon resulting in significant cost savings as compared to traditional modular chassis routers. Now, ISP/CSP's do not have to be bound to a specific router vendor's hardware/software roadmap and can freely mix and match different whitebox server hardware with custom vendor software applications/containers.

Multiple modern network operating systems (NOSs) are based on Linux, however, most of them use a custom-built command-line interface which means they don't have a look and feel of Linux. However, since operating system components are running as services in containers, engineers should learn how to deploy container management software to create, deploy, and scale and them.

This combination also promotes more open-source configurations and concepts where different API's can "hook" into this custom routing platform. Some NOS's even make the whole disaggregated cluster made up of different components seem like a "virtual" chassis, providing the best of both worlds [9].

These options allow potential for seamless integration into the existing network while maintaining small failure domains or a blast radius. While N+1 redundancy gives you an extra component, node, or link to failover to, N+M design provides a whole node or cluster redundancy giving a whole other level failover capability. This allows hitless maintenances and updates. Now, it is possible to direct traffic away from a segment of spines and leaves and upgrade the software or configurations while having traffic run across the rest of the S/L cluster. This reduces the failover impact of traffic and does not put such a hazardous condition strain on the system, like it used to with a single chassis system. This S/L design makes an In-Service Software Upgrade (ISSU) more reliable than in single chassis routers. This allows any time maintenance, even daytime since it will be Non-Service Affecting. Now, a single failure of a component doesn't impact traffic in multiple directions and reduces the impact to services.

2.3. Processes and skills needed

2.3.1. Skills

It is no secret that to keep up with innovation, engineers and technologists must set out for a lifelong learning and development journey. Having the right skills has always been important in IT, especially if you want to stand out in the industry. This is certainly going to become even more critical as the pace of change and innovation continues to accelerate. Technical experts will come to find that the technology they have come to know on such a deep level is constantly evolving. Carrier-grade networks are highly dependent on the expertise of network professionals who build, operate, and maintain them. The skillset of these engineers had been evolving for a while and now with the proliferation of new innovative disaggregated network solutions there is a new set of skills and capabilities which they must master. They now need more DevOps, programmability, and scripting skills to leverage automation tools because the newer open software is more API/GNMI/Model driven. There is a trend and requirement by management to move away from traditional CLI network device management, where you have the chance of repetitive human error. Based on the "Annual outage analysis 2021" report by Uptime Institute an aggregated year-on-year average of 63% of failures are due to human error [32]. Network engineers now need to blend their skill sets and learn and utilize server sysadmin skills to make them future-prone, ready, and stay marketable in the job market.

2.3.1.1. Microservices

Traditional architecture has been a classic software design pattern since the origins of the industry in which the user interface and the source code are combined into a single program. Though in the past this was convenient to have the networking vendor be responsible for any software issues, there are many drawbacks to this approach such as complexity of the code, scale limitations, reliability among many others. A software code bug may affect both the SW and HW, but when decoupling the two and disaggregating them, there is now a clear demarcation point for the hardware components versus the software. The industry is moving towards the microservices architectural style for developing applications. Figure 5 provides an illustrative example of these two approaches.

Microservices allow a relatively large application to be divided into smaller parts, having their own autonomy and realm of responsibility. With microservices in containers, it's simpler to take advantage of hardware and easily orchestrate services, including networking. Network engineers and architects should learn how to operate and maintain the container orchestration platforms like Kubernetes and Docker because many NOSs designated for disaggregated networks are based on the cloud-native, distributed architecture. Cloud Native Computing Foundation (CNCF), which is the vendor-neutral hub of cloud native computing, states that microservices and containers form the foundation for the cloud-native application development [10]. Therefore, it is paramount for network professionals to develop their Linux skills and System/Server Admin skills so they can easily operate on both sides of the domain.

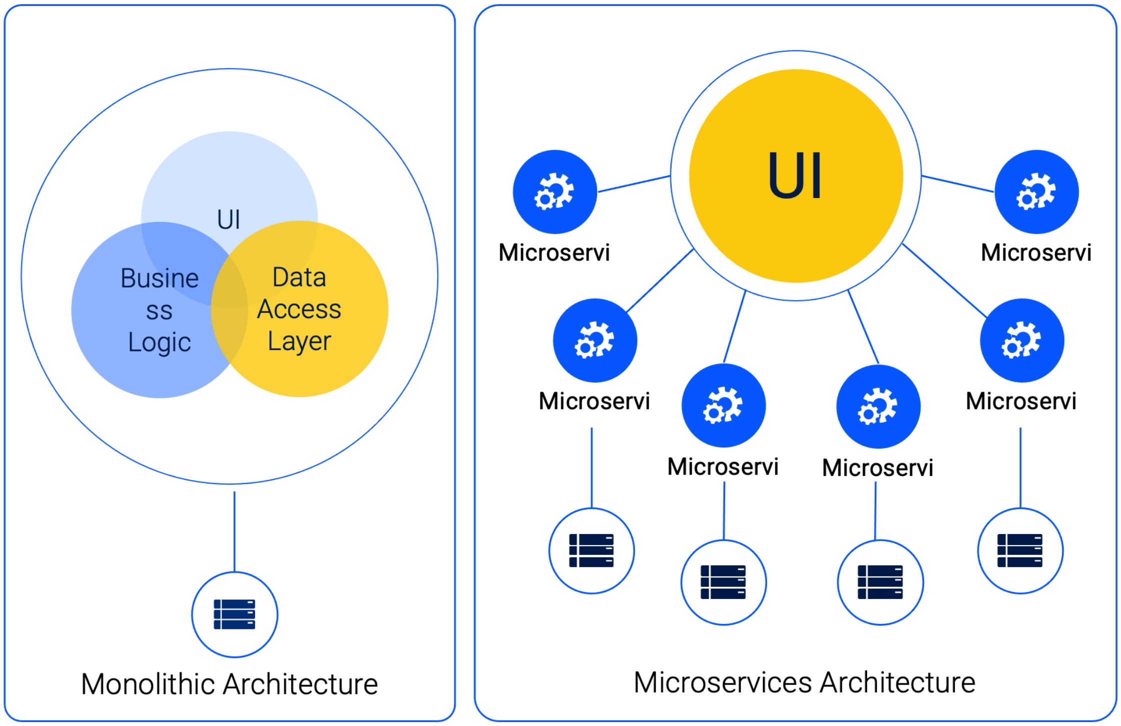

Figure 5 - Traditional versus Microservices Architecture

2.3.1.2. Linux

Refer to Figure 6 for the high-level view of Linux containers. Open organizations like OCP are actively developing and deploying new Linux distributions that are specifically targeted and tailored for network equipment [11]. Many vendors in the ecosystem have already started offering NOSs based on these distributions. Network engineers and architects are expected to understand and implement the fundamentals like navigation in the filesystem from the shell, which is the server CLI interface, manipulation of files and directories, running programs and working with background services, also known as daemons. On top of that, engineers must learn how to manipulate network interfaces, how to view and manage routing on a Linux system because interface and routing configuration go together. Also, due to the openness of the hardware, a huge number of logs and event traces are available which can be viewed "under the hood" via the Linux shell. To be able to troubleshoot and analyze the sequence of events or even take the packet captures of control and data plane traffic, an engineer must be familiar with text editors which offer the tools to sift through large amounts of data and even observe events in real-time. Additionally, many of the tools that we will discuss in the future section have their origins in Linux and require to be run from the Linux system.

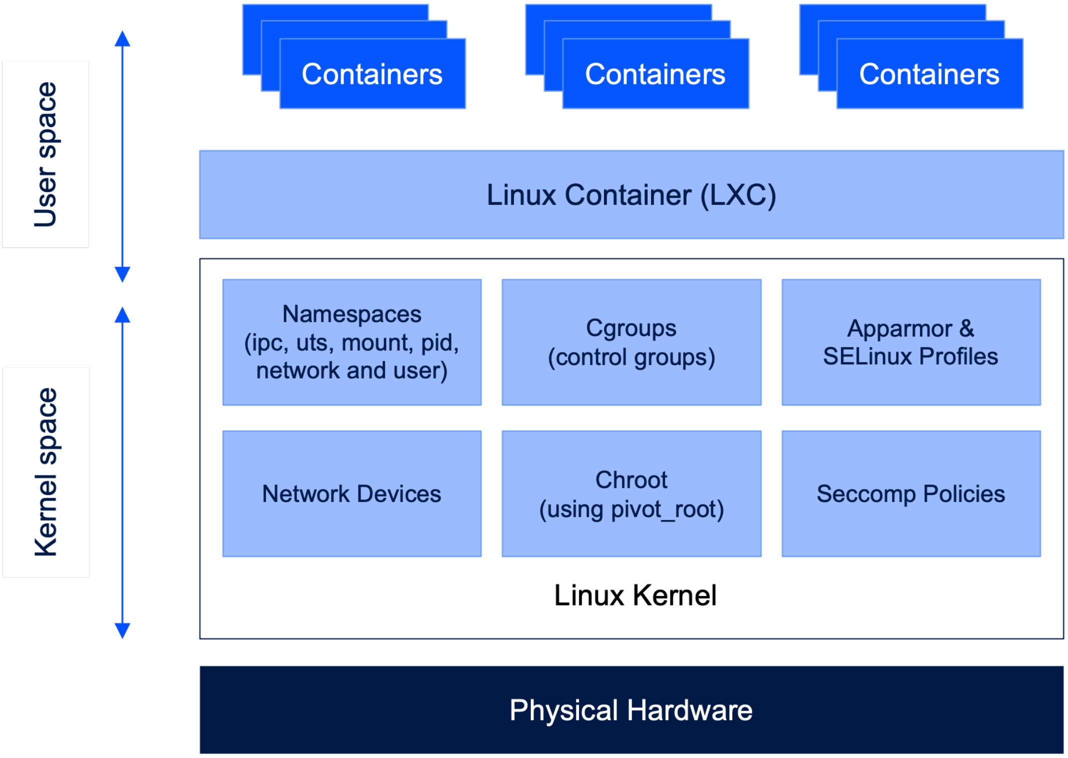

Figure 6 - High-level view of Linux Containers

2.3.1.3. *Data Formats and Models*

Protocols like BGP, OSPF, ISIS, and TCP/IP were created out of need for devices on the network to have a single language to communicate across a globally distributed system – the Internet! The data formats were conceived for a similar reason – for computer systems to freely understand each other.

With the significant increase of the quantity of network devices and the proliferation of the Internet of Things (IoT) it became apparent that it is beyond the bounds of possibility for humans to manage them [12]. The solution here is to automate configuration management and maintenance as much as possible and streamline the deployment of new services. Engineers should get familiar with network automation tools and methodologies as many modern NOSs expose programmatic interfaces that offer an API. Traditionally, each network device is closed (locked from installing third-party software, as an example) and only has a command-line interface (CLI). Although the CLI is still a well-known and even preferred method of access to the router by network professionals, it does not offer the flexibility required to truly manage and operate this brave new Internet.

Imagine operating a disaggregated cluster comprised of fifty or so components or nodes. Each node may need its own Loopback IP for management and its own configuration file. In the past with traditional routers, the network engineer had to create the fifty separate config files and load them onto each component in the cluster. This also opens the system up to human error and the configuration files can be prone to typos and errors. Pushing configurations and updates to multiple nodes in a S/L system might result in unexpected errors, takes a lot of time to prepare configurations or writing scripts to automate the process. One way to easily automate it is to set up a central system and network platform or orchestrator. This system can now treat all the nodes like a virtual cluster, where the Orchestrator pushes the software and configurations to all the components of the cluster. This buys the best of both worlds where there are benefits of a traditional system with the flexibility of S/L. Thusly, automation becomes key and "table-stakes" to implement any type of disaggregation.

One of the most optimal ways to push and retrieve the configuration from the devices is via the HTTP-based APIs like RESTful and non-RESTful APIs [13]. Another one is NETCONF [14] which is a network management protocol conceived specifically for configuration management and retrieving operational state data. In order to leverage these tools, professionals need to understand the data formats like XML and JSON [15, 16], YAML [17] and data modeling language YANG [18]. Refer to the Figure 7 for the overview of these tools.

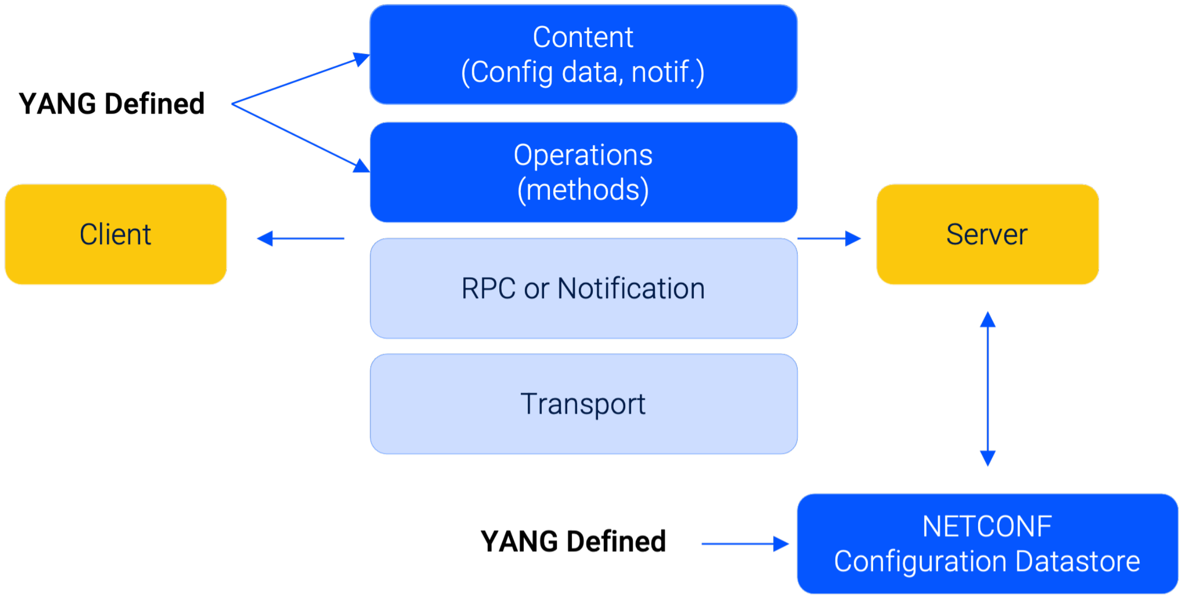

Figure 7 - Overview of YANG and NETCONF

2.3.2. Processes

The processes to deploy and maintain the disaggregated solutions are simplified due to multiple factors like less planning efforts and faster installation time.

2.3.2.1. Less planning efforts

IP network planning can be quite complicated and take months. The planning time is heavily influenced and considers the number of components, service requirements, capacity forecasting, site constraints, and more. While traditional chassis solutions are limited in their abilities to simplify the overall process, disaggregated distributed solutions offer one design process and ease of expansion by relying on only two standard data plane building blocks: Packet Forwarders and Fabric Forwarders, as compared to five to ten integrated router solutions with multiple line card models which must be compatible with fabric modules and different software revisions for different features on a line card.

2.3.2.2. Installation Time Considerations

In this high-paced, modern era of networking, field installations and maintenance are some of the most comprehensive and costly operational activities. They include the engagement of multiple teams, materials, and dedicated planning to adapt to each physical location's constraints like space, power, and cooling. In the disaggregated model, installation process is identical across all node sizes, eliminating the need to constantly train staff and avoid costly installation errors. This cookie-cutter method of installation can be used for the deployments across all fields of use in the network (peering exchanges, cloud exchanges, edge cloud etc.). However, there is a lot of backend fabric fiber and ethernet connections (fondly referred to as "spaghetti-wiring") since it looks like strands of spaghetti that must be installed to connect the nodes as a cluster as opposed to simply sliding in a linecard in traditional chassis case where only the service ports must be connected as the fabric is hidden in the backplane.

There is a lot of up-front cost and effort in a DDC/DDBR architecture in performing the physical install work initially to obtain a longer roadmap, but the benefit is savings in port migration to the new chassis. When forecasting growth and installing with the mindset of room to grow, this eliminates the need to come back and install any additional fabric forwarders and disrupt the system later. Also, unlike scale up in a single chassis router, when the system grows beyond its size, all the client ports ought to be replumbed to another chassis whereas in a disaggregated model scale out consists of only adding another PF or two. This positions the network to consider any un-forecasted growth surges, such as the one a lot of networks experienced during the COVID 19 pandemic. This follows the "set it and forget it" ideals.

3. Operational Considerations and Challenges

DDC/DDBR is an optimal solution that overcomes the most relevant challenges that ISPs/CSPs are facing today when deploying and scaling their IP backbone/aggregation networks. The role of these networks is to route the mobile, voice, broadband & commercial traffic between different network segments at a national and regional level while providing connectivity with external networks such as other service and cloud providers, content data networks, Internet exchange peers and IP transit providers.

The IP backbone networks must regularly scale to support the internet traffic growth, to improve resiliency and reliability, and to meet the expectations of mission-critical types of communications. The essential objective is to lower the cost per bit and improve the overall customer experience and satisfaction.

In the following sections, we will dig deep into the key challenges that exist in the IP backbone networking space, how DDC/DDBR architecture can help in overcoming them, and comprehensive comparison of DDC/DDBR and Spine-Leaf architecture.

3.1. Challenges with traditional router architectures

3.1.1. *Supply chain concerns*

Akin to all sectors of the telco and cableco networks (Transport, Core, Aggregation, Access), the continuous mergers and acquisitions in the core routers market have resulted in the following limitations, no less important of which is a lack of diversity:

- Notable reliance on a very limited number of suppliers.
- Market which is extremely difficult to enter and compete in leading to an ever-increasing risk from soaring costs.
- Insubstantial innovation and time-to-market speed.
- Unsatisfactory interoperability across different hardware components. For instance, line cards and fabric modules compatibility issues.

3.1.2. *Traditional nature of the components*

Historically the cablecos have been deploying traditional IP backbone/aggregation routers which are based on proprietary components as demonstrated in Figure-8:

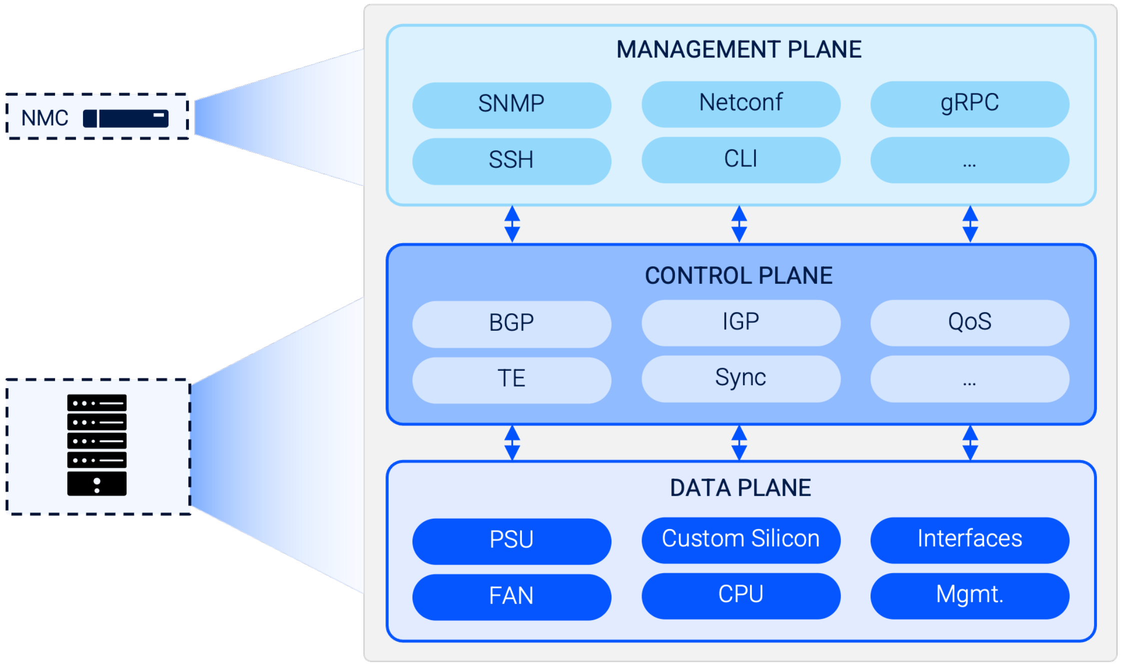

Figure 8 - Traditional Routers

The data plane hardware which contains line-cards is where custom silicon chips are implemented to manipulate the packet processing, traffic management and forwarding. Multiple line cards serve different purposes and have differing applications depending on the place in the network which introduces additional operational complexity and overhead. For example, there are availability concerns when replacing or upgrading a line card inside the chassis. In the live production environment, an issue with a fabric module or a line card can have an adverse impact on the whole system rather than being limited only to that component consequently increasing the blast radius.

The disaggregated model has a cloud-like pay-as-you grow approach where fixed form-factor PF whitebox servers can be purchased as needed from multiple ODMs for a significantly lower price. Packet forwarders enable operators to utilize any port on the whitebox for any service regardless of the implementation area. This adds tremendous flexibility to enable multiple services and reduce their time to market. The ports on these whiteboxes come with variable native interface speeds which can be reconfigured to operate at different speeds. For example, native 400GE port can be reconfigured to be utilized as 100GE port or can be broken down into 4x100GE via a breakout cable. For the access use case, 100GE can equivalently be adjusted to operate at 10GE speed or, also, broken down into 4x10GE or 4x25GE ports. This allows for gradual growth into the port step functions depending on your deployment budget.

A custom NOS which is specifically designed to efficiently run only on the custom hardware and is comprised of proprietary code and licensing runs the control plane. That includes the drivers which are in control of the hardware components like power, cooling, etc. The firmware is responsible for loading the NOS image when the router boots up including the networking software stack.

In the disaggregated model, multiple NOS vendors in the ecosystem can install their software onto the open networking hardware. The Open Network Install Environment (ONIE) [19] is an open-source initiative that defines an open “install environment”. Before the invention of ONIE, routing equipment was procured with pre-installed NOSs, essentially creating networking devices that locked operators with vendors whose supply chain is integrated and owned completely by them. ONIE runs in the management subsystem of the whiteboxes utilizing capabilities in a Linux kernel. This allows ISPs/CSPs to install target NOS as part of the provisioning process, in the same way the COTS servers are provisioned.

The third component is the management plane, which takes care of the overall platform management: for instance, the interfaces configuration, services provisioning, inventory management, alarm reporting, fault handling, and performance monitoring, all which is tightly coupled with control and data plane; thereby introducing a single point of failure.

This traditional architecture has met the essential needs of operators (capacity, availability, etc.) and has served them for an extended time. Nonetheless, it has impeded them from unleashing the true potential of open networking and has significantly decelerated the innovation in the backbone/aggregation networks.

For instance, having the data and control plane so closely tied together leads to a significant and uncomfortable dependency on the incumbent vendor’s roadmap and prevented them to benefit from features available in a 3rd party NOS supplier. Operators are dependent on the vendor to deliver the HW or SW roadmap on time to upgrade the environment for addressing the growth pressure. The below graph depicts this port growth need versus availability.

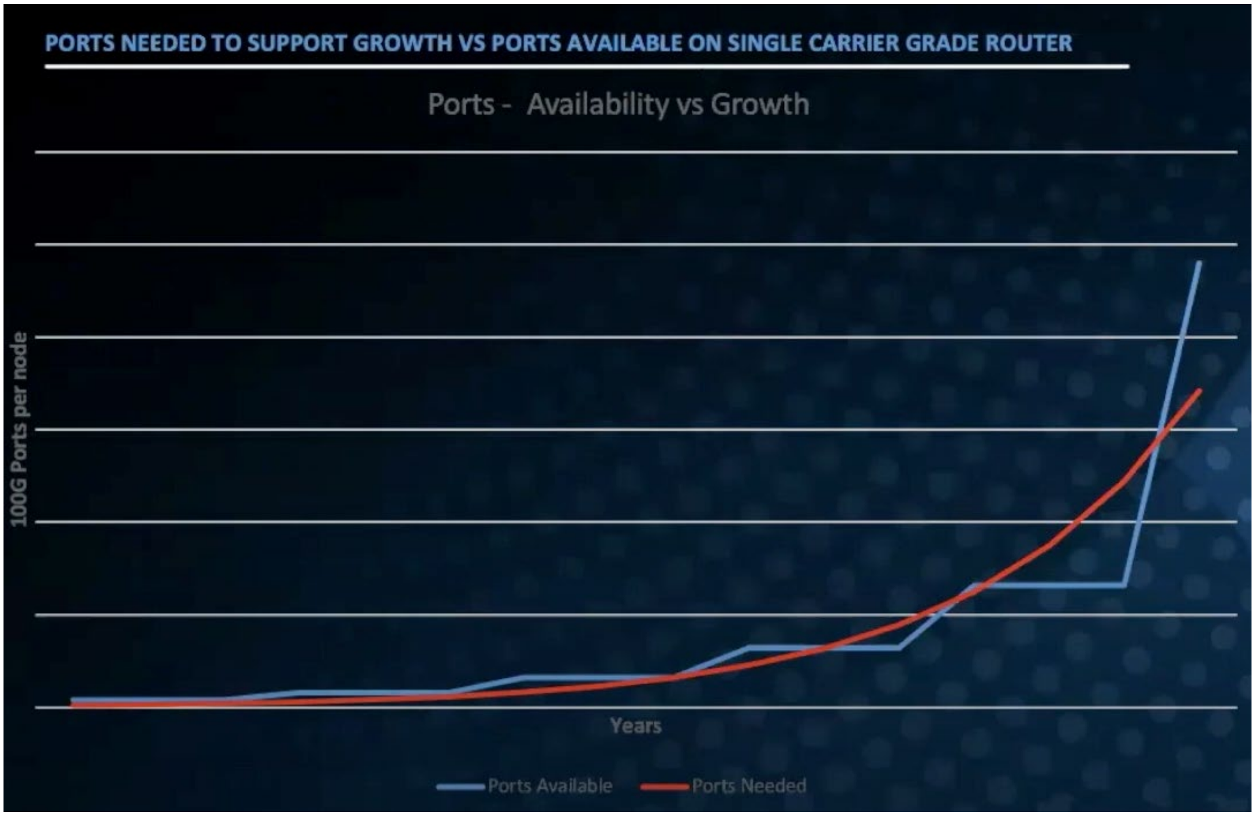

Figure 9 - Non-linear growth of ports [4]

Even though multiple software companies have offered powerful Network Management Systems (NMS) and software controllers to manage the traditional routers' management plane, ISPs/CSPs are still exposed to various challenges. One of the major one is the complicated and high integration cost to manage third party products via their existing homegrown NMS. Even though the promising NETCONF protocol is capable to address them, there is a colossal amount of work yet to be done to come up with a vendor-neutral data model for network and device configuration.

3.1.3. Chassis limitations

The current backbone routers are primarily designed based on a chassis structure with front access where the user network interfaces (UNI), network-to-network interfaces (NNI) and the control boards are interconnected into a backplane.

Based on the crucial role of these routers in the network and the huge volume of traffic they pass through, they were required to offer:

- Resiliency to maintain uninterrupted connectivity to the mobile, broadband, and business customers.
- Significant computing and storage capability to store IPv4 and IPv6 global routing tables.
- Port density and capacity enough to support the growth of the customers and services.

Therefore, the chassis had to be equipped with:

- High availability at all levels: control processing, switching fabric, cooling, and power.
- Large TCAM (Ternary Content Addressable Memory), strong computing capability, deep buffers.
- Variety of sophisticated control plane features which includes NSR (Non-Stop-Routing) & ISSU (In-Service Software Upgrade).
- Large port density, number of slots for line cards and backplane switching capacity.

The above requirements have resulted in a platform of considerable size with high cost and huge power consumption, and which necessitates an upfront capital allocation without a guarantee of a reasonable return. This system takes a lot of space and demands a lot of advanced cooling system deployments to ensure adequate operating thermal levels in datacenters and head ends and lacks a capability to adequately grow based on the current capacity needs.

Additionally, due to the limited number of slots in the chassis, running out of ports may turn out to be quite disruptive to an ISPs/CSPs' operational model. This could lead to having to purchase an entirely new chassis resulting in an unnecessarily complex network topology, possible suboptimal traffic flows and a non-linear cost per port model as depicted in Figure-9. This chassis upgrade process is not even remotely agile to empower service providers with a capability to react to unplanned upgrade requests in a timely manner, which results in missing the opportunity to gain more market share by increasing the customer base. This chassis upgrade becomes akin to forklifting the chassis out from the rack and performing "open-heart surgery" to rip-and-replace with an upgraded traditional chassis.

Furthermore, all the NNI & UNI interfaces are centralized on one chassis which creates undesirable operational risks of losing the entire node in the event of a software failure, power issue, executing the Method of Procedure (MoP) document in the wrong order etc. Since telcos and cablecos are dependent on a single chassis in the core of their network, they have the potential to isolate services in case the redundant site goes down increasing the blast radius once again. Even an un-expected fiber cut or an environmental issue that no one has control over can blackhole all services in the network.

3.1.4. Openness and ease of upgrades

According to Yole Développement's "Optical Transceivers for Datacom & Telecom Market 2021" report [20], the optical transceiver market will highly likely grow 14 percent in the next 5 years. This growth is driven by the need of the operators to utilize high data rate modules above 100G.

With the industry trends in the optical pluggable transceivers and the dawn of 400G and 800G QSFP-DD (Quad Small Form Factor Pluggable-Double Density) optics, the ISPs/CSPs need to replace the existing hardware with higher capacity, more compact proportions, elastic thermal management ports which facilitate supporting higher capacity links with advantageous port density per rack unit.

Notably, the 10G/25G/40G user network interfaces need to be upgraded to 100G or even to 400G, and the 100G network to network interfaces or fabric interfaces to 400G and 800G. This means line cards or entire chassis replacement needs to happen to take advantage of the new interfaces and benefit the most from the interface's capacity through the backplane.

To that end, the disaggregated networking approach gives telcos and cablecos an upper hand when replacing or upgrading the installed base to protect their investment in IP backbone/aggregation networks, as the components are based on open hardware. There is no need to use specialized proprietary optics between packet and fabric forwarders because of cell-based packets, keeping with the open-standard theme.

3.2. Comparison of DDC/DDBR and Spine-Leaf architectures

A 3-stage Clos network is the smallest version of a Clos network, and it is relevant to modern scalable carrier and hyperscalers' networks. As its name implies, this network has 3 stages: ingress, middle and egress. Figure-4 from the earlier chapter gives a high-level view of that architecture.

A Spine and Leaf architecture is a derivative from the 3-stage Clos network. Occasionally, it is referred to as a "Folded 3-stage Clos Network", where the ingress and egress points are folded back on top of each other [22] as shown below in Figure-10:

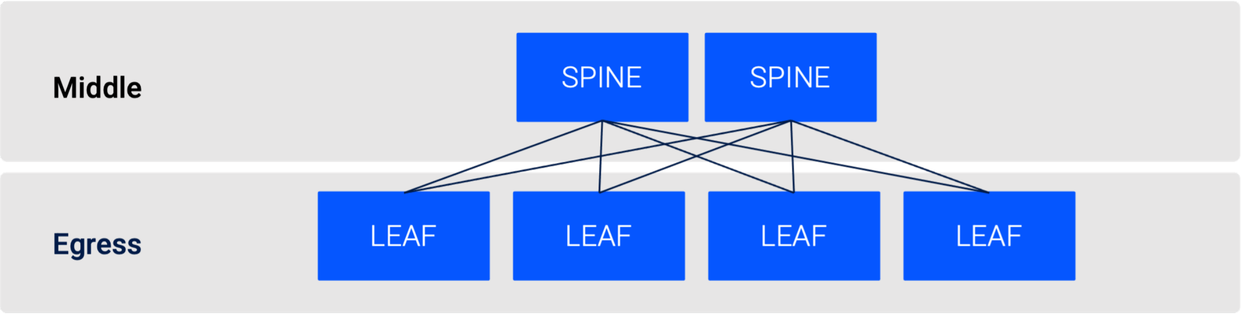

Figure 10 - Folded 3-stage Clos network (also known as Spine-Leaf)

Let's dive into a detailed comparison of DDC/DDBR architecture and Spine/Leaf:

- Over-subscription of Leaf to Spine links in S/L versus equal or over-provisioned fabric to leaf ports bandwidth in the disaggregated solution:

  In the S/L architecture oversubscription of links is a common occurrence causing a lot of operational overhead in cases when more bandwidth is needed.

  When oversubscription of links ensues (meaning, if there is more traffic on the ingress than the egress on the active link at one time), the procedure for enlarging capacity is complicated. A whole new spine switch or router must be added, and uplinks need to be expanded to every leaf switch thereby adding more cable density within and between the racks. In instances where device port capacity reaches its limit, a totally new leaf switch/router must be added accruing to the operational complexity introduced by the addition of a spine.

In the disaggregated model on the other hand, clusters are built in a way where bandwidth is equal or over-provisioned to leaf ports [23]. Below is the table with variable cluster sizes ranging from a standalone whitebox with 4Tb capacity up to the large cluster with 192Tb throughput:

Table 1 - DDC/DDBR cluster sizes

	Standalone	Small Cluster	Medium Cluster	Large Cluster
Max Capacity	**4Tb**	**16Tb**	**96Tb**	**192Tb**
Port Density	40x100G	160x100G	960x100G	1920x100G
	10x400G	40x400G	240x400G	480x400G
	80x10G/25G	320x10G/25G	1920x10G/25G	3840x10G/25G
Packet Forwarder White box	1	4	24	48
Fabric White box	-	2	7	13

- Elephant flows:

In a large Clos network, a typical approach for using the built-in multipath on NNI links is to utilize reliable Layer 3 routing, including on the leaves. Point-to-point links between the spines and leaves are in a specifically allocated subnet and common Equal Cost Multipath (ECMP) routing can be implemented to distribute traffic equally over them. This leads to maintaining large numbers of subnet routes in the tables of the nodes in the topology which results in a requirement to purchase expensive custom hardware with large enough ternary content-addressable memory (TCAM). Although, this architecture is designed using Layer 3 routing, ECMP and provides distinguished bandwidth and latency performance, the preeminent complication here is insensitivity to the workload. Carrier network traffic tends to feature a mix of latency sensitive “mice” flows, and bandwidth intensive longer-lived flows, also known as “elephants”, for which throughput is of higher priority than latency and which constitute only 10% of flows, while accounting for nearly 80% of traffic.

Mice flows are very bursty and short lived, however, elephant flows tend to “pile up” on certain paths even though others are available and ready to be used, filling network buffers, creating congestion events and suppressing mice flows on these links.

This leads to a detrimental degradation of application performance such as online gaming, web requests, multimedia broadcasting and VoIP which are of a particular importance for cable operators who continuously strive to improve quality of experience for their customers [24].

In the DDC/DDBR solution, the NNI links do not rely on ECMP or any sort of Layer 3 routing rather taking advantage of the capabilities enabled by merchant silicon for redistributing traffic equally with remarkable precision.

- Variable sized ethernet frames versus cell-based fabric traffic:

  In the Spine-Leaf network architecture packets that are passing through fabric links (NNI links) are of variable size.

  Packets transiting the fabric in the DDC/DDBR architecture are proprietary formatted fixed sized cells [23]. Thus, monitoring systems must account for differences in the way the payload is transmitted.

- Spreading of traffic over fabric ports:

  In the Spine-Leaf network architecture the traffic is spread over disparate NNIs via ECMP and the routing protocols' logic, and the processing occurs in the control-plane by the NOS where the routes are programmed in the tables of the forwarding ASIC.

  In the disaggregated solution, cell-based traffic is used on fabric ports and the spreading logic is being executed at the microcode level of the forwarding ASIC and is completely transparent to the NOS. Fairly sophisticated credits/tokens mechanism is involved in the implementation of this feature [5, 23].

- Managing a folded Clos topology of N-independently functioning elements versus all elements are under control of a single NOS:

  In the DC Leaf-Spine design, each node ordinarily runs a NOS that is independent. In this case, forwarding decisions are made individually on each node which lack a global view of the network, so the best path computation algorithms are at best locally optimized.

  In the DDC-RS, the Packet and Fabric Forwarders function under the oversight of a single NOS. The virtual router needs only a single network management interface and is considered as one big virtual chassis. Subsystems of the operating system can be distributed to the elements of the cluster but the "Brains" of it are centralized on the powerful compute nodes. This also makes monitoring and onboarding automation frameworks easier.

  From this point of view, multiple routing algorithms are optimized through a single routing system, The traffic is just traversing through one router – from one port to another port on a separate Packet Forwarder from the end-to-end network perspective [5].

- Easier upgrade process:

  Software and firmware upgrades usually take more time and require multiple maintenance windows in the S/L case.
  Due to multiple independent components, each of them must be upgraded separately. In many cases, not all traffic can be migrated away from the pod which introduces a need to move traffic away from certain leaves and spines within it to proceed with the upgrade which negatively contributes to the already complex process.

Due to a single point of management, disaggregated backbone/aggregation router upgrades have the potential to upgrade either some or all the elements of the cluster within one maintenance window considerably reducing the operational overhead and the amount of scheduled maintenance windows. This gives the user the flexibility of shifting traffic off the cluster, where in the past with traditional routers or in the S/L network upgrading sets of spines and leaves left the potential to be exposed to being in a hazardous condition.

Furthermore, in the traditional S/L model convergence of the IGP takes more time before the traffic is soaked out of nodes but there is a minimal negative impact on Quality of Experience (QoE) for customers. However, in the DDC/DDBR case due to this being a single virtual chassis traffic can be diverted immediately and converges instantaneously.

There is of course a tradeoff between a graceful traffic soak from S/L with minimal negative impact on the customer Quality of Experience which can take more time and leave the system exposed, whereas in the yank and replace method the protocol must fail the traffic over and traffic reconverges more abruptly but is quicker. There is a certain impact on the customer's QoE in either method.

4. Orchestration, Automation and Analytics

Despite bringing meaningful scale and cost advantages to the table, this new disaggregated network operational model might potentially be considered a risk. This approach has emerged recently in the industry and service providers are weighing the extra complexity related to the orchestration, automation, and management of this open model, specifically in the areas of field installation, upgrades, capacity growth, and troubleshooting.

Cloud orchestration is a solution that is required to deliver the operational simplicity, automation, and visibility to the DDC/DDBR architecture to drive the acceleration of the deployment of this cloud-native networking solution. It should offer detailed visibility into the system's internal architecture including hardware and software components, KPIs related to SLA, alarm, and fault management.

4.1. Automated Operations

This encompasses the lifecycle management of resources and services – from provisioning to decommissioning, which includes:

Zero-touch provisioning (ZTP) – automatically integrates multi-vendor white box hardware and NOS into a working routing platform supporting a secure deployment prone to less errors with limited manual intervention. Ahead of entering the protected and controlled operator network's environment, white boxes go through an extra security measure which is the bootstrap process to ensure system integrity.

Hardware inventory management – grants detailed data on every element within the cluster, including location, model, serial number, firmware version etc. This becomes even more so critical as the cluster scales out and the inventory management gets too complicated to manage manually.

Modular software orchestration – covers the entire stack and can be done selectively per specific software component, counting firmware, base OS, NOS image/container. This capability becomes extremely handy because upgrades can be initiated per component without affecting the overall software stack reducing the risk of failure in cases when the whole cluster is upgraded. Another advantage is the real-time orchestration status which helps engineers to monitor the activity and potentially prevent issues before they occur. Software rollback can also be easily performed in cases when the upgrade was unsuccessful.

4.2. Health Monitoring and Assurance

The orchestration tool automates event and KPI monitoring and ensures availability and performance SLAs such as:

Cluster topology – live view of the cluster's nodes, their states, formation and connectivity across clusters and the entire network, including:

- Hardware components: CPU, memory, PSU, fan, temperature, ports, and interfaces.
- Software components: base OS, firmware, processes, containers, and microservices.

Fault, performance, and alarm management: on top of 3rd party applications that collect data for events management, the cloud orchestration system can provide all these details for the clusters under its management:

- Supports alarms and KPIs at every level of the system – from hardware components to software containers.
- Alarm dashboard to monitor and categorize system alarms.
- Real-time and time-series alarm view.

Tech support integration for in-depth system diagnosis and debugging: being able to retrieve tech-support files when there is a potential software or hardware bug is paramount for network operations teams. The orchestration tool should allow a simple way to retrieve such a bulky file from the clusters for analysis by the NOS and hardware vendors.

4.3. Telemetry

Historically, SNMP has been the de-facto protocol for data collection, however, it may not be suitable for the management of truly large networks because of the performance limitations of polling.

gRPC is Google's project for Remote Procedure Calls between applications. gRPC based telemetry streaming allows to export performance monitor counters and operational-state parameters in a flexible and scalable way which will help operators tremendously as the number of devices in their networks grows exponentially [25]. Unlike traditional performance monitoring (PM) collection methods such as SNMP walk, gRPC based telemetry uses push method for delivering PM data from router to the PM collector. Such an approach gives the following advantages:

- PM Collector does not need to poll each router individually.
- PM Collector can define different set of counters for collection from specific routers.
- Push parameters can be configured to each pushed counter specifically (e.g., sample rate, telemetry packet DSCP value etc.).
- gRPC interface when combined with Protobuf encoding is more efficient in terms of network channel utilization than SNMP and other interfaces such as NetConf and RestConf.

5. Real-life deployment of the DDC/DDBR model

DDC/DDBR architecture is a field-proven concept deployed in the core and aggregation layer of the largest telcos and cablecos in the world [26].

The implementation started with the channel partners wiping the received whiteboxes and installing the ONIE and BaseOS on top of it in preparation for the NOS integration. After this step had been completed, all the elements were shipped to the respective data centers for the site installation and wiring.

Once the cabling was finalized and all elements within the cluster powered up, the route controllers were discovered by the orchestration system which initiated the cluster creation via the Zero-Touch Provisioning process. At this point, the cluster is ready for the enablement of the control plane and for passing traffic.

There are several tools and dashboards needed to monitor production grade Clos networks including the need to monitor the load sharing over ECMP paths, scale of the databases in the IGP domain due to the large number of nodes. On the other hand, with a virtual cluster:

- It became easier for the NOC team to manage the network and respond accordingly to the events within the system.
    - For example, the load sharing on fabric links is astonishingly equal and does not require any monitoring because it is done at the microcode level, as we have mentioned earlier.
- No IGP or any other routing protocol required between the components of the cluster as is needed in a S/L system, thus reducing the number of nodes in the IGP domain.
- There is also no need for a multi-level BGP design within the cluster and consequently no requirement for BGP enhancements and features to improve recovery time during convergence.

Now, an abstraction layer is introduced which hides all the complexity of the Clos network and appears as a single router which is so familiar to network engineers. The next figure demonstrates an example of a large national backbone network:

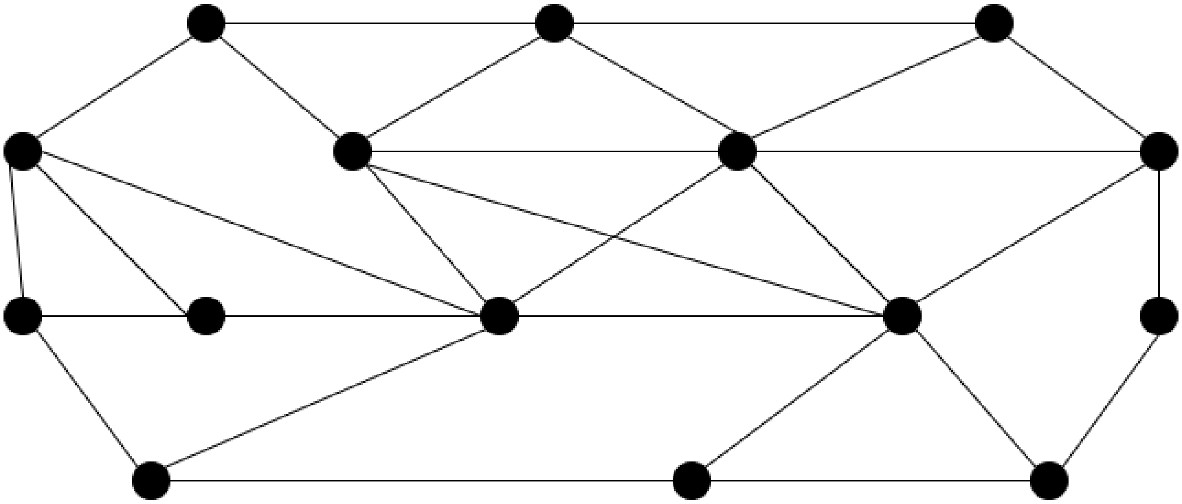

Figure 11 - Large National Backbone Network

Every backbone site either has a traditional router or a spine-leaf cluster. Let’s zoom in into one of the sites that hypothetically contains a spine-leaf cluster:

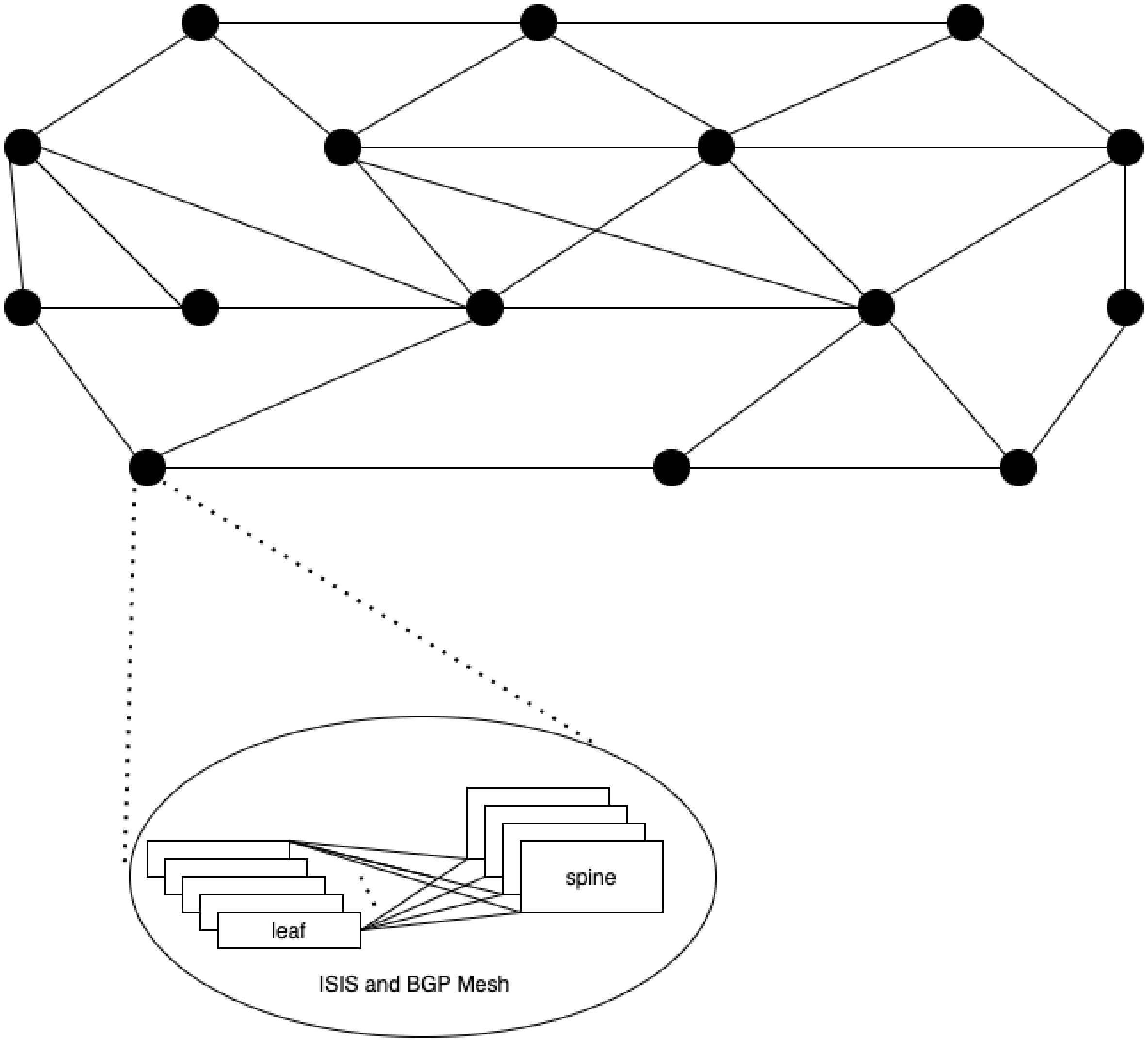

Figure 12 - Spine-Leaf cluster within a Backbone Site

As can be seen from the above figure, there is an ISIS and BGP mesh that must be maintained within the cluster on top of interconnections between different backbone, aggregation, and peer networks. The next image showcases a similar topology but with a virtual chassis:

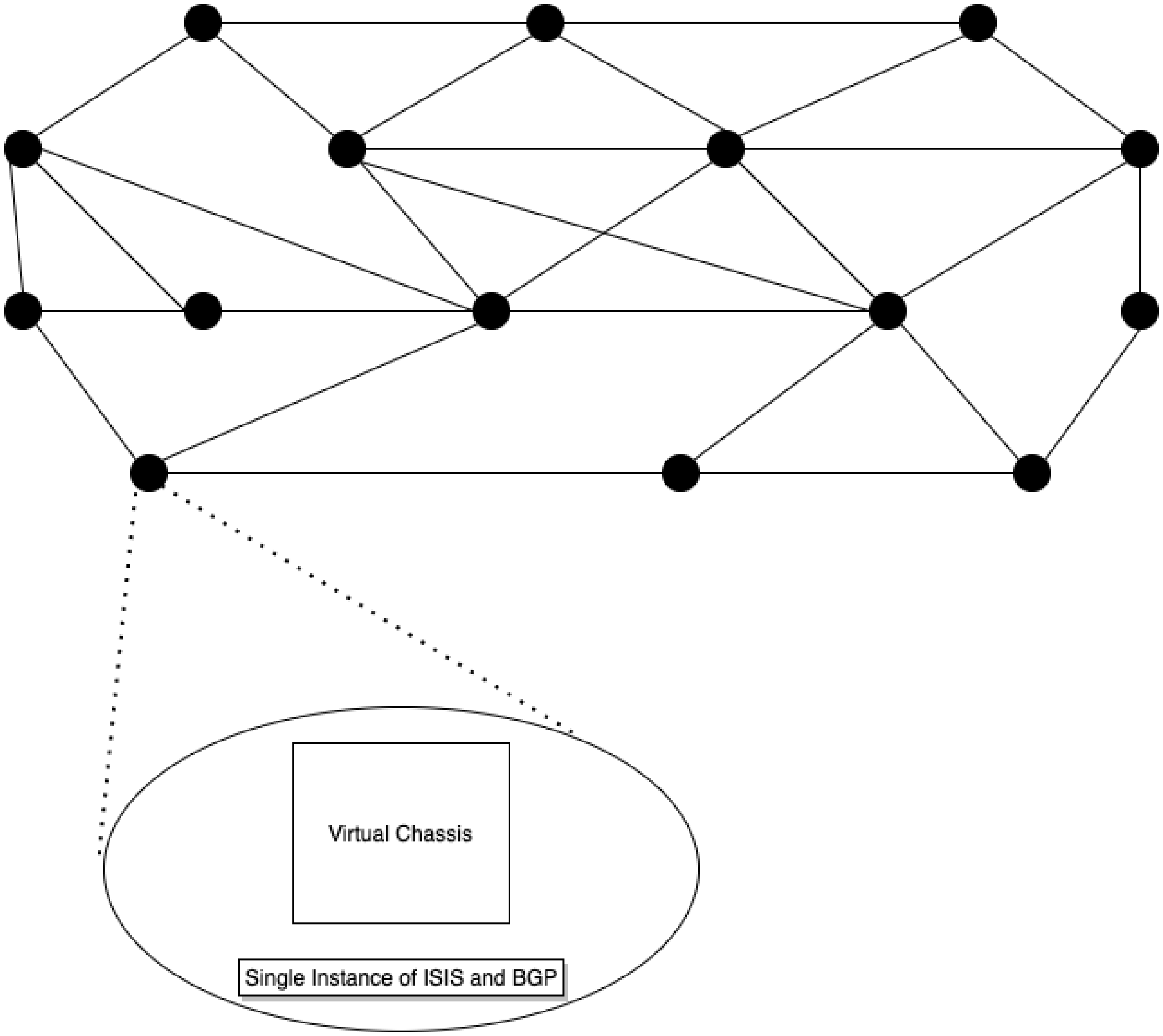

Figure 13 - Virtual Chassis within a Backbone Site

The virtual chassis depicted in Figure-13 eliminates the requirement for intra-cluster control-plane. Now, network architects can focus on mission critical services rather than assuring the SLAs within the cluster.

Any future launches of a virtual chassis in the network will be done by the orchestration system which supports comprehensive validation methods of spaghetti-wiring fabric cabling, management of the code upgrades and device configurations. Another important activity of the engineering and operations teams is auditing the provisioned components count against the actual operational count so that N+M redundancy can be maintained which is already incorporated into the orchestration system alleviating the need for engineers to do it manually.

With a disaggregated approach comes a challenge of the identification of the responsibility domain: hardware and software. Since there are 2 vendors delivering these components, occurrence of the issues requires an understanding of which domain it belongs to, to deliver necessary tech-support files and logs to the respective TAC. There is a need for cooperative collaboration between HW and SW vendors during troubleshooting of such systems so that the root cause of an issue can be reached quickly and amicably.

6. The future of the dissaggregated solutions

In the words of the ancient Greek philosopher Heraclitus, “The only constant in life is change”. Nowadays, this most vividly applies to the technology world. The pace of innovation must keep up with the demand and new emerging use cases. Telecommunications companies along with vendors will keep driving innovation in disaggregated network solutions. Essential future developments are multiservice functionality on top of the shared pool of resources, OpenOffload and data center sustainability.

The average number of devices connected to the internet per household and per capita is increasing. Expanding machine-to-machine applications contribute in a major way to device and connection growth and push the expansion of data center infrastructure. With the growth of IoT and “smart” appliances, almost any type of device needs connectivity to the home network. Examples of this are laundry machines, refrigerators, stoves, microwaves, thermostats, garage openers, and even the front door keypad of a home! If one would check the number of devices connected to their home Wi-Fi network, they would be surprised at the number.

One of the promising concepts being developed is utilizing the network as a cloud resource leveraging this model’s High-Availability (HA) capabilities. By applying modern, cloud-based, shared resource methodologies to networks, telcos and cablecos can enhance their network resource utilization by taking advantage of the unified infrastructure that supports multiple network functions as software-based services, such as enterprise, broadband, mobile, firewall, load-balancer etc. [27]. Any port of the cluster, designated for a specific function, can be used to enable this service. Sharing the physical infrastructure for various services extensively lowers the physical footprint within data centers and reduces the number of unused ports, resulting in a more efficient employment of compute and networking resources. In the past we had core, edge, and backbone nodes that were separate units. Now they can be aggregated onto a unified cloud-native infrastructure.

Spinning up various network functions is very demanding from the compute resources perspective which calls for original approaches in dealing with processing. There is an initiative in the industry that defines APIs to accelerate network functions and applications by offloading packet processing to the packet forwarders rather than x86 servers which already have their fair share of load. One of the APIs is for applications like Virtual Firewalls and Intrusion Detection Systems called OpenOffload [28], and another one is for functions like VPN Gateways that are designed to offload IPSEC and GENEVE [29, 30] tunnel processing to the hardware. This development will allow ISPs/CSPs to efficiently use the disaggregated clusters by running virtual functions on top without a compromise in throughput.

Network operators are continuously seeking technologies to reduce carbon footprint. One of the sustainable solutions can be found when what was once a sophisticated deployment becomes antiquated for its environment due to the changing conditions on the ground. Decommissioning a disaggregated cluster and re-using its hardware in a developing country where it can be practical for many years to come paves the way for more sustainability. It is the function/capacity separation of the DDC/DDBR architecture that facilitates resiliency in reusing the existing hardware to serve different geographical regions [31].

7. Conclusion

The explosion of the traffic growth over the last two decades has been somewhat unexpected and has urged telecommunications companies to explore alternative routing solutions. The disaggregated approach to the traditional chassis routers has been gaining significant interest from ISPs/CSPs' driven by a variety of motives such as cost reduction, the removal of vendor lock-in and service innovation. Regarding the talent pool, due to the proliferation of the innovative networking solutions, network architects and engineers should master new skills to stay relevant in the industry. Traditional chassis router architectures, having dutifully served for decades, impose multiple challenges like the traditional nature of the components, and the hurdles to overcome upgrade simplicity. DDC/DDBR architecture solves many of the outlined issues while presenting service agility and faster innovation. Spine-Leaf architecture has been deployed in the data centers of hyperscalers and backbone networks of the large telcos and cablecos. Although this approach has multiple drawbacks which are now addressed by the disaggregated solution, this approach is not only valid on paper but there are multiple scalable DDC/DDBR deployments in the backbone and aggregation networks of the largest cable and telecommunications companies in the world. On the automation side, cloud orchestration system is leveraged to streamline cluster management, device configuration and critical files retrieval. In the foreseeable future, development in the network disaggregation domain will focus on multiservice, session offload by leveraging open source OpenOffload API all while trying to remain cognizant of sustainability of the network. It seems as if we are getting closer to a more complete automated, portable, and easily scalable network for the future!

Abbreviations

ASIC	Application Specific Integrated Circuits
API	Application Programming Interface
BGP	Border Gateway Protocol
CLI	Command-line Interface
CNCF	Cloud Native Computing Foundation
COTS	Commercial Off-The-Shelf
CPU	Central Processing Unit
CSP	Communications Service Provider

DC	Data Center
DDBR	Disaggregated Distributed Backbone Router
DDC	Disaggregated Distributed Chassis
DOCSIS	Data Over Cable Service Interface Specification
DSCP	Differentiated Services Field Codepoints
E2E	End-to-end
ECMP	Equal Cost Multipath
FF	Fabric Forwarder
FRU	Field-Replaceable Unit
GE	Gigabit Ethernet
GENEVE	Generic Network Virtualization Encapsulation
GNMI	gRPC Network Management Interface
HA	High-Availability
HTTP	Hypertext Transfer Protocol
HW	Hardware
IGP	Internal Gateway Protocol
IGW	Internet Gateway Router
IoT	Internet of Things
IP	Internet Protocol
IPSEC	Internet Protocol Security
ISIS	Intermediate System - Intermediate System
ISP	Internet Service Provider
ISSU	In-Service Software Upgrade
JSON	JavaScript Object Notation
KPI	Key Performance Indicator
MoP	Method of Procedure
MPLS	Multiprotocol Label Switching
NETCONF	Network Configuration Protocol
NMS	Network Management Systems
NNI	Network-to-Network Interfaces
NOS	Network Operating System
NSR	Non-Stop-Routing

OCP	Open Compute Project
ODM	Original Design Manufacturers
OEM	Original Equipment Manufacturers
OIR	Online Insertion and Removal
ONIE	Open Network Install Environment
OSPF	Open Shortest Path First
PF	Packet Forwarder
PM	Performance Monitoring
Protobuf	Protocol Buffers
PSU	Power Supply Unit
QoE	Quality of Experience
QSFP-DD	Quad Small Form Factor Pluggable Double Density
REST	Representational state transfer
RPC	Remote Procedure Call
RPM	RPM Package Manager
RS	Routing System
S/L	Spine-Leaf
SCTE	Society of Cable Telecommunications Engineers
SLA	Service-level Agreement
SNMP	Simple Network Management Protocol
SW	Software
TAC	Technical Assistance Center
TCAM	Ternary Content Addressable Memory
TCP	Transmission Control Protocol
TIP	Telecom Infra Project
UNI	User Network Interfaces
VPN	Virtual Private Network
WAN	Wide Area Network
WIFI	Family of wireless network protocols/Wireless Fidelity
XML	Extensible Markup Language
YAML	YAML Ain't Markup Language
YANG	data modeling language
ZTP	Zero-touch Provisioning

Bibliography & References

[1] https://www.broadbandsearch.net/blog/internet-statistics

[2] https://corporate.comcast.com/stories/announcing-another-10g-milestone-amidst-a-flurry-of-innovation

[3] https://www.broadcom.com/blog/800g-optical-platform-solutions

[4] https://www.youtube.com/watch?v=d90uK4WW0Po

[5] *Hardware Specifications and Use Case Description for J2-DDC Routing System*, Tuan Duong; Open Compute Project

[6] *Distributed Disaggregated Backbone Router (DDBR) Technical Requirements Document*, Eva Rossi, Jose Angel Perez, Kenji Kumaki, Ryuji Matsunaga, Yuji Sonoki, Ahmed Hatem, Diego Marí Moretón; Telecom Infra Project

[7] https://www.opencompute.org/

[8] https://telecominfraproject.com/

[9] GigaOm Radar for Network Operating Systems: Network Service Providers, https://research.gigaom.com/report/gigaom-radar-for-network-operating-systems-network-service-providers/?utm_content=208592782&utm_medium=social&utm_source=linkedin&hss_channel=lcp-11282464

[10] https://www.cncf.io/

[11] https://www.opencompute.org/wiki/Networking/ONL

[12] *The Internet of Things: Catching up to an accelerating opportunity,* Michael Chui, Mark Collins, Mark Patel; McKinsey & Company

[13] https://restfulapi.net/

[14] https://datatracker.ietf.org/doc/html/rfc6241

[15] https://www.w3schools.com/xml/xml_whatis.asp

[16] https://www.w3schools.com/js/js_json_intro.asp

[17] https://yaml.org/

[18] https://datatracker.ietf.org/doc/html/rfc7950

[19] https://opencomputeproject.github.io/onie/overview/index.html

[20] *Optical Transceivers for Datacom & Telecom Market 2021: Market and Technology Report 2021,* Dr. Martin Vallo, Pars Mukish, Dr. Eric Mounier; Yole Development

[21] *A Study of Non-Blocking Switching Networks,* Charles Clos; The Bell System Technical Journal, March 1953

[22] *On Nonblocking Folded-Clos Networks in Computer Communication Environments,* Xin Yuan, Department of Computer Science, Florida State University, Tallahassee, FL 32306, 2011

[23] https://www.ufispace.com/company/blog/what-is-a-distributed-disaggregated-chassis-ddc

[24] *Engineered Elephant Flows for Boosting Application Performance in Large-Scale CLOS Networks,* Broadcom Corporation, March 2014

[25] https://grpc.io/

[26] https://about.att.com/story/2020/open_disaggregated_core_router.html

[27] *Multiservice: Maximizing Infrastructure Utilization;* DriveNets Whitepaper

[28] https://github.com/att/sessionOffload

[29] https://datatracker.ietf.org/doc/html/rfc6071

[30] https://datatracker.ietf.org/doc/html/rfc8926

[31] *Topology and component description of DriveNets Network Cloud as a sustainable solution for multipurpose networking,* Run Almog; Open Compute Project

[32] *Annual outage analysis 2021,* Andy Lawrence; Uptime Institute
https://uptimeinstitute.com/uptime_assets/25ff186d278b32c202fc782e60a0d473bd72bfbc6d4d65afedfa15dd406c7656-annual-outage-analysis-2021.pdf