Beyond Edge Cloud: Distributed Edge Computing

Nihel Benzaoui
Nokia Bell-Labs France, Route de Villejust, Nozay France
Nihel_djoher.benzaoui@nokia-bell-labs.com

Abstract: High bandwidth demands combined with low latency applications lead the move from centralized cloud to distributed Edge Computing. We discuss how this paradigm shift impacts network interconnects design and the key network features to truly enable 5G and beyond. © 2020 Nokia Bell Labs

1. From centralized to distributed Edge Data Centers:

In the past decade, research has been active on web-scale data center switching fabrics, especially on networking solutions capable of delivering high switching capacity while ensuring low latency. Today, large data centers (DC) is still a hot topic, and players such as Facebook [1] and Google [2] are periodically (e.g., every 5 years) redesigning their intra DC network to cope with the ever-growing demands of scale. Bandwidth demand increase is not only due to users' proliferation but also to the start of intensive use of machine learning (ML) and artificial intelligence (AI); Facebook declared that AI inference demands are doubling each year in their DCs [3]. Google stated that a workflow currently communicates with thousands of servers and each server is hosting hundreds of workflows, and in the next decade, those numbers are expected to be 100 folds. The main foreseen possible solutions to help large DCs to scale are disaggregated [4] and distributed [5] computing.

In parallel with large DC evolution, the emergence of 5G opened the opportunity for a new generation of time-sensitive and contextualized-experience applications. A new type of DC had to be proposed; the Edge DC for Edge Computing. The principle of Edge Computing is to reduce propagation delay (1ms target) by storing and processing time-sensitive data closer to the user. The consequence is a DC architecture shift, from a fully centralized to a highly distributed computing environment. In [6] the assumption is that by 2025, 80% of enterprises will have shut down their traditional DC, versus 10% today. [7] describes the ultimate Edge Computing as an aggregate of storage and compute resources, regardless of their distributed nature, acting as a single fabric used to deliver services.

Even if the scale of a single Edge DC is many orders below a large DC, large scale distributed Edge Computing is comparable to large DC in terms of scale and complexity. Edge computing will inherit from technological advances such as hardware disaggregation but will be confronted to performance issues such as latency control. [8] explains the impact of the lack of latency control on large scale DCs. For an online service, user requests must be satisfied within a specified latency target; the completeness of the responses directly impacts the quality of service and in turn, the operator revenue. Furthermore, horizontal scalability entails that online services have a partition-aggregate workflow [9] (e.g., Hadoop). Application latency targets cascade down to targets for workflows at each layer. Any network workflow is associated with a deadline. The workflow is useful and contributes to the application throughput if, and only if, it completes within the deadline. Targets are in the range of 10 to 100ms [9], which leaves only a fraction of this latency target to the network.

In [10], authors further explain that not only average latency matters but its variation can have a strong impact inside the DC. Latency control is considered so crucial for interactive applications, that software builders have used very complex techniques to compensate for the network performance uncertainty. In [10], Google considers software techniques that tolerate latency variability vital for building responsive large-scale services. Authors make the strong statement that even rare performance hiccups affect a significant fraction of all requests in large-scale distributed systems. They claim that it is challenging for service providers to keep the tail of latency distribution short for interactive services as the size and complexity of the system grows. This is even true in an environment running complex time-sensitive applications over a large distributed Edge Computing network.
This sets a clear challenge on distributed DC environment, especially on network interconnects, to deliver a deterministic low latency, as highlighted by Google [11], Facebook [3], and Alibaba [12]. We highlight that in the context of distributed Edge Computing, the "network interconnect" must be considered as an end-to-end structure combining both intra and inter DC interconnects.
In the following, in Section 2 we review the directions of technological evolution in distributed DC area and their impact on network interconnects' design. In Sections 3 and 4, we discuss the key enablers of distributed DCs and present Dynamic and Deterministic Network (DDN) as a candidate for the Edge Computing network interconnect.

2. Impact of distributed Edge Computing on DC network design:

In this section, we discuss the impact of DC hardware design and applications' migration towards distributed Edge Computing on network interconnects.

Online gaming: Cloud gaming (e.g., xCloud by Microsoft, Stadia by Google), relies entirely on Edge Computing to replace users' hardware, and on the network to interface the Edge with users' display. The burden is on the network to deliver fast communication. In [13], Deutsche Telekom explains that an end-to-end latency of 50-80 ms ensures a smooth gaming experience, while a latency of 120 ms noticeably impacts responsiveness. In cloud gaming, not only latency matters, but jitter is even more important: average latency may be low (below 50 ms), but peaks at 100 ms will heavily impact the gameplay. Moreover, to keep the gameplay fair across players, the network must deliver consistent latency and jitter to all users.

Data analytics: "Data is priceless in the seconds after it's collected, any latency introduced by the network will make that data losing of its value" [14]. Real-time data analytics requires low latency since the delivered output is used to interact with a user/system. Hence, latency drove the shift of data analytics execution from a centralized approach in large scale DC that was delaying timeliness of the analytics (up to 19 folds) [14], to a distributed approach over Edge Computing. Data analytics incurs hard challenges on the network interconnect: 1) High bandwidth combined with low latency, as required for Video analytics – qualified as the killer application for large distributed Edge Computing by Microsoft [15]; 2) Achieving real-time analytics systems with high accuracy outputs; accuracy is directly impacted by the delay between a video frame capture and its processing; 3) Synchronization of data access between data analytics jobs to avoid job blocking by the slowest one. A common solution to face these challenges lies in controlling latency. Network interconnects should provide deterministic latency that can be tuned through resource scheduling (e.g., [15] for analytics accuracy).

Distributed Deep Learning (DL): Originally, DL models training (e.g., used for data analytics and intelligent manufacturing) moved from centralized DC to Edge Computing to avoid costly migration of large amounts of data across the network, and enable faster inference [16]. Recent DL technique advances made online and incremental learning possible. By moving to the edge, DL can take advantage of the large and contextualized data generated by the geo-distributed devices. Additionally, training a large DL model generally requires important computation power (thousands of CPUs). Distributed computing facilitates the training process by taking full advantage of parallel GPUs [17]. The network interconnect needs to deliver low and controlled latency over distributed Edge Computing to enable a single computing fabric view for DL applications.

Disaggregated hardware and high-performance storage: For high-performance computing using disaggregated hardware, applications such as large-scale machine learning training (e.g., running on GPUs) transfer large volumes of data. Given that data storage and computation speeds are considered very fast, the performance bottleneck is the network interconnect [12]. In [12], Alibaba pointed out the challenge to reconcile low latency and high bandwidth utilization. In a high-speed DC environment, congestion can rapidly occur when flows start at line rate and aggressively grab available network capacity. Congestion cannot be afforded for applications such as machine learning that typically ask for an average latency of 100 µs and expect a tail latency of 50 µs for remote memory access. More generally, according to [12], resource disaggregation requires 5 µs network latency to maintain good application-level performance. Optimally, networks should not be optimized only to deliver traffic as fast as possible, but they should aim to deliver it just in time to compute resources [18]. A data flow can be prioritized by the network to reach a busy compute resource at the expense of other flows. The overall performance could be improved if the network was aware of computing resource state and be scheduled to deliver data just in time.

Industry 4.0: The main benefit of introducing edge cloud (for software and control execution) to the industry would be an optimized and timely coordinated production chain with reduced downtimes [19]. In [20] two challenges are highlighted in the factory floor to support real-time services (Augmented Reality, Remote robot system, and motion control): 1) achieve low and strictly deterministic latency, and 2) achieve high-precision time synchronization. To truly enable the 4th revolution of the industry, network interconnects with deterministic latency is key.

3. Enablers of deterministic distributed Edge Computing:

Given the observations previously made on the impact of distributed Edge Computing on network interconnects, we claim that three network features are essential to enable a deterministic DC environment.

Time slotted access: Handling low latency applications with fixed jitter and low packet loss ratio will be complicated and expensive through L2/L3 mechanisms only without any form of hard pipe enforcements [19].

Inside the DC, many efforts to control congestion [9][12][21][22] have been proposed, but none was sufficient to deliver the required deterministic performance. The only way to achieve strict control of latency is through time-slotted networks where a fraction of network resources can be scheduled on a per-flow basis.

Real-time control plane and scheduling: In Edge Computing, processing and scheduling latency cannot directly benefit from the closer deployment of DCs. When so many efforts have been made to reduce end-to-end propagation latency to less than 1 ms on the one hand and to increase link speed, on the other hand, it becomes essential to reduce the time for control plane communication and resource scheduling decision [23]

Cross-domain, cross-layer end-to-end orchestration: As mentioned by A. Vahdat in [11], performance only matters if it can be ensured end-to-end. This applies to both 1) cross-network domains [24]: there is a need to coordinate allocation from computing resource down to wireless capacity in order to improve quality of experience; and 2) cross-network elements: each layer or disaggregated hardware crossed by the data will impact its latency, thus a coordination of resource scheduling is needed to respect the required deterministic latency.

4. Dynamic Deterministic Network (DDN), a candidate for distributed Edge Computing:

As a solution to truly enable the distributed Edge Computing, we proposed DDN [25], a homogeneous time-slotted network fabric for inter (Optical Ethernet) and intra (Cloud Burst Optical Slot Switching) edge DC interconnect. In DDN, client packets are aggregated into short time slots (few microseconds) that may either be used opportunistically or reserved (scheduled) to carry time-sensitive data traffic to guarantee channel access in time and/or capacity. Opportunism decreases scheduling complexity while allowing statistical multiplexing. Resource allocation is centrally managed by a real-time controller [26]. The controller of each network domain calculates and distributes a slot reservations schedule to DDN nodes in its perimeter. Through the collaboration of real-time controllers from all domains, slots may be dynamically reserved end-to-end to deliver slot-based virtual circuits. Time-sensitive flows can, therefore, be physically isolated and carried across the network without interaction with best-effort traffic or between themselves, hence ensuring end-to-end deterministic and low latency per application.

5. Acknowledgment:

The author would like to thank his colleague Yvan Pointurier for the helpful discussions and technical insights.

6. References:

[1] A. Andreyev et al., “Reinventing Facebook’s data center network,” Facebook Engineering, Mar. 2019,
[2] A. Singh et al., "Jupiter Rising: A Decade of Clos Topologies and Centralized Control in Google's Datacenter Network," SIGCOMM Comput. Commun. Rev., vol. 45, no. 4, pp. 183-197, Oct. 2015.
[3] K. Schmidtke, “Cloud scale-computing,” Keynote talk, ECOC, Dublin, Ireland, Sept. 2019. URL: https://tv.theiet.org/?videoid=13393
[4] G. Zervas et al., “Optically disaggregated data centers with minimal remote memory latency: Technologies, architectures, and resource allocation [invited],” JOCN, vol. 10, no. 2, pp. A270–A285, Feb. 2018.
[5] D. Sood, et al., “Survey of Computing Technologies: Distributed, Utility, Cluster, Grid and Cloud Computing,” JNCET, vol. 6, no. 5, 2016.
[6] D. Cappuccio, “The Data Center is Dead,” Gartner Blog Network, July 2018.
[7] 5G Americas, “5G at the Edge,” 5G Americas whitepaper, Oct. 2019. URL: https://www.5gamericas.org/5g-at-the-edge/
[8] C. Wilson et al., “Better Never Than Late: Meeting Deadlines in Datacenter Networks,” SIGCOMM Comput. Commun. Rev., vol. 41, no. 4, pp. 50-61, Aug. 2011.
[9] M. Alizadeh et al., “Data center TCP (DCTCP),” SIGCOMM Comput. Commun. Rev., vol. 40, no. 4, pp. 63-74, Oct. 2010.
[10] J. Dean et al., “The tail at scale,” Commun. ACM, vol. 56, no. 2, pp. 74–80, Feb. 2013.
[11] A. Vahdat, “Networking challenges for the next decade,” Google Networking Research Summit Keynote Talks, Feb. 2017. URL: https://www.youtube.com/watch?v=5N7QS5vP68o
[12] Y. Li, “HPCC: High Precision Congestion Control”, in Proc. SIGCOMM, Aug. 2019, Beijing, China, pp 44-58.
[13] R. Rubenstein, “Deutsche Telekom’s edge for cloud gaming,” Gazettabyte, Oct. 2019.
[14] D. Huang, “Edge Clouds – Pushing the Boundary of Mobile Clouds,” Mobile Cloud Computing, Morgan Kaufmann, 2018, pp. 153-176.
[15] G. Ananthanarayanan et al., "Real-Time Video Analytics: The Killer App for Edge Computing," Computer, vol. 50, no. 10, pp. 58-67, 2017.
[16] R. Hong et al., “DLion: Decentralized Distributed Deep Learning in Micro-Clouds,” HotCloud, Renton, Washington, DC, Jul. 2019.
[17] Y. Han et al., “Convergence of Edge Computing and Deep Learning: A Comprehensive Survey,” ArXiv, Jul. 2019, no abs/1907.08349.
[18] A. Ousterhout et al., “Just In Time Delivery Leveraging Operating Systems Knowledge for Better Datacenter Congestion Control,” HotCloud, Renton, Washington, DC, Jul. 2019.
[19] C. Pallasch et al., "Edge Powered Industrial Control: Concept for Combining Cloud and Automation Technologies," EDGE, San Francisco, CA, Jul. 2018, pp. 130-134.
[20] L. Geng et al., “Problem Statement of Edge Computing on Premises for Industrial IoT,” Internet Engineering Task Force, Mar. 2018.
[21] N. Cardwell et al., “BBR: Congestion-Based Congestion Control,” ACM Queue, vol. 14, no. 5, pp. 20-53, Sep.-Oct. 2016.
[22] R. Mittal et al., “TIMELY: RTT-based Congestion Control for the Datacenter,” SIGCOMM Comput. Commun. Rev., vol. 45, no. 4, pp. 537-550, Aug. 2015.
[23] V. Shrivastav, “Fast, Scalable, and Programmable Packet Scheduler in Hardware,” in Proc. SIGCOMM, Beijing, Aug. 2019, pp. 367-379.
[24] S. Yi et al., “A Survey of Fog Computing: Concepts, Applications, and Issues,” in Proc. Mobidata, Hangzhou, China, Aug. 2015, pp. 37-42.
[25] N. Benzaoui et al., "Deterministic Dynamic Networks (DDN)," JLT, vol. 37, no. 14, pp. 3465-3474, Jul. 2019.
[26] M. Szczerban et al., “Real-time Control for Deterministic and Dynamic Networks”, in Proc. ECOC, Dublin, Ireland, Sep. 2019, Tu.3.E.3.