open EDGE computing

How Close to the Edge?

Edge Computing and Carrier Interexchange

An Open Edge Computing Whitepaper

CONTENTS

EXECUTIVE SUMMARY

In edge computing [1], there are many scenarios where users and edge computing nodes or cloudlets [2] need to interact even though they may be connected to different carrier networks. Ubiquitous edge cloud federation [4], when it is implemented, will address some interaction scenarios but not others. In these exception cases, traffic between users and between a user and cloudlet will pass through a carrier interexchange point (IXP). Prior to edge computing, IXPs were not systematically placed close to users. Distant IXP placement can cause long latencies between users and cloudlets even when the serving cloudlet and users are in the same metropolitan area. These long latencies are a high barrier to the feasibility of edge-native applications.

In early 2020, the Open Edge Computing (OEC) Initiative [3] established a workstream to investigate this IXP placement challenge. This whitepaper presents the key findings from the workstream and our recommendations for carrier action. The business success of edge computing depends on addressing limitations imposed by the industry's legacy approach to carrier interconnect. These limitations render many edge-native applications unusable in the many scenarios. Our workstream results show that viable edge computing requires:

- Regardless of IXP location, edge computing "cloudlets" must be located in the same metro/region as the application users. Without this, end-to-end latency becomes unacceptable for many applications.
- Once metro cloudlets are deployed, IXPs must be established within the metro area and networks engineered to prevent user to cloudlet data paths outside of the metro. Since cloudlets will often be hosted on wired metro networks, metro IXPs will increase significantly in importance.
- Within the metro area, the marginal performance benefit to moving IXPs closer to the user (e.g., to the cell tower) is small and may not justify cost. This conclusion, however, depends on the value and requirements of the full set of edge applications to be deployed. For example, edge apps like augmented and virtual reality games that rely on very fast user and display responses require very low and consistent round-trip times to be acceptable. Achieving this will necessitate moving the edge closer to the gamer. IOT sensor applications requiring real-time or near real-time control responses will also likely need closer placements.
- Metro third-party neutral host IXPs will provide equivalent performance to direct carrier-to-carrier IXPs with potentially lower complexity.
- While application performance is the main criteria for IXP placement decisions, other requirements like lawful intercept and data geofencing also need to be considered. For example, many widely distributed IXPs make it more difficult for carriers to assure full compliance with lawful intercept regulations.

Given long planning and implementation cycles, carriers should begin work immediately to enable edge computing by deploying metro based IXPs with other carriers as soon as possible.

For further information on the work conducted to produce this whitepaper, see the "Interconnect Work Stream Report" from the Open Edge Computing Initiative [10].

THE INTEREXCHANGE IMPERATIVE

KEY TERMINOLOGY

Internet Exchange Point or Interexchange Point (IXP)
A physical location through which Internet infrastructure companies connect with each other. An IXP between carriers may be through the public internet or may be a private peering point.

Edge Computing
The use of a small data center or other facility at much closer network proximity than the cloud to end points such as IoT devices and mobile users.

Cloudlet
A trusted, resource-rich edge computer or cluster of computers well-connected to the Internet and available for use by nearby devices.

Edge Cloud
A set of geographically distributed cloudlets that are interconnected and configured into a cloud-like IaaS or PaaS service.

Edge Cloud Federation
Interconnection and interoperation between Edge Clouds from different operators to provide common edge services to users.

The simplest conception of carrier-hosted *edge computing* is a collection of geographically distributed edge computing nodes, or *cloudlets [2]*, operated by a carrier. These cloudlets may be integrated into an *edge cloud* that offers a set of IaaS or PaaS services to application providers. Edge clouds may be federated to offer edge services that span multiple carriers. This vision is outlined, for example, in the GSMA Operator Platform specification [4].

However, there are many scenarios where users and cloudlets need to interact even though they may be connected to different carrier networks. Ubiquitous edge cloud federation, when it is implemented, will address some interaction scenarios but not others. Some examples:

- Two or more gamers, on different mobile carrier networks, want to play together. The edge game server can only be on the same carrier network as one of them.
- A user roams into a carrier network without a local edge cloud.
- A user sometimes accesses an edge application from their home WiFi network and sometimes in their car over a mobile network. They want the same low-latency quality of experience for both environments, but the application's cloudlet is only attached to one of the networks.
- A wholesale edge cloud operator provides edge services to users on retail mobile networks without their own edge cloud.

In these scenarios, traffic between users and between a user and cloudlet will pass through a carrier interexchange point (IXP). An IXP is a physical location through which carriers connect with each other to exchange data. Prior to edge computing, IXPs were not systematically placed close to users. Scale economies drove IXP geographic centralization just as they drove the centralization of cloud computing. For this reason, many metropolitan areas may not have local IXPs between local access providers. This gap can cause long latencies between users and cloudlets even when the serving cloudlet is in the same metropolitan area as the user. For example, the figure below shows the actual path of packets between a mobile phone on a Pittsburgh wireless carrier to a cloudlet on a Pittsburgh wired network. These long paths lead to long round trip times for applications. For multi-user applications, traffic between users on different networks may also be similarly impacted. Depending on the locations of the users, cloudlet and IXP, this added end-to-end latency can be 10s to 100s of milliseconds. These long interconnect paths are a high barrier to the feasibility of low latency edge-native applications.

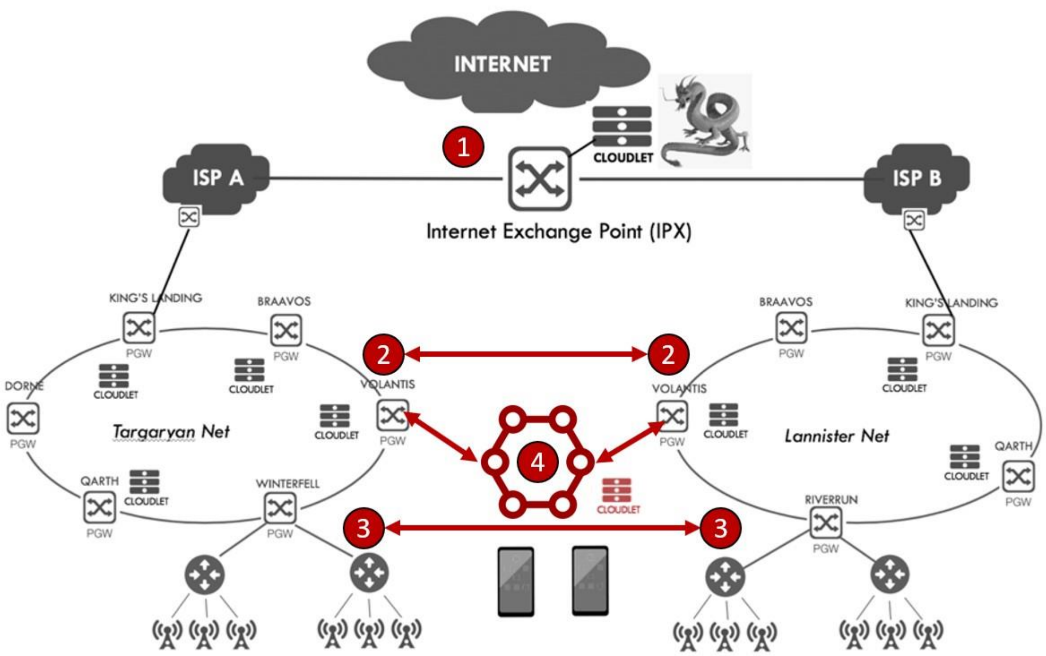
The solution is obvious: implement IXPs closer in the network to cloudlets and users. But, how close? MobileEdgeX's Tomasz Gerszberg, in his January 2020 Blog [5], outlined the options for edge friendly IXP placement. Placing IXPs near users, say, close to cell towers, will obviously result in the lowest round trip times between devices and cloudlets. However, this comes at a high cost in increased transport infrastructure and management. Insight on optimal IXP placement has not been available.

In early 2020, the Open Edge Computing (OEC) Initiative established a workstream to investigate the IXP placement challenge. A full OEC workstream report is available at [10]. This whitepaper presents the key findings from the workstream and our recommendations for carrier action.

BUT, WHERE?

Gerszberg's blog [5] posed 4 potential locations for IXP placement to better enable edge-native applications [6]. As shown below, these locations are:

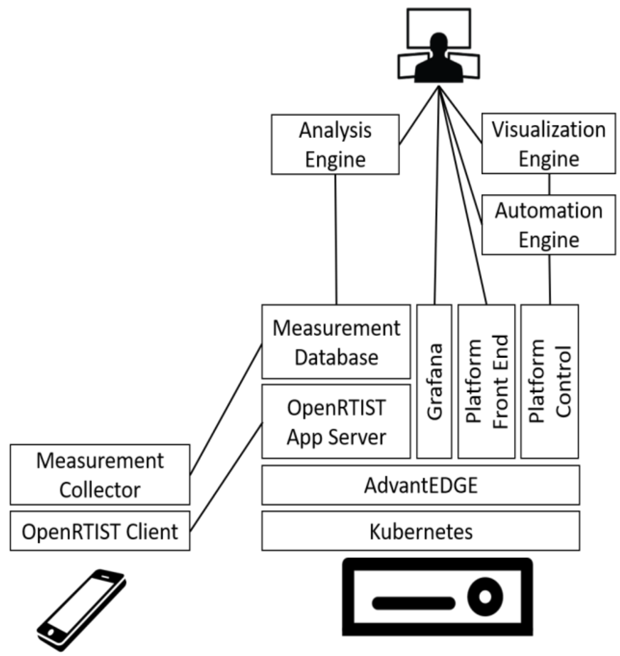
1. An IXP position geographically remote from the user and the cloudlet. This position is currently typical of many existing mobile networks.
2. An IXP position within the same metropolitan area as the user and cloudlet. This position would typically be somewhere in the metro core infrastructure. The figure shows the IXP relatively far from users. This location might typically be in one of several carrier co-location centers in that metro.
3. An IXP position in the radio access network (RAN). This position puts the IXP very close to the UE and the cloudlet. These IXPs might be at individual cell towers or, more likely, at some aggregation point in the RAN.
4. Two IXPs within the metro area core. In this case, the IXPs are provided by a third-party neutral host who transfers the data between the two carriers.



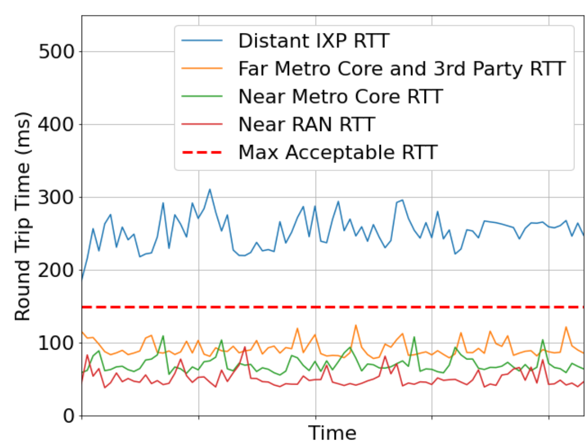
Carrier costs increase as the IXP is moved closer to the network edge. This cost increase derives from the increased number of IXPs and increased transport infrastructure required to connect to the IXPs. In the OEC workstream, we assumed that the optimal IXP position is the location furthest from the edge where the application user experience meets the minimum acceptable requirement. Application user experience, of course, depends on the application. A gaming application has much different experience metrics than a video analytics application. An edge cloud's application mix and the corresponding metrics will be highly diverse. However, we have found that for most latency sensitive edge-native applications, end-to-end network latency needs to be in the 10-60ms range. High end gaming and ultra-reliable and low-latency communications (URLLC) applications are exceptions that require sub-10ms end-to-end network latency. Our analysis focused on applications that require 10-60ms network latencies which we believe will represent most edge-native applications for the foreseeable future.

Evaluating the Options

To better understand the tradeoffs of these IXP placement options, the OEC chartered the Carnegie Mellon University Living Edge Lab (LEL) [7] to develop and run a simulation that evaluated application performance in different IXP and cloudlet placement scenarios. To execute the simulation, the LEL created a simulation environment around the AdvantEDGE emulation platform [8]. This environment (shown at right) was calibrated with real network characteristics supplied by OEC members and from Pittsburgh commercial mobile networks. The simulation collected user experience metrics from the OpenRTiST demonstration application [9] in different emulated and real network conditions. The IXP locations were varied through the four positions shown in the figure above.



For OpenRTiST, the primary user experience metric is the total round-trip time (RTT) including network and application latency. Long RTTs cause the application display to appear jerky and excessively delayed. Subjective observation allowed us to set an acceptability threshold at a somewhat arbitrary round-trip time of 150ms. Application latency, added by the processing time at the device and cloudlet, is approximately 40ms. That left about 110ms (or 55ms each way) for the network latency component of RTT.[1]



[1] Round-trip time is the sum of client and cloudlet application processing time and twice the end-to-end network latency. So, for OpenRTiST, 150ms = 40ms + 2 * 55ms.

The simulation measurements above show that all IXP placement locations within the metro area were adequate to meet OpenRTiST user experience requirements. However, the distant IXP location gave RTTs of ~250ms – unacceptable for this application.

This result is consistent with other edge computing network latency work but, for the first time, places it in the context of interexchange point placement.

KEY CONCLUSIONS

While this experiment cannot be considered a general study of the IXP placement problem, it does provide some key insights and conclusions that are likely to be broadly applicable.

The business success of edge computing depends on addressing limitations imposed by the industry's legacy approach to carrier interconnect. These limitations render many edge-native applications unusable in the many scenarios. Our workstream results show that viable edge computing requires:

- Regardless of IXP location, edge computing "cloudlets" must be located in the same metro/region as the application users. Without this, end-to-end latency becomes unacceptable for many applications.
- Once metro cloudlets are deployed, IXPs must be established within the metro area and networks engineered to prevent user to cloudlet data paths outside of the metro. Since cloudlets will often be hosted on wired metro networks, metro IXPs will increase significantly in importance.
- Within the metro area, the marginal performance benefit to moving IXPs and cloudlets closer to the user (e.g., to the cell tower) is small and may not justify cost. This conclusion, however, depends on the value and requirements of the full set of edge applications to be deployed. For example, edge apps like augmented and virtual reality games that rely on very fast user and display responses require very low and consistent round-trip times to be acceptable. Achieving this will necessitate moving the edge closer to the gamer. IOT sensor applications requiring real-time or near real-time control responses will also likely need closer placements.
- Metro third-party neutral host IXPs will provide equivalent performance to direct carrier-to-carrier IXPs with potentially lower complexity.
- While application performance is the main criteria for IXP placement decisions, other requirements like lawful intercept and data geofencing also need to be considered. For example, many widely distributed IXPs make it more difficult for carriers to assure full compliance with lawful intercept regulations.

Given long planning and implementation cycles, carriers should begin work immediately to enable edge computing by deploying metro based IXPs with other carriers as soon as possible.

This work is described in greater detail in [10] and [11]. If you have further questions, please reach out to us at info@openedgecomputing.org, visit www.openedgecomputing.org or follow us on Twitter @openedgecomput1. Thank you to OEC Members Vodafone, InterDigital and VaporIO for their assistance in this workstream.

This research was supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR001117C0051 and by the National Science Foundation (NSF) under grant number CNS-1518865 and the NSF Graduate Research Fellowship under grant numbers DGE1252522 and DGE1745016. Additional support was provided by Intel, Vodafone, Deutsche Telekom, Crown Castle, InterDigital, Seagate, Microsoft, VMware and the Conklin Kistler family fund. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the view(s) of their employers or the above funding sources.

REFERENCES

1. M. Satyanarayanan. "The Emergence of Edge Computing". In: *Computer* 50.1 (2017), pp. 30–39.
2. M. Satyanarayanan et al. "Cloudlets: at the leading edge of mobile-cloud convergence". In: *6th International Conference on Mobile Computing, Applications and Services*. 2014, pp. 1–9.
3. The Open Edge Computing Initiative. https://openedgecomputing.org/
4. "Operator Platform Telco Edge Proposal", Version 1.0, OPG.01, GSMA, 22 October 2020
5. Tomasz Gerszberg. Shared Edge Experience. https://www.linkedin.com/ pulse/shared-edge-experience-tomasz-gerszberg.
6. M. Satyanarayanan et al. "The Seminal Role of Edge-Native Applications". In: *2019 IEEE International Conference on Edge Computing (EDGE)*. 2019, pp. 33–40.
7. Carnegie Mellon University. *The Living Edge Lab*. https://openedgecomputing. org/living-edge-lab/ and https://www.cmu.edu/scs/edgecomputing/index.html
8. Michel Roy, Kevin Di Lallo, and Robert Gazda. AdvantEDGE: A Mobile Edge Emulation Platform (MEEP). https://github.com/InterDigitalInc/AdvantEDGE.
9. S. George et al. "OpenRTiST: End-to-End Benchmarking for Edge Computing". In: IEEE Pervasive Computing (2020), pp. 1–9. DOI: 10.1109/MPRV.2020.3028781.
10. The Open Edge Computing Initiative, "Interconnect Work Stream Report", February 2021. Contact info@openedgecomputing.org.
11. J. Blakley et al. "Simulating Edge Computing Environments to Optimize Application Experience", School of Computer Science Carnegie Mellon University, Technical Report CMU-CS-20-135, November 2020.

