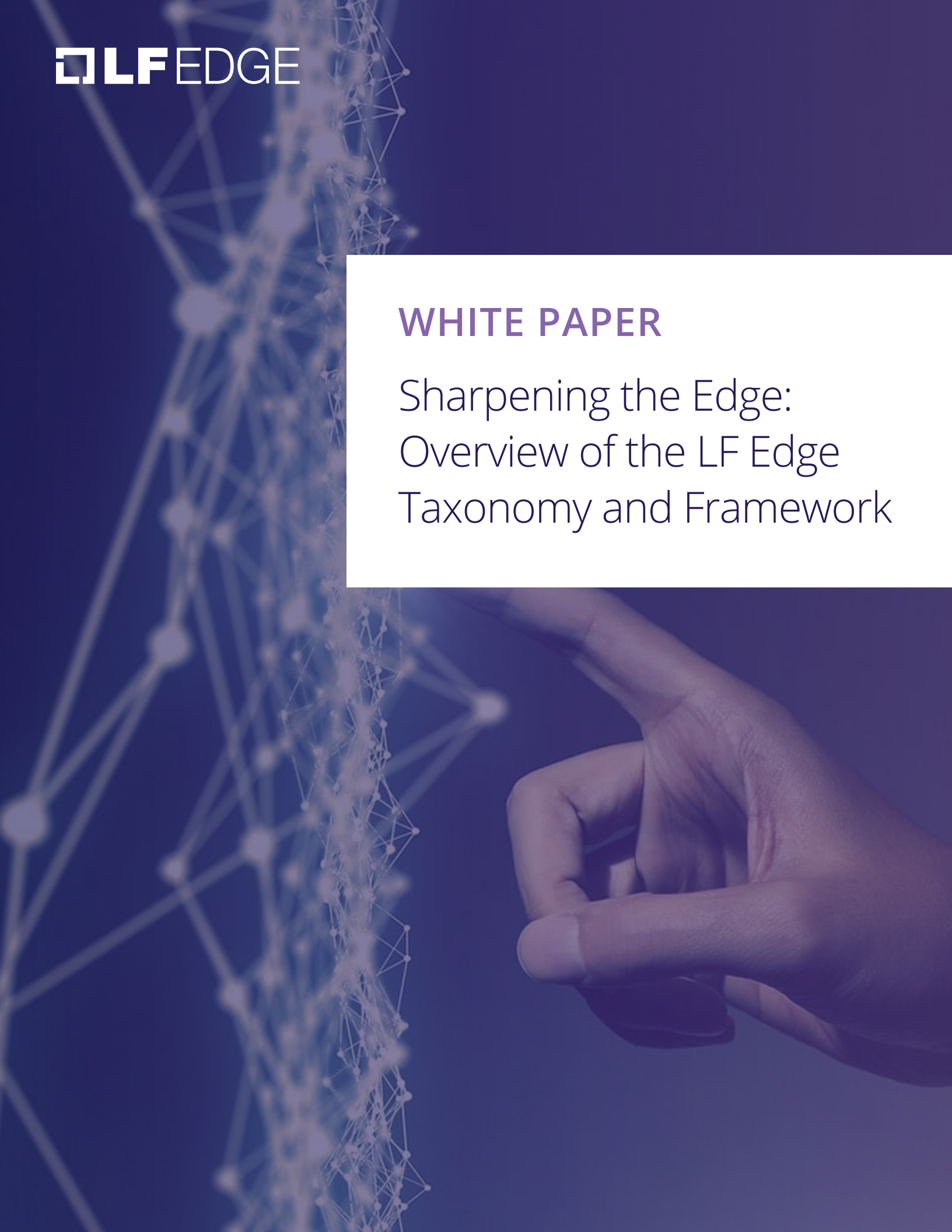
LF EDGE

WHITE PAPER

Sharpening the Edge: Overview of the LF Edge Taxonomy and Framework



LF EDGE

Table of Contents

1 Executive Summary 3
2 Introduction 3
2.1 Introducing the Edge Continuum 4
2.2 Extending Cloud Native Principles to the Edge 7
2.3 Considerations for the Service Provider Edge 8
2.3.1 Architectural Trends at the Service Provider Edge 9
2.3.2 Edge Application Deployment at the Service Provider Edge 10
2.3.3 Design Strategy for Backend Application Mobility Workloads at the Service Provider Edge 11
2.3.4 Design Strategy for User Device Mobility at the Service Provider Edge 12
2.3.5 Identifying the Optimum Edge Location to Serve a User 13
2.4 Considerations for the User Edge 14
2.4.1 Securing and Managing Distributed Devices 15
2.4.2 Accommodating both Legacy and Modern Applications 16
2.4.3 Addressing Protocol Fragmentation in IoT Use Cases 16
2.4.4 Latency-Critical Applications 16
2.4.5 Separation of Concerns in IT and OT Environments 17
2.5 Edge Deployment Patterns 17
2.6 Trends for Edge AI 18
2.7 Edge Computing Use Cases 19
2.7.1 Industrial IoT (IIoT) 19
2.7.2 Computer Vision 20
2.7.3 Augmented Reality (AR) 21
2.7.4 Retail 22
2.7.5 Gaming 23
2.7.6 Assisted Driving 24
2.7.7 Summary of the Edge Continuum 25
3 LF Edge Project Portfolio 26
3.1 LF Edge Project Summaries 26
3.1.1 Stage 3: Impact Projects 26
3.1.2 Stage 2: Growth Projects 27
3.1.3 Stage 1: At Large Projects 27
3.2 Project Focus Across the Edge Continuum 28
3.3 For more Information on LFE Projects 29
4 Summary 30

1 Executive Summary

Companies in a wide range of vertical markets are aggressively exploring new commercial opportunities that are enabled by extending cloud computing to the edge of the network. The concept of edge computing promises exciting new revenue opportunities resulting from the delivery of new types of services to new types of customers, in both consumer and enterprise segments.

Intended for readers interested in both the technical and business aspects of edge computing, this white paper introduces a set of open-source software projects hosted by the Linux Foundation (LF) and its subsidiary organization LF Edge (LFE). It outlines the LF Edge taxonomy and framework and describes opportunities for companies to participate in and benefit from these projects, accelerating the development, deployment and monetization of edge compute applications. The paper also introduces a set of open-source software projects hosted by the Linux Foundation (LF) and its subsidiary organization LF Edge (LFE).

The paper includes references to online resources associated with each project, providing developers with access to a wealth of technical information as well as the open-source software itself.

2 Introduction

Edge computing represents a new paradigm in which compute and storage are located at the edge of the network, as close as both necessary and feasible to the location where data is generated and consumed, and where actions are taken in the physical world. The optimal location of these compute resources is determined by the inherent tradeoffs between the benefits of centralization and decentralization.

This white paper introduces the LF Edge taxonomy and the key concepts of edge computing, highlighting emerging use cases in telecom, industrial, enterprise and consumer markets.

The paper also provides details of eight open source edge projects hosted by The Linux Foundation (LF) and its subsidiary LF Edge (LFE) umbrella organization. The LF is a non-profit technology consortium founded in 2000 to standardize Linux, support its growth and promote its commercial adoption. The LF and its projects have more than 1,500 corporate members from over 40 countries. The LF also benefits from over 30,000 individual contributors supporting more than 200 open source projects.

Founded in 2019, the mission of LF Edge is to establish an open, interoperable framework for edge computing independent of hardware, silicon, cloud or operating system.

2.1 Introducing the Edge Continuum

As defined in the Linux Foundation's **Open Glossary of Edge Computing**, edge computing is the delivery of computing capabilities to the logical extremes of a network in order to improve the performance, security, operating cost and reliability of applications and services. By shortening the distance between devices and the cloud resources that serve them, and also reducing the number of network hops, edge computing mitigates the latency and bandwidth constraints of today's internet, ushering in new classes of applications. In practical terms, this means distributing new resources and software stacks along the path between today's centralized data centers and the increasingly large number of deployed nodes in the field, on both the service provider and user sides of the last mile network.

In essence, edge computing is distributed cloud computing, comprising multiple application components interconnected by a network. Many of today's applications are already distributed, such as (1) a smartphone application with a cloud backend, (2) a consumer device, such as thermostat or a voice control system that connects directly to the cloud, (3) a smartwatch or sensor connected to a smartphone and then to the cloud and, (4) an industrial IoT (IIoT) system connected to an edge gateway and then to an on-premise system and/or the cloud. In addition, many LTE and 5G network functions are increasingly being distributed to the edge, enabling new business models and use cases, including those for dedicated private networks, fixed wireless access, SD-WAN and network slicing, thereby catering to the needs of many enterprises and vertical industries.

It helps to visualize edge computing through the continuum of physical infrastructure that comprises the internet, from centralized data centers to devices. By locating services at key points along this continuum, developers can better satisfy the latency requirements of their applications. Historically, cloud providers and Content Delivery Networks (CDNs) have reduced overall end-to-end latency by moving some services (such as the ability to cache data) out of centralized data centers and into distributed Points of Presence (POPs) closer to the devices being served. This has created a "cloud edge" or "internet edge" capable of improving the performance of traditional applications, such as streaming video and rich web content, but has not been enough to address many emerging applications, especially those that require a more sophisticated distribution of resources along the edge continuum for reasons of latency, bandwidth, autonomy, security and privacy.

This paper focuses on the two main edge tiers that straddle the last mile networks, the "**Service Provider Edge**" and the "**User Edge**", with each being further broken down into subcategories. Figure 1 summarizes the edge computing continuum, spanning from discrete distributed devices to centralized data centers, along with key trends that define the boundaries of each category. This includes the increasingly complex design tradeoffs that architects need to make the closer compute resources get to the physical world.

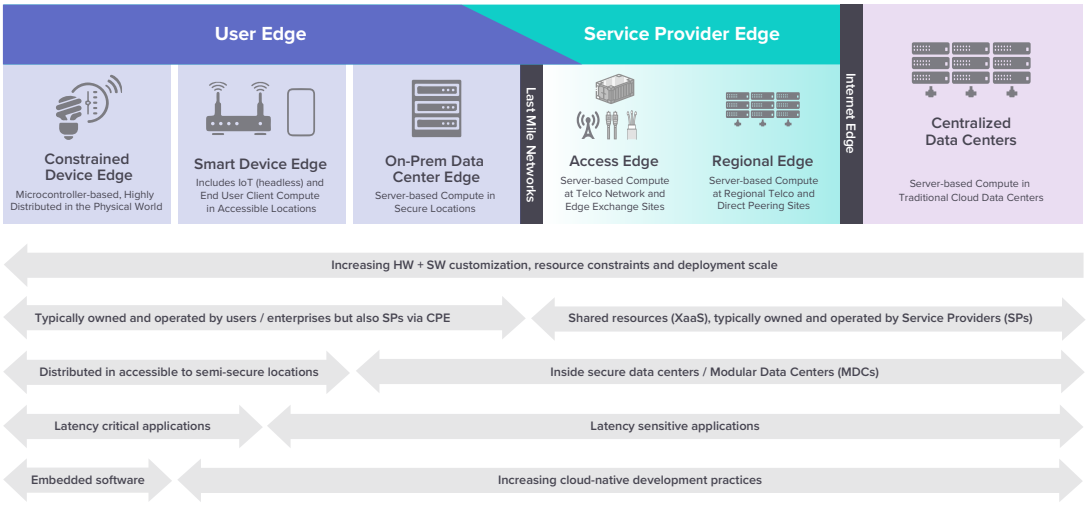


Figure 1: Summary of edge continuum

The far right of the diagram shows centralized data centers representing cloud-based compute. These facilities offer economies of scale and flexibility that are not possible or appropriate on a device. Centralized cloud resources are practically unlimited, whereas device resources are inherently constrained. A centralized cloud can oversee the collective behavior of a large number of devices, for example configuring, tracking and managing them, but it's limited by the centralized location of the data centers and the fact that the resources are shared.

Moving along the continuum from centralized data centers toward devices, the first main edge tier is the **Service Provider (SP) Edge**, providing services delivered over the global fixed/mobile networking infrastructure. Like the public cloud, infrastructure (compute, storage and networking) at the Service Provider Edge is often consumed as a service. Solutions at the Service Provider Edge can provide more security and privacy than the public cloud because of differences between the public internet and the private networks, including mobile cellular systems, operated by service providers. It leverages the existing trillion-dollar investments by Communications Service Providers (CSPs), who will have their own commodity servers in place at the network edge and will also cross-connect with cloud providers and bare-metal operators in nearby locations. Infrastructure at the Service Provider Edge is generally more standardized than infrastructure at the User Edge but there are still unique requirements for regulatory compliance and ruggedization, depending on where it is deployed.

The Service Provider Edge is distributed and brings edge computing resources much closer to end users. For example, CSPs can leverage their fixed and mobile networks at the edge and provide a platform for many edge applications, thus creating new business models and use cases as part of a network's evolution, such as when a wireless provider upgrades their network to 5G. Also, in the case of fixed networks, CSPs terminate their networks within enterprise buildings and homes in the form of Customer Premise Equipment (CPE) and these resources can be leveraged further to deliver various edge services.

The second top-level edge tier is the **User Edge** which is delineated from the Service Provider Edge by being on the other side of the last mile network. Sometimes it is a necessity to use on-premise and highly distributed compute resources that are closer to end-users and processes in the physical world in order to further reduce latency. Another common reason for placing compute at the User Edge is to conserve broadband network bandwidth, reducing the need to unnecessarily backhaul data across the last mile network, whether to compute and storage at the Service Provider Edge or all the way back to centralized data centers. Additional reasons for locating compute at the User Edge include autonomy, increased security and privacy, and lower overall cost, if the available resources match the need of the application workload. Compared to the Service Provider Edge, the User Edge represents a highly diverse mix of resources. As a general rule, the closer that edge compute resources get to the physical world, the more constrained and specialized they become.

As indicated in Figure 1, a key difference between the edge tiers is who owns the computing assets. Resources at the Service Provider Edge and within the public cloud are typically not owned by the end user but are, instead, shared across many users. In contrast, resources at the User Edge are typically dedicated and customer-owned and -operated. Applications that only use resources on the User Edge often result in a business model based on CAPEX rather than OPEX, with the infrastructure and technology acquisition, operational complexity and scaling being the responsibility of the user rather than delivered as a managed service. Increasingly, though, service providers (and cloud providers) are building managed service offerings that support and even include on-premise compute and networking infrastructure, making it possible to deliver applications that combine resources at both the User Edge and Service Provider Edge. Examples include a provider operating private cellular base stations for connectivity across a remote mining site or an analytics company providing Artificial Intelligence (AI) analysis and decision-support from the Service Provider Edge, supporting devices on the User Edge.

The edge computing taxonomy and associated terminology presented in this document were developed with careful consideration, seeking to balance various market lenses (e.g. cloud, telecom, cable, IT, OT/industrial, consumer) while also creating high-level taxonomy categories based on key technical and logistical tradeoffs. These tradeoffs include whether a compute resource is capable of supporting application abstraction (e.g. through containers and/or virtual machines), whether it is in a physically-secure data center or accessible, and whether it is on a LAN or a WAN relative to the process/user it serves (an important consideration if a use case is latency-critical vs. -sensitive). This document seeks to provide a holistic point of view without using edge terminology that may mean something to one entity but can be confusing to another. For example, the terms “near” and “far” edge are commonly used by telecom providers to distinguish between infrastructure closer to users/subscribers (far edge) versus infrastructure further upstream (near edge). This can be confusing because relative location is viewed through the eyes of the service provider instead of the user. In another example, the terms “thin” and “thick” have been used in some circles to characterize degrees of on-premise edge compute capability, however these terms do not delineate between resources at the User Edge that are physically secured in a data center versus distributed in accessible locations.

2.2 Extending Cloud Native Principles to the Edge

With the introduction of containerization and Kubernetes, a rapidly increasing number cloud-native software development based on platform-independent, microservice-based architecture and Continuous Integration / Continuous Delivery (CI/CD) practices for software enhancements. The same benefits of cloud-native development in the data center apply at the edge, enabling applications to be composed on the fly from best-in-class components, scaling up and out in a distributed fashion and evolving over time as developers continue to innovate.

In a perfect world, developers would have a universal foundation that enables them to deploy containerized workloads anywhere along the device to cloud data center continuum as needed, in order to balance the benefits of distributed and centralized computing depending on the use case and context. However, this isn't universally possible due to inherent technical and logistical tradeoffs, including the need to accommodate legacy investments while protecting safety-critical systems and processes.

As defined by the Open Glossary of Edge Computing, an "**edge-native application**" is one which is impractical or undesirable to operate entirely in a centralized data center. Edge-native applications leverage cloud-native principles while taking into account the unique characteristics of the edge in areas such as resource constraints, security, latency and autonomy. It is important to note that the term "edge-native" does not mean that an application isn't developed with the cloud in mind, rather, edge-native applications are designed to work in concert with upstream resources. An edge-optimized application that doesn't comprehend centralized cloud compute resources, remote management and orchestration, or leverage CI/CD isn't truly "edge native," rather it is a more traditional on-premise application. An example is a traditional Supervisory Control and Data Acquisition (SCADA) application within a nuclear power plant that has no connection to the cloud for security purposes. Ultimately, developers need a foundation that extends cloud-native principles as far down the edge continuum as feasible while accounting for inherent tradeoffs such as latency- and safety-critical processes.

2.3 Considerations for the Service Provider Edge

The Service Provider Edge consists of infrastructure on the other side of the last mile network from the User Edge. It consists of two subcategories, the Regional Edge and the Access Edge, with the former traditionally being associated with backhaul networks and the latter with front- and mid-haul networks.

To understand how the Service Provider Edge and its subcategories relate to each other and the rest of the internet, it helps to review how traffic routes to and from centralized data centers. Centralized data centers, such as those in Amazon's US West and US East regions, exist in specific locations that are far from most major metropolitan areas. These public cloud data centers connect to edge resources over internet backbones, which fan out across the continents and terminate in regional Internet Exchange Points (IXPs). IXPs exist in major cities and are the primary bridge between the access networks and the rest of the internet. For many reasons, centralized data centers are not well-suited for time-sensitive workloads, mainly because traffic from edge locations would need to travel relatively long distances and traverse multiple network hops, both of which add latency and jitter. An emerging trend, however, is for public cloud operators to create regional caches to address this issue.

As a result, providers are increasingly locating compute resources in data centers at the Regional Edge to reduce network hops while still retaining moderate scalability benefits compared to resources located at the User Edge. These edge sites are sometimes owned by telco network operators but equally common are the Multi-Tenant Colocation (MTCO) facilities owned by companies like Equinix and Digital Realty. These MTCO companies have built large regional data centers adjacent to the IXPs, often in the same building, and lease space for servers and other IT equipment to multiple tenants, including the major public clouds. A rich confluence of data passes through these locations. There is an emerging trend for non-telco providers to build direct peering sites that bridge compute resources in regional data centers to centralized cloud data centers through the Internet Exchange, for example a provider like Equinix peering with a public cloud. Further, CDN operators are evolving to enable customers to run custom applications at IXP sites. As a general rule, Regional Edge data centers are capable of supporting edge workloads that can tolerate latencies in the 30ms - 100ms range.

Also within the Service Provider Edge is the Access Edge which spans the "middle mile" between regional data centers and the actual last mile network. Access Edge sites include front- and mid-haul infrastructure spanning cell towers, cable head-ends, aggregation and pre-aggregation hubs and central offices, and other facilities which house network access equipment such as cellular radio base stations, as well as xDSL and xPON equipment. Service providers and edge colocation companies are repurposing existing facilities and deploying small-to-medium-scale micro data centers at or near these access site locations to provide "one-hop" proximity to the last mile network. These data center facilities support low-latency workloads, including those that require a predictable connection to the last mile network with latencies below one millisecond.

As with the Regional Edge, there are many companies deploying IT equipment at the Access Edge, including telcos at their network access sites, but there is also an emerging trend for new business models operated by edge co-location companies. Compute resources on the Access Edge and the Regional Edge can work in concert to balance trade-offs between scalability, cost, complexity and latency.

2.3.1 Architectural Trends at the Service Provider Edge

Many webscale design principles can be applied to implement cloud-like compute capabilities at the Service Provider Edge. Over the last few years, orchestration technologies like Kubernetes have made it possible to run cloud-native workloads in on-premise, hybrid or multi-cloud environments. Most applications offloaded to the Service Provider Edge will not require significant changes in their design or code and will retain continuous delivery pipelines that can deploy specific workloads at Service Provider Edge sites, such as those which have low latency, high bandwidth, or strict privacy needs. In addition, workloads may interact with networks in complex ways, such as to prioritize Quality of Service (QoS) for specific applications based on needs such as giving priority to life safety applications.

Major content owners like Netflix, Apple and YouTube are expected to retain their cache-based distribution models, which entail storing states in the centralized public cloud along with Authentication and Authorization (AA) functions, while redirecting the delivery of content from the “best” cache as determined by Quality of Experience (QoE) at the client device, where “best” doesn’t always means the nearest cache. This approach will be retained for other distributed workloads utilizing edge acceleration like Augmented Reality (AR), Virtual Reality (VR), Massively Multiplayer Gaming (MMPG) etc.

Content delivery networks such as Akamai and Cloudflare will maintain their existing distribution models but will also likely increase the number of PoPs in their network as well as extend their networks farther out in the Service Provider Edge as they look to enhance their content caching capabilities while expanding their other lines of business, such as their security and distributed workload products, which benefit from being farther out in the network and closer to the devices.

Cloud providers, including Amazon Web Services (AWS), Google Cloud Platform (GCP) and Microsoft Azure, are also expected to increase the number of PoPs in their network as well as extend their networks farther out to the Service Provider Edge. Each cloud provider will likely seek to differentiate their edge offerings in unique ways: some will focus on AI workloads, others will look to simply expand the number of regions in which users can provision resources, and others will look to build edge capabilities into their IoT toolchains.

According to the design principles mentioned above, the Service Provider Edge will need to ensure a deterministic method of measuring and enforcing QoE based on key application needs such as latency and bandwidth. As most internet traffic is encrypted, these guarantees will likely be based on the transport layer, leading to the evolution of congestion control algorithms which determine the rate of delivery. A similar design

principle will evolve for geographical data isolation policies for stores and workloads, beyond just complying with global data protection regulations.

Figure 2 shows an example deployment of highly-available edge applications at the Service Provider Edge, which could be federated across multiple service provider networks at peering sites while also cooperating with public cloud workloads.

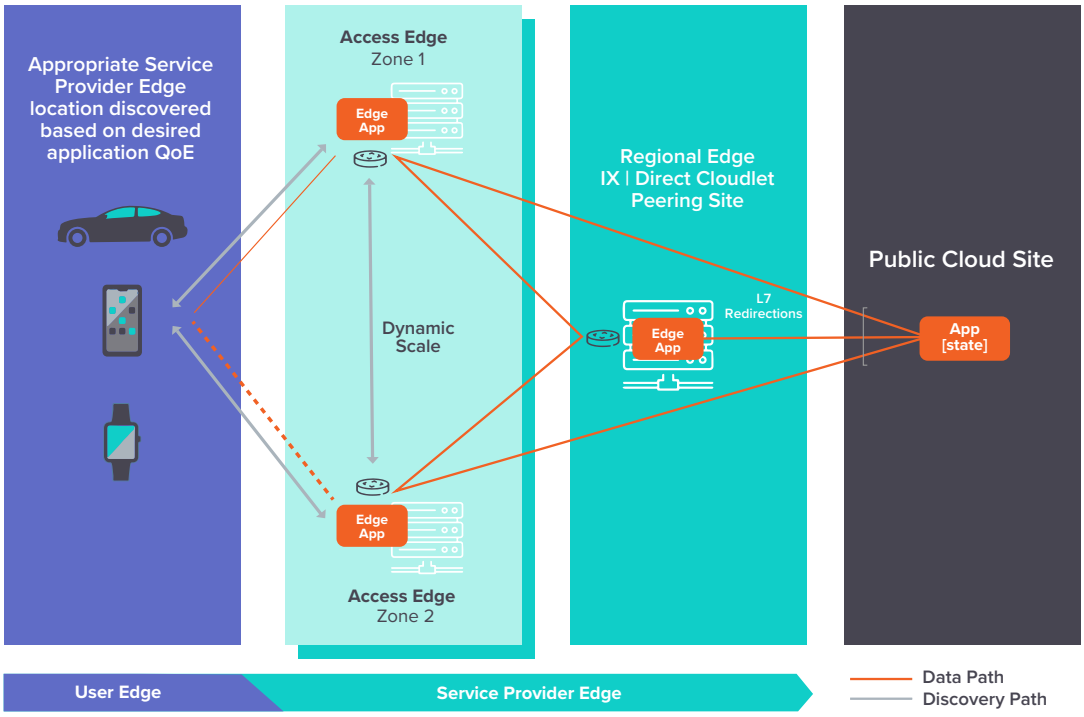


Figure 2. Example Service Provider Edge Application Deployment

2.3.2 Edge Application Deployment at the Service Provider Edge

Developers can study the geographical consumption patterns of their customers, as well as determine the optimal latencies and QOS requirements of their applications. Using Machine Learning (ML) algorithms, they can even predict how these patterns might change over time for advanced planning purposes. Orchestration services (such as custom Kubernetes schedulers) will emerge, and these will allow developers to specify their workload requirements in order to provide automated placement.

The deployment of application backends can be independent of network mobility or specific device attachment. Backend services deployment can be based on a number of different strategies to enable mobility of edge applications, including:

- Static, whereby the developer chooses the specific edge sites and the specific services for each site.

- Dynamic, whereby the developer submits criteria to an orchestration service and the orchestration service makes best-effort decisions about workload placement on behalf of the developer. One implementation of this would have developers choose a region in which they yield control to a system operator's or cloud operator's orchestration system in order to determine the optimum placement of workloads based on the number of requested compute instances, the number of users and any specialized resource policies.

The Akraino project is working on blueprints for the lifecycle management of edge applications based on the following workflow for deployment:

1. Create the cluster, deploying microservices as a set of containers or Virtual Machines (VMs);
2. Create the application manifest, defining an application mobility strategy that includes QoE, geographical store and privacy policies;
3. Create the application instance, launching the Edge Application and autoscaling.
4. For more information on this topic, please visit **the developer section for the Akraino Edge Stack project**.

2.3.3 Design Strategy for Backend Application Mobility Workloads at the Service Provider Edge

Workloads at the Service Provider Edge should instantiate and migrate based on demand and resource availability. For example, a backend for stateless applications might need to move across zones based on compute capacity, specialized resources and/or Service Level Agreement (SLA) boundaries. Stateful workloads can synchronize states from centralized servers and redirect them at layer 7 to edge applications, operating consistently, regardless of the orchestration system employed. The orchestration platform may offer periodic QoE hints to centralized servers to assist with the redirection process, but they can also operate independently.

2.3.4 Design Strategy for User Device Mobility at the Service Provider Edge

Since device mobility is based on route awareness, it's important to review how data moves across mobile networks before explaining the design principles of device mobility. A mobile device connected to a wireless network attaches to the nearest tower, then tunnels all application data to the nearest gateway which is further tunneled to the regional gateway, which is then transferred over the internet exchange to a public cloud and back. Regional gateways called packet gateways (PGWs) can be viewed as anchors, which CSPs utilize for enforcing centralized subscriber control, like policies, billing and management. Routing data in this way, however, is sub-optimal and cannot enforce the latency, bandwidth and privacy guarantees which edge applications require. Application backends at the Service Provider Edge are then challenged to follow individual consuming devices as they move from one region to another.

An easier solution is provided by local breakout , which allows service providers to place their anchors at edge sites, near the location of devices. Control and User Plane Separation (CUPS) for these packet gateways is a key step, deploying lightweight cost-effective distributed user plane functions (UPFs) at each edge site. Obtaining the GPS location for the UPF, if exposed from a centralized control plane, assists in identifying the nearest application backend. Another approach is for devices to attach to a geographically co-located anchor based on the physical location of the device, in which case local breakout works seamlessly with the edge cloud orchestration scheme behind these anchors.

Recent trends in 5G CUPS allows for local breakout and anchor redistribution, which is being deployed today. Network appliance vendors have started to virtualize their network functions, disaggregating their hardware from the software, and running network functions in virtual machines or containers. These are called Virtual Network Functions (VNFs) and Container Network Functions (CNFs). Network operators may use a common orchestration plane, such as that provided by Kubernetes, enabling CNF lifecycle management with a continuous delivery pipeline. The life cycle management techniques can be extended not only to anchors but to virtualized radio heads at the access edge.

The control plane separation will also allow Software Defined Networking (SDN)-like programming of tunnels to redirect traffic from devices to distributed endpoints. These tunnels can carry user application traffic as Packet Data User (PDU) sessions. Organizations like 3GPP are working on standards for redirecting edge application IP flows within PDU sessions so that they may be routed to the nearest anchors. The PDU sessions with embedded tunnel IDs as transport state present state synchronization issues, thus existing 3GPP session continuity procedures are not viable because they expect that a device will maintain PDU sessions across thousands of distributed anchors. Fortunately, the anchored routing structure can be changed by leveraging container

[1] Tunnels are GTP-U encapsulations over IP. For more information, see this Wikipedia entry.

[2] Local Breakout enables the Mobile Network Operator (MNO) to break out internet sessions into the Home network, to provide inbound roamers with an ability to order data, which is provided directly by the visited network.

mobility techniques used by web scale companies, but that requires not just virtualizing the compute (VNF/CNF) but also virtualizing the networks such that underlying IP routing is based on the identity of application and location of device. Identifier Locator Addressing is a means to implement network overlays without the use of encapsulation can help achieve anchorless device mobility.

2.3.5 Identifying the Optimum Edge Location to Serve a User

The nearest edge location is not always the best. Instead, clients must be steered to application backends based on the most recently recorded QoE for the application at each geographically-located edge site. The network may provide QoS mapping to improve QoE.

Based on this design, an application discovery engine could be embedded across multiple CSPs which records the health of the application backend and the QoE for each application, across all edge sites within a region, exposing a control API to identify the best location. This API can also be used to tune the rate of content delivery for the best experience. For example, content services like Netflix and YouTube maintain dozens of different bitrate encodings for the same movie or TV show, so that the optimal resolution can be delivered based on device characteristics, network congestion and other factors. A discovery engine can be employed that would return a ranked list of Uniform Resource Identifiers (URIs), identifying the optimum sites nearby, based on selection criteria that include:

- Edge application instances in sites geo-located based on the client's location;
- URI rank based on recent Layer 4 QoE measurements (latency and bitrate).

The LF Edge Akraino Edge Stack project has defined such an Application Discovery engine. Please visit the Find Cloudlet section for the Akraino Edge Stack project for more information and the definition of a control API implementation.

Later in the document there is a comprehensive list of use cases and workload attributes which individual CSPs can use today to serve enterprise- and privacy-centric edge use cases. However, to truly unlock the next generation of applications, developers must be able to deploy applications across multiple operator networks. One solution to this problem involves operators linking their edge cloud resources using a smart federation scheme at the last mile. Traditionally CSPs have federated to provide us global coverage, where sometimes they adopted the sub optimal approach of rerouting traffic to anchors in the home network. A more efficient strategy is for CSPs to federate directly via a peering exchange as described above, or even across last mile Radio Access Networks (RANs).

2.4 Considerations for the User Edge

The User Edge consists of a diverse mix of compute form factors and capabilities that get increasingly unique as deployments get closer and closer to the physical world. Technical tradeoffs at the User Edge include varying degrees of compute capability (especially system memory) as well as the need for specific I/O functionality to support both legacy and modern data sources and actuators. User Edge compute resources also require various degrees of ruggedization, including extreme temperature support with fanless design for reliability, specialized certifications (e.g. Class 1 / Division 2 for explosion proof), highly specific form factors, unique needs for Management and Orchestration (MANO) and security, and so forth. Moreover, while consumer-oriented devices tend to have a typical lifespan of 12-18 months, enterprise and industrial edge computing assets in the field need to support a long service life of 5 to 7+ years. Given all of these inherent tradeoffs, it is helpful to break the User Edge down further into several subcategories.

At the far-right end of the User Edge tier is the **On-Premise Data Center Edge** subcategory which can be considered server-class infrastructure located within traditional, physically-secure data centers and Modular Data Centers (MDCs), both inside and close to buildings like offices and factories. These resources tend to be owned and operated by a given enterprise and are moderately scalable within the confines of available real estate, power and cooling. Tools for security and MANO here are similar to those used in a centralized cloud data center, however there is some evolution required to support coordination across multiple locations, such as with Kubernetes clusters.

In the middle of the User Edge is the **Smart Device Edge**, which consists of hardware located outside of physically-secure data centers but still capable of supporting virtualization and/or containerization to support cloud-native software development. These resources span consumer-grade mobile devices and PCs to hardened, headless gateways and servers that are deployed for IoT use cases in challenging environments such as factory floors, building equipment rooms, farms and weatherproof enclosures distributed within a city. While capable of general-purpose compute, these devices are performance-constrained for various reasons including cost, battery life, form factor and ruggedization (both thermal and physical) and therefore have a practical limit to processing expandability when compared to resources in an upstream data center. As in the data center, there is an increasing trend for these systems to feature coprocessing to accelerate analytics, with the added benefit of distributing thermal dissipation which is beneficial in extreme environments. Resources at the Smart Device Edge can be deployed and used standalone (e.g. a smartphone, IoT gateway on a factory floor) or embedded into distributed, self-contained systems such as connected/autonomous vehicles, kiosks, oil wells and wind turbines.

At the farthest extreme of the User Edge tier is the **Constrained Device Edge** subcategory, represented by microcontroller-based devices that are highly distributed in the physical world. These devices range from simple, fixed-function sensors and actuators that perform little-to-no localized compute to more capable devices such as

Programmable-Logic Controllers (PLCs), Remote Terminal Units (RTUs) and Engine Control Units (ECUs) addressing latency-, time- and safety-critical applications. Devices at this tier leverage embedded software and have the most unique form factors to conform to highly specific environments and user experiences.

The Smart Device Edge includes both headless compute resources targeted at IoT use cases (e.g. gateways, embedded PCs, routers, ruggedized servers) and client devices that have a user interface (e.g., smartphones, tablets, PCs, gaming consoles, smart TVs). Together, headless, constrained and smart devices represent the "things" in IoT solutions, with smart devices providing localized general-purpose compute capability. The spectrum of compute devices targeting IoT workloads is often referred to as the "IoT Edge".

As a general trend in the area of networking, IoT use cases tend to be constrained by the upload of data collected from the physical world, whereas end user client use cases tend to be constrained by content download. This results in different considerations for applications, storage, network topologies and so forth, depending on the use case and available resources.

2.4.1 Securing and Managing Distributed Devices

Resources at the Constrained and Smart Device Edges are typically deployed and used in semi-secure to easily accessible locations in the field. As such, it is important to adopt a zero-trust security model and not pre-suppose a device is behind a network firewall. In all cases, distributed computing resources need a remote software update capability to avoid costly truck rolls, and in the case of on-premise data centers and smart devices, evolve their capability over time through modular, software-defined architecture. However, MANO and security solutions optimized for the data center are not suitable for the Constrained and Smart Device Edges due to the available compute footprint, deployment scale factor, potentially intermittent connectivity and typical lack of physical and network security. Solutions should also leverage techniques like Zero-Touch Provisioning (ZTP) to avoid requiring IT skill sets for secure deployment in the field.

IoT and client-centric compute resources at the Smart Device Edge are capable of leveraging MANO tools that support abstraction through containerization and virtualization and have headroom for security features like data encryption. Meanwhile, constrained devices leverage embedded software images that are typically tailored to the host hardware and may need to rely on a more capable device immediately upstream for added security measures. As a result, MANO tools for devices at the Constrained Devices Edge are often specific by manufacturer and silicon used. Meanwhile smart devices can afford the necessary abstraction to make MANO tools more standardized and platform independent, such as through the use of Linux with containers working across both x86 and Arm-based hardware or through a mobile operating system such as Android supporting applications on a variety of manufacturers' smartphones. Whenever possible, all key security functions (e.g. authentication, boot, encryption) should be enabled by a

hardware-based root of trust, such as Trusted Platform Module (TPM) or Arm TrustZone, but this is not always an option for highly-constrained devices.

2.4.2 Accommodating both Legacy and Modern Applications

As with centralized cloud data centers, many User Edge compute resources need to accommodate legacy applications in parallel with modern, cloud-native workloads. This is relatively straightforward in an on-premise data center through the use of well-established enterprise virtualization software together with solutions such as Kubernetes, however it is not feasible to leverage these same tools on more constrained hardware deployed in the field. Special consideration must be made for an abstraction layer that is optimized for resource-constrained hardware and comprehends the unique security needs for devices distributed outside a secure data center. The ability to abstract virtualized and/or containerized workloads on a given compute node is typically limited by available memory, with the practical lower limit being roughly 256MB: just enough to host an abstraction layer together with a workload. This memory constraint is the primary delineator between the Smart and Constrained Device Edges and is generally the limit for extending cloud-native software development practices closer to the source of data. Below this memory capacity, software needs to be embedded with tight coupling to hardware which limits flexibility and reduces the scope for expandability through abstracted, modular applications.

2.4.3 Addressing Protocol Fragmentation in IoT Use Cases

Compared to the entirely IP-based data flow spanning the Cloud, Service Provider and On-Premise Data Center Edges, resources for IoT workloads serving constrained and smart devices must comprehend a diverse mix of legacy and modern connectivity protocols, spanning wired and wireless transport as well as both standard and proprietary formats. This is especially the case in the IIoT space where there are hundreds of legacy protocol formats to comprehend. Rather than expecting one connectivity standard to dominate, it is important to have edge software frameworks that can normalize a variety of IoT data sources into desired IP-based formats for further processing upstream. Openness here enables users to retain control over their data by not getting locked into any particular backend service.

2.4.4 Latency-Critical Applications

Safety- and latency-critical applications that require “hard” real time operation for deterministic response comprise another key driver for running workloads at the User Edge. Resources like PLCs, RTUs and ECUs have been used in industrial process control, machinery, aircraft, vehicles and drones for many years, requiring a Real-Time Operating System (RTOS) and specialized, fixed-function logic. Time- and safety-critical processes such as controlling a machine, applying a vehicle’s brakes or deploying an airbag are

universally operated locally because they can't rely on control over a last-mile network, regardless of the speed and reliability of that connection. This scenario is contrasted with latency-sensitive applications such as video streaming that operate in "soft" real time and are often delivered by the Service Provider Edge for scalability. With latency-sensitive applications a networking issue can result in a poor user experience but will not cause a critical, potentially life-threatening failure.

2.4.5 Separation of Concerns in IT and OT Environments

OT organizations have historically isolated their layered industrial control infrastructure (e.g. PLCs, SCADA, DCS and MES systems) from broader networks, to ensure uptime, safety and security. However, a key aspect of Industrial IoT (IIoT) involves connecting these assets and the associated processes to networked intelligence to drive new outcomes. In order to create a separation of concerns from control systems with no risk of disrupting existing processes, industrial operators rely on network segmentation, typically installing a secondary "overlay network" that taps into data from existing control systems, in addition to new sensors installed throughout their environment to enable analytics workloads. Meanwhile, there is a trend for consolidation of mixed-criticality workloads on common infrastructure, for example with a virtualized "soft PLC" providing control functionality while additional virtualized and/or containerized data management, security and analytics applications run in parallel and interact with higher edge tiers. This consolidation requires specific considerations in the abstraction layer to ensure separation of concerns between these mixed-criticality workloads.

In summary, developers need flexible tools at the User Edge that enable them to run legacy, safety-critical, latency-critical, time-critical and modern containerized workloads concurrently while protecting their operations from undue risk, all while taking advantage of the scale benefits of working together with both the Service Provider Edge and the Cloud.

2.5 Edge Deployment Patterns

The sub-categories under the User Edge work with the Service Provider Edge and Cloud as part of a tiered compute continuum, but not necessary in series. Constrained and smart devices distributed in the physical world (such as smart thermostats, smartphones and connected vehicles) often communicate directly with the Service Provider Edge and Cloud through a router, bypassing all On-Premise Data Center infrastructure. Devices can also be deployed on-premise and interact with more capable local edge compute, which in turn interacts with the Service Provider Edge and Cloud. The continuum is a complex matrix of locality, capability, form factor and ownership. Figure 3 illustrates examples of various edge deployment patterns.

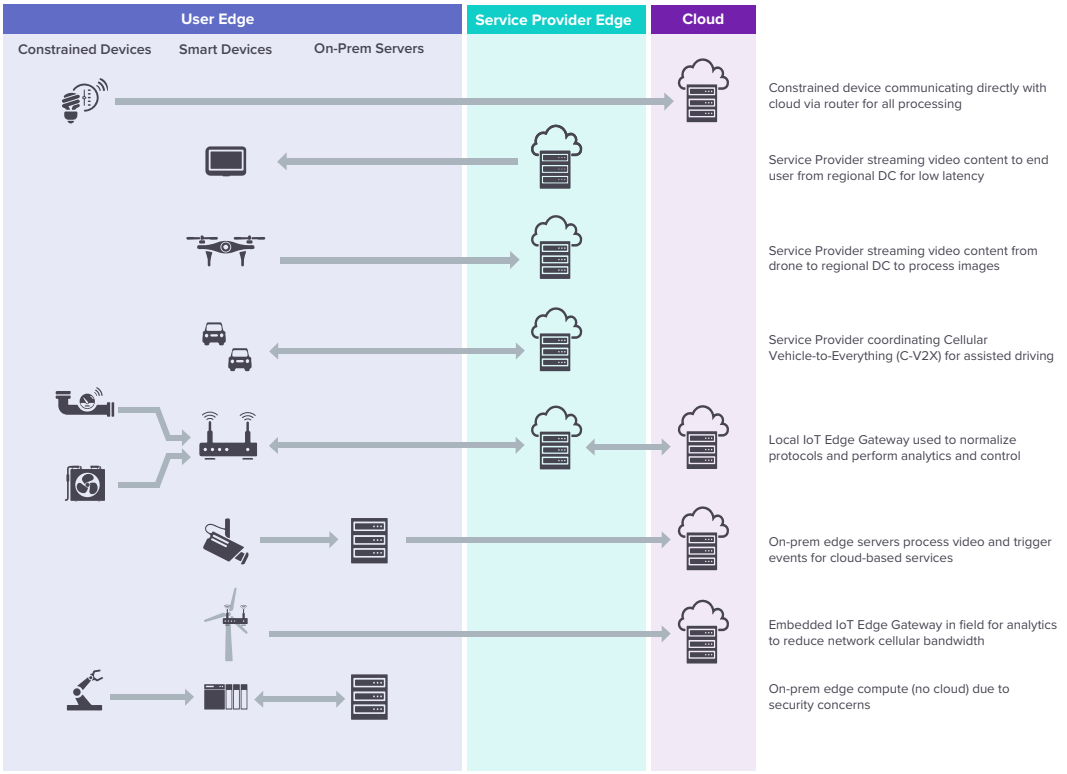


Figure 3. Example deployment patterns across the edge continuum.

Note that this diagram is simplified in the sense that it does not take into account that resources will be communicating "north, south, east and west" with multiple peers across the continuum, depending on use case.

2.6 Trends for Edge AI

Regarding Artificial Intelligence and Machine Learning (AI/ML) at the edge, the general trend is for deep learning and model training to occur where resources are plentiful, as in centralized cloud data centers, with models subsequently being pushed to more constrained resources at the Service Provider and User Edges for performing inferencing on data locally. The location of model execution along the edge continuum depends on a variety of factors, including addressing latency issues, ensuring autonomy, reducing network bandwidth consumption, improving end user privacy and meeting requirements for data sovereignty.

There is an emerging trend for running federated learning and even training models at the edge to address privacy and data sovereignty issues, although the potential for regional bias then needs to be considered. Another emerging trend at the Constrained Device Edge is deploying ML inferencing models in microcontroller-based resources. An example is a ML model that enables a smart speaker to recognize a wake word (e.g. "Hey Google" or "Hey Alexa") locally before subsequent voice interactions are powered by servers further up the compute continuum. Dubbed "Tiny ML", this requires specialized toolsets to accommodate the available processing resources and is outside of the scope of LF Edge at the time of this writing.

2.7 Edge Computing Use Cases

Enterprises in numerous market segments are deploying edge-hosted applications in order to capitalize on new business opportunities that are enabled by provisioning local compute as an extension of centralized cloud architectures. Figure 4 provides some examples of the wide number of use cases that benefit from edge computing and related enabling technologies.

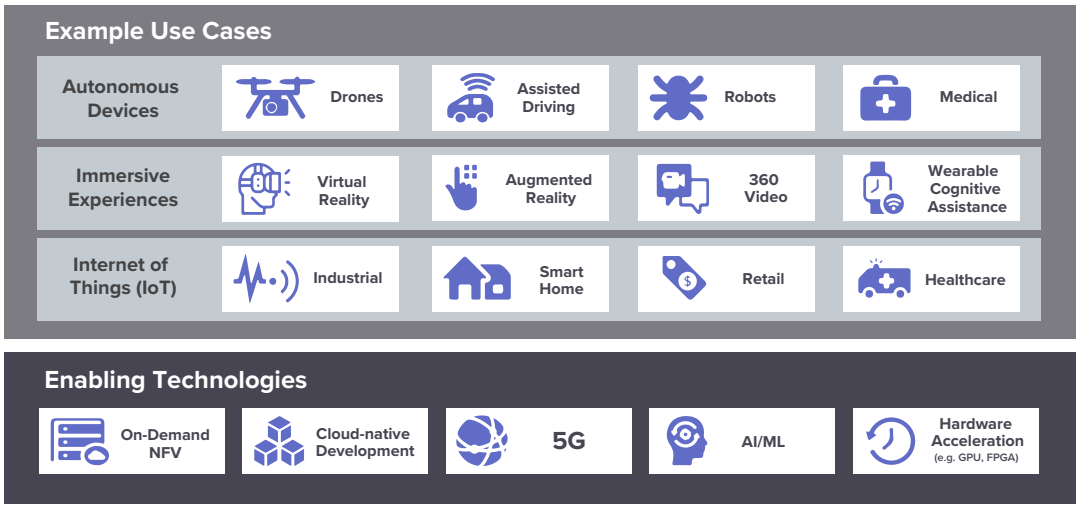


Figure 4. Example deployment patterns across the edge continuum.

This section discusses a variety of use cases to highlight key considerations and benefits:

- Industrial IoT;
- Computer Vision;
- Augmented Reality
- Retail;
- Gaming;
- Assisted Driving.

2.7.1 Industrial IoT (IIoT)

Edge compute delivers a number of important benefits for IIoT use cases in markets such as manufacturing, utilities, oil and gas, agriculture and mining.
With edge computing, industrial operators can perform time-critical analytics close to their sensors, machines and robots, reducing the latency for operational decision-making. This makes their processes more agile and responsive to changes. To maximize efficiency and minimize OPEX, functions necessary for real-time operation are hosted on-premise while those that are less time-critical may run in the public, private or hybrid cloud.

An edge compute architecture can ensure that high-value, proprietary information never leaves a factory. This can minimize the security threats associated with transmitting data to the cloud over public networks that are vulnerable to hacking, as well as help organizations meet data sovereignty requirements.

Remote operations such as mines or oil rigs typically have intermittent connectivity to the cloud and must be able to function autonomously. In these scenarios, on-premise compute enables real-time operational decisions based on the local analysis of sensor data without unnecessarily backhauling data over expensive wide area connections. Data required for long-term process optimization or multi-site aggregation is sent to the cloud whenever connectivity is available, which in certain locations might only be over a satellite link available at irregular intervals.

A significant amount of data is "perishable", meaning it is only valuable if acted on in the moment. The cost of connectivity through the service provider edge is minimized by processing data from sensors locally and sending only relevant information to the cloud, instead of raw streams of data. This is critical for high-bandwidth vibration data used for predictive maintenance use cases or devices like smart meters used in agriculture or utilities that connect to the cloud via low-bandwidth Narrowband IoT (NB-IoT) networks. In these cases, additional data might periodically be centralized in the cloud for the training of AI models that are then pushed close to operations at the edge for inferencing.

2.7.2 Computer Vision

Computer vision technology is widely used in video surveillance for law enforcement and building security, as well as monitoring industrial processes. Modern high-resolution IP cameras, however, generate significant volumes of data, for example 4 Mbps per device for an array of 4-megapixel (MP) cameras. While cameras can be configured to minimize bandwidth requirements by transmitting only when motion is detected, this is of no help in environments such as a city street or factory production line where the surveillance system network connection must be provisioned to cope with constant motion. Analyzing video in the cloud therefore requires a high-bandwidth network connection to transmit a continuous, high-resolution stream. If network bandwidth is constrained, the accuracy of the analysis is limited by the lower resolution of compressed video.

With edge compute, however, high-resolution video data is processed either within the smart camera itself as the edge node, or on a nearby edge server. High-end IP cameras have sufficient processing power to run algorithms such as facial recognition, leveraging analytics based on AI and/or deep learning technologies. Only selected events and/or video sequences that are flagged as important are transmitted to the cloud, for example an individual of interest, a vehicle with a specific license plate or a defective component. This significantly reduces the required network bandwidth while ensuring high quality, high accuracy analytics.

Edge compute also reduces latency, which is important for any time-critical vision-based detection scenarios such as factory automation or facial recognition. Continuous process control, for example, leverages the low latency associated with edge compute to ensure the near-real time detection of process deviations or manufacturing flaws, enabling production lines to be stopped or control parameters to be adjusted quickly enough to minimize wastage.

Drones used in applications such as surveying, package delivery and surveillance will leverage low-latency computer vision systems that perform object recognition for navigation within edge compute nodes on the ground rather than in heavy, power-hungry systems on the drone itself. This reduces the cost of the drones while also minimizing their power consumption, thereby maximizing both battery life and flight time.

Processing video at the User Edge also mitigates privacy concerns, especially in surveillance applications that are subject to regulatory constraints or in commercial applications where process information is valuable intellectual property.

2.7.3 Augmented Reality (AR)

Enterprises are increasingly adopting Augmented Reality (AR) in order to improve the efficiency of their operations, leveraging technology familiar to consumers through applications like Pokémon Go and its more sophisticated successors.

In industrial environments, AR can guide lesser-skilled workers through maintenance tasks without having an expert engineer on site at all times. This can either be done with a pre-scripted instruction overlay in the worker's field of view, or with a remote expert talking the on-site worker through performing a complicated task through their eyes. Similarly, in aerospace AR provides technicians with maintenance and diagnostic information via smart goggles, eliminating the need to physically reference bulky, complex and possibly outdated manuals in hard-to-reach locations inside a wing, fuselage or engine cowling.

AR applications typically analyze the output from a device's camera to supplement the user's experience. The application is aware of the user's position and the direction they are looking in, with this information provided via the camera view and/or positioning techniques. The application is then able to offer information in real-time to the user, but as soon as the user moves that information must be refreshed. Additionally, for many use cases it is valuable to update critical real time data from sensors in the user's field of view, for example the temperature and pressure of a tank while an operator is working through a maintenance procedure.

Edge compute improves the efficiency of enterprise AR by reducing the dizziness associated with high latency and slow frame refresh rates, that can otherwise lead to an experience that is frustrating, potentially nausea inducing and ultimately disorienting. Edge-hosted systems ensure predictable latency, resulting in a consistent experience for users instead of the constantly-changing delays that result from cloud-hosted

implementations. Moving compute power into edge servers located close to the user allows an AR application to eliminate the need for high processing bandwidth on goggles that therefore become expensive, power-hungry and too heavy for comfortable use over an extended period.

In another example, edge computing and AR are poised to deliver truly immersive media experiences for sports fans while at the game. Sports such as baseball, cricket, football and soccer have already held successful trials in "smart stadia" enabling spectators to stream video from unique, custom camera angles, including drones and spider-cams. "Virtual cameras" present views from within the field of play, giving spectators the opportunity to experience the action from the perspective of the players themselves. All these use cases require edge compute in order to guarantee the responsiveness that spectators expect while eliminating the need to backhaul prohibitive amounts of data to the cloud.

2.7.4 Retail

For brick-and-mortar retailers, almost 90% of global retail sales occur in physical stores so most retailers are investing in computing infrastructure located closer to the buyer, with edge computing as an extension of their centralized cloud environments. In-store edge environments focus on the digital experience of the customer, through edge applications supporting local devices such as smart signage, AR-based mirrors, kiosks and advanced self-checkout.

Retailers can deliver personalized coupons when shoppers walk into stores as WiFi, beaconing and computer vision systems recognize customers who previously signed up to connect while in-store. Smart fitting rooms equipped with AR mirrors can show shoppers in different clothing without the requirement to physically try them on. Meanwhile, infrared beacon and computer vision technology can generate heat maps that provide retailers with insights on in-store traffic patterns, allowing them to better configure their space and optimize their revenue-per-square-foot.

Infusing self-checkout systems with computer vision capability and integrating them with RFID and Point-of-Sale (PoS) systems gives them the ability to confirm that the item scanned by a customer matches what's in their bag, improving loss prevention. Vision algorithms can also be used to enable facial recognition to authenticate payments and gesture recognition for touchless commands, as well as the delivery of personalized offers at the point of sale.

Through the use of edge compute, retailers are able to ensure improved security for sensitive customer information. When data is transferred from devices to the cloud, security and compliance risks increase, but edge compute applications can filter information locally and only transfer data to the cloud that is required for strategic operational planning.

Edge compute provides a lean, highly reliable IT infrastructure for retailers that can run multiple applications while supporting the control and flexibility of cloud-based services. High-resiliency in-store micro data centers have become the solution of choice, managed and orchestrated remotely so that IT staff aren't required on-site. A chain of retail stores can be treated as an entire ecosystem rather than just a collection of individual locations.

Most large retailers have tremendous investments in both cloud-native and mobile applications, for the benefit of their customers, their associates and their employees. The edge continuum provides them with the opportunity to use the same software development tools for both environments, as well as the same deployment tools for deploying applications to the data center, the cloud, or elsewhere along the continuum to the on-premise User Edge. If retailers fail to leverage the edge continuum this way and continue to manage their on-premise investments as traditional enterprise IT assets, they will deprive themselves of the flexible, responsive and dynamic attributes that their cloud and mobile teams already enjoy.

2.7.5 Gaming

Massively Multiplayer Games (MMPGs) served up by the cloud typically involve players controlling their avatars, with any movement of an avatar needing to be communicated as quickly as possible to all players who have that avatar in their field of view. Latency has a major impact on the overall user experience, to the point where perceptible delays can render a game effectively unplayable. A video game must appear to respond instantaneously to keystrokes and controller movements, implying that any commands issued must complete a round-trip over the network and be processed fast enough by the data center for the player to feel like the game is responding in real time. For the best multiplayer experience, the latency must be consistent across all players, otherwise those with the lowest latency have the opportunity to react faster than their competitors.

Edge compute improves the experience of cloud-enabled gaming by significantly reducing latency and providing the necessary storage and processing power in edge data centers. With processing centers for a game running at the edge of the network, for example in each metro area, the ultra-low latency results in reduced lag-time. This enables a more interactive and fully immersive experience than if the game is hosted in a remote cloud data center.

Edge compute is expected to trigger new subscription-based MMPG business models along with reduced hardware costs for end-users: with edge processing enabling high-quality experiences, less processing power is required in the users' hardware itself. The gaming industry hopes that this reduction in hardware costs will spur greater user investment in new subscriptions, driving overall growth in this segment.

2.7.6 Assisted Driving

While edge compute will be a critical enabler for the holy grail of fully-autonomous driving, that vision is many years away from being realized, for a host of reasons beyond the scope of this paper. Assisted driving technologies, however, are being deployed today and edge compute is key to their viability.

The number of sensors in a vehicle grows with each model year, along with each introduction of new capabilities in safety, performance, efficiency, comfort and infotainment. Although most of the sensor data is processed in the vehicle itself due to autonomy and the safety considerations of latency critical applications, some capabilities, such as alerting in the event of deviations from the norm, require data to be moved to the cloud for analysis and follow-up. Edge computing helps to limit the amount of data that is sent to the cloud, reducing the data transmission cost and minimizing the amount of sensitive data such as Personally Identifiable Information (PII) leaving the vehicle.
The infotainment system in a vehicle is the most prominent user interface besides the driving controls. To learn what functions and applications users are really using and where the design of interactions should be optimized, ML algorithms represent an important tool for uncovering relevant insights within the vast amount of available data. Edge compute brings ML models, which were trained in the cloud, to the vehicle itself, so that the available behavioral and sensor data can be used locally for predictions that improve overall user interaction.

Efficient battery monitoring and predictive maintenance are key to the long-term customer experience for vehicle owners and operators. Edge compute addresses these challenges through the ability to aggregate data and perform the real-time evaluation of relevant battery parameters and sensor values. Appropriate information can be automatically uploaded to backend operational systems in the cloud, enabling dealers or fleet operators to automatically schedule preventative maintenance at a time and place that balances convenience for the user against the severity of problems that have been detected.
Edge compute technologies can enable secure, frictionless entry to a vehicle based on multi-factor authentication, for example using a camera for face recognition, an infrared camera for spoofing detection and a Bluetooth sensor to detect the proximity of the driver's smartphone.

Finally, once the proportion of smart vehicles reaches a critical threshold within a certain geography, smart traffic management will become feasible, enabled by roadside edge compute. In one example, if a road intersection has an edge node deployed to which the majority of vehicles can communicate while coming towards the intersection, the edge node can aggregate the location and speed data from nearby vehicles, optimize traffic light timing for efficient traffic flow and notify the smart vehicles in advance about the situation at the intersection. Widespread deployments of such edge nodes will enable Cellular Vehicle-to-Everything (C-V2X) applications that optimize traffic flows not only for individual intersections, but over wider areas thanks to the cloud-based analysis of edge data and the centralized orchestration of the individual intersections.

2.7.7 Summary of the Edge Continuum

Each edge tier represents unique tradeoffs between scalability, reliability, latency,

cost, security and autonomy. In general, compute at the User Edge reflects dedicated, operated resources on a wired or wireless local area network (LAN) relative to the users and processes they serve. Meanwhile, the Service Provider Edge and Public Cloud generally represent shared resources (XaaS) on a wide area network relative to users and processes.

In many applications, User Edge workloads will run in concert with Service Provider Edge workloads. Workloads on the User Edge will be optimized for latency criticality, bandwidth savings, autonomy, safety, security and privacy, whereas workloads on the Service Provide Edge will be optimized for scale. For example, an AI/ML model might be trained in a centralized cloud data center or on the Service Provider Edge but pushed down to the

Attribute	User Edge			Last Mile Networks	Service Provider Edge		Centralized Cloud Data Centers
	Constrained Device Edge	Smart Device Edge	On-prem Data Center Edge		Access Edge	Regional Edge	
Hardware Class	Constrained microcontroller-based embedded devices (e.g. voice control speakers, thermostats, light switches, sensors, actuators, controllers). KBs to low MBs of available memory.	Arm and x86-based gateways, embedded PCs, hubs, routers, servers, small clusters. >256MB of available memory but still constrained. Accelerators (e.g. GPU, FPGA, TPU) depending on need.	Standard servers and networking with accelerators		Standard servers and networking with accelerators, telco radio infrastructure	Standard servers and networking with accelerators	Standard servers and networking with accelerators
Deployment Locations	Highly distributed in the physical world, embedded in discrete products and systems	Distributed in field, outside of secure data centers (e.g. factory floor, equipment closet, smart home) or embedded within distributed systems (e.g. connected vehicle, wind turbine, streetlight in public R.O.W.)	Secure, on-premise data-centers and micro-data centers (MDCs), e.g. located within an office building or factory. Typically owned and operated by enterprises.		CO, RO, Satellite DCs, owned and operated by service providers (e.g. ISPs, CSPs). Resources can also be located at User Edge in the case of CPE owned and managed by a service provider	CO, RO, Satellite DCs, owned and operated by service providers (e.g. ISPs, CSPs).	Centralized DCs, Zones, Regions owned and operated by CSPs. Compute in DCs located near key network
Global Node Footprint	Trillions	Billions	Millions		Hundreds of Thousands	Tens of Thousands	Hundreds
Role/ Function	Fixed to limited function applications, rely on higher-class-es of compute for advanced processing. Emerging simple ML capability via TinyML.	Hyperlocal general compute for apps and services. Dynamic, SW-defined configuration with limited scalability. Includes IoT Compute Edge (headless systems) and End User devices	Local general compute for applications and services with moderate scalability. Dedicated to a specific enterprise.		Providing last mile access to the internet for users/enterprises. High availability, pubic and private, general and special. Broad scalability. Shared resources for IaaS, PaaS, SaaS, SDN (XaaS).	High availability, pubic and private, general and special. Broad scalability. Shared resources for IaaS, PaaS, SaaS, SDN (XaaS).	Hyperscale or webscale, public, general purpose. Public cloud involved shared resources for IaaS, PaaS, SaaS, SDN (XaaS).
Software Architecture	Embedded software/ firmware, Real-time Operating Systems (RTOS) for time-critical applications.	Bare metal to containerized/ virtualized depending on capability and use case. Linux, Windows and mobile OS'es (e.g. Android, iOS).	Virtualized, containerized and clustered compute. Linux and Windows.		Virtualized, containerized and clustered compute. VNF, CNF, managed services, networking. Linux and Windows.	Virtualized, containerized and clustered compute. VNF, CNF, managed services, networking. Linux and Windows.	Bare metal, VMs, Clusters, Containers, all architectures, all services. Linux and Windows.
Security, M&O	Specialized OTA M&O tools, often custom by device/manu-facturer. May rely on higher-class compute for security.	Require specific security and M&O tools due to resource constraints, unique functionality, accessibility and limited field technical expertise. Often unable to rely on a network firewall.	Evolution of cloud data center security and M&O tools to support distributed Kubernetes clusters. Benefits from physical and network security of purpose-built data centers.		Evolution of cloud data center security and M&O tools to support distributed Kubernetes clusters in regional locations	Evolution of cloud data center security and M&O tools to support distributed Kubernetes clusters in regional locations	Traditional cloud data-center security and M&O tools
Physical Attributes	Highly-specific form factors for every device	Diverse mix of specialized form-factors with unique I/O, industrial ruggedization, regulatory certifications, etc. based on use case	General purpose server-class infrastructure with some ruggedization and regulatory considerations (e.g. for MDCs)		Purpose-built radio infrastructure. General purpose server and networking hardware. Power, thermal, ruggediza-tion and regulatory considerations for localized resources.	General purpose server and networking infrastructure with power, thermal, ruggedization and regulatory considerations for localized resources	General purpose server infrastructure

User Edge for execution. Table 1 summarizes key attributes of each edge.

Table 1: Summary of Edge Attributes.

The boundaries between edge tiers are not rigid. As mentioned previously, the Service Provider Edge can blend into the User Edge when CPE resources are deployed on-premise in order to provide a user with connectivity and compute as a managed service. Meanwhile, the User Edge can also extend to the other side of the last mile network, as in the case of enterprise-owned private cloud data centers. While the edge boundaries are fluid, they are instructive: certain technical and logistical limitations will always dictate where workloads are best run across the continuum based on any given context.

Regardless of the definitions of various edge tiers, the ultimate goal is to provide developers with maximum flexibility, enabling them to extend cloud-native development practices as far down the cloud-to-edge continuum as possible, while recognizing the

practical limitations. The following sections dive deeper into LF Edge and how each project within the umbrella is working to realize this goal.

3 LF Edge Project Portfolio

The Linux Foundation's LF Edge (LFE) was founded in 2019 as an umbrella organization to establish an open, interoperable framework for edge computing independent of hardware, silicon, cloud or operating system. The project offers structured, vendor neutral governance and has the following mission:

- Foster cross-industry collaboration across IoT, Telecom, Enterprise and Cloud ecosystems;
- Enable organizations to accelerate adoption and the pace of innovation for edge computing;
- Deliver value to end users by providing a neutral platform to capture and distribute requirements across the umbrella;
- Seek to facilitate harmonization across edge projects.

As with other LF umbrella projects, LF Edge is a technical meritocracy and has a Technical Advisory Committee (TAC) that helps align project efforts and encourages structured growth and advancement by following the **Project Lifecycle Document (PLD)** process. All new projects enter as Stage 1 "At Large" projects which are projects that the TAC believes are, or have the potential to be, important to the ecosystem of Top-Level Projects, or the edge ecosystem as a whole. The second "Growth Stage" is for projects that are interested in reaching the Impact Stage, and have identified a growth plan for doing so. Finally, the third "Impact Stage" is for projects that have reached their growth goals and are now on a self-sustaining cycle of development, maintenance, and long-term support.

3.1 LF Edge Project Summaries

LFE comprises the following open source projects, explained in more detail in the online resources:

3.1.1 Stage 3: Impact Projects

- Akraino Edge Stack is a software stack that supports high-availability cloud services optimized for edge computing systems and applications. It offers users new levels of flexibility to scale edge cloud services quickly, to maximize the applications and functions supported at the edge and to help ensure the reliability of systems that must be completely functional at all times. Akraino Edge Stack delivers a deployable and fully-functional edge stack for edge use cases including IIoT, telco 5G core, virtual Radio Access Network (vRAN), Universal Customer Premises Equipment (uCPE), Software-Defined Wide Area Networking (SD-WAN) and edge media processing. It creates a framework for defining and standardizing APIs across stacks, via upstream/

downstream collaboration. Akraino Edge Stack is currently composed of multiple blueprint families that include specific blueprints under development. The community tests and validates the blueprints on real hardware labs supported by users and community members.

- EdgeX Foundry is a vendor-neutral, loosely-coupled microservices framework that enables flexible, plug-and-play deployments that leverage a growing ecosystem of available third-party offerings or to include proprietary innovations. At the heart of the project is an interoperability framework hosted within a full hardware- and OS-agnostic reference software platform. The reference platform helps enable the ecosystem of plug-and-play components that unifies the marketplace and accelerates the deployment of IoT solutions. EdgeX Foundry is an open platform for developers to build custom IoT solutions, either by feeding data into it from their own devices and sensors, or consuming and processing data coming out.

3.1.2 Stage 2: Growth Projects

- EVE is an edge computing engine that enables the development, orchestration and security of cloud-native and legacy applications on distributed edge compute nodes. Supporting containers, clusters, VMs and unikernels, it provides a flexible foundation for IoT edge deployments with a choice of hardware, applications and clouds.
- Home Edge is a robust, reliable and intelligent home edge computing open source framework, platform and ecosystem. It provides an interoperable, flexible and scalable edge computing services platform with APIs that can also be used with libraries and runtimes.
- State of the Edge is a vendor-neutral platform for open research on edge computing dedicated to accelerating innovation by publishing free, shareable research and analysis on edge computing. The project publishes the yearly State of the Edge reports, maintains the Open Glossary of Edge Computing and oversees the LF Edge Interactive Landscape.

3.1.3 Stage 1: At Large Projects

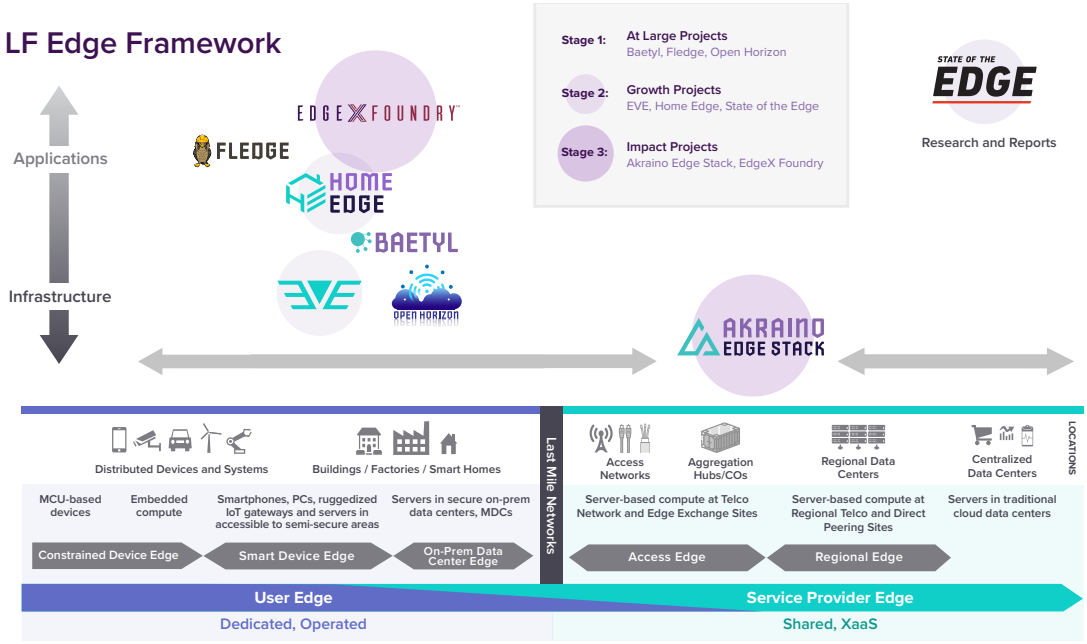
- Baetyl (pronounced “Beetle”) is a general-purpose platform for edge computing that manipulates different types of hardware facilities and device capabilities into a standardized container runtime environment and API, enabling the efficient management of application, service and data flow through a remote console both in the cloud and on-premise.
- Fledge is a proven software framework for the industrial edge focused on critical operations, predictive maintenance, situational awareness and safety. Fledge has been deployed in industrial use cases since early 2018. Fledge is architected to integrate IIoT, sensors, machines, ML/AI tools-processes-workloads and clouds with industrial production process (Level 0), sensing and manipulating (Level 1), monitoring

and supervising (Level 2), manufacturing operations management (Level 3) and business planning logistics (level 4), as per ISA95.

- Open Horizon is a platform for managing the service software lifecycle of containerized workloads and related machine learning assets. It enables management of applications deployed to distributed webscale fleets of edge computing nodes and devices without requiring on-premise administrators.

3.2 Project Focus Across the Edge Continuum

The general focus area for each project along the edge continuum is depicted in Figure 5, though the scope of each project tends to span further across the spectrum as it integrates with various upstream and downstream efforts. This includes extending



up and down the compute continuum and offering varying degrees of application- vs. infrastructure-centric benefits.

Figure 5: LF Edge project framework.

In terms of general project focus, Akraino addresses the unique infrastructure needs of the Service Provider Edge through holistic blueprints, with reach into the various subcategories of the User Edge.

The mission of Project EVE is to create a universal orchestration foundation for enterprise and IIoT edge computing use cases at the Smart Device Edge, like Android has provided for smartphones. EVE addresses the need to accommodate both legacy and modern applications on constrained IoT edge compute resources while meeting the unique security and scale requirements for devices deployed outside of the data center.

Baetyl and Open Horizon are focused on enabling the delivery of containerized workloads to resources distributed across the Smart Device Edge but also have a footprint that extends through the Service Provider Edge to the Cloud. The Open Horizon controller can

be deployed centrally in the cloud, regionally at the Service Provider Edge or locally at the On-Premise Data Center Edge.
EdgeX Foundry and Fledge serve as application frameworks for IoT use cases at the Smart Device Edge to address the fragmentation in the market stemming from diverse technology choices spanning hardware, operating system and connectivity protocols. These frameworks provide an open foundation for deploying analytics and other value-added services, with each taking a slightly different architectural approach that balances tradeoffs between flexibility, portability, footprint and performance. Their efforts bridge to the Constrained Device Edge, facilitate local data processing and in turn relay data to and from higher edge tiers.

Home Edge is focused at the Smart Device Edge for consumer use cases in the home.

The State of the Edge project spans the entire edge computing continuum, conducting research and producing free reports on edge computing and related topics. The project also oversees the Open Glossary of Edge Computing, which seeks to be an industry-wide lexicon for edge computing as well as a tool to align terminology across all LF Edge projects. Finally, the project maintains the LF Edge Interactive Landscape, which is a database-driven taxonomical landscape of edge-related vendors, organizations, projects, standards and technologies.

LFE will add more projects over time with a philosophy of being inclusive but also offering structure and promoting increasing harmonization. Per the project mission, the community aims to develop common best practices and eventual unification of APIs as appropriate. The result will be an open ecosystem for edge computing with infrastructure that can be context-aware of the needs of workloads running above, regardless of who wrote them. As an example, imagine a world where infrastructure could prioritize QoS for a healthcare app running right next to one that delivers entertainment content.

3.3 For more Information on LFE Projects

For more information on LFE projects, refer to their respective websites:

- Akraino Edge Stack
- Baetyl
- EdgeX Foundry
- EVE
- Fledge
- Home Edge
- Open Horizon
- State of the Edge

4 Summary

The concept of edge computing promises exciting new revenue opportunities resulting from the delivery of new types of services to new types of customers, in both consumer and enterprise segments. Compelling use cases include applications such as industrial IoT, computer vision, augmented reality, retail, gaming and assisted driving.

The Linux Foundation (LF) and its subsidiary organization LF Edge (LFE) have initiated a range of open-source software projects that enable companies of all types to collaborate around solutions for developing, deploying and monetizing edge applications and services. Recognizing the compelling business potential that results from extending cloud computing to the edge of the network, hundreds of developers from industry-leading organizations worldwide are participating in these projects that result in edge-optimized solutions for orchestration, management cloud services, frameworks and more.

This white paper has provided an overview of the architectures, use cases and LFE projects associated with edge compute. In-depth technical information is available via the individual projects' websites and interested developers are encouraged to participate in the LFE community.

The Join page on the LFE website provides information on joining LFE, explaining the processes for both existing LF members and non-members. There's also a link to an Inquiry page where interested parties can ask specific questions and obtain additional information.