1. INTRODUCTION

The increase of IoT devices at the edge of the network is producing a massive amount of data to be computed to data centers, pushing network bandwidth requirements to the limit. Despite the improvements of network technology, data centers cannot guarantee acceptable transfer rates and response times, which could be a critical requirement for many applications.[2] Furthermore devices at the edge constantly consume data coming from the cloud, forcing companies to build content delivery networks to decentralize data and service provisioning, leveraging physical proximity to the end user. In a similar way, the aim of Edge Computing is to move the computation away from data centers towards the edge of the network, exploiting smart objects, mobile phones or network gateways to perform tasks and provide services on behalf of the cloud. By moving services to the edge, it is possible to provide content caching, service delivery, storage and IoT management resulting in better response times and transfer rates. At the same time, distributing the logic in different network nodes introduces new issues and challenges. Edge Computing is the process of performing computing tasks physically close to target devices, rather than in the cloud or on the device itself. Over the past decades we've seen different architectural patterns for systems. Depending on the bottleneck of the system it was designed as a centralized or decentralized system. The growing amount of data (IoT) and the limitations of the networking layer (and computation) currently lead to a decentralized system like Edge Computing.

Cloud computing is a general term for anything that involves delivering hosted services over the Internet. These services are broadly divided into three categories: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS). The name cloud computing was inspired by the cloud symbol that's often used to represent the Internet in flowcharts and diagrams. Private cloud services are delivered from a business's data center to internal users. This model offers the versatility and convenience of the cloud, while preserving the management, control and security common to local data centers

In the public cloud model, a third-party cloud service provider delivers the cloud service over the internet. Public cloud services are sold on demand, typically by the minute or hour, though long-term commitments are available for many services. Customers only pay for the CPU cycles, storage or bandwidth they consume.

A hybrid cloud is a combination of public cloud services and an on-premises private cloud, with orchestration and automation between the two. Companies can run mission-critical workloads or sensitive applications on the private cloud and use the public cloud to handle workload bursts or spikes in demand.The goal of a hybrid cloud is to create a unified, automated, scalable environment that takes advantage of all that a public cloud infrastructure can provide, while still maintaining control over mission-critical data.

The pendulum of the IT industry is swinging away from centralised services – such as those in the cloud – and back towards distributed systems, especially for real-time data processing in IoT, sensor array, AI, data triage, and machine-to-machine instant messaging applications. the pendulum of the IT industry is swinging away from centralised services – such as those in the cloud – and back towards distributed systems, especially for real-time data processing in IoT, sensor array, AI, data triage, and machine-to-machine instant messaging applications. These increasing data and real-time analysis requirements have given rise to edge computing, and other distributed approaches. By placing data storage and analysis capabilities at the edge – as close as possible to the sensors, pumps, generators, or whatever hardware is crucial to your operations – users are given several advantages over cloud-based solutions. A combination of both cloud and edge computing suits many organisations. And, as these and other hybrid approaches proliferate, supporting software will become ever smarter, helping decide which workload gets handled where, according to business strategy, regulations, economics, and more.

In the context of IIoT, 'edge' refers to the computing infrastructure that exists close to the sources of data, for example, industrial machines (e.g. wind turbine, magnetic resonance (MR) scanner, undersea blowout preventers), industrial controllers such as SCADA systems, and time series databases aggregating data from a variety of equipment and sensors. These edge computing devices typically reside away from the centralize computing available in the cloud.

2.WHAT IS EDGE COMPUTING

Pushing the frontier of computing applications, data, and services away from centralized nodes to the logical extremes of a network. It enables analytics and data gathering to occur at the source of the data. This approach requires leveraging resources that may not be continuously connected to a network such as laptops, smartphones, tablets and sensors.

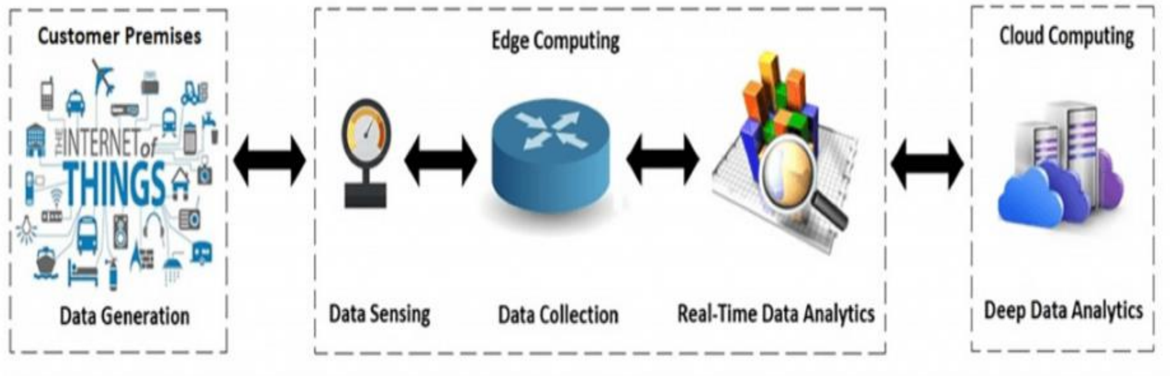
The role of edge computing to date has mostly been used to ingest, store, filter, and send data to cloud systems. We are at a point in time, however, where edge computing systems are packing more compute, storage, and analytic power to consume and act on the data at the machine location. This capability of edge computing will be more than valuable to industrial organizations—it will be indispensable. The name "edge" in edge computing is derived from network diagrams; typically, the edge in a network diagram signifies the point at which traffic enters or exits the network. The edge is also the point at which the underlying protocol for transporting data may change. For example, a smart sensor might use a low-latency protocol like MQTT to transmit data to a message broker located on the network edge, and the broker would use the hypertext transfer protocol (HTTP) to transmit valuable data from the sensor to a remote server over the Internet.

The OpenFog consortium uses the term fog computing to describe edge computing. The word "fog" is meant to convey the idea that the advantages of cloud computing should be brought closer to the data source.

Edge computing can also benefit remote office/branch office (ROBO) environments and organizations that have a geographically dispersed user base. In such a scenario, intermediary micro data centers or high-performance servers can be installed at remote locations to replicate cloud services locally, improving performance and the ability for a device to act upon perishable data in fractions of a second. Depending upon the vendor and technical implementation, the intermediary may be referred to by one of several names including edge gateway, base station, hub, cloudlet or aggregator.

2.1.WHY DO WE NEED EDGE COMPUTING

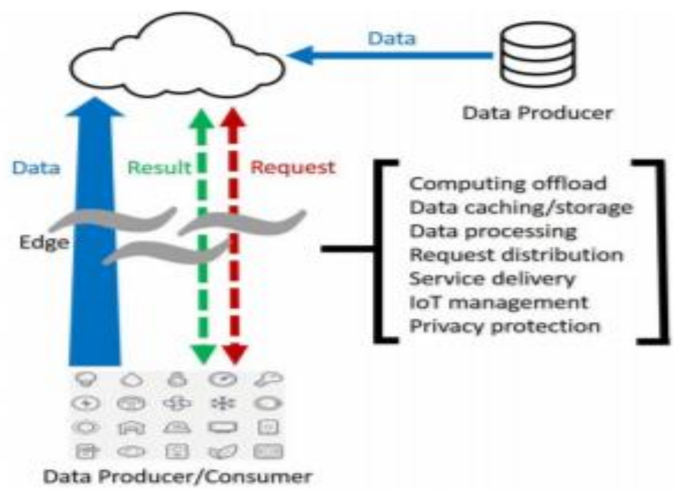
The growth of the wireless industry and new technology implementations over the past two decades has seen a rapid migration from on-premise data centers to cloud servers. However, with the increasing number of Industrial Internet of Things (IIoT) applications and devices, performing computation at either data centers or cloud servers may not be an efficient approach. Cloud computing requires significant bandwidth to move the data from the customer premises to the cloud and back, further increasing latency. With stringent latency requirements for IIoT applications and devices requiring real-time computation, the computing capabilities need to be at the edge—closer to the source of data generation. The word "edge" precisely relates to the geographic distribution of network resources. Edge computation enables the ability to perform data computation close to the data source instead of going through multiple hops and relying on the cloud network to perform computing and relay the data back. Does this mean we don't need the cloud network anymore? No, but it means that instead of data traversing through the cloud, the cloud is now closer to the source generating the data.Edge computing refers to sensing, collecting and analyzing data at the source of data generation, and not necessarily at a centralized computing environment such as a data center. Edge computing uses digital devices, often placed at different locations, to transmit the data in real time or later to a central data repository. Edge computing is the ability to use distributed infrastructure as a shared resource.



NEED OF EDGE COMPUTING

2.1.1. Push From Cloud Services

Putting all the computing tasks on the cloud has been proved to be an efficient way for data processing since the computing power on the cloud outclasses the capability of the things at the edge. However, compared to the fast developing data processing speed, the bandwidth of the network has come to a standstill. With the growing quantity of data generated at the edge, speed of data transportation is becoming the bottleneck for the cloud-based computing paradigm. For example, about 5 Gigabyte data will be generated by a Boeing 787 every second [16], but the bandwidth between the airplane and either satellite or base station on the ground is not large enough for data transmission. Consider an autonomous vehicle as another example. One Gigabyte data will be generated by the car every second and it requires real-time processing for the vehicle to make correct decisions [17]. If all the data needs to be sent to the cloud for processing, the response time would be too long. Not to mention that current network bandwidth and reliability would be challenged for its capability of supporting a large number of vehicles in one area. In this case, the data needs to be processed at the edge for shorter response time, more efficient processing and smaller network pressure.



EDGE COMPUTING PARADIGM

2.1.2. Pull From IoT

Almost all kinds of electrical devices will become part of IoT, and they will play the role of data producers as well as consumers, such as air quality sensors, LED bars, streetlights and even an Internet-connected microwave oven. It is safe to infer that the number of things at the edge of the network will develop to more than billions in a few years. Thus, raw data produced by them will be enormous, making conventional cloud computing not efficient enough to handle all these data. This means most of the data produced by IoT will never be transmitted to the cloud, instead it will be consumed at the edge of the network.

2.1.3. Change From Data Consumer to Producer

In the cloud computing paradigm, the end devices at the edge usually play as data consumer, for example, watching a YouTube video on your smart phone. However, people are also producing data nowadays from their mobile devices. The change from data consumer to data producer/consumer requires more function placement at the edge. For example, it is very normal that people today take photos or do video recording then share the data through a cloud service such as YouTube, Facebook, Twitter, or Instagram. Moreover, every single minute, YouTube users upload 72 h of new video content; Facebook users share nearly 2.5 million pieces of content; Twitter users tweet nearly 300 000 times; Instagram users post nearly 220 000 new photos [18]. However, the image or video clip could be fairly large and it would occupy a lot of bandwidth for uploading. In this case, the video clip should be demised and adjusted to suitable resolution at the edge before uploading to cloud.

2.2. EDGE COMPUTING BENEFITS

Edge Computing Benefits In edge computing we want to put the computing at the proximity of data sources. This have several benefits compared to traditional cloud-based computing paradigm. Here we use several early results from the community to demonstrate the potential benefits. Researchers built a proof-of-concept platform to run face recognition application in , and the response time is reduced from 900 to 169 ms by moving computation from cloud to the edge. Ha et al used cloudlets to offload computing tasks for wearable cognitive assistance, and the result shows that the improvement of response time is between 80 and 200ms. Moreover, the energy consumption could also be reduced by 30%–40% by cloudlet offloading. clonecloud in combine partitioning, migration with merging, and on-demand instantiation of partitioning between mobile and the cloud, and their prototype could reduce 20× running time and energy for tested applications.

Clearly, **speed** is a huge factor in using edge computing, and there are plentiful use cases that solve for speed. Factories can use edge computing to drastically reduce the incidence of on-the-job injuries by detecting human flesh. TSA checkpoints can gather data on chemicals coming through different gates that could be combined to create bombs. Cities can use edge computing to address maintenance of roads and intersections before problems occur.

Another big benefit is **process optimization**. If self-driving cars, factories, and TSA checkpoints were to use the cloud instead of the edge, they'd be pushing all the data they gather up to the cloud. But if the edge makes local decisions, the cloud may not need all that data immediately — or even at all.

With edge computing, data centers can execute rules that are time sensitive (like "stop the car"), and then stream data to the cloud in batches when bandwidth needs aren't as high. The cloud can then take the time to analyze data from the edge, and send back recommended rule changes — like "decelerate slowly when the car senses human activity within 50 feet."

In addition to speed and optimization, **outage reduction** is also a major reason to use edge computing. By pushing everything to the cloud, you're leaving your business open to ISP failures and cloud server downtime. Many mission critical operations like railroads and chemical plants won't even use the cloud today. Their own server farms are the only way to guarantee uptime.

Edge computing relies on the connection between individual sensors and a data center in a local area, which drastically reduces the opportunity for outages. Our partner NXP experienced this firsthand at CES when a power outage brought down the internet for the entire conference — except NXP's computing experience, which was running on the edge.

2.2.1. Reduce the Amount of Data Transmitted and Stored in the Cloud

The amount of data constantly being generated at the edge is growing exponentially faster than the ability for networks to process it. Instead of sending data to the cloud or a remote data center to do the work, endpoints should transmit data to an Edge Computing device that processes or analyzes that data.

Bringing this computing power to the edge of the network helps address the challenge of data build-up, mostly in closed IoT systems. The ultimate goal is to minimize cost and latency, while controlling network bandwidth. A major benefit Edge Computing brings to the table is the reduction of data needing to be transmitted and stored in the cloud. This can typically costs around $4,000 per petabyte for long-term Cloud storage and around ten times that for real-time access storage. Being able to use a technology to reduce these costs is a real benefit for businesses to ultimately help save them money.

2.2.2. Reduce the Amount of Data Transmitted and Stored in the Cloud

The amount of data constantly being generated at the edge is growing exponentially faster than the ability for networks to process it. Instead of sending data to the cloud or a remote data center to do the work, endpoints should transmit data to an Edge Computing device that processes or analyzes that data.

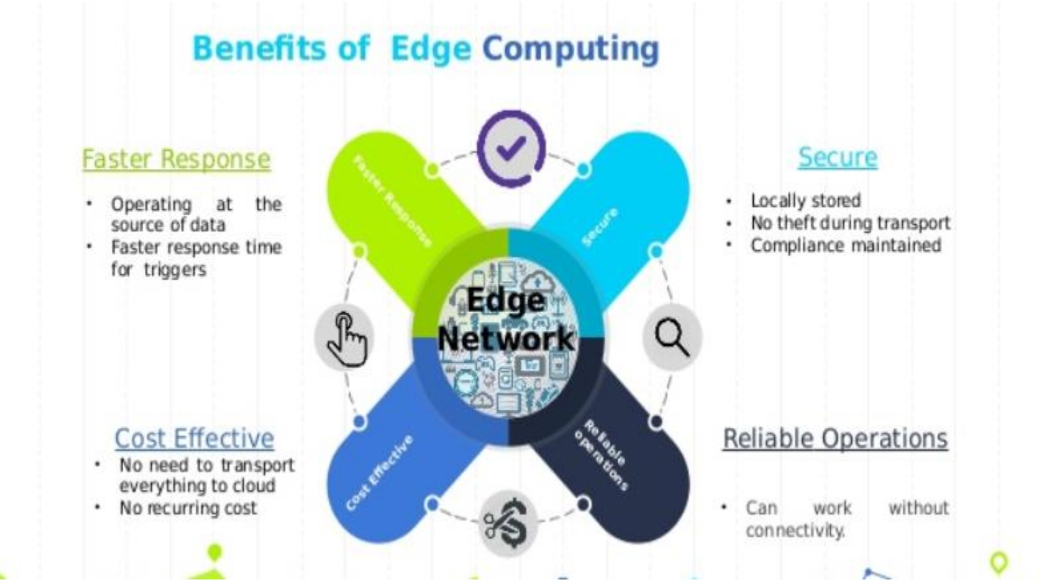
Bringing this computing power to the edge of the network helps address the challenge of data build-up, mostly in closed IoT systems. The ultimate goal is to minimize cost and latency, while controlling network bandwidth. A major benefit Edge Computing brings to the table is the reduction of data needing to be transmitted and stored in the cloud. This can typically costs around $4,000 per petabyte for long-term Cloud storage and around ten times that for real-time access storage. Being able to use a technology to reduce these costs is a real benefit for businesses to ultimately help save them money.

2.2.3. Reduce the Lag Time in Data Transmission/Processing

Edge Computing also reduces the lag that can occur between data transmission, processing and the action required at the end. Analysis and event processing can also be done more quickly and cost effectively because much of the raw data does not need to be streamed up to the Cloud to be processed and analyzed. Cloud data centers can be hundreds – if not thousands – of miles away from a connected device, thus resulting in round-trip latency of tens to hundreds of milliseconds. This kind of latency for robotic surgery, autonomous vehicles and precision manufacturing are a relative lifetime. Edge Computing can reduce the cycle to just a few milliseconds.

2.2.4. Reduce the Signal to Noise Ratio

Lastly, Edge Computing helps reduce the signal to noise ratio to allow companies to prioritize data, such as focus on critical data that needs to be analyzed, stored and processed imminently. Take the monitoring of a commercial refrigeration unit for example. The data collected is machine generated and dominated by "I'm OK" telemetry state data. Every once in a while, the machine will generate an "I'm not OK" event – this is what the monitoring company really cares about. Everything else is superfluous "noise" data that drowns out the signal event. Edge Computing helps prioritize data that needs attention.



EDGE COMPUTING BENEFITS

3.EVOLUTION OF EDGE COMPUTING

How did edge computing begin — and what made its start possible? In just a few decades, the IT world has evolved from the mainframes, to client/servers, to the cloud, and now to edge computing. How do these eras interconnect, what spurred these evolutionary transitions, and where will we go next? The historical perspective outlined below explains how we landed where we are today: outside of the traditional data center, and at the rise of the era of edge computing.

3.1 1960s-1970s: The Mainframe Era

Computers were invented during the era of the mainframe. Large, monolithic systems were a treasured commodity that only sizable organizations could afford to deploy and maintain. Companies conducted all network computing in a physical datacenter. Computing was evolving to meet the rising demands for information access and availability.

Imagine an antiquated world where computers were completely void of font style and size options, graphics — and didn't even include a mouse. Data entry employees relied on "dumb" terminals to capture digital data at the user level. For example, airline reservations during this time were manually entered by a data entry clerk — with one font color option and on a system with very slow processing power. In fact, some airlines still have these legacy systems in place today.

3.2 1980s-1990s: The Client/Server Era

Speed and power were key drivers of the 1980s-1990s. Intel Corporation was formed and revolutionized the IT world with the launch of standard microprocessors. Companies like Dell and Hewlett-Packard were founded, and delivered computing into smaller organizations, and even our homes, with the introduction of small, powerful servers, and desktop and laptop computers.

During this era, Intel founder Gordon Moore observed that transistor densities were doubling every 18 months based on incremental technology improvements. "Moore's Law" findings held true well into the 21st century, and countless IT vendor organizations have relied on the Law for research, development and accurate roadmap planning.

The IT landscape had to accommodate the increasingly mobile processing demands as business boomed in the 1980s. Computing power was by now directly in the hands of users, with the advent of applications like word processing, spreadsheets, and databases. While some compute was conducted locally, organizations still relied on connections to a primary datacenter for large projects and data storage. Envision an engineering or accounting firm with multiple personal computers for design and file creation sending locally to a network server, then connecting out to a larger data center for further processing, backup, and storage or archival.

3.3 2000s-2010s: The Cloud Era

Fast forward to the new century. Y2K is behind us, computers small and large are a staple of every organization, and the Internet is now in every household. Smartphones and tablets enter the market and instantly turn all users into professional photographers and social media members; everyone’s briefcase, backpack, and back pocket is filled with GBs of storage with multiple applications to choose from and manage quickly.

Businesses have by now undergone large economies of scale and are using dozens or hundreds of servers onsite in one location, similar to practices in the mainframe days. With digital data explosion on a significant rise, and regulations now in place holding organizations legally accountable for responsible data use; businesses are looking for a lily pad to migrate the data to while protecting it from the increasing amount of cyber crimes, front-ended by a seamless experience to the IT administrator (or personal user).

The cloud enters as a beautiful, buttoned-up option during this time — users can access applications locally from a computer, phone, or tablet, with all of the processing and storage happening offsite in the cloud. Many organizations around the world have significantly reduced the size of their own on-site datacenter and instead leverage public cloud sites to run applications and store their data. The benefits of the cloud are that CAPEX goes almost to zero and admin time to take care of those systems is cut significantly. However, networking costs and cloud provider fees can really add up fast. Industry behemoths that arose during the Cloud era include Software-as-a-Service companies like Amazon, Netflix, Salesforce, and Spotify and public cloud platforms like Amazon's AWS, Microsoft's Azure, and Google's GCP.

3.4 Today: The Edge Computing Era

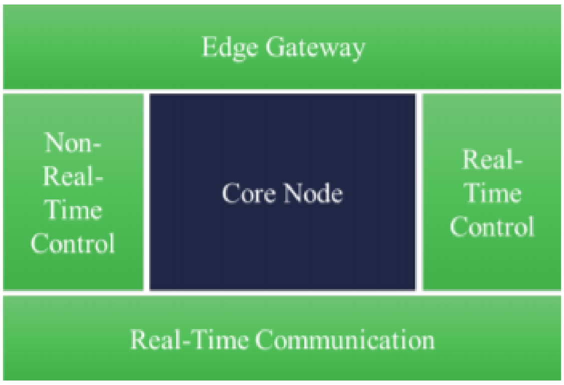
This leads us to where we are today: the edge computing era, where there is a corollary back to the client/server days. With the rise of IoT, the increase in the number of businesses with multiple sites, and the fact that more and more data is being generated outside the data center, computing systems need to deployed at the edge. These systems must be super low-cost and lightweight so they can be deployed in many small locations and can be managed from one central location since IT resources at the edge are sparse.

The natural evolution of IT transitioned from a centralized model during the mainframe era, to a decentralized model during the client/server era, back to centralized with cloud computing, and today, the trend sits steadily with a decentralized computing model with growing edge adoption.

Because edge computing technology is making the complex simple for IT generalists, the adoption of decentralized edge computing will continue to rise into the foreseen future.

4.EDGE COMPUTING ARCHITECTURE

Edge Computing adds an additional tier between the Cloud and IoT devices for computing and communication. The data produced by the devices themselves are not directly sent to the Cloud or back-end infrastructure, but initial computing is performed on this tier. Considering the number of connected devices and the data they produced, this tier is used to aggregate, analyse, and process the data before sending it into the upper layer, the infrastructure.



The proposed Edge Server architecture is to be designed modular and should provide functionalities for real-time and non-real-time control, as well as real-time communication. Core node runs on an operating system and tracks resources and makes decisions on where to execute a task. In the proposed architecture, addition of a new hardware or software modules enable new functionalities and improve the usability of the server. For example, in the case that machine learning algorithms are desired to be executed on the server, connecting a dedicated artificial intelligence (AI) module with dedicated Graphics Processing Unit (GPU) should require none to minimal configuration to be active.

4.1 Function View

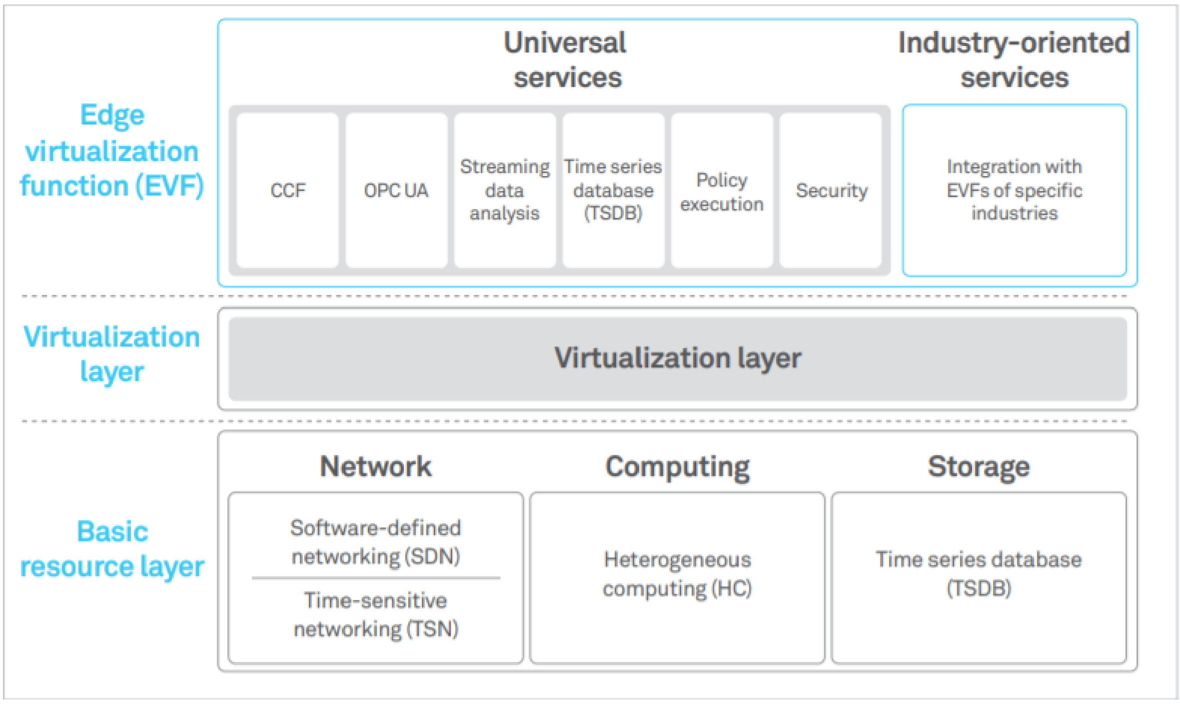
The ECN architecture comprises three main layers:

1) Basic Resource Layer

This layer includes network, computing, and storage modules.

• Network

SDN architectures separate a network's control plane from the forwarding plane to make the network programmable. When SDN is applied to edge computing, the network can support access to millions ofnetwork devices as well as flexible expansion, enabling efficient and low-cost automatic O&M. Additionally, this approach helps achieve network and security policy association and integration.



• Computing

Heterogeneous Computing (HC) is a crucial aspect of the computing hardware architecture at the network edge. Even as Moore's Law continues to hold true for breakthroughs in chip technologies, the popularity of IoT applications has brought explosive growth in information volume, and the application of Artificial Intelligence (AI) has increased computing complexity. These developments place higher requirements on computing capabilities. The types of data to be processed are also becoming more diversified. As a result, edge devices need to process both structured and unstructured data. Therefore, a new computing architecture is proposed that combines compute units that handle different types of instruction sets and have different architectures, that is, heterogeneous computing. Such an architecture gives full play to the advantages of various compute units, achieving a balance between performance, cost, power consumption, and portability.

• Storage

The digital world needs to keep track of the dynamics of the physical world in real time and store complete historical data in chronological order. A new generation of Time Series Database (TSDB) offers efficient storage for time series data (including information such as timestamps of the data). TSDBs need to support basic functions of time series data, such as fast write, persistence, and multi-dimensional aggregated query. To ensure data accuracy and completeness, TSDBs need to continuously add new time series data instead of updating the original data.

2) Virtualization Layer

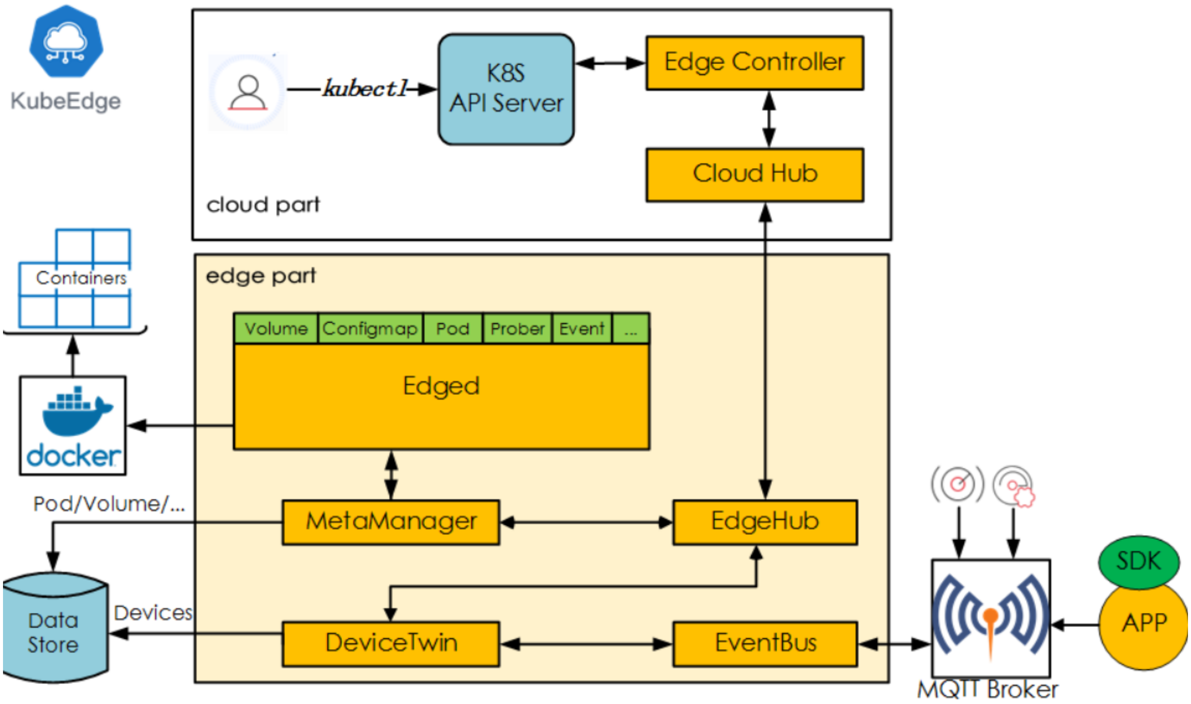
Virtualization technology reduces system development and deployment costs, and has been adopted into embedded system applications from server applications. Typical virtualization technologies include the bare metal architecture and host architecture. In the bare metal architecture, virtualization-layer functions such as the hypervisor run directly on the system hardware platform, and then operating system and virtualization functions run under the hypervisor. In the host architecture, virtualization-layer functions run under the host operating system. The bare metal architecture has better real-time performance and is generally used by smart assets and smart gateways.

3) EVF Layer

Edge Virtualization Functions (EVFs) are software-based and service-based functions that are decoupled from a proprietary hardware platform. Based on virtualization technology, the hardware, system, and specific EVFs can be vertically combined based on services on the same hardware platform. In this manner, multiple independent service zones can be virtualized and isolated from each other. ECNs' service scalability reduces CAPEX and extends a system's lifecycle. EVFs can be flexibly combined and orchestrated, and migrated and expanded on different hardware platforms and devices, enabling dynamic resource scheduling and service agility.

4.2. KubeEdge Architecture: Building Kuberenetes Native Edge computing

The core architecture tenet for KubeEdge is to build interfaces that are consistent with Kubernetes, be it on the cloud side or edge side.



Edged: Manages containerized Applications at the Edge.
EdgeHub: Communication interface module at the Edge. It is a web socket client responsible for interacting with Cloud Service for edge computing.
CloudHub: Communication interface module at the Cloud. A web socket server responsible for watching changes on the cloud side, caching and sending messages to EdgeHub.
EdgeController: Manages the Edge nodes. It is an extended Kubernetes controller which manages edge nodes and pods metadata so that the data can be targeted to a specific edge node.

EventBus: Handles the internal edge communications using MQTT. It is an MQTT client to interact with MQTT servers (mosquitto), offering publish and subscribe capabilities to other components.

DeviceTwin: It is software mirror for devices that handles the device metadata. This module helps in handling device status and syncing the same to cloud. It also provides query interfaces for applications, as it interfaces to a lightweight database (SQLite).

MetaManager: It manages the metadata at the edge node. This is the message processor between edged and edgehub. It is also responsible for storing/retrieving metadata to/from a lightweight database (SQLite).

Even if you want to add more control plane modules based on the architecture refinement and improvement (for example enhanced security), it is simple as it uses consistent registration and modular communication within these modules.

Kubernetes is fast becoming the universal scheduler for scheduling and managing resources that go beyond containers. The control plane of Kubernetes is designed to handle tens of thousands of containers running across hundreds of nodes. This architecture is well-suited to manage scalable, distributed edge deployments. Each edge computing device can be treated as a node while one or more connected devices can be mapped to pods. Developers and operators can use the familiar kubectl tool or Helm charts to push containerized IoT applications that run on one or more edge devices. This approach makes Kubernetes the control plane not just for containers but also for millions of devices managed through an autonomous edge computing layer.

5.IMPORTANT USE CASES

The number of our enterprises who are saying edge is part of their core strategy has doubled in a year. We think by next year about half of enterprises will have edge as a part of their strategy.

The rise of edge computing has helped companies analyze information in near real-time, and create new value around Internet of Things (IoT) devices and data. However, there is no standard formula for implementing edge computing.

5.1 Autonomous vehicles

Self-driving cars need to be able to learn things without having to connect back to the cloud to process data, Bittman said. Machine learning techniques such as reinforcement learning don't rely on training large models with big data sets -- instead, you can run inferences directly in the car, which is essentially edge computing, he added.

"It's not in a cloud, it's not in data centers, it's right in the computer in the car. Engines can learn how to drive themselves without being reliant on connectivity," Bittman said.

"You don't want your autonomous vehicle to be asking the cloud all the time what it's supposed to be doing, or asking a remote server," said Ian Hughes, senior IoT analyst at 451 Research. "It may communicate information to those, it may talk to the infrastructure, it may talk to the other vehicles around it, but most of its processing is done onboard."

5.2. Industrial automation

Edge computing can help create machines that can sense, detect, and learn things without having to be programmed, Bittman said. For example, if sun shining through a window hits a machine for part of the day, the machine will eventually be able to tell that the temperature change doesn't mean that something is wrong.

In industrial processes, machines generally need to be adjusted based on the environment or the quality of materials coming in, Hughes said. "If you are monitoring the process closely and locally, you can extend the life of that machine and extend the operational efficiency of that machine by tweaking what it's doing," he added.

5.3. Augmented reality (AR) and virtual reality (VR)

AR and VR tools that are used for employee training need to understand the environment around them, Hughes said.

"It has to scan the physical status of the world where the services are, and keep an internal model of where you are and what you're looking at," Hughes said. "You can push that up to the cloud, but it's a very localized thing to be able to do that. And so it's a high-end computing piece, done very, very close to the edge."

5.4. Retail

Several retail chains, including Nordstrom, are creating more immersive in-store environments with technologies like AR to attract additional shoppers, Bittman said. This requires lower latency, which is where edge computing capabilities come in, he added.

5.5. Connected homes and offices

Many people use Amazon Alexa or Google Assistant to complete tasks like turning on lights on command, or changing the temperature, Bittman said. However, right now those tasks tend to take a few seconds to occur. With edge computing, it will be possible for them to happen in near real-time.

5.6. Predictive maintenance

Edge computing can help detect machines that are in danger of breaking, and find the right fix before they do, Hughes said. Alerts for what's happening with a machine are best done close to that machine, he added.

5.7. Video monitoring

Video cameras can gather gigabytes of data at a time. Shipping all of that data to a remote processor takes time and money -- especially if you want to use motion detection or facial tracking. But edge computing can handle the sort of detection that would traditionally have to be done on a large-scale computer, Hughes said.

5.8. Software-defined networking

Software-defined networking technologies, some of which will power the move to 5G, require local processing to determine the best route to send data at each point of the journey, Hughes said.

"Each node and network can make a decision about the quality of service it needs to give a particular piece of information that comes to it, and then route that in a different way. It might jump protocols, and it might go from wi-fi to cellular or back again all over."

5.9. Blockchain

Distributed ledger technology like blockchain requires decentralized computing models, Hughes said. "If you're going to do your blockchain, you need to be able to process these ledgers locally, and you need to be able to house them locally," he added. "Each node in a blockchain is a compute unit, so blockchain isn't a centralized ledger, it's a distributed ledger. Therefore it's edge."

5.10. Fog computing

Fog computing is an architecture that uses edge devices to connect to a distributed computing model, Hughes said. Distributed computing systems are able to harness underused cycles across the edge and the continuum to the cloud.

6.CHARACTERISTICS OF EDGE COMPUTING

Edge computing possess various characteristics, some of them are listed below:

• Heterogeneity: Edge Computing is a highly virtualized platform that yields compute, storage, and networking services between end devices and traditional Cloud Computing Data Centers, typically, but not elite located at the edge of network. Compute, storage, and networking resources are the building blocks of both the Cloud and the Fog .

• Edge location: The origins of the Edge can be traced to early proposals to support endpoints with rich services at the edge of the network, including applications with low latency requirements (e.g. gaming, video streaming, augmented reality.

• Geographical distribution: In sharp contrast to the more centralized Cloud, the services and applications targeted by the Fog demand widely distributed deployments. The Edge, will play an active role in delivering high quality streaming to moving vehicles, through proxies along highways and tracks [12].

• Large-scale sensor networks: To monitor the environment and the Smart Grid are other examples of inherently distributed systems, requiring distributed computing and storage resources.

• Very large number of nodes, as a consequence of the wide geo-distribution, as evidenced in sensor networks in general and the Smart Grid in particular.

• Support for mobility. It is essential for many Edge applications to communicate directly with mobile devices, and therefore support mobility techniques, such as the LISP protocol, that decouple host identity from location identity, and require a distributed directory system.
• Real-time interactions. Important Edge applications involve real-time interactions rather than batch processing.

• Interoperability and federation. Seamless support of certain services (streaming is a good example) requires the cooperation of dierent providers. Hence, Edge components must be able to interoperate, and services must be federated across domains.

7.CHALLENGES OF EDGE COMPUTING

There are many problems that will have to be addressed to make the Edge a reality . First we need to identify such problems so that researcher can concentrate on them. Some of open challenges for the fog can be listed as below:

- **Discovery/Sync:** Applications running on devices may require either some agreed, centralized point (e.g. to establish an upstream backup if there are too few peers in our storage application;
- **Compute/Storage limitation:** Current trends are improving this fact with smaller, more energy-efficient and more powerful devices (e.g. one of today‘s phones is more powerful than many high end desktops from 15 years ago). Still new improvements are granted for non-consumer devices;
- Management : Having potentially billions of small devices to be configured, the fog will heavily rely on decentralized (scalable) management mechanisms that are yet to be tested at this unprecedented scale;
- **Security:** The same security concerns that apply to current virtualized environments can be foreseen to affect fog devices hosting applications. The presence of secure sandboxes for the execution of droplet applications poses new interesting challenges: Trust and Privacy. The fog will allow applications to process user‘s data in third party‘s hardware/software. This of course introduces strong concerns about data privacy and its visibility to those third parties;
- **Standardization**: Today no standardized mechanisms are available so each member of the network (terminal, edge point...) can announce its availability to host others software components, and for others to send it their software to be run; 6) Programmability: Controlling application lifecycle is already a challenge in cloud environments. The presence of small functional units (droplets) in more locations (devices) calls for the right abstractions to be in place, so that programmers do not need to deal with these difficult issues .

8.FUTURE OF EDGE COMPUTING

Many industry analysts and technology pundits have been focusing on IoT as the primary driver for edge computing. This makes a lot of sense since billions of things will need to interact with a nearby edge compute resource. But when you consider that a ‘thing’ isn’t just a sensor, but could be a car or a drone or even a mobile phone, it becomes clear why we must think way beyond IoT when it comes to the edge.

With the ensuing data avalanche -- predictions are that the global datasphere is expected to grow 10x to 163 Zettabytes by 2025 -- moving compute and data closer to the user is now a necessity. This, in turn, will lead to many different edge computing "platforms"; the wireless edge, mobile edge, building edge. And yes, the IoT edge as well.

In a world where billions of devices are all collecting and processing data simultaneously, edge computing is a feasible solution that would complement existing cloud computing. Organizations that want to keep pace with future IoT disruption will need to carefully implement and optimize edge computing within their networks.

Edge Computing is a recent term which moves the services from the Cloud to the device as close as possible. It is a borderline between the Cloud and the device tier. Although the Cloud Computing has brought many advantages in the previous years, increased number in the connected devices raised some issues, such as latency and low QoS problems. Edge Computing is believed to solve these issues by analysing the issues and considering the requirements of real world use cases.

There are already several existing proposed architectures in the domain of Edge Computing, such as EdgeX Foundry, Liota, and OpenFog Reference Architecture. Although they are also extensible and they allow inter-connectivity, they do not talk about the real-timeliness of the architectures. This work will be focusing on real-time computing and communication for the given tasks. Of course, it will also be available for non-realtime tasks. The work is being developed by considering the real-world use cases of the industrial partners. The validation will be performed with these use cases and the comparison with the legacy systems will be made. In the future, internal software and hardware components for the Edge Server will be decided. Later, they will be simulated as an initial work for the architecture design. Next, the software components will be individually implemented in the simulation environment. By analysing the simulator results, a hardware benchmarking will be performed and a hardware will be chosen to be used as the Edge Server solution. The final task will be to realize the components by deploying them on the chosen hardware.

9.CONCLUSION

Nowadays, more and more services are pushed from the cloud to the edge of the network because processing data at the edge can ensure shorter response time and better reliability. Moreover, bandwidth could also be saved if a larger portion of data could be handled at the edge rather than uploaded to the cloud. The burgeoning of IoT and the universalized mobile devices changed the role of edge in the computing paradigm from data consumer to data producer/consumer. It would be more efficient to process or massage data at the edge of the network. In this paper, we came up with our understanding of edge computing, with the rationale that computing should happen at the proximity of data sources. Then we list several cases whereby edge computing could flourish from cloud offloading to a smart environment such as home and city. We also introduce collaborative edge, since edge can connect end user and cloud both physically and logically so not only is theconventional cloud computing paradigm still supported, but also it can connect long distance networks together for data sharing and collaboration because of the closeness of data. At last, we put forward the challenges and opportunities that are worth working on, including programmability, naming, data abstraction, service management, privacy and security, as well as optimization metrics. edge computing is here, and we hope this paper will bring this to the attention of the community.

10.BIBLIOGRAPHY

- Shabnam Kumari , Surender Singh and Radha ,"International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)", *Volume 6, Issue 2*.
- Volkan Gezer and Jumyung Um and Martin Ruskowski, "An Extensible Edge Computing Architecture: Definition, Requirements and Enablers", *a Conference Paper on UBICOMM 2017 : The Eleventh International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies.*
- Weisong Shi, Fellow, "IEEE.Edge Computing: Vision and Challenges"-*IEEE INTERNET OF THINGS JOURNAL, VOL. 3, NO. 5.*
- Eric Simone," Why Do We Need Edge Computing?", *https://medium.com.*
- Roopak Parikh and Vamsi Chemitiganti "Edge Computing: Challenges and Opportunities", *https://platform9.com*
- Omkar Dharmadhikari, "Moving Beyond Cloud Computing to Edge Computing", *https://www.cablelabs.com*
- Bruce Kornfeld ,"A Brief History of Edge", *https://dzone.com*
- Alison DeNisco-Rayome, "Ten scenarios where edge computing can bring new value ",*https://www.zdnet.com*