Ref. Ares(2023)4658183 - 05/07/2023

rEaLIzing healthcare 4.0 eXploIting the 6G netwoRk evolution

Marie Skłodowska-Curie Actions (MSCA) & Support to Experts

Doctoral Networks

HORIZON-MSCA-2022-DN-01

101120135 – ELIXIRION

WP2 – Fully-distributed compute continuum for low latency healthcare applications

D2.1: SoA review on the compute continuum and methodological tools

Contractual Date of Delivery:	M12
Actual Date of Delivery:	15/11/2024
Responsible Beneficiary:	BSC
Responsible Person:	Elli Kartsakli
Contact email:	elli.kartsakli@bsc.es
Contributing Beneficiaries:	BSC, NBC, AUMC
Security:	Public
Nature:	R
Version:	V0.6

PROPRIETARY RIGHTS STATEMENT

Document Information	
Version Date:	15/11/2024
Total Number of Pages:	58

List of Authors	
Organization	Authors
NBC	Lazaros Liatsas, Angelos Antonopoulos
BSC	Danial Shafaie, Elli Kartsakli
AMC	Bhavesh Sonwani, Henk A. Marquering

Document History			
Revision	Date	Modification	Contact Person
0.1	10/09/24	TOC Created	Elli Kartsakli
0.2	10/10/24	First round of contributions	Elli Kartsakli
0.3	25/10/24	Second round of contributions	Elli Kartsakli
0.4	29/10/24	Introduction added	Elli Kartsakli
0.5	31/10/24	Revised version to be checked by coordinators	Elli Kartsakli
0.6	014/11/24	Checked by the coordinators	Agapi Mesodiakaki and Amalia Miliou

Table of ContentsF

Lists of Figures

Executive Summary

D2.1 is the first deliverable of the second Work Package (WP2) of the ELIXIRION project named *"Fully-distributed compute continuum for low latency healthcare applications".* The overall aim of WP2 is to bring advanced capabilities to support cloud-native, reliable and explainable applications at the edge fully exploiting the edge-to-cloud compute continuum and leveraging secure big data analytics.

In this context, the work developed within WP2 tackles three main research challenges: i) explore zero-touch multi-domain edge orchestration solutions and service monitoring for low-latency healthcare applications; ii) leverage distributed and parallel computing paradigms, coupled with advanced task-based scheduling solutions, to optimize and orchestrate application workflows across the compute continuum, and iii) develop solutions for secure big data medical applications, focusing on the specific challenges associated with emergency medicine.

This deliverable provides an overview of the state-of-the-art on relevant technologies, existing approaches and challenges on these three specific topics explored by the ELIXIRION Doctoral Candidates within WP2.

ABBREVIATIONS

AI	Artificial Intelligence
AoI	Age Of Information
AR	Augmented Reality
CNCF	Cloud Native Computing Foundation
CNF	Cloud-Native Network Function
CNF	Cloud-Native Network Function
DRL	Deep Reinforcement Learning
E2E	End-To-End
ECG	Electrocardiogram
ED	Emergency Department
EM	Emergency Medicine
EMR	Electronic Medical Records
EMS	Emergency Medical Services
ETSI	European Telecommunications Standards Institute
GA	Genetic Algorithms
GLAD	Graph Layout Scheduling
H4.0	Healthcare 4.0
HER	Electronic Health Records
HIE	Health Information Exchange
HPC	High Performance Computing
ICT	Information And Communication Technology
IIoT	Industrial Internet Of Things
IoMT	Internet Of Medical Things
IoT	Internet Of Things
KPI	Key Performance Indicators
LSTM	Long Short-Term Memory
MDP	Markov Decision Process
MEC	Multi-Access Edge Computing
ML	Machine Learning
NFC	Near-Field Communication
NFV	Network Functions Virtualization
NOMA	Non-Orthogonal Multiple Access
NTN	Non-Terrestrial Networks
OFDMA	Orthogonal Frequency-Division Multiple Access
ONAP	Open Network Automation Platform
PSO	Particle Swarm Optimization
QoE	Quality Of Experience
QoS	Quality Of Service
RFID	Radio-Frequency Identification

SBA	Service-Based Architecture
SDN	Software-Defined Networking
UAV	Unmanned Aerial Vehicle
UE	User Equipment
URLLC	Ultra-Reliable Low Latency Communication
VIM	Virtualized Infrastructure Manager
VM	Virtual Machine
VNF	Virtual Network Function
VR	Virtual Reality
WBAN	Wireless Body Area Networks
WP	Work Package
XAI	Explainable Ai
ZSM	Zero-Touch Network And Service Management

1 Introduction

ELIXIRION aims to set the foundations of the emerging Healthcare 4.0 (H4.0) paradigm by leveraging 6G technologies. The overall project objectives are targeting to: i) enable a wide range of services of different requirements, such as ultra-low latency for latency-critical applications, high speed for data hungry services and ubiquitous secure access to healthcare resources, anytime, anywhere, respecting all privacy aspects, and ii) ensure a secure, efficient, and profitable healthcare ecosystem to all involved stakeholders, while creating a sustainable open market easing access to new players.

The overall vision and organization of ELIXIRION is shown in Figure 1, where the three key research areas of the project, mapped to the three technical work packages (WPs) are depicted:

- WP1 deals with high reliability and high capacity green 6G infrastructures, tackling challenges related to advanced access and connectivity through emerging 6G technologies such as Non-Terrestrial Networks (NTNs), joint access and X-haul and multi-GHz bands.
- WP2 focuses on creating a fully distributed compute continuum from edge to cloud, ensuring ultra-high performance of cloud-native applications, leveraging Artificial Intelligence (AI) for advanced orchestration and relevant big data technologies for secure emergency medicine applications.
- WP3 provides an end-to-end vision on healthcare provisioning, leveraging explainable AI (XAI) and semantic information to improve service management, and introducing mechanisms for secure collaboration of H4.0 stakeholders.

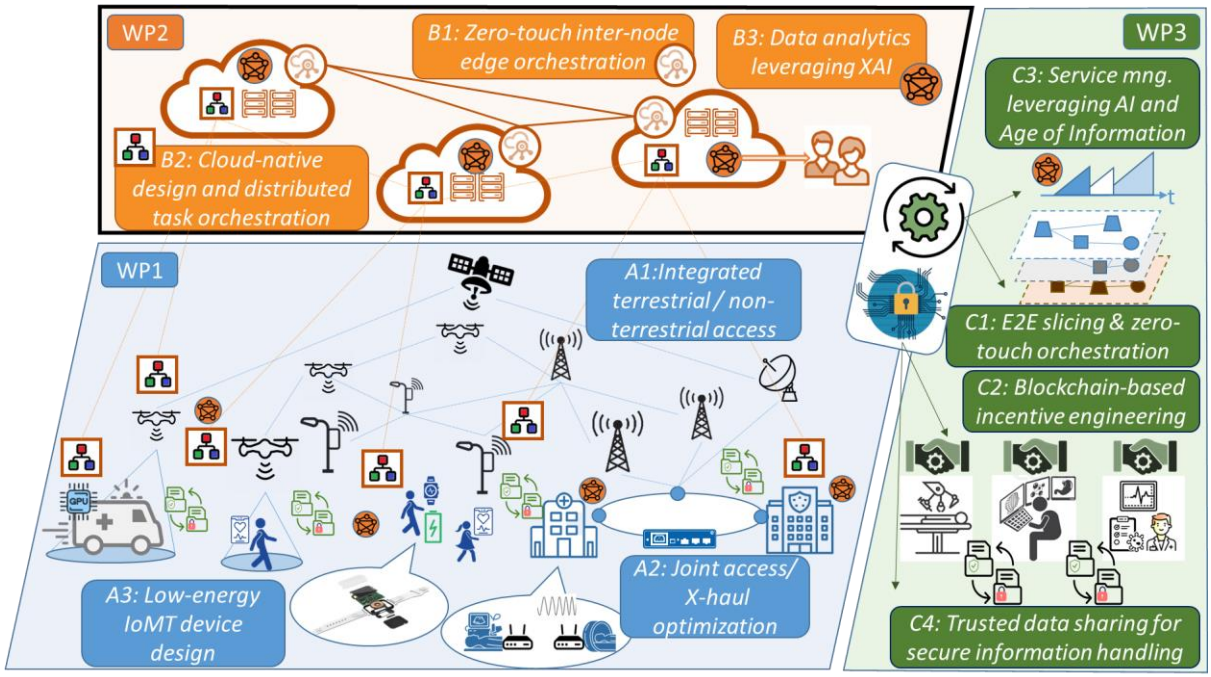

Figure 1. The overall ELIXIRION concept

The overall aim of WP2 is to bring advanced capabilities to support cloud-native, reliable, and explainable applications at the edge, fully exploiting the edge-to-cloud compute continuum and leveraging secure big data analytics. The edge-to-cloud compute continuum consists of a pool of heterogeneous computing resources, ranging from small embedded processors to powerful data centers, and diverse processing architectures

(CPUs, GPUs, etc.) that can be situated anywhere between the end users and the cloud. In this dynamic computing environment, the Internet of Medical Things (IoMT) plays a pivotal role, as big data technologies combined with wireless communications are essential for gathering, analyzing, and interpreting vast amounts of health-related data in real-time. This integration ultimately leads to improved patient outcomes and the development of more personalized care strategies. Focusing on WP2, the following technical limitations have been identified, calling for novel solutions and approaches to be developed within the ELIXIRION project:

- ***Latency & Real-time Service Provision***: Although the 5G Ultra-Reliable Low Latency Communication (URLLC) latency target of 1 ms is adequate to meet even the strictest telehealth demands (e.g., telesurgery/haptic feedback), it has not been realized yet, and it is not expected for the next years to come. However, in such delay intolerant services, even a few ms difference could be life-threatening for the patient. To that end, Multi-Access Edge Computing (MEC) enables the processing of a massive amount of patient data closer to the end devices and the execution of resource-hungry AI/data analytics. However, its integration into existing 5G networks is still on its infancy, esp. in remote areas with low network penetration.
- ***AI explainability***: The massive generated health data can be transformed to useful information/insights for improving the delivery of healthcare. However, the lack of interpretability of such AI-based decisions can greatly compromise the trust of people in healthcare. This calls for novel XAI models that enable people to understand the algorithmic logic and how specific inputs affect the results, leading to trustworthy performance, e.g., through data corrections to avoid bias in training or identifying entry points of adversarial attacks.
- ***Security and Data Transparency***: In H4.0, the involved stakeholders need to securely share sensitive information, e.g., Electronic Health/Medical Records (EHRs/EMRs). Existing distributed ledger solutions for secure data management are only limited to few applications on private networks. AI approaches could be also used to effectively detect security breaches of EHRs/EMRs by analyzing a massive number of events. In parallel, in systems where intensive cryptographical computation may be prohibitive, e.g., in low-complexity IoMT devices, physical layer security solutions could be used leveraging physical channels properties. However, the limited uptake of these methods in 5G, prevents them from reaching their full potential.

The work conducted within WP2 will contribute in overcoming the aforementioned limitations by: i) exploring zero-touch multi-domain edge orchestration solutions and service monitoring for low-latency healthcare applications, leveraging AI-driven and explainable solutions; ii) employing distributed and parallel computing paradigms, coupled with advanced AI-enabled task-based scheduling solutions, to optimize and orchestrate application workflows across the compute continuum, and iii) developing solutions for secure big data medical applications, focusing on the specific challenges associated with Emergency Medicine (EM). This deliverable aims to set the scene, describing the main challenges, existing frameworks and relevant state-of-the art works in the literature.

To that end, the remaining of D2.1 is organized as follows.

- Section 2 first gives an overview of the edge-cloud compute continuum and relevant enabled applications, and then identifies key challenges in the domains of orchestration and emergency medicine.

- Section 3 focuses on explainable zero-touch orchestration of cloud-native services, providing an overview of relevant standardization activities, existing tools and frameworks, as well as relevant contributions in the context of explainable AI.
- Section 4 discusses the deployment and management of cloud-native services across the compute continuum based on AI-driven solutions and specifically graph neural networks. Furthermore, the challenges imposed by applications with increased computational requirements are also discussed.
- Section 5 tackles the challenges in big healthcare data and EM, explores the use of blockchain technology for potentially mitigating the challenges associated with big data security and identifies key tools and frameworks for securely and timely performing analytics on the big healthcare data at the edge.

2 The ELIXIRION compute continuum ecosystem

2.1 Overview of the edge-cloud compute continuum

Cloud computing revolutionized computer services by addressing challenges such as scalability, flexibility, and cost efficiency. It shifted storage, processing power, and applications from local servers to the internet, enabling users to access these resources remotely. This shift played a key role in realizing the vision of the Internet of Things (IoT) on a large scale, driving its rapid expansion in fields like smart homes, manufacturing, healthcare, and agriculture. IoT refers to a network of interconnected devices that collect, communicate, and exchange data via the internet. Equipped with sensors, these devices generate vast amounts of data, fuelling various automation applications.

In traditional cloud-based architectures, large volumes of IoT data are sent to the cloud for processing, analysis, and decision-making. However, this can place a significant burden on the cloud, lead to network congestion, and introduce unacceptable levels of latency, ultimately reducing the effectiveness of cloud computing. The growing volume of data from IoT devices has highlighted the need for more efficient resource allocation and better communication between devices and servers. To address these issues, edge computing was introduced, bringing processing power and storage closer to the source of data—at the network's edge, as depicted in Figure 2.

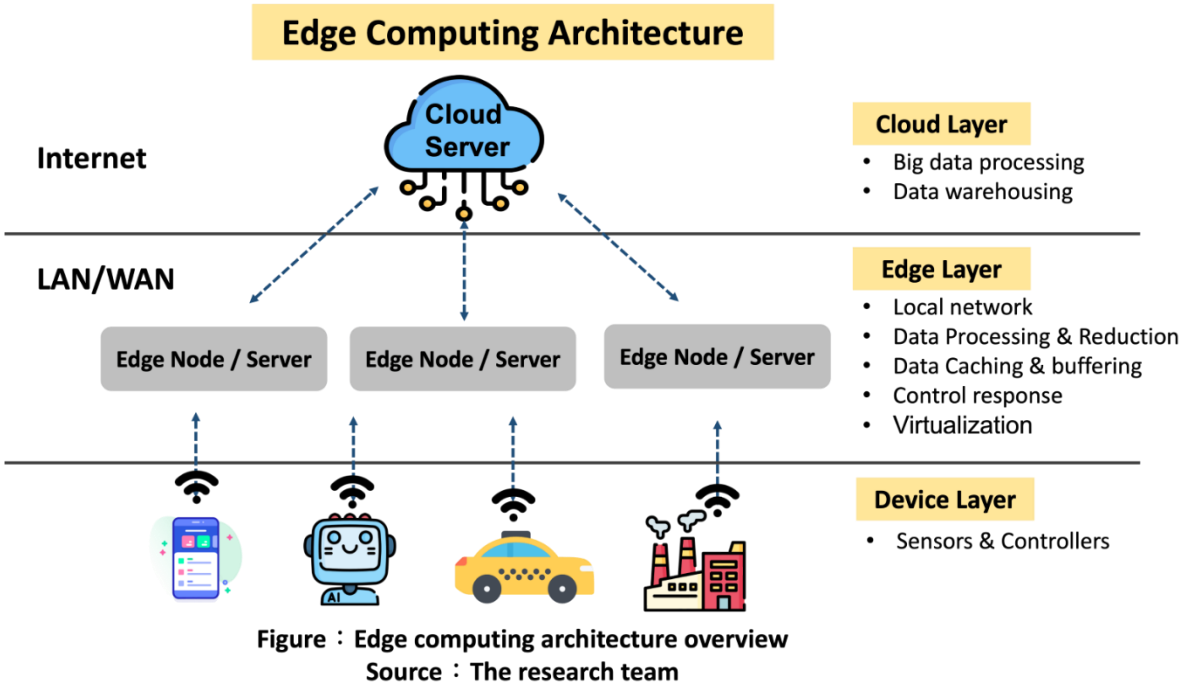

Figure 2. Edge computing overall architecture[1]

As a result, the current trend is shifting from purely centralized, cloud-based architectures to distributed architectures that integrate ***edge computing***. Cloud computing alone cannot handle the immense amounts of data generated by the billions of IoT devices, nor can it meet the needs of latency-sensitive applications.

[1] Source : *"What Is Edge Computing? 8 Examples and Architecture You Should Know"*: https://www.fsp-group.com/en/knowledge-app-42.html

However, a fully edge-dependent approach is also not feasible for most systems. Instead, a hybrid model is emerging, combining both edge and cloud computing. This allows systems to leverage the cloud for resource-intensive tasks while using edge computing for lightweight processing and real-time, localized insights.

2.1.1 Telco edge cloud

In the telecommunications industry, efforts are underway to develop a global platform that can expose, manage, and market edge computing, network resources, and capabilities across various mobile network operators and international borders. This initiative aims to leverage both existing and future network assets. As illustrated in Figure 3, IoT, edge, and cloud infrastructures need to be highly disaggregated into multiple layers of access points, depending on factors such as physical location and latency requirements. Unlike other solutions, telecom operators are focused on delivering a platform that combines optimal computing and networking performance to ensure the highest quality of experience for applications, particularly those with stringent mobility and security requirements.

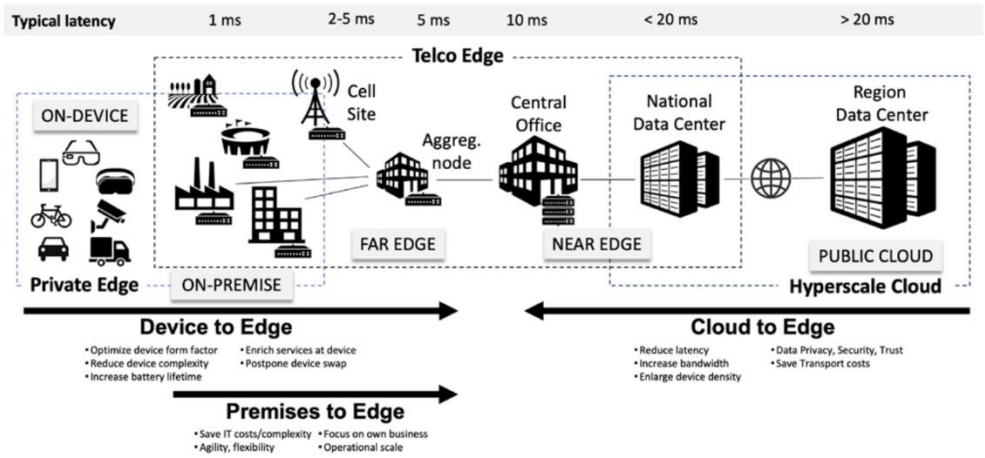

Figure 3. Disaggregated Infrastructure[2]

In recent years, the telecommunications industry has been rapidly evolving, with new trends enabling a shift toward edge-cloud continuum systems. Specifically, the Telco industry has been transitioning from traditional Virtual Machine (VM)-based architectures to cloud-native solutions. This transformation is driven by the need for greater flexibility, scalability, and efficiency in delivering telecom services. It involves adopting containerization, microservices, and orchestration tools to streamline the deployment and management of applications across both cloud and edge environments.

2.2 Emerging edge/cloud-enabled applications

The IoT concept has been applied across a wide range of fields, from industrial and business sectors to everyday applications that impact the daily lives of ordinary citizens. The integration of edge and cloud computing has greatly enhanced the potential of these applications, which are summarized next.

- **IoT in Agriculture:** Agricultural processes, including irrigation, fertilization, and pest control, are increasingly automated through IoT devices that use advanced sensors to collect data on crop health,

[2] Source: "Telco Edge Cloud Value & Achievements": https://www.gsma.com/solutions-and-impact/technologies/networks/wp-content/uploads/2022/03/GSMA-TEC-Value-Whitepaper-v13.pdf

moisture levels, temperature, and more. Robotic systems, such as weeding robots or Unmanned Aerial Vehicles (UAVs) are also used to intervene in these processes. Given the large-scale nature of agricultural applications and the substantial computational requirements for processing sensor data, IoT-edge-cloud computing solutions have proven highly beneficial.

- **Smart Cities:** Modern urban environments generate vast amounts of data from various sources and sensors. This data can be harnessed to improve citizens' quality of life through applications related to traffic management, transportation, and energy consumption, contributing to more sustainable and "greener" cities. Research has highlighted the benefits of edge computing in these applications, particularly in supporting critical low-latency tasks and optimizing resource allocation through the use of edge servers.
- **Smart Manufacturing:** The integration of IoT, commonly referred to as the Industrial Internet of Things (IIoT) in this context, plays a crucial role in the Industry 4.0 revolution. Localized processing capabilities provided by edge computing enhance predictive maintenance, improve quality control, and generate valuable production insights through data analytics.
- **Autonomous Vehicles:** Autonomous driving remains one of the most significant challenges today, attracting research from fields such as image processing, geo-localization, telecommunications, and embedded hardware. Autonomous vehicles are equipped with a range of sensors like cameras and LiDAR scanners and require extensive computational power to perform complex tasks such as object detection, path planning, and 3D mapping, all essential for safe navigation.
- **Healthcare:** Healthcare is a prime example of applications with critical low-latency requirements. IoT has made inroads into healthcare with the advent of e-health applications, such as real-time patient monitoring, personalized treatments based on data analytics, and even remote robotic surgeries. Since Healthcare is the key vertical targeted by ELIXIRION, some additional discussion on relevant medical applications, especially in the domain of emergency medicine will be given.

2.2.1 Emergency medicine

ICT is rapidly transforming the way emergency medicine operates, especially within the realm of Emergency Medical Services (EMS). By integrating advanced technologies into pre-hospital care, EMS providers can enhance patient outcomes, streamline operations, and improve efficiency. ICT offers a wide range of applications within the EMS context. Some key areas include:

- ***Telemedicine and Remote Consultation***: Real-time communication with medical specialists enables remote diagnosis and treatment guidance in rural or remote areas.
- ***Mobile Health (mHealth) Technologies***: Mobile devices equipped with sensors and communication capabilities allow for remote patient monitoring, data collection, and timely intervention.
- ***Electronic Patient Care Report Systems***: Digital patient records facilitate seamless information sharing between EMS providers, hospitals, and other healthcare professionals, improving care coordination and reducing medical errors.
- ***Geographic Information Systems (GIS)***: GIS technology helps optimize resource allocation and response times by mapping incident locations, identifying traffic patterns, and predicting potential hazards.

- ***Wearable Devices***: Wearable health devices can monitor vital signs, detect early warning signs of deterioration, and provide real-time data to EMS providers.

2.3 Technical challenges and opportunities considered in WP2

2.3.1 Challenges relevant to service and resource orchestration

With the rise of MEC, the traditional client-server model of application development has evolved into a three-tiered framework comprising the client, edge (near server), and cloud (far server). This shift presents new challenges for developers, who must distinguish between application components that need low-latency, real-time processing—best suited for deployment at the edge—and those requiring substantial computational resources without real-time constraints, which can be managed in the cloud. To tackle these complexities, developers are increasingly turning to virtualization-driven approaches like microservices and container-based architectures, which are also highly relevant for MEC environments [1]. To propose a 6G cloud-native microservices architecture tailored for MEC-enabled network slicing, considering H4.0, we need to study the inherent challenges of orchestration. Different works try to tackle different challenges of deployment, but summing up from different surveys [2], [3], [4], the challenges can be summed up as followed.

- ***Resource Allocation and Management:*** Allocating resources effectively across heterogeneous cloud and edge environments is challenging due to limited resources at the edge and the need to optimize across cloud and MEC layers
- ***Mobility Management:*** Maintaining seamless service continuity and quality while users move between MEC and cloud environments requires complex mobility management, especially with varying network conditions and user mobility patterns
- ***Latency and Real-time Processing:*** Achieving low-latency responses for applications requiring real-time processing is difficult due to the inherent latency introduced when offloading tasks to the cloud, which may not meet MEC's ultra-low-latency requirements
- ***Network Heterogeneity:*** Integrating various networking technologies (e.g., 6G, Wi-Fi, and fiber) adds complexity to the orchestration, as each technology has different capabilities, latencies, and operational characteristics
- ***Energy Efficiency:*** Efficiently managing energy consumption, especially for edge devices, is critical, given that edge nodes typically have lower energy capacity than centralized cloud servers
- ***Data Privacy and Security:*** Ensuring data security and privacy in distributed MEC environments is challenging due to potential exposure at various network points, including edge nodes and data transmission between edge and cloud layers
- ***Scalability:*** MEC systems need to scale services dynamically as demand fluctuates, which requires efficient orchestration of virtualized resources across edge and cloud infrastructures
- ***Interoperability and Standardization:*** The lack of standardization across vendors and technologies complicates the orchestration process, particularly when integrating new services across multiple platforms

- ***Cost Optimization:*** Balancing computational offloading between cloud and edge to minimize operational costs, while maintaining performance, presents an ongoing orchestration challenge
- ***Quality of Service (QoS) and Quality of Experience (QoE):*** Guaranteeing consistent QoS and QoE for end-users requires sophisticated orchestration mechanisms that can dynamically adapt to changing network conditions and service demands

2.3.2 Challenges relevant to emergency medicine

While the Information and Communication Technology (ICT) sector presents numerous opportunities, its implementation in EMS also faces several challenges:

- ***Interoperability***: Ensuring seamless data exchange between different systems and organizations remains a significant hurdle.
- ***Data Security and Privacy:*** Protecting sensitive patient information is crucial, and robust cybersecurity measures are essential.
- ***Technological Literacy:*** EMS providers need adequate training and education to utilize ICT tools effectively.
- ***Infrastructure and Connectivity:*** Reliable connectivity, especially in remote areas, is essential for optimal ICT performance.

Despite these challenges, the potential benefits of ICT in EMS are substantial. By embracing technological advancements, EMS agencies can:

- ***Improve Patient Outcomes***: Timely and accurate diagnosis, treatment, and transfer can significantly impact patient survival rates and recovery times.
- ***Enhance Operational Efficiency:*** Streamlined communication, data sharing, and resource allocation can optimize response times and reduce costs.
- ***Facilitate Research and Evidence-Based Practice:*** Data collected through ICT can be analyzed to identify trends, evaluate interventions, and inform future practices.
- ***Strengthen Community Engagement:*** ICT can be used to educate the public about emergency preparedness, promote healthy lifestyles, and disseminate vital health information.

ICT has the potential to revolutionize emergency medicine by empowering EMS providers with innovative tools and technologies. We can create a more efficient, effective, and patient-centred EMS system by addressing the challenges and capitalising on the opportunities.

3 Towards explainable zero-touch orchestration

We have already witnessed the transition from traditional physical network functions to virtualized ones and, through the introduction of the microservices architecture, to the containerized network functions. This is a key factor towards enabling large scale automation into managing, deploying, and monitoring networks and services in the cloud-edge continuum. This task has been defined as zero-touch service orchestration and is going to be crucial as we move beyond 5G into 6G networks.

AI and machine learning (ML) techniques can be employed towards enabling such automation and provide, amongst others, dynamic resource allocation, self-healing capabilities and real-time optimization across the continuum. Intelligent orchestration systems decision making consists of actions such as service scaling, service migration and resource allocation between edge and cloud nodes, based on fluctuations in user demand, network and infrastructure conditions.

In ELIXIRION's context, healthcare applications are characterized by strict ultra-low latency requirements and the critical services that require close to zero downtime and reliability. While this highlights the benefits from exploiting AI in service orchestration, it also poses the question of the reliability and trustworthiness of these models, due to their black box nature. Explainable AI (XAI) has been introduced to tackle this limitation of AI models, aiming to shed light on the inner workings of these model providing users and designers with more insights on how they produce their results.

In this section, we start by presenting the main relevant standardization efforts to the task of service orchestration in Section 3.1. Next, in Section 3.2, we present popular tools and frameworks, along with latest research works on AI-driven closed loop orchestration. Finally, in Section 3.3 we conclude by steering towards explainability, a key factor to provide more reliable and trustworthy AI solutions to the task at hand.

3.1 Zero-touch orchestration standardization efforts

The European Telecommunications Standards Institute (ETSI) has initiated three working groups relevant to this section's research area, tasked with creating, developing, and promoting standards that ensure consistency, quality, and compatibility across various products and services. This section will provide a brief overview of these working groups, namely Zero-touch network and Service Management (ZSM), Network Functions Virtualization (NFV), and MEC, along with their key features.

3.1.1 ETSI Zero-touch network and Service Management

The ETSI ZSM working group's main scope is automating the orchestration and management of services in next-generation networks. It aims to address the growing complexity of service management in environments such as 5G, IoT, and cloud-native infrastructures, where traditional manual management methods struggle to meet the performance, scalability, and agility demands. By reducing human intervention and achieving fully automated, end-to-end management of network services across diverse domains, service agility and quality can be increased while operational costs decrease.

Figure 4 illustrates the reference architecture of ETSI ZSM as published in [5].

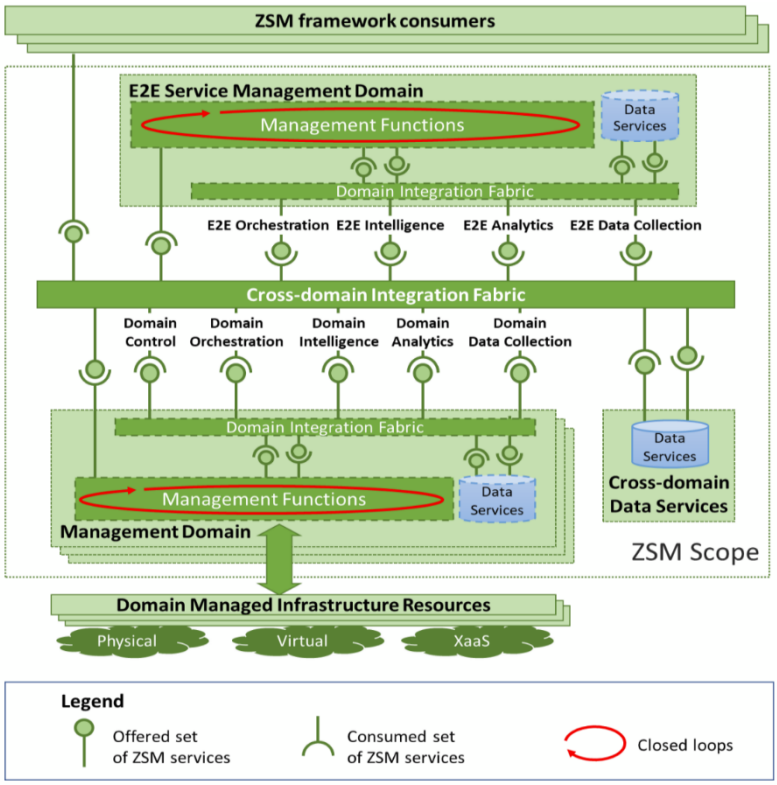

Figure 4. ETSI ZSM reference architecture [1]

To achieve this, ZSM introduces several key features:

1. ***Automation and zero-touch operations**:* Decision making for operations such as configuration, scaling, fault management, performance monitoring and healing is part of a closed-loop processes that utilizes AI and ML techniques to continuously adapt and optimize the system behaviour without human intervention.

2. ***End-to-end service management**:* Services are consistently managed from creation through decommissioning across multiple domains (e.g., access, core, and transport networks). Cross-domain orchestration enables management of services over hybrid infrastructures that may span multiple network operators, cloud providers, and service platforms.

3. ***Modular and extensible architecture**:* Service orchestrators need to be highly modular, to make easy integration of new components, technologies, or vendor solutions possible. The architecture consists of well-defined interfaces between components, ensuring interoperability and extensibility.

4. ***Service-based Architecture (SBA)**:* Every function within the management system is exposed as a service, enhancing flexibility and reusability, as services can be composed, orchestrated, and re-used to fulfil different requirements. This enables a microservices-based deployment model, making it well-suited for cloud-native environments.

3.1.2 ETSI Network Function Virtualization

Introduced by ETSI in 2012, the NFV working group's main scope is to decouple network functions from proprietary hardware, enabling them to run on standard, off-the-shelf hardware through virtualization technologies. This can lead to significant gains, including reduced capital and operational expenses, while also increasing flexibility and scalability of networks and services. NFVs are the crucial building block towards the shift to software-defined, cloud-native, and automated networks.

Figure 5 illustrates the framework architecture of ETSI NFV as published in [6].

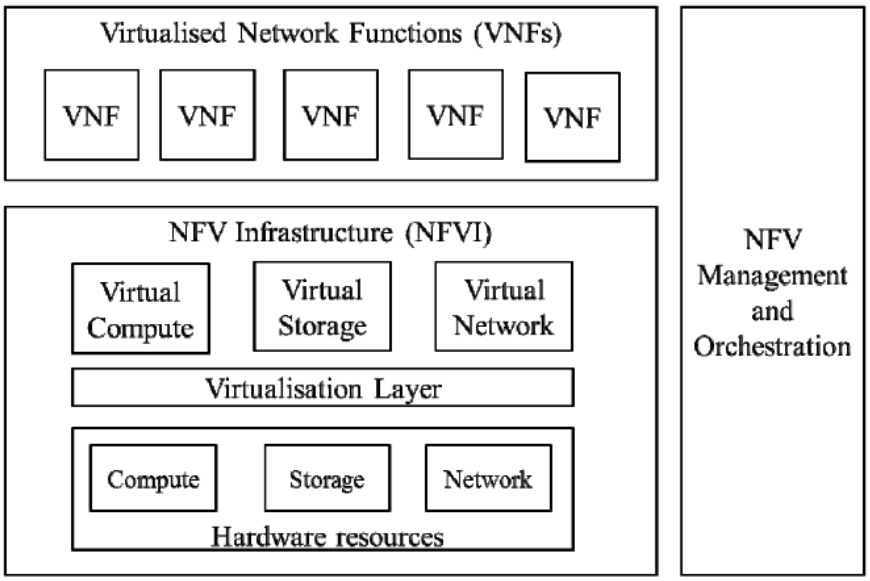

Figure 5. ETSI NFV framework architecture [6]

To achieve this, NFV provides a framework that introduces several key features:

1. ***Decoupling of Network Functions from Hardware**:* Network functions that were once tightly integrated with specific hardware appliances (e.g., firewalls, routers, load balancers) are now virtualized and can run on general-purpose computing hardware (e.g., x86 servers).
2. ***Service Agility and Elasticity**:* NFV allows service providers to rapidly deploy new network services, modify existing services, and scale them based on demand. This service agility is essential for keeping up with the dynamic needs of modern applications, such as those in 5G and IoT, which may experience fluctuating traffic or require the allocation of specific resources to QoS requirements.
3. ***Automation and Orchestration**:* NFV requires advanced orchestration and automation to manage the lifecycle of Virtual Network Functions (VNFs), which can be instantiated, updated, or decommissioned dynamically. To this end, NFV integrates closely with other frameworks, such as software-defined networking (SDN), to ensure efficient traffic routing, scaling, and fault management across virtualized environments.

3.1.3 ETSI Multi-access Edge Computing

ETSI MEC is a working group with the main scope of bringing computation and data storage closer to the end user, specifically at the edge of the network. The MEC framework, developed by ETSI, enables the deployment of applications and services at or near the network edge, reducing latency and improving service efficiency. This can enable ultra-low-latency services, high-bandwidth applications, and real-time processing requirements, which are critical in use cases like autonomous vehicles, IoT, augmented reality (AR), virtual reality (VR), and Industry 4.0.

Figure 6 illustrates the reference framework architecture of ETSI MEC as published in [7].

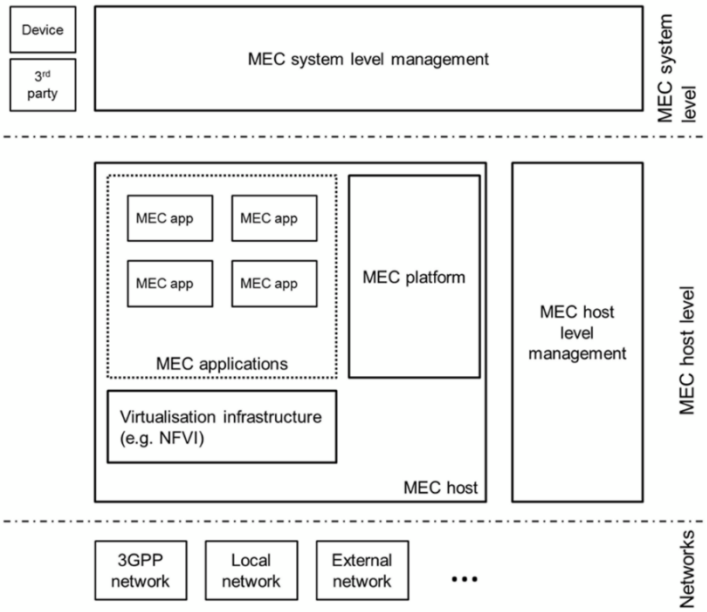

Figure 6. ETSI MEC reference architecture [7]

By offloading critical processing from centralized cloud environments to the edge, MEC can lead to faster response times, reduced network congestion, and greater scalability. ETSI MEC is designed to complement existing cloud and network architectures by extending cloud capabilities to the network edge. The key objectives and features of MEC include:

1. ***Low-latency and Real-time Processing**:* By positioning computing resources at the edge of the network, MEC reduces the distance between end devices and the processing infrastructure. Thus, it drastically reduces processing times and latency, both critical factors for real-time, mission-critical IoT applications.

2. ***Localized Data Processing**:* Data generated at the network edge (e.g., by IoT devices or sensors) can be processed and analyzed locally, without needing to send it back to the cloud. Aside from latency and processing efficiency, this also leads to enhanced privacy and security in the case of sensitive data.

3. ***Integration with 5G and Beyond**:* MEC is designed to integrate with 5G networks and make deployment of edge services within the 5G architecture seamless. By combining it with key 5G features like network slicing, MEC can also be utilized to benefit network management operations.

3.2 Frameworks for zero-touch management and orchestration

This section provides an overview of essential tools and frameworks used in managing and orchestrating virtualized infrastructures, particularly in the context of cloud-native environments. First, it introduces Virtualized Infrastructure Managers (VIMs), such as Kubernetes and OpenStack, which oversee the allocation of compute, storage, and network resources. Following this, it explores network and service orchestrators, including solutions like OSM, ONAP, and NearbyOne, Nearby Computing's edge orchestration solution.

3.2.1 Virtualized Infrastructure Managers

Virtualized Infrastructure Managers are essential components in the orchestration and management of virtualized environments. They are responsible for managing the compute, storage, and network resources within virtualized infrastructure, ensuring that these resources are properly allocated to applications and services.

3.2.1.1 Kubernetes

Kubernetes[3] is a powerful, open-source platform designed to automate the deployment, scaling, and management of containerized applications. Originally developed by Google and now maintained by the Cloud Native Computing Foundation (CNCF), Kubernetes has become the de facto standard for container orchestration in cloud-native architectures. In the context of NFV and MEC, Kubernetes is increasingly being used as a VIM to manage containers that run microservices-based VNFs or other services in the cloud-edge continuum, providing scalability, and efficiency for next-generation telecom services.

Figure 7 illustrates the general Kubernetes architecture.

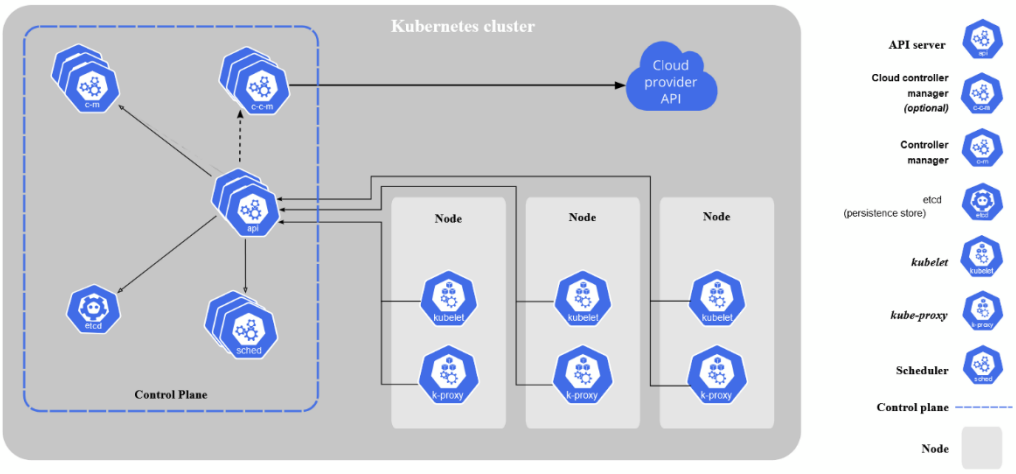

Figure 7. General Kubernetes architecture

The key features of Kubernetes include:

1. ***Service Discovery and Load Balancing***: Kubernetes provides built-in mechanisms for service discovery, allowing microservices and containers to easily communicate with each other. Its built-in

[3] https://kubernetes.io/

load balancer ensures that traffic is evenly distributed across containers, improving performance and reliability in cloud-native environments.

2. ***Scaling and Self-Healing***: Kubernetes enables both manual and automatic scaling of containers by adding or removing container instances based on traffic load or resource utilization. Additionally, Kubernetes has self-healing capabilities, automatically restarting failed containers, rescheduling them to healthy nodes, and maintaining overall service continuity.

3. ***Declarative Configuration and Automation***: Using YAML configuration files, Kubernetes allows infrastructure and applications to be defined in a declarative manner. This simplifies the orchestration and management of services, as changes in the user-defined requirements can be implemented automatically without manual intervention.

3.2.1.2 OpenStack

OpenStack[4] is an open-source cloud platform that is widely used as a VIM for managing large-scale virtualized environments. It provides a set of modular components that handle different aspects of cloud infrastructure, including compute, storage, networking, and identity management. OpenStack is particularly popular in NFV deployments due to its ability to manage large-scale, VM-based VNFs, making it a reliable and scalable option for telecom operators and service providers.

Figure 8 presents an overview of the OpenStack framework.

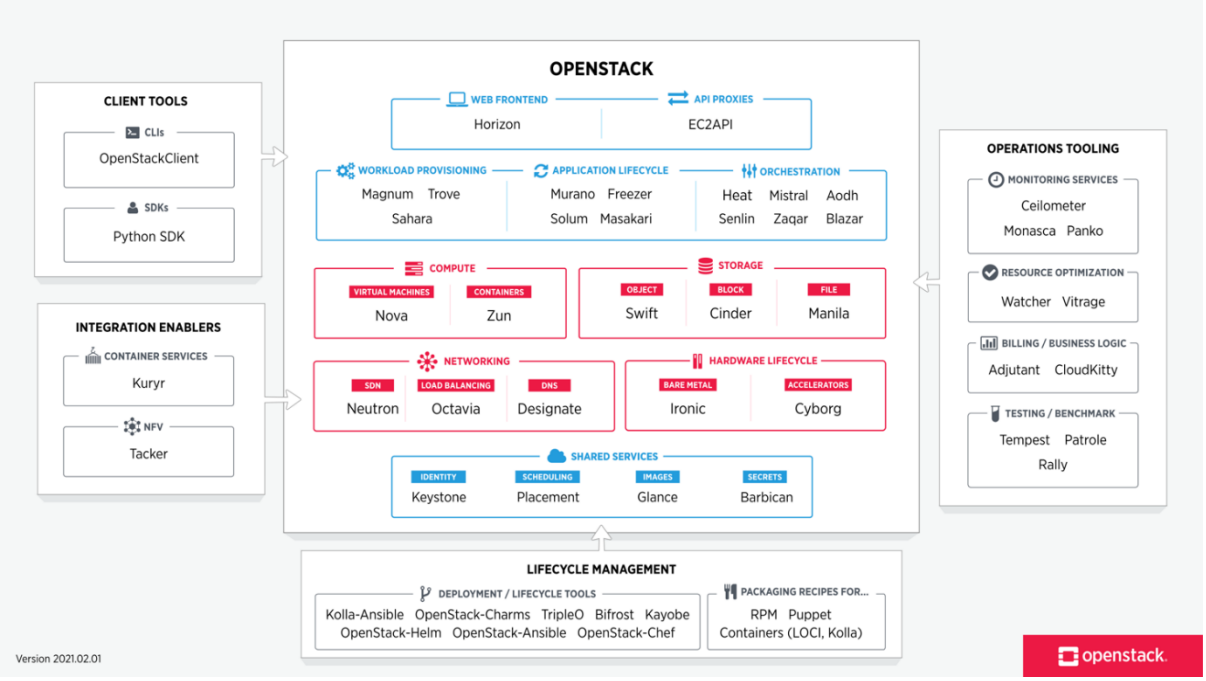

Figure 8. OpenStack framework

The key components of OpenStack include:

1. ***Compute Management (Nova):*** Nova is the compute service responsible for provisioning and managing the lifecycle of VMs and other compute resources like containers and bare-metal servers.

[4] https://www.openstack.org/

Nova allows operators to allocate and manage compute resources based on the needs of applications, enabling scalable deployment of workloads in a cloud environment.

2. ***Networking as a Service (Neutron):*** Neutron provides advanced networking capabilities, enabling users to create and manage networks, subnets, and routers in a cloud environment. Neutron supports various networking technologies, including software-defined networking (SDN) controllers, load balancing, and virtual routers. It also allows for flexible network configurations and QoS management.

3. ***Storage Management (Cinder and Swift):*** OpenStack offers two primary storage solutions: Cinder for block storage and Swift for object storage. Cinder provides persistent storage for VMs, allowing for stateful applications and large data operations. Swift enables scalable object storage, which is ideal for large unstructured data sets like media content or backups. Together, these storage solutions provide flexible, reliable, and scalable storage for a wide range of use cases in cloud environments.

4. ***Orchestration (Heat):*** Heat is the orchestration service that allows for automated deployment and management of cloud resources through templates (Infrastructure as Code). Heat simplifies the process of defining and managing complex infrastructure by enabling users to specify resources such as servers, networks, and storage volumes in a single configuration file. This capability is useful for automating the deployment of cloud applications, ensuring repeatability and consistency.

5. ***Identity and Access Management (Keystone):*** Keystone handles authentication, authorization, and identity management for cloud resources. It supports multi-tenancy and provides secure access control by managing user roles and permissions. Keystone ensures that different projects and users can securely coexist in a shared cloud environment, offering robust security for both private and public cloud deployments.

3.2.2 Network and Service Orchestrators

Network and service orchestration play a critical role in modern telecommunications, enabling automated, efficient, and scalable management of network services and virtualized infrastructures. In this section, we provide an overview of 3 such network and service orchestration solutions.

3.2.2.1 Open-Source MANO (OSM)

OSM[5] is an open-source project under the ETSI aimed at developing a production-grade management and orchestration tack for NFV. OSM provides a framework for end-to-end orchestration of VNFs and network services, aligning with the ETSI NFV MANO architecture to manage the complete lifecycle of network services across diverse VIMs and software-defined networking (SDN) controllers.

The key features of OSM include:

1. ***ETSI NFV MANO Compliance***: OSM is designed to be fully compliant with the ETSI NFV MANO specifications, making it a natural fit for telecom operators and service providers looking to deploy standardized NFV services. It provides key functional blocks, including the NFV Orchestrator, VNF Manager, and SDN controller integration, all within a modular and extensible framework.

[5] https://osm.etsi.org/

2. ***Multi-VIM/SDN Support**:* OSM supports orchestration across multiple VIMs and SDN controllers, making it versatile for different infrastructure environments, whether based on OpenStack, Kubernetes, VMware, or other platforms. This allows operators to manage hybrid cloud and multi-cloud environments efficiently while supporting different types of workloads (e.g., VNFs, CNFs).
3. ***End-to-End Network Service Orchestration**:* OSM offers full lifecycle management of network services, including the deployment, scaling, healing, and termination of VNFs or Cloud-Native Network Functions (CNFs). OSM orchestrates multi-site deployments and enables service chaining enabling complex network services.

3.2.2.2 Open Network Automation Platform (ONAP)

ONAP[6] is an open-source project hosted by the Linux Foundation, designed to provide a comprehensive platform for real-time, policy-driven orchestration and automation of physical and virtual network functions. It is used for managing the lifecycle of complex network services and supports automation across multiple domains.

Figure 9 presents a high-level overview of the ONAP architecture.

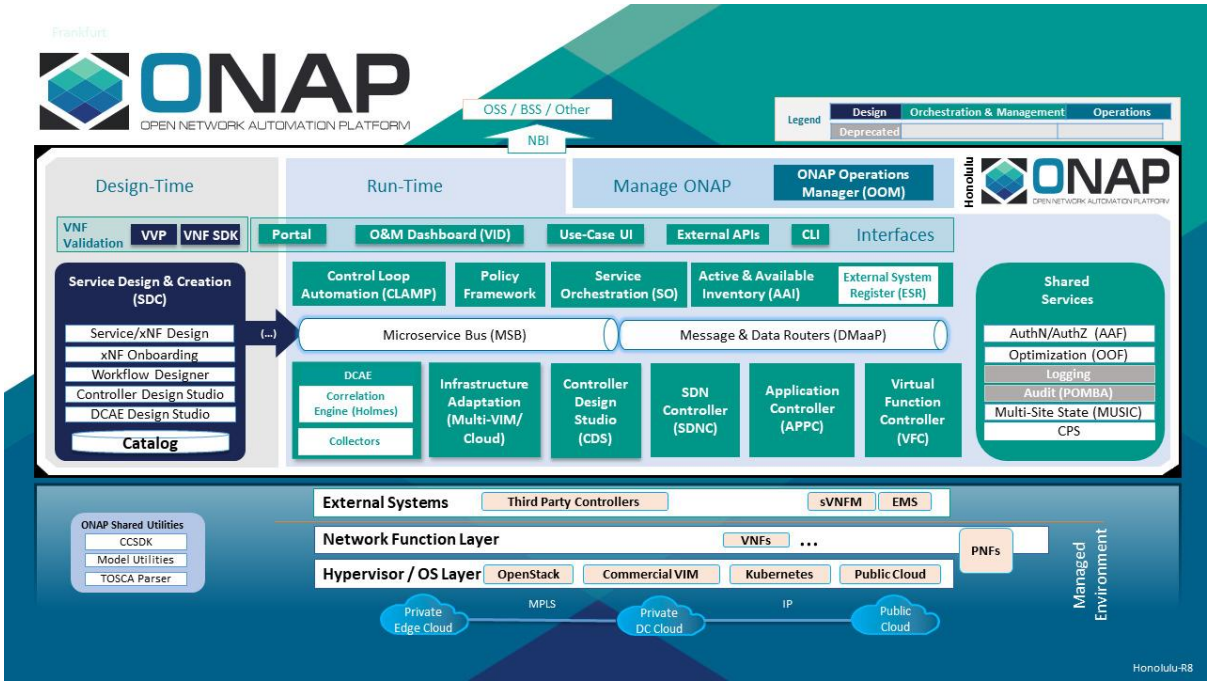

Figure 9. ONAP architecture

The key features of ONAP include:

1. ***Service Design and Creation**:* ONAP provides tools for the design, testing, and validation of network services through its Service Design and Creation (SDC) component. Operators can model services and their underlying components, enabling the rapid design and deployment of complex services, such as 5G network slices or MEC applications.

[6] https://www.onap.org/

2. ***Policy-driven Automation**:* One of ONAP's key strengths is its policy-driven orchestration, which allows network behaviours and responses to be automated based on predefined policies. This ensures that services are automatically adjusted based on real-time network conditions, such as traffic load, resource availability, or failure events, without requiring manual intervention.
3. ***Closed-loop Automation**:* ONAP's closed-loop automation framework enables real-time monitoring of network services, automatically detecting anomalies or performance issues and triggering corrective actions.

3.2.2.3 NearbyOne

NearbyOne[7] is an end-to-end cross-domain orchestration platform that is built to solve the challenges to deploy workloads at the Edge. It covers the full lifecycle management of distributed systems, networks and applications encompassing all layers, from the Edge to the Cloud, seamlessly managing all the areas of operations that take part in edge deployments: the infrastructure, the network, and the applications (see Figure 10).

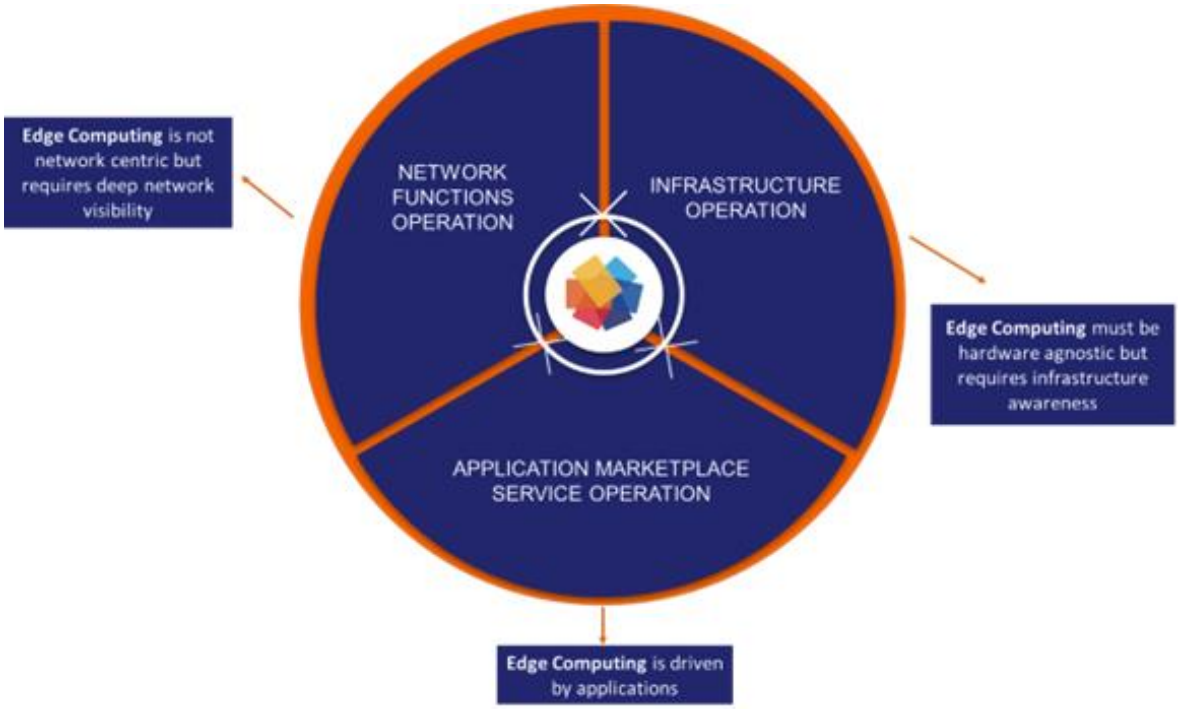

Figure 10. Overview of NearbyOne

1. ***Infrastructure**:* NearbyOne allows to fully manage the lifecycle of the edge nodes as a service. It supports all kinds of infrastructure providers including Edge devices, Hyperscalers, Public, Private and Hybrid clouds.
2. ***xNFs**,* both VNFs and Cloud-Native Network Functions (CNFs): Onboarding of network functions that are needed to enable access to the workloads to be deployed. This includes different access technologies (ranging from IoT to 5G networks), core network components (LTE/5G cores as well as SD-WAN components), and all types of other auxiliary functions, such as firewalls or virtual routers.

[7] https://www.nearbycomputing.com/nearbyone/

3. ***Application and Service Operations***: Applications and services are the central component of any edge deployment, and they require the seamless integration of components provided by third-party software developers or vendors. Application developers are not required to make any changes to their application code for proper integration. The integration points between the applications and the platform (i.e. telemetry collectors or log parsers) need to be defined to provide means to NearbyOne to assess the operation of the applications and to provide reactive actions if needed. All configurations are made using declarative languages (YAML).

NearbyOne is intent-driven: it allows users to onboard new applications and xNFs into the platform through the definition of a number of declarative files that define the behaviour that the orchestration platform must follow to fulfil user-defined requirements.

3.2.3 AI-driven Closed-loop Automation

Closed-loop automation in the context of service orchestration across the cloud-edge continuum refers to the use of automated feedback loops to continuously monitor, analyze, and adjust services or infrastructure in real-time. In order to handle the complexity of the dynamic workloads in MEC environments, researchers have been investigating the use of AI methods towards proactive, self-optimizing procedures to enable real-time service management. This section explores two fundamental components of AI-driven closed-loop automation: workload and resource utilization forecasting and service auto-scaling.

3.2.3.1 Workload and Resource Utilization Forecasting

In AI-driven networks, predictive analytics powered by AI/ML algorithms can be utilized forecast resource demands and optimize the allocation of cloud and network resources. Accurate forecasting enables operators to proactively scale resources, to deal with added latencies related to under-provisioned services, while also avoiding over-provisioning or underutilization. In the paragraphs below, we will follow the analysis in [8], that provides a review on research works in the scope of workload and resource demand prediction.

Long Short-Term Memory (LSTM) models are really popular predictors in such forecasting cases, as in many other fields, because of their ability to handle long sequential data (time-series), BG-LSTM is introduced in [9], which combines the BiLSTM and GridLSTM models, to handle sequence data of CPU and RAM utilizations, as well as task arrival rates, by extracting patterns. The authors in [10] propose the LSTM-ED model, an enhanced version of the traditional LSTM model able to capture dependencies in the host-load data more accurately.

Hybrid forecasting models are also a common approach, where neural networks are combined with the traditional statistical ARIMA model, excessively used in forecasting for their ability to identify trends and seasonal patterns in time-series data. The proposed methods in [11] utilizes ARIMA to detect linear trends and employed triple exponential smoothing to capture nonlinear patterns, [12] introduces the ARIMA-ANN model to forecast CPU and memory utilization across multiple time intervals. This hybrid approach outperforms single models by offering greater accuracy in multi-step predictions, making it highly effective for forecasting in cloud environments.

Advanced AI techniques, such as deep neural networks, encoder-decoder models, and attention mechanisms, are also gaining a lot of attention. The authors in [13] developed an attention-based seq2seq model that predicts CPU and memory usage, splitting prediction sequences into smaller subintervals to manage the volatility of cloud workloads. The Evolutionary Quantum Neural Network (EQNN) is proposed in [14], employing

quantum methods to improve the accuracy of workload and resource utilization predictions in cloud data centres. This approach leverages evolutionary algorithms to optimize the prediction process, demonstrating significant performance improvements over traditional methods.

Multi-predictor models can be a suitable alternative to single predictors, as all the previously mentioned ones, considering generalization capabilities. The COIN model is introduced in [15], which predicts container workloads using transfer learning and online learning. This model adjusts to limited data availability and workload fluctuations by selecting the most appropriate predictors based on past performance, making it highly adaptable in dynamic environments.

3.2.3.2 Service Auto-scaling

Service auto-scaling is defined as the task of dynamically adjusting resources (e.g., CPU, memory, networking) based on real-time demand. Without auto-scaling, resources are statically provisioned, leading to inefficiencies, such as underutilized resources during low-demand periods or resource shortages during peak times. Auto-scaling mitigates these challenges by adjusting resource allocation to maintain service quality while optimizing costs. In this section we will follow the analysis in [16], a state-of-the-art review of auto-scaling techniques in cloud-edge computing environments.

Auto-scaling's main goal is to optimizes resource allocation in real time, targeting to simultaneously prevent performance degradation while reducing costs from overprovisioning. Auto-scaling decisions are being driven by key metrics, such as CPU usage, memory consumption, and HTTP request rates, ensuring that resources scale in response to workload changes. Even small improvements in resource utilization can lead to significant cost savings in pay-per-use cloud environments [17], [18].

Auto-scaling methods typically fall into two major categories: horizontal and vertical. Horizontal methods adjust the number of instances (e.g., containers) to match demand by scaling up (adding instances) or scaling down (removing instances) [19], [20]. Vertical methods, on the other hand, adjusts the resources assigned to each instance by removing or adding resources to the already deployed instance [21], [22]. Horizontal scaling is more effective for handling sudden workload spikes, while vertical scaling is used for finer resource adjustments.

The time of the decisions to scale also dictates another important categorization between auto-scaling methods, dividing them into reactive and proactive methods. Reactive auto-scaling methods adjust system resources based on real-time metrics like CPU usage, memory consumption, request volume, and queue length. When these metrics surpass predefined thresholds, additional resources are either provisioned or deallocated accordingly [24][23], [25]. For instance, if CPU utilization exceeds 70%, a scale-up is initiated, while usage falling below 40% triggers a scale-down. However, setting appropriate threshold values in reactive systems can be complex due to constantly shifting workloads [26]. Another challenge with reactive methods is the delay between the resource request and the time it becomes available, which can take several minutes, affecting system efficiency [27]. Additionally, since resources are only allocated once demand is detected, there is a risk of losing incoming requests during peak periods.

Although most auto-scaling strategies rely on reactive mechanisms, which adjust resources only after workload changes occur, proactive methods offer a more forward-looking approach. Proactive auto-scaling uses historical data and predictive models to anticipate future demand, allowing systems to adjust resources ahead of time. In the context of cloud computing, proactive container-based auto-scaling is primarily implemented using platforms like Docker and Kubernetes. Several attempts have been made to develop

proactive scaling methods for these platforms using machine learning techniques [28][29], [30] and analytical queuing models [31]. By identifying workload patterns in advance, proactive approaches aim to allocate resources pre-emptively, ensuring performance is maintained with minimal response time fluctuations.

3.3 Explainable AI

The transparency of AI model's decision-making decreases as their complexity increases, leading to modern deep learning solution appear as complete black boxes to the users and engineers designing them. However, understanding of the "thought" process of those methods is crucial towards providing trust to their decisions as well as improving their performance and tackling their limitations. In this section, we first provide a brief overview, using a general taxonomy often used in relevant surveys [32], [33]. Finally, we present a summary of works employing XAI techniques in the field of networking and cloud-edge computing.

3.3.1 General taxonomy of XAI methods

A common way to distinguish XAI techniques into categories is based on their scope of explanations, their implementation level, and their model compatibility.

XAI methods can provide insights with either a local, sample-based, scope or on a global level characterizing the AI model in a more general manner.

- ***Globally explainable*** methods summarize the behavior of black-box models across multiple inputs, providing insights into feature attributions for the entire model rather than individual cases. This global explainability is crucial for understanding the model's overall behavior with diverse input distributions and unseen data.
- ***Locally explainable*** methods focus on generating explanations for a classifier's decisions based on individual input data instances by leveraging various data features. Early research in local explanations utilized heatmaps, rule-based approaches, Bayesian methods, and feature importance matrices to assess feature correlations and their significance in output predictions, resulting in positive real-valued matrices or vectors. More recent advancements enhance these techniques through attribution maps, graph-based models, and game-theoretic approaches, providing feature-wise scores that indicate positive or negative correlations with output classifications. A positive attribution score suggests that a feature enhances the output class probability, while a negative score indicates a reduction in probability.

While there are methods designed specifically for certain AI model architectures, there are also XAI techniques for which the underlying model is irrelevant, namely model-agnostic techniques.

- ***Model specific*** interpretability methods are constrained to model classes, with intrinsic methods inherently being model-specific. This limitation means that when specific interpretations are needed, users must rely on models that can provide them, which may sacrifice the use of more predictive and representative models. Consequently, there has been a growing interest in model-agnostic interpretability methods, as they are not tied to any specific model.
- ***Model agnostic*** methods are independent of specific machine learning models, allowing for a separation between prediction and explanation. Typically, post-hoc, these methods are often employed to interpret neural networks and can be either locally or globally interpretable. To enhance

the interpretability of AI models, numerous model-agnostic methods have been developed recently, utilizing a variety of techniques from statistics, machine learning, and data analysis.

Another distinguishing factor between XAI techniques is the implementation level where the explanation is generated during the model's lifecycle. According to that, model can either be classified as post-hoc or ante-hoc.

- ***Post-hoc*** explanation methodologies are particularly valuable, as they enhance the interpretability of existing accurate models. Most post-hoc XAI algorithms are model agnostic, which is a key advantage. For instance, a well-established, pre-trained neural network's decisions can be explained without compromising the model's accuracy.
- On the implementation level, ***ante-hoc*** explainable methods incorporate interpretable elements directly into their design. These models are inherently interpretable through mechanisms like strict axioms, rule-based decisions, and detailed explanations for their outcomes. By definition, intrinsic explanation methods are model-specific, meaning the explanation relies on the model architecture and cannot be reused across different classifiers without tailoring the explanation algorithm to the new architecture.

3.3.2 XAI in network management and the cloud-edge continuum

The fact that AI is expected to have a key role in next-generation networks, it is only natural that research would also steer towards explainability in this area. In this context, explainability is considered to be crucial towards trustworthiness of AI decisions from users and stakeholders, as well as regulatory compliance, especially with recent evolution in the matter of regulation around AI. This has led many recent works to explore the applications and potential of explainability techniques while we transition beyond 5G networks [34], [35], along with possible challenges and limitations.

Recently, there has been a variety of works adopting XAI techniques in the telecommunications domain, proposing explainable solutions in challenging topics, such as network slicing, zero-touch network and service management and short-term resource reservation. More specifically, authors in [36] introduce an explainable reinforcement learning (XRL) approach to improve resource allocation transparency and reliability in 6G network slicing. Reference [37] proposes a neuro-symbolic XAI twin framework for trustworthy zero-touch network management in 6G IoE, integrating implicit and explicit learning to improve automation and interpretability. In the context of short-term resource reservation (STRR) for 5G network slicing, [38] applies Explainable AI (XAI) techniques, enabling human-in-the-loop understanding of AI-driven decisions. The significance of explainability for simplifying and compressing AI models in radio resource management is also highlighted in [39], including a case study on how XAI can reduce the input size for a Deep Reinforcement Learning (DRL) agent. Similarly, in the domain of cloud resource management, [40] advocates for using XAI to interpret and decrease the complexity of a DRL agent responsible for cloud resource allocation.

4 Towards distributed and task-based orchestration

In this section we will be looking into relevant literature for cloud-native design and task orchestration for real-time performance guarantees over the 6G compute continuum. The objective of this section is to bring us to the state of art on the research and works done on the matter, and act as a departure point for novelties carried out.

Edge computing has been a familiar term in telecommunication for a while, but despite the huge hype for the previous generation of cellular networks, the architecture did not manage to deliver the full potential it had. There were different problems keeping the promised performance metrics unattainable. Firstly, the edge computing ecosystem was not mature enough [41], also transplanting the available applications for cloud to the new platform is still a challenge [42], as service providers have been developing applications specific to the cloud ecosystem for many years now.

To tackle these, ELIXIRION proposes a twofold approach: i) first to use the emerging programming models for distributed computation, which should support the inherent heterogeneity and scalability of the architecture, ii) secondly, after setting up the new architecture, novel management mechanisms are also required, aiming to reduce workflow execution time, which in turn, would make the real-time service provision possible.

Therefore, this section is organized as followed. At first, in Section 4.1, we will take into account works that contribute to the deployment of servers and applications on the compute continuum. Works that provide implementations on the edge computing will also be studied and some frameworks that facilitate the transplant into edge ecosystem will be mentioned. After laying down the basic platform, in Section 4.2 we will give insights into the works that cover the management aspect of such systems. Finally, Section 4.3 will take a deeper look into application-related challenges, focusing on works that use edge computing for specific applications, considering the context defined by ELIXIRION.

Also, as both ELIXIRION and the new research works put great emphasis on AI based methods, for all parts, we will look at works that specifically propose AI based solutions to the research challenges relevant to each section.

4.1 Challenges and frameworks for compute continuum deployment

To move away from the bulky processing units conventionally used towards the new design paradigms which make the stringent medical applications feasible, a definition of the compute continuum is required. Although lacking a global consensus on the meaning, it can broadly be defined as a conceptual all-encompassing framework connecting this diverse range of computing environments, spanning from large-scale, centralized cloud data centres to smaller, decentralized edge devices, as well as high-speed High Performance Computing (HPC) systems [43].

Considering the innate heterogeneity and dispersity of such systems, legacy approaches towards deployment of servers and services would not suffice. To unleash the full potential of processing units, novel frameworks should be studied. In the next part we will look into some works that contribute to the challenge. After that some real cases of deployment and frameworks that ease the implementation are mentioned, finishing with AI for edge deployment.

4.1.1 Challenges

The paper *"Joint Edge Server Placement and Service Placement in Mobile-Edge Computing"* [44] presents a study of the joint problem of edge server placement and service placement in MEC. It highlights the importance of considering both tasks jointly for improving the overall profit of MEC platforms. The authors propose a solution that optimizes the placement of both edge servers and services while considering various constraints, such as edge server capacity and service request rates. They develop a two-step method that includes a clustering algorithm and nonlinear programming to maximize the economic benefits of edge servers. The proposed system architecture is presented in Figure 11.

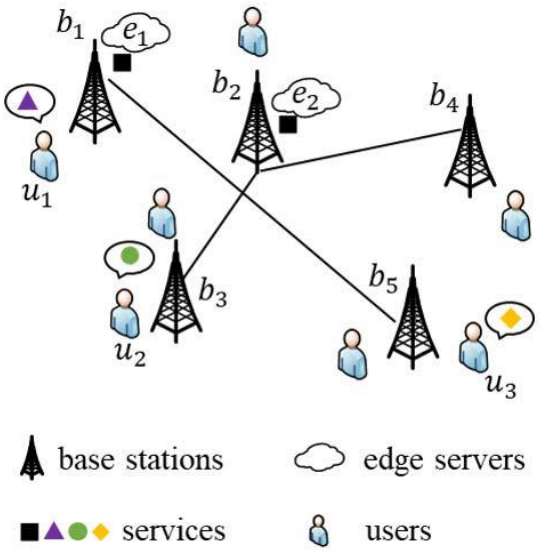

Figure 11. System architecture proposed in [44]

Each edge server is responsible for multiple base stations, and the service requests vary based on user demand. The maximization of the overall profit in the MEC system is formulated as an NP-hard optimization problem. The first step involves using a clustering algorithm to determine optimal edge server locations based on the distribution of base stations. The second step uses nonlinear programming to decide the placement of services on these edge servers while considering constraints such as storage and computing capacity. The experimental setup uses real-world data from the EUA dataset[8] and shows that this work outperforms the other methods in terms of profit, storage efficiency, and service delay.

The paper *"Graph Partition and Multiple Choice-UCB Based Algorithms for Edge Server Placement in MEC Environment"* [45] addresses two challenges in MEC environments, namely interference management between edge servers and the optimal MEC placement. This work lays out a typical system architecture for MEC deployment as shown in Figure 12.

[8] https://github.com/PhuLai/eua-dataset/tree/master

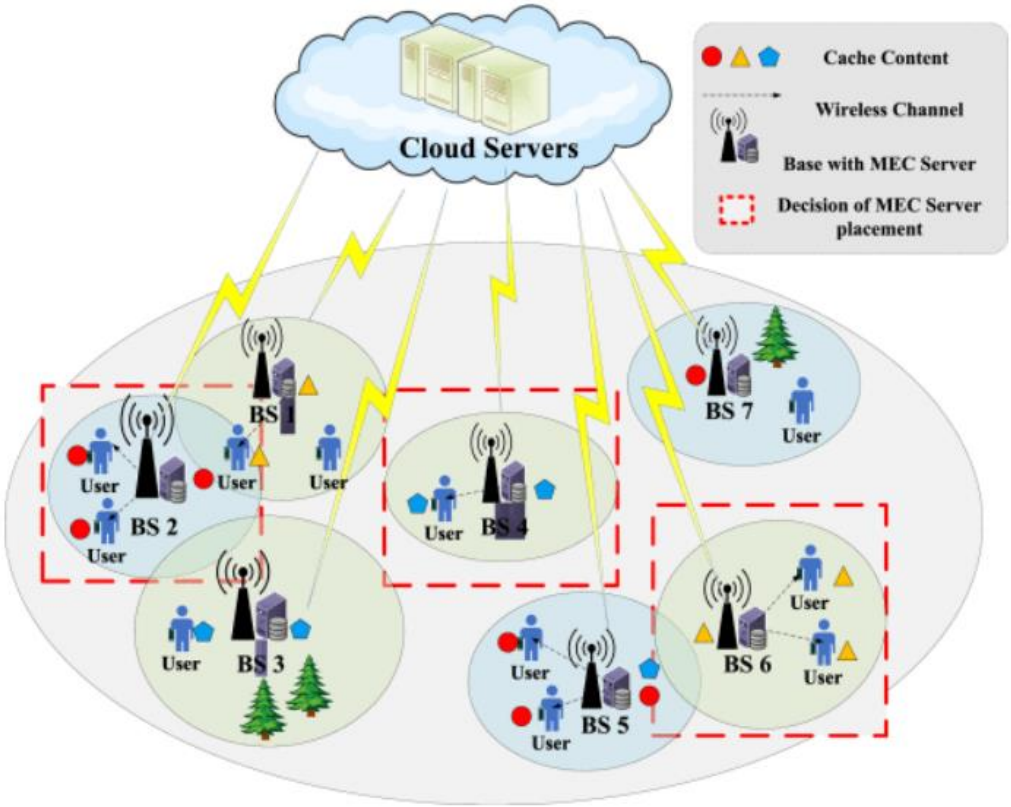

Figure 12. MEC deployment considered in [45]

The two objectives are tackled sequentially in the paper, and as interference management is out of our scope, we will let it out. Authors propose an approach based on Multiple Choice Upper Confidence Bound (MC-UCB) for optimizing MEC server placement. The performance of the proposed algorithm is compared with existing methods like Particle Swarm Optimization (PSO) and Genetic Algorithms (GA), showing notable improvements in QoS metrics. The system is modelled as a multi-user, multi-MEC server scenario. MEC servers have varying hardware configurations and coverage areas, and user devices request services (e.g., video streaming) from nearby MEC servers. The QoS is evaluated based on metrics of Transmission Delay, Throughput and User Density. These metrics are combined to formulate a QoS function, which is used to guide server placement decisions. Once non-interfering subsets of servers are identified, the Multiple Choice-UCB algorithm selects the optimal servers to deploy by balancing exploration (testing less-known options) and exploitation (choosing servers with historically high rewards). The algorithm iteratively improves server placement by minimizing cumulative regret. They test the proposed algorithm in a simulated MEC environment with 50 servers and 1000 users. The results showed better decision-making over time, lowered the transmission delay and guaranteed time thresholds.

4.1.2 Frameworks and implementations

Most works take into account only the mathematical modelling of the compute continuum and leave the more practical concerns open or up to the developers/users. To get closer to a real case of implementation and examination of such systems, in this section we will survey some works that contribute to that direction.

The paper *"On the Continuous Processing of Health Data in Edge-Fog-Cloud Computing by Using Micro/Nanoservice Composition"*[46] focuses on creating an efficient and scalable architecture for processing

large volumes of health data across various computing environments, specifically edge, fog, and cloud. The architecture uses microservices and nanoservices to fulfil non-functional requirements such as security, reliability, and efficiency, which are crucial for healthcare applications. The authors present an architectural model for managing health data in edge-fog-cloud computing environments. This model processes large volumes of data in a modular and infrastructure-agnostic manner using microservices and nanoservices. The overall system definition is presented Figure 13.

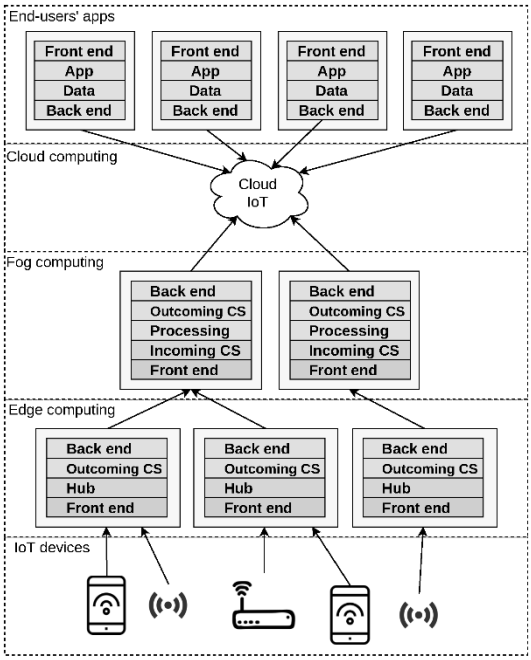

Figure 13. Architecture stack of edge-fog-cloud structures management proposed in [46]

The paper highlights the importance of IoT devices in healthcare, noting that 40% of IoT devices are used for health-related purposes. Following the challenges in processing health data, authors propose a model based on recursive modular blocks. An example of their defined blocks is presented in Figure 14.

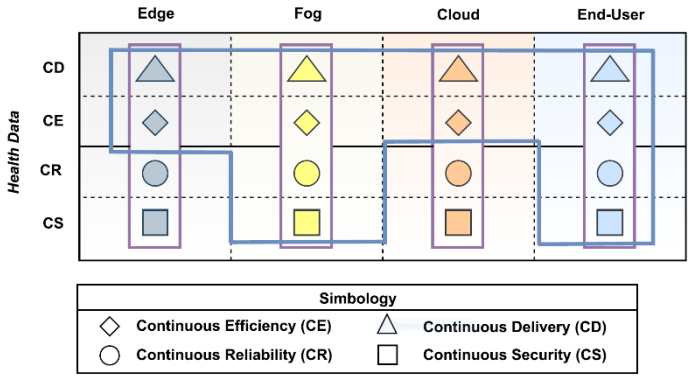

Figure 14. Conceptual representation of a requirements matrix considered in [46]

The authors evaluated the model using a case study involving remote patient monitoring, specifically the processing of electrocardiogram (ECG) data from IoT sensors. The prototype was tested in various configurations, including scenarios where data was processed and shared with internal and external staff. Apart from the model definition and development of the architecture, the prototype evaluation done by the authors

gives good insights to software solutions for task handling in the compute continuum. A table comparing different frameworks is presented in the paper and is shown in Figure 15.

Work	IoT oriented	Reliability			Efficiency		Security		
		Transport	Storage	Processing (fault-tolerance)	Parallelism	Compressing	Integrity	Confidentiality	Access control
Sacbe [86]			✓	✓	✓				
Comps [118]				✓	✓				
PyComps [119]				✓	✓				
Makeflow [120]				✓	✓				
Parsl [121]				✓	✓				
Apache Kafka [122]	✓	✓		✓	✓	✓		✓	✓
Amazon kinesis [123]	✓	✓		✓	✓	✓			✓
DagOnStar [124]	✓			✓	✓				✓
Blocks and Microblocks	✓	✓	✓	✓	✓	✓	✓	✓	✓

Figure 15. Qualitative comparison between different frameworks [46]

As a further explanation, COMPSs (and PyCOMPSs implementation for Python) is a task-based programming model developed and maintained by Barcelona Supercomputing Center (BSC) which aims to ease the development of applications for distributed infrastructures, such as large HPC, clouds and container managed clusters. COMPSs provides a programming interface for the development of the applications and a runtime system that exploits the inherent parallelism of applications at execution time [47]. COMPSs is agnostic of the actual computing infrastructure and provides an abstraction for single memory and storage space, uses standard programming languages of Java with bindings for Python and C/C++ and does not need any API calls.

As an example of real implementation of applications, the thesis "*5G-enabled edge deployments for emerging real-time services*" [48], investigates how 5G technology combined with edge computing can improve latency and performance for real-time applications like autonomous driving, telemedicine, and smart city services. In the work, a practical 5G testbed with edge computing capabilities is deployed to demonstrate substantial latency reductions for delay-sensitive applications. The hypothesis is that hosting applications and processing data at the network's edge can significantly reduce transmission paths and delays, benefiting different areas. Tackling the challenge for the lack of experimental 5G platforms with edge computing capabilities, a testbed is deployed using open-source frameworks (Open5GS and UERANSIM) and hardware components. The overall architecture for the deployed system is presented in Figure 16.

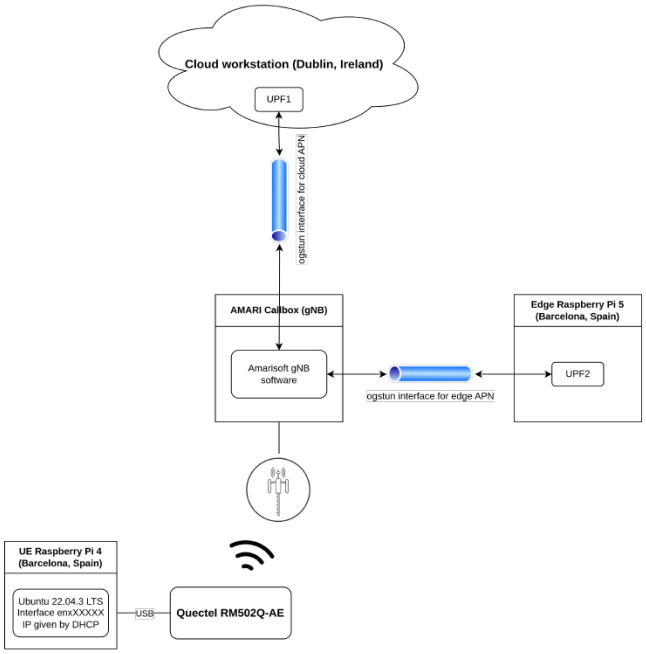

Figure 16. Deployed testbed architecture considered in [48]

Performance metrics such as delay, throughput, and jitter were used to evaluate the system. To showcase the edge computing ecosystem and quantify the benefits of process data closer to the end user, three real-time applications (messaging services, video streaming, and real-time object detection) are implemented and studied. Each of these applications benefited from edge computing by having improved response times, reduced load on core networks, and therefore better performance. An example of an AI application for image segmentation, run on deployed architecture is presented:

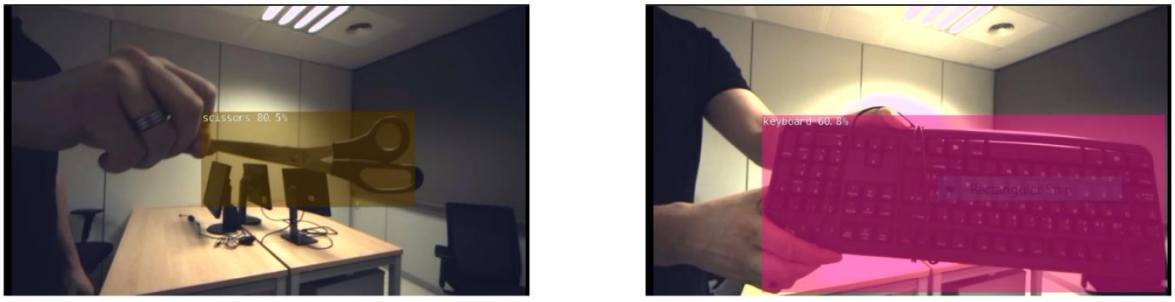

Figure 17. Sample AI tasks detecting a pair of scissors and a keyboard run on the testbed in [48]

4.1.3 AI for edge deployment

The paper *"An Efficient Multi-Edge Server Coalition Computation Offloading Scheme of Sensor-Edge-Cloud"* [49] proposes a multi-mobile edge server collaborative computation offloading scheme to address the high latency and energy consumption challenges in Wireless Body Area Networks (WBANs) for computationally intensive tasks. The scheme leverages multiple edge servers to distribute computing tasks, minimizing system latency and energy consumption. The proposed model is framed as a Markov Decision Process (MDP) and solved using a Deep Q-network developed by the authors, named m4m-PDQN algorithm.

WBANs have become critical in medical care, enabling real-time health monitoring through wearable sensors. However, the energy and bandwidth limitations in WBANs degrade QoS. This paper addresses the inadequacy of single-server offloading systems by proposing a multi-server collaborative approach to optimize energy and latency performance. In the architecture proposed by this study, user equipment (UE) can offload tasks to multiple edge servers, and tasks are split between local execution and offloading. Each user offloads a portion of their computational task to an edge server, while the remaining portion is processed locally.

The objective is to minimize the total cost, comprising both time and energy costs, while ensuring tasks are processed within delay tolerances. The optimization problem is solved using a mixed discrete-continuous action space, and task offloading decisions are learned using the m4m-PDQN algorithm. Details on the Parametrized DQN are not analyzed in this scope and is left for the reader. Model and algorithm interaction are presented in Figure 18. Simulation experiments demonstrate reduced total system latency and energy consumption, and an increased task execution success rate.

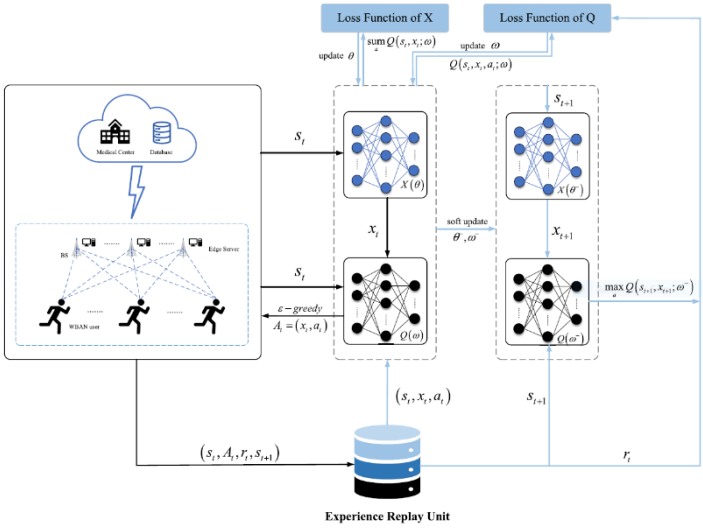

Figure 18. Framework of the m4m-PDQN optimization strategy considered in [49]

4.2 Challenges and frameworks for edge management

To build the full stack for service provision, after setting up the MEC servers and services and coming up with the policies for task offloading, now we must take care of network management. Therefore, in scope of SoA review, we will present some works that contribute to this domain.

4.2.1 Challenges

The paper *"Energy Efficiency and Delay Tradeoff in an MEC-Enabled Mobile IoT Network"* [50] addresses the tradeoff between energy efficiency and service delay in a MEC-enabled IoT network. The authors propose a stochastic optimization problem that jointly optimizes resource allocation and task offloading in multiuser multiserver environments while accounting for user mobility, wireless channel conditions, and task queue stability. The solution is developed using Lyapunov optimization and solved through an online resource allocation algorithm called OORAA.

The dynamicity in mobile environment such as varying wireless channel conditions, unpredictable task arrivals, and mobility that changes user-server associations over time, motivate this work. The system consists of multiple base stations, each equipped with an MEC server that provides computation and radio access services to mobile IoT devices. The system, which is presented in Figure 19, accounts for user mobility, random channel states, and task arrivals.

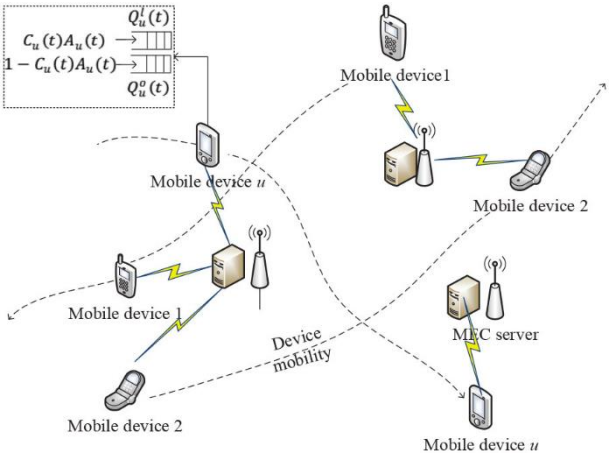

Figure 19. System model considered in [50]

The goal is to minimize the long-term average energy consumption while maintaining queue stability and balancing the tradeoff between energy and delay. The problem is formulated as a stochastic optimization problem with constraints on CPU frequency, transmit power, and maximum queue lengths. The authors then employ a Lyapunov optimization framework to decompose the long-term optimization problem into subproblems for each time slot. These subproblems involve deciding task partitioning, local computation resource allocation, and offloading resource allocation (including bandwidth and power allocation). After that an algorithm named by the authors as OORAA (Online Offloading and Resource Allocation Algorithm) solves the per-time-slot optimization problems, for task partitioning, local computation allocation and offloading resource allocation.

The simulation evaluates the performance of the proposed OORAA algorithm which shows reduced energy consumption compared to baseline methods, managing of the queue lengths, balancing the delay and energy tradeoff, and improves energy efficiency at the cost of increased delay. Comparison with other schemes is also presented.

4.2.2 AI for Edge management

Here, focusing on AI methods, we will name some works that contribute to Edge management and give a short summary on them.

In *"Digital Twin Driven Service Self-Healing with Graph Neural Networks in 6G Edge Networks"* [51] presents a self-healing mechanism for 6G edge networks, designed to ensure network reliability and service continuity in the face of anomalies, caused by heterogeneity and rapidly changing network states. In the architecture, Digital Twin (DT) models monitor and predict network states (performance status and abnormalities), and Graph Neural Network (GNN)-based service redeployment mechanisms adjust service deployments to pre-emptively address issues such as network congestion or failure.

The system, presented in Figure 20, is modelled as an undirected graph, where network nodes and links are represented, and each node has associated resources (e.g., computing and memory). The network supports distributed services, which are decomposed into subtasks that must be processed at different nodes. The model focuses on minimizing end-to-end (E2E) delay and balancing load across the network.

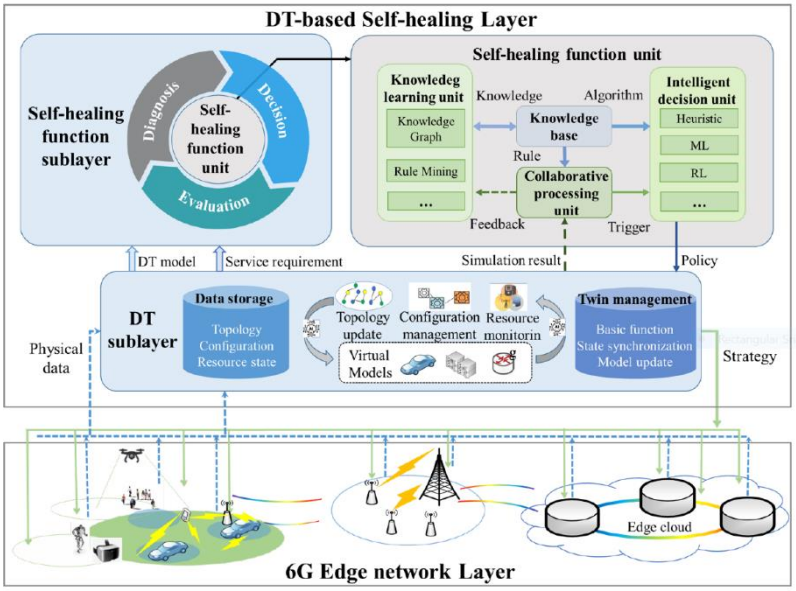

Figure 20. The DT-driven service self-healing architecture in [51]

Details on the Dynamic Graph Attention Network (DGAT), the subpart that predicts network performance is left out, due to the scope of this review and the same applies for the GNN based Service Redeployment Mechanism. The paper concludes that their work significantly enhances the reliability and performance of 6G edge networks. By integrating GNNs for performance prediction and service redeployment, the solution enables proactive management of network anomalies, ensuring continuous service delivery and reducing delays.

The work *"MVSTGN: A Multi-View Spatial-Temporal Graph Network for Cellular Traffic Prediction"* [52] introduces MVSTGN, a novel architecture for cellular traffic prediction that addresses the challenges of capturing complex spatial-temporal correlations in cellular networks. MVSTGN combines attention and convolution mechanisms to model the global and local spatial-temporal dependencies of cellular traffic.

Pointing out the importance of optimizing resource allocation and reducing energy consumption in mobile networks and the dynamic and complex nature of cellular traffic, accurate predictions are much demanded. Considering the importance of topic and a deficit in the previous literature, authors propose the new architecture that captures Global Spatial View, Global Temporal View and Local Spatial-Temporal View: A dense convolution module that further explores the local spatial-temporal dependencies within the network, combining the global and local information.

More in depth analysis is available in the paper, but overall it consists of Node-Level Attention mechanism which captures spatial correlations between nodes by analyzing the traffic state similarity, Trend-Level Attention mechanism to capture correlations based on traffic demand trends across different regions, a Spatial Gated Fusion mechanism to fuse the two previous information, and then the two parts defined previously, Global Temporal View and Local Spatial-Temporal View. A visualization of the architecture is provided in Figure 21.

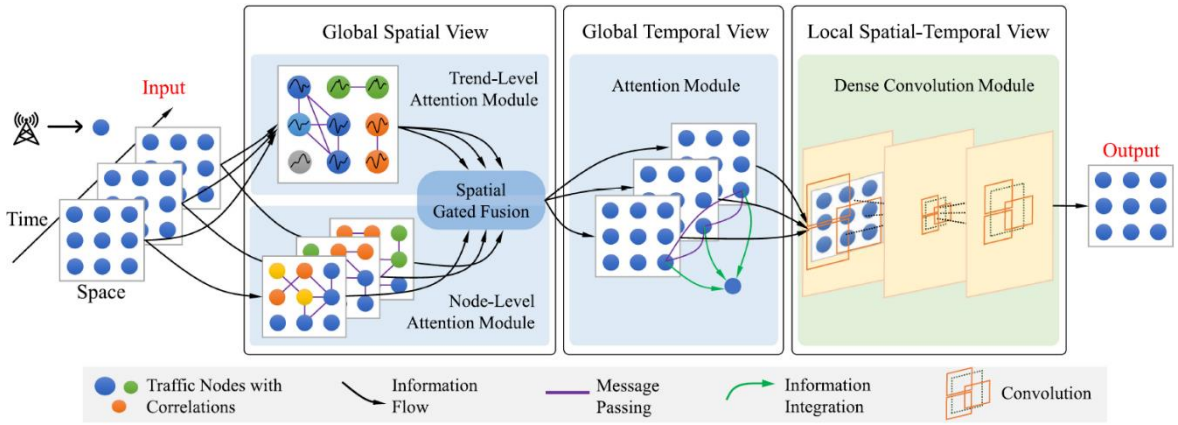

Figure 21. MVSTGN model proposed in [52]

A real-world dataset from the Telecom Italia Big Data Challenge[9] is used, which contains cellular traffic data in Milan. The dataset covers a citywide area and includes traffic data collected at 10-minute intervals. The experimental results show that MVSTGN outperforms all baseline models in terms of prediction accuracy, so the authors conclude that MVSTGN is a highly effective model for cellular traffic prediction, offering a comprehensive approach to capture complex spatial-temporal dependencies.

4.3 Edge-enabled application challenges

Up to this point, we have studied the deployment of servers and services and have briefly looked into works that cover different management aspect of the compute continuum. Supposedly, the system is up and running, but still, application specific challenges might rise, therefore a deeper understanding of such service difficulties is required. In the context of ELIXIRION, we will look into measures taken in the compute continuum to handle medical application requirements. At the next sub-section, compute-continuum policies towards serving AI applications are presented, referred to as Edge4AI.

4.3.1 Challenges

In *"Mobile Edge Computing Enabled 5G Health Monitoring for Internet of Medical Things: A Decentralized Game Theoretic Approach presents a Mobile Edge Computing (MEC)-enabled 5G health monitoring system"* [53] the excessive spectrum requirements and communication overload inherent to in-home health monitoring are addressed. The paper identifies several challenges, including the real-time transmission of healthcare data, minimizing energy consumption, and maintaining the freshness of medical information represented by the Age of Information (AoI).

The proposed solution involves two networks, Intra-WBANs and Beyond-WBANs with different approaches. For Intra-WBANs, body sensors collect healthcare data and transmit it to gateways using orthogonal frequency-division multiple access (OFDMA). These sensors cooperate to ensure timely data transmission, medical criticality and AoI guide bandwidth allocation. For Beyond-WBANs, once the data reaches the gateways, decisions must be made on whether to process the data locally or offload it to an MEC server using Non-

[9] https://theodi.fbk.eu/openbigdata/

Orthogonal Multiple Access (NOMA). Patients aim to minimize their overall costs (energy and AoI) by choosing the best processing strategy. A visualization of the concepts is presented in Figure 22.

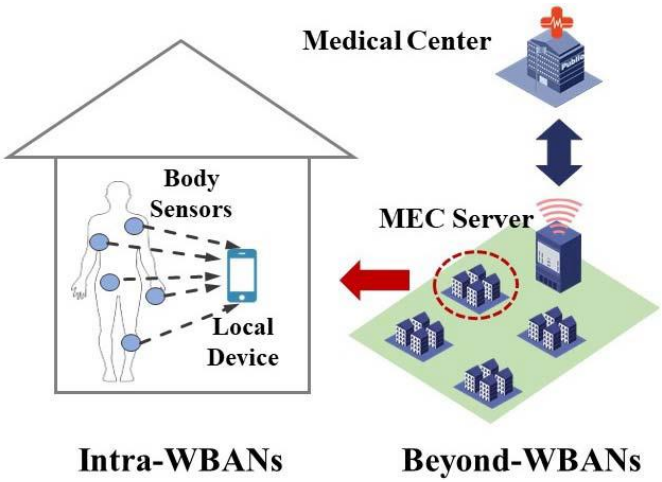

Figure 22. An illustration of MEC-enabled 5G health monitoring system considered in [53]

The optimization problem is formulated to minimize the system-wide cost of healthcare monitoring, according to medical criticality, AoI and Energy consumption. The problem is then solved in the two sub-networks, A cooperative game is formulated for Intra-WBANs and a non-cooperative game for beyond-WBANs modelling the interaction between patients competing for MEC resources. Nash Bargaining is used for the cooperative game, also non-cooperative game for beyond-WBANs is solved separately. Finally, a mixed algorithm combining the two is developed reaching a Nash equilibrium for the general problem. Simulations are conducted in a scenario with 30 patients using 5 edge servers, which show significant reductions in system-wide costs, outperforming baseline methods in terms of energy efficiency and AoI, keeping transmission delays within acceptable limits.

4.3.2 Edge for AI

The paper *"GNN at the Edge: Cost-Efficient Graph Neural Network Processing Over Distributed Edge Servers"* [54] is reviewed as an example for edge resources, orchestrated with specific application requirements in mind, in this case to provide Graph Neural Network Processing service. In this work, a cost-efficient approach for distributed GNN processing over heterogeneous edge networks is developed, defining system's cost metrics as data collection, GNN computation, cross-edge communication, and server maintenance. The problem is formulated as an NP-hard graph layout optimization problem, and then, the authors introduce their solution algorithm, called GLAD framework (Graph LAyout scheDuling). Cost reduction and convergence of the proposed framework is studied using real-world datasets.

The main problem addressed is the cost-efficient deployment of GNNs across multiple edge servers. This involves balancing several objectives: minimizing computation and communication costs while maintaining service performance. The complexity of this problem arises from the distributed nature of GNN inputs, server heterogeneity, and the need for frequent inter-server data exchange due to GNN's neighbour aggregation pattern. The authors introduce GLAD, a framework for GNN deployment optimization in two main scenarios: Static Input Graphs and Dynamic Input Graphs.

The system model is composed of a distributed edge network, a data graph (representing the input data, vertex being a client, and edges representing relationships) and a graph layout: The placement of client data on edge

servers is presented, which must be optimized to minimize costs. An abstracted system model is depicted in Figure 23.

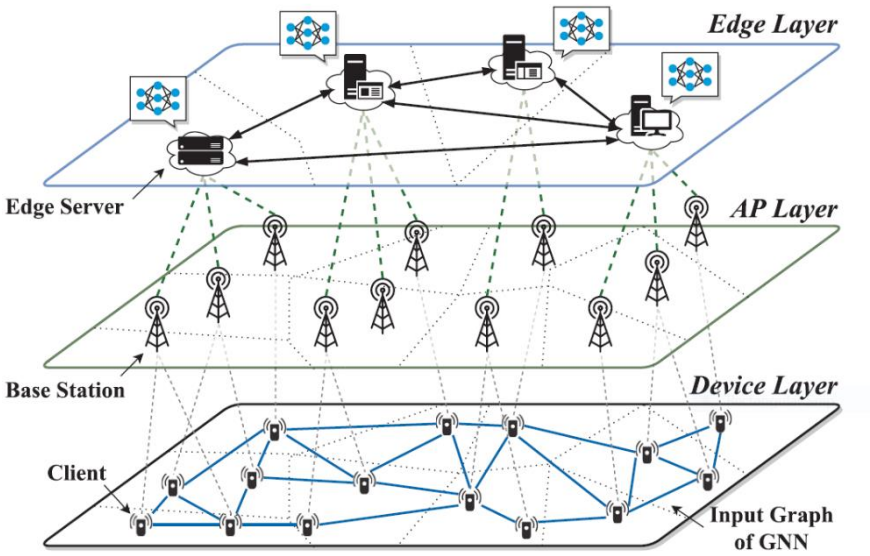

Figure 23. GNN processing network architecture considered in [54]

Considering the cost factors previously enumerated the two GLAD frameworks are formed as the GLAD-S (For Static Graphs:) which iteratively applies graph cuts to optimize the graph layout across pairs of edge servers. The GLAD-E (For Dynamic Graphs) that incrementally updates the graph layout to accommodate changes in the input graph. Trying to present a balanced algorithm, a hybrid approach is also presented, called GLAD-A by authors, which switches between previous frameworks according to performance drift, balancing system cost and update overhead.

Dataset for a socially-coupled IoT graph and a customer review graph from Yelp[10] are used for evaluation purposes which shows a great cost reduction compared to baseline methods, scalability and adaptability to changing input graphs and Superior performance in terms of both computational efficiency and communication costs.

[10] https://www.yelp.com/dataset

5 ICT for Secure Big Healthcare Data Analytics and Emergency Medicine

5.1 Introduction

The past decade has witnessed an exponential increase in the generation of big healthcare data by users from various sources like wearables and health monitoring equipment, electronic health records (EHRs), genomics data, health information exchange (HIE), social media, behavioural data.

Performing analytics on these data timely is crucial to make informed medical decisions such as preventive actions and making timely treatment decisions. Patients' healthcare data are subjected to privacy threats and data breaches. Such data contains patients' sensitive information which is worth high monetary value for illegitimate businesses [55].

Performing data analytics on the patients' personal devices like on their smartphones, wristwatches, and personal computers without sharing data with third-party systems is one of the ways to ensure data protection. However, such devices have resource constraints due to limited computational power, limited memory and storage. Designing efficient algorithms to work efficiently on such devices is challenging.

Sending patients' healthcare data to cloud is challenging in numerous ways. First, it requires a high throughput network to transmit a high volume of data. Second, a reliable connection for timely transmission of such data. Third, a secure connection to cloud for sending such sensitive data securely.

Besides, there are two major latencies associated with the usage of edge and cloud computing – communication latency and computation latency. When computation is performed at the local or edge device, there will be low communication latency due to the low proximity of the device form the data source but high computation latency due the limited hardware capabilities. On the other hand, the processing of data at cloud leads to high communication latency due to their typical high distance from the data source and low communication latency due to their much superior computational resources. Hence, the right strategies must be employed to find the right trade-offs between these latencies by sending the right data to the right computing frameworks [56].

Utilizing the edge-cloud continuum for efficiently allocating the right task to the right computing device is essential for big healthcare data. For example, analytics on time-series data or textual data can be performed on the user devices or edge devices. But it would be challenging to perform resource-heavy tasks like video analytics on such devices. Such resource-heavy tasks must be offloaded to the cloud [57].

It is crucial to adopt relevant ICT for big healthcare data. In this report, we discussed briefly the challenges in big healthcare data and the use of blockchain technology for potentially mitigating the challenges associated with big data security. Then we discussed the emergency medicine (EM) use case of ICT. Finally, we identified and listed key tools and frameworks for securely and timely performing analytics on the big healthcare data on the edge framework.

5.2 Big Data in Healthcare

Big healthcare data refers to the vast amounts of health-related information that are generated, collected, and analyzed across various healthcare settings. This data is comprised of everything from patient records, medical

imaging, clinical trial data, to real-time monitoring from wearable devices, and social media. It can be structured (e.g., lab results, patient demographics, or EHRs in general) or unstructured (e.g., doctor's notes, lab results, clinical notes, radiology reports or medical scans), semi-structured (e.g., HL7 messages), multimedia data (e.g., images, audios and videos), time-series data (e.g., EEG, vital signs), etc.

Big data processing has more technological challenges, such as management, cleaning, addressing imbalances, limited computational capabilities, security and privacy, analytics, and extracting valuable information from those.

The big healthcare data is characterized by the 4V features:

- ***Volume:*** The volume in big data refers to the sheer quantity of data generated in healthcare. Healthcare data is produced in massive quantities from numerous sources, such as electronic health records (EHRs), clinical trials, medical imaging (e.g., X-rays, MRIs), wearable devices, and genomic sequencing. Processing big healthcare data is subjected to several challenges, including dimensionality, modularity, and class imbalance. Besides, there are more challenges in processing high volumes of big data, such as non-linearity of data, bias and variance, and limited computing resource availability.
- ***Variety:*** The variety in big data refers to the different types and formats of healthcare data. Such variations in the healthcare data types lead to several challenges, such as data heterogeneity, data locality, and noisy and dirty data. Managing and integrating this diverse array of data types is crucial for a holistic understanding of patient care.
- ***Velocity:*** The velocity in big data refers to the speed of data generation from different sources. Data in healthcare is constantly being produced in real-time or near-real-time. For example, wearable devices and health apps collect continuous streams of data, such as heart rates or glucose levels, while emergency room systems handle data that needs immediate attention. This requires more computational resources to meet these data's processing time for making informed decisions.
- ***Veracity:*** Veracity refers to the accuracy, trustworthiness, and reliability of healthcare data or in short, the quality of big healthcare data. Not all healthcare data is clean or reliable. There can be errors in EHRs, inconsistent entries, or inaccurate patient records. Veracity is critical for clinical decision-making, as incorrect data can lead to misdiagnoses or inappropriate treatments. The transmission of real-time multimedia data using the user datagram protocol, like audio and video from noisy channels, also leads to their degradation at the receiver.

5.3 Wireless Communications for Big Healthcare Data

Wireless communication technology plays a crucial role in the management, collection, and transmission of big healthcare data, especially as healthcare systems increasingly adopt digital solutions for patient care, diagnostics, and monitoring. Wireless communication is a key enabling technology for wearables and for telemedicine, helping to overcome challenges related to data volume, speed, mobility, and accessibility.

For big healthcare data, there are stringent requirements for wireless communications in terms of QoS, especially in terms of their key performance indicators (KPIs) like throughput, latency, reliability, connection density, mobility support, and for network security.

Different wireless communication technologies will be required for big healthcare data depending on the specific use cases. These use cases differ in terms of the size of the data and the distances between the sender and receiver. In the following section, we stratify the use cases on wireless communication technologies for healthcare in terms of their range of communication below:

5.3.1 Short-range Wireless Communication Technologies

The short-range wireless communications are widely used in healthcare for seamless interaction and transmission of data between medical devices. These technologies offer secure, reliable, and low-power communication over relatively short distances, making them ideal for clinical settings and specifically crucial for HIE systems [58].

5.3.1.1 Bluetooth

Bluetooth technology typically offers a range of 10 meters and even 100 meters for certain versions. This technology is useful in transmission of data from wearables and medical devices to smartphones and computers [59].

5.3.1.2 Near-Field Communication (NFC)

NFC offers a transmission range of up to 4 cm. This technology is used in fast authentication of medical devices, such as in blood glucose meters and insulin pumps with a simple tap, data transfer between patient records and smartcards or wearable devices, and contactless payment for healthcare services (e.g., patient registration, billing) [60].

5.3.1.3 Wi-Fi

Wi-Fi offers a typical range of up to 50 meters and offers high throughput of transmission rate for transferring high volume of medical data (e.g., MRI or CT scans) and other multimedia data (e.g., patient monitoring video) [61].

5.3.1.4 Zigbee

Zigbee typically offers a communication range of 10-100 meters. The use cases of Zigbee in healthcare includes wireless sensors for monitoring patient conditions such as heart rate, temperature, and oxygen levels, IoT for managing hospital environments like lighting, medical equipment tracking, etc., and connecting medical devices in a low-power mesh network for continuous monitoring [62].

5.3.1.5 Radio-Frequency Identification (RFID)

The communication range of RFID is from a few centimeters to several meters depending on the antenna size and operating frequency. Some use cases of RFID in healthcare includes, tracking patients, medical equipment, and medication in hospitals; verifying and managing patient records and prescriptions; and monitoring the cold chain for pharmaceutical products, ensuring temperature-sensitive items are stored properly [63].

5.3.1.6 Infrared

The infrared WC technology typically offers a communication range of up to 5 meters. The typical use cases are remote control of medical devices (e.g., ventilators, infusion pumps) in sterile environments without physical contact; and communication between medical devices, such as a thermometer and a computer [64].

5.3.1.7 Z-Wave

The Z-wave offers a range of around 30 meters for indoor environments. The typical use cases are smart hospital environments for remote control of medical devices and systems (e.g., lighting, alarms); home healthcare automation, where devices like blood pressure monitors or motion detectors communicate with central hubs [65].

5.3.2 Long-range Wireless Communication Technologies

5.3.2.1 Cellular Networks (4G/5G)

A single base station of the cellular networks offers a coverage range from a few hundred meters to 100 kilometers, depending on the operating frequency and other antenna design technology. The typical use cases are Telemedicine for real-time consultations, remote diagnostics, and video conferencing between healthcare providers and patients, etc.; remote monitoring using wearable health devices, such as heart rate monitors and glucose sensors, transmitting data to healthcare providers; mobile health applications using apps that provide real-time data, reminders, and alerts to patients and doctors; ambulance services for connecting paramedics with hospital systems to share vital signs and prepare for patient arrival [66].

5.3.2.2 LoRaWAN

The typical range of the LoRaWAN WC is up to 10 kilometers in urban areas, and up to 15-30 kilometers in rural or open environments. The LoRaWAN is a long-range, low-power, low-cost communication technology that operates on a low-frequency spectrum and thus offers low throughput. The typical use cases of LoRaWAN WC are remote patient monitoring by collecting data from wearable devices or sensors like heart rate, temperature in remote areas and transmitting it to healthcare providers over long distances; asset tracking by monitoring the location of medical equipment and inventory management in large hospitals or healthcare facilities; elderly care by tracking patients with chronic conditions or elderly patients living at home, sending alerts in case of abnormal conditions like fall detection, irregular heartbeats [67].

5.3.2.3 Satellite Communications

Satellite communications offer a global range of coverage, covering remote or rural regions where other communication networks may not be available. The typical use cases are telemedicine in remote areas by providing healthcare consultations and data transfer in regions with no cellular or internet connectivity e.g., disaster zones, rural clinics, or offshore; remote diagnostics using the transmission of diagnostic data from isolated areas e.g., remote clinics, maritime or aerospace environments to specialists in urban centers; and emergency response by connecting emergency medical teams in remote locations with hospitals to provide real-time updates [68].

5.3.2.4 WiMAX (Worldwide Interoperability for Microwave Access)

WiMAX offers a communication range typically up to 50 kilometers. The typical use cases of WiMAX are telemedicine by supporting video consultations, large data transfers e.g., medical imaging, and real-time monitoring of patients in rural or underserved areas); mobile clinics by providing connectivity to mobile healthcare units in rural areas, enabling them to communicate with hospitals and transmit patient data; and expanding healthcare facilities by connecting remote clinics or healthcare facilities to central hospital systems [69].

5.3.2.5 Mesh Networks

Mesh networks offer scalable range, depending on the number of devices (nodes), typically covering large hospitals or healthcare campuses. The typical use cases are hospital networks as a mesh network can connect multiple healthcare devices, sensors, and systems in a large facility without relying on central hubs, ensuring reliable communication even if some nodes fail; home healthcare system by creating a reliable communication network in home care settings to connect multiple devices, such as patient monitoring systems, alarms, and medication dispensers; and creating community healthcare system in rural or remote settings, mesh networks can be used to link healthcare centres or homes to a central hospital or clinic [70].

5.4 Blockchain for Big Healthcare Data

A blockchain is a decentralized digital ledger system that records transactions shared across different nodes or blocks in the network in such a way that the recorded transactions cannot be altered retroactively. Each participating block or node has the complete monitoring authority of records of transactions in a peer-to-peer manner.

Blockchain technology can potentially revolutionize how medical data is stored, shared, and accessed, offering enhanced security, privacy, and interoperability [71]. Big healthcare data requires the utmost privacy, security, and integrity. Integrating blockchain for big healthcare data can potentially ensure safe patient data management.

The advantages of the blockchain integration for big healthcare data are summarized next:

- ***Improved privacy and security:*** The big healthcare data are potentially vulnerable to untrusted entities. The Firewall technology cannot address this vulnerability because these data sources are not part of an organization's network perimeter. Blockchain technology can potentially address such issues by utilizing decentralized and encrypted peer-to-peer storage technology to prevent any unauthorized access to personal healthcare data.
- ***Improved data integrity:*** Blockchain technology ensures that personal healthcare data is immutable to tempering. The untrusted entities can only mutate the data if they can control more than 50% of the nodes or blocks in the blockchain network, which is infeasible in practice.
- ***Real-time data analytics:*** Blockchain technology involves transactions in real-time across distributed nodes. This makes big healthcare data analytics and decision-making in real-time possible. The use of blockchain for big healthcare data also ensures the monitoring of potential changes in these data in real-time.

- ***Enhanced sharing of big healthcare data:*** Integrating blockchain technology in big healthcare data minimizes the risk of data leakage to third-party entities not part of the blockchain network, as the blockchain technology records the transactions carried out on them.

5.5 ICT for Emergency Medicine

Emergency medicine (EM) is a type of healthcare scenario where patients are subjected to losing their long-term health or even in a matter of minutes of delay in initiating appropriate medical intervention. Hence, ICT plays a crucial role in saving the lives of EM patients. For example, during medical emergencies, the family members of those patients require a reliable communication medium to call 112 for prompt EMS response. While en route to the receiving hospitals, the EMS also requires a reliable communication channel to share patient information with the receiving emergency department (ED) and seek guidance from remote doctors to stabilize the patient in an ambulance.

With the advancement of ICT in the last two decades, it is possible to exchange numerous data between different stakeholders in EM. We identified the three crucial technologies in EM that can potentially improve patient outcomes in EM, which we will discuss below:

5.5.1 Types of Data in Emergency Medicine

Timely sharing of patient-related data is crucial for all the stakeholders in the EM-patient's family members, EMS crews, and ED. This data sharing ensures informed, in-ambulance decision-making by the EMS personnel, timely preparation by the receiving EDs, and timely information reception to the patient's family members and, thus, the overall patient outcome. From the peer-reviewed literature, we identified the following data for EM, described in the next subsections.

5.5.1.1 Text and time-series data

- ***Electronic Healthcare Records (EHR):*** EHR includes the patient demographics, illness and treatment history, allergies, vaccination history, etc. The EHR data is essential for making informed decisions in EM scenarios [72]. This data is accessible to the physicians of the receiving EDs. However, the accessibility of this data to the EMS personnel is subject to local laws.
- ***Vital signs:*** The vital signs data include blood pressure, respiratory rate, respiratory state, heart rate, pulse rate, body temperature, blood oxygen saturation level, end-tidal carbon dioxide level, consciousness information and more. These data help the EM stakeholders monitor the emergency patients in real-time and thus help make prompt decisions and interventions [72], [73].
- ***Electroencephalogram:*** Electroencephalogram (EEG) helps in timely diagnosing the emergency patients presented to the EDs for their suspicions on acute symptoms like seizures, acute stroke, epilepsy history, etc. and thus helpful in initiating timely and accurate treatments [74].
- ***Triaging:*** The prehospital triaging of emergency patients helps classify the patient's acuity level and prioritise the appropriate resources in a timely manner based on the patient's acuity levels [75].

- ***Time logging:*** The automatic time logging of various events is crucial in EM. For example, the time logging of ambulance arrival at the patient's site and estimated arrival time at the hospital are helpful in the timely preparation of the receiving ED [72], [73].

5.5.1.2 Multimedia and medical imaging data

- ***Speech:*** The patients' speech data can potentially help test the patients for aphasia associated with stroke. The advancement of deep learning-based diagnostic methods can potentially improve the recognition of aphasia caused by stroke [76].
- ***Conversational audio:*** The conversational voice between the EM stakeholders can be potentially helpful for electronic patient care record (ePCR) and patient report generation by utilizing automatic speech recognition (ASR) and natural language processing (NLP) technologies [77].
- ***Patient images:*** Patient image sharing between the EMS and ED can improve the EM by helping the receiving ED better understand the injury mechanisms and other acute illness-related symptoms [78].
- ***Diagnostic imaging:*** The computed tomography (CT) scans performed in the mobile stroke units (MSUs) are beneficial in the prehospital diagnosis of suspected stroke patients. This saves time and resources as the patients do not have to be transported to the suboptimal hospital, which does not have appropriate treatment facilities for the different types of stroke patients [79]. However, the prevalence of MSUs is very low due to their very high costs.
- ***Video:*** Remote or receiving ED doctors can monitor the in-ambulance emergency patients in real time and help the EMS personnel with appropriate in-ambulance interventions for the time-critical EM [73].
- ***Diagnostic videos:*** The videos of portable ultrasonography machines are helpful in the rapid identification of imminent life-threatening injuries and thus help in performing immediate medical intervention for needful emergency patients [80].

5.5.2 Wireless Communication for EM

Sharing crucial patient data in EM requires a reliable communication medium that ensures no information is missed or corrupted. Hence, data integrity is not lost when shared between the stakeholders in EM. The following are the wireless technologies that are crucial for EM.

5.5.2.1 Cellular network (4G/5G/B5G)

Since the advent of 4G cellular networks with much higher key performance indicators than their predecessors, transmitting high-definition patient videos from moving ambulances to hospitals has become possible [79]. Hence, this high-speed communication made it possible for effective telemedicine to help the EMS personnel perform in-ambulance, informed and timely patient interventions while en route to the receiving ED.

5.5.2.2 WiMAX

WiMAX operates in the microwave range of the broadband frequency spectrum. The advantage of this technology is that it offers an extended range of communications with high throughput [81]. Furthermore, the communication infrastructure for this technology can be rapidly deployed in remote locations that lack permanent infrastructure for mobile communications [82].

5.5.2.3 LoRaWAN

It could be helpful for telemedicine, which requires sharing data with low storage and throughput requirements, like vital signs, voice messages, and images [83], [84].

5.5.2.4 SatCom

The SatCom technology is crucial for EM in disaster scenarios and remote locations where communication infrastructure is unavailable. This technology is also useful when terrestrial network-based communication infrastructures get damaged due to war and natural disasters [85].

5.5.2.5 Radio Communication

The radio communication technology in EM uses ultra-high frequency and very high-frequency radio communication for the voice interactions between different EMS personnel and the dispatch centers in the events of mass casualties like stampedes, wars and natural disasters. This technology ensures direct and timely interaction for effective and timely coordination between the EMS personnel [86].

5.5.3 AI for Emergency Medicine

AI is a disruptive technology for EM. AI has numerous potential uses in EM. However, for our study, we identified two key areas- decision-making and documentation, which we will discuss briefly below:

5.5.3.1 AI for EM decision making

Timely and accurate decision-making in EM is crucial. AI can support the EMS personnel in numerous ways [87]. Some of these are listed below:

- Automating the triage process and ensuring that high-risk patients are prioritized in terms of resources.
- AI models can revolutionize the diagnosis of emergency patients based on several types of data like CT, X-rays, EEG, and vital signs and thus initiate timely treatment.
- AI can utilize the emergency patient's EHR data and current vital signs for performing predictive analytics for potential complications such as strokes, sepsis, myocardial infarctions, etc.
- AI-driven clinical decision support systems can provide real-time recommendations to emergency patients based on evidence-based recommendations and interventions, thus reducing the scope of errors in time-critical EM scenarios.

5.5.3.2 AI for EM documentation

ASR and NLP combined can utilize conversational speech between EM stakeholders and can be crucial for several documentation-related EM applications [88]. Thus, this technology can significantly improve emergency patient-related information retention and retrieval and reduce the cognitive burden on EMS personnel, ED doctors, and healthcare administrative personnel. So, if the potential applications of these are given below:

- Automatic reporting of ePCR
- Automatic clinical notes generation
- Automatic ED handover report generation
- Automatic documentation for billing and insurance claims

5.6 Frameworks for medical big data analytics

5.6.1 AI for local computation

It is crucial to compute the user's vulnerable healthcare data right at their location to prevent sharing them to the third-party computing entities. The local computation also has key advantages like low-latency and offline computation capability. The following are the list of edge-aware tools and frameworks for performing medical data analytics locally:

- ***PyTorch Mobile:*** PyTorch Mobile[11] is a light version of PyTorch's deep learning framework which is made to run machine learning models on mobile and edge devices. This framework is optimized for low-latency on-device data processing.
- ***LiteRT:*** LiteRT[12] is a light-weight framework of Google's deep learning framework, formerly known as TensorFlow Lite, which is suitable for low-latency on device machine learning with support for wearables, smartphones and IoT devices.
- ***Intel's OpenVINO:*** OpenVINO[13] is Intel's toolkit for deploying AI-models on the edge device with applications including inference on the medical imaging data for performing diagnostics thus making it suitable for low-latency healthcare applications.
- ***Microsoft Azure IoT Edge:*** Microsoft Azure IoT Edge[14] is a framework for performing data analytics securely on IoT devices. It also supports integration with Microsoft's Azure cloud platform.
- ***Amazon SageMaker Neo:*** Amazon SageMaker Neo[15] is an Amazon Web Services tool that offers optimization of machine learning models for edge devices especially for less powerful patient care devices like wearables, remote monitoring equipment, etc.

5.6.2 Big data

[11] https://pytorch.org/mobile/home/
[12] https://ai.google.dev/edge/litert
[13] https://www.intel.com/content/www/us/en/developer/tools/openvino-toolkit/overview.html
[14] https://azure.microsoft.com/en-us/products/iot-edge
[15] https://aws.amazon.com/sagemaker/neo/

- ***Apache Kafka:*** Apache Kafka[16] is a distributed data streaming platform making it suitable for use cases where there is a requirement of continuous streaming of the user's healthcare data from wearables and other sensors. It supports real-time data ingestion and streaming and it can process high-throughput data with low latency.
- ***Apache Flink:*** Apache Flink[17] is a stream processing framework that requires handling real-time data at scale. This supports low-latency analytics and processing of complex events. It offers batch processing capabilities and a distributed framework for edge deployment.
- ***Apache Spark:*** Apache Spark[18] offers fast in-memory computation of big data which is capable of handling batch and stream processing. Also offers an ML framework, MLlib, which is built into Spark. Potential healthcare use cases are the analysis of massive health datasets such as medical images, genomic data, or patient records, and using ML for predictive healthcare applications.
- ***Apache Hadoop:*** The Apache Hadoop[19] framework is developed for distributed data storage and processing large-scale data. It was originally developed for cloud environments, but specialized configurations can also be optimized for edge computing scenarios.
- ***KubeEdge:*** KubeEdge[20] was developed by Kubernetes for orchestration and management of containerized applications on the edge devices for running the big data and AI-applications. Supports low-latency interactions between edge and cloud.

[16] https://kafka.apache.org/
[17] https://flink.apache.org/
[18] https://spark.apache.org/
[19] https://hadoop.apache.org/
[20] https://kubeedge.io/

6 Conclusions

This deliverable provided an overview of relevant concepts, technological frameworks, and methodologies towards enhancing low-latency healthcare applications, particularly in Emergency Medicine. By examining zero-touch multi-domain edge orchestration, deployment and management of distributed computing workflows, and secure big data solutions, we have identified key challenges and opportunities within the edge-cloud compute continuum. The discussions on explainable AI, edge-enabled application requirements and blockchain technology further underscore the importance of integrating advanced methodologies to address the unique demands of healthcare data management

The insights will be leveraged by the ELIXIRION project to further advance the technological contributions within WP2, to be reported in the future deliverable D2.2.

7 REFERENCES

[1] Developing software for multi-access edge computing," ETSI White Paper 20, Feb. 2019.

[2] H. Gu, L. Zhao, Z. Han, G. Zheng, and S. Song, "AI-Enhanced Cloud-Edge-Terminal Collaborative Network: Survey, Applications, and Future Directions," *IEEE Communications Surveys & Tutorials*, vol. 26, no. 2, pp. 1322–1348, 2024. doi: 10.1109/COMST.2023.3338153.

[3] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On Multi-Access Edge Computing: A Survey of the Emerging 5G Network Edge Cloud Architecture and Orchestration," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1657–1681, 2017. doi: 10.1109/COMST.2017.2705720.

[4] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A Survey on Mobile Edge Computing: The Communication Perspective," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017. doi: 10.1109/COMST.2017.2745201.

[5] ETSI GS ZSM 002 Version 1.1.1, 2019. https://www.etsi.org/

[6] ETSI GS NFV 002 Version 1.2.1, 2014. https://www.etsi.org/

[7] ETSI GS MEC 003 Version 3.1.1, 2022. https://www.etsi.org/

[8] C. Kuang, Y. Qiu, W. Cao, Z. Xiao and Z. Ming, "A Brief Review on Prediction Methods for Cloud Resource Management," *2024 9th IEEE International Conference on Smart Cloud (SmartCloud)*, NYC, NY, USA, 2024, pp. 92-96

[9] J. Bi, S. Li, H. Yuan, and M. Zhou, "Integrated deep learning method for workload and resource prediction in cloud systems," Neurocomputing, vol. 424, pp. 35–48, 2021.

[10] H. M. Nguyen, G. Kalra, and D. Kim, "Host load prediction in cloud computing using long short-term memory encoder–decoder," The Journal of Supercomputing, vol. 75, no. 11, pp. 7592–7605, 2019

[11] Y. Xie, M. Jin, Z. Zou, G. Xu, D. Feng, W. Liu, and D. Long, "Real-time prediction of docker container resource load based on a hybrid model of arima and triple exponential smoothing," IEEE Transactions on Cloud Computing, vol. PP, pp. 1–1, 2020.

[12] Devi, K.L., Valli, S. Time series-based workload prediction using the statistical hybrid model for the cloud environment. *Computing* 105, 353–374

[13] M. M. Al-Sayed, "Workload time series cumulative prediction mechanism for cloud resources using neural machine translation technique," *Journal of Grid Computing*, vol. 20, no. 2, p. 16, 2022

[14] A. Singh, D. Saxena, J. Kumar, and V. Gupta, "A quantum approach towards the adaptive prediction of cloud workloads," IEEE Transactions on Parallel and Distributed Systems, vol. PP, pp. 1–1, 2021

[15] Z. Ding, B. Feng, and C. Jiang, "Coin: A container workload prediction model focusing on common and individual changes in workloads," IEEE Transactions on Parallel and Distributed Systems, vol. 33, no. 12, pp. 4738–4751, 2022

[16] Javad Dogani, Reza Namvar, Farshad Khunjush, "Auto-scaling techniques in container-based cloud and edge/fog computing: Taxonomy and survey", Computer Communications, Volume 209, 2023, pp. 120-150

[17] A. Ullah, J. Li, Y. Shen, A. Hussain, A control theoretical view of cloud elasticity: taxonomy, survey and challenges, Cluster Comput. 21 (2018) 1735–1764

[18] Y. Shin, W. Yang, S. Kim, J.-M. Chung, Multiple adaptive-resource-allocation real-time supervisor (MARS) for elastic IIoT hybrid cloud services, IEEE Trans. Netw. Sci. Eng. 9 (3) (2022) 1462–1476

[19] D. Perri, M. Simonetti, S. Tasso, F. Ragni, O. Gervasi, Implementing a scalable and elastic computing environment based on cloud containers, in: Computational Science and Its Applications–ICCSA 2021: 21st International Conference, Cagliari, Italy, September 13–16, 2021, Proceedings, Part I 21, Springer, 2021, pp. 676–689

[20] K. Ray, A. Banerjee, Horizontal auto-scaling for multi-access edge computing using safe reinforcement learning, ACM Trans. Embed. Comput. Syst. (TECS) 20 (6) (2021) 1–33
[21] Y. Al-Dhuraibi, F. Paraiso, N. Djarallah, P. Merle, Elasticity in cloud computing: state of the art and research challenges, IEEE Trans. Serv. Comput. 11 (2) (2017) 430–447
[22] G.R. Russo, V. Cardellini, G. Casale, F.L. Presti, MEAD: Model-based vertical auto-scaling for data stream processing, in: 2021 IEEE/ACM 21st International Symposium on Cluster, Cloud and Internet Computing, CCGrid, IEEE, 2021, pp. 314–323
[23] V. Rampérez, J. Soriano, D. Lizcano, J.A. Lara, FLAS: A combination of proactive and reactive auto-scaling architecture for distributed services, Future Gener. Comput. Syst. 118 (2021) 56–72
[24] D.R. Augustyn, Improvements of the reactive auto scaling method for cloud platform, in: Computer Networks: 24th International Conference, CN 2017, Lądek Zdrój, Poland, June 2023, 2017, Proceedings, Springer, 2017, pp. 422–431
[25] A. Bento, J. Correia, R. Filipe, F. Araujo, J. Cardoso, Automated analysis of distributed tracing: Challenges and research directions, J. Grid Comput. 19 (2021) 1–15
[26] E. Radhika, G.S. Sadasivam, A review on prediction based autoscaling techniques for heterogeneous applications in cloud environment, Mater. Today: Proc. 45 (2021) 2793–2800
[27] M. Yadav, H.A. Akarte, D.K. Yadav, Container elasticity: Based on response time using docker, Recent Adv. Comput. Sci. Commun. (Formerly: Recent Pat. Comput. Sci.) 15 (5) (2022) 773–785
[28] K. Cho, et al., Learning phrase representations using RNN encoder–decoder for statistical machine translation, arXiv preprint arXiv:1406.1078
[29] N.-M. Dang-Quang, M. Yoo, Deep learning-based autoscaling using bidirectional long short-term memory for kubernetes, Appl. Sci. 11 (9) (2021) 3835
[30] D. Saxena, A.K. Singh, A proactive autoscaling and energy-efficient VM allocation framework using online multi-resource neural network for cloud data center, Neurocomputing 426 (2021) 248–264
[31] Z. Cai, D. Liu, Y. Lu, R. Buyya, Unequal-interval based loosely coupled control method for auto-scaling heterogeneous cloud resources for web applications, Concurr. Comput.: Pract. Exper. 32 (23) (2020) e5926
[32] Das, A.; Rad, P. Opportunities and challenges in explainable artificial intelligence (xai): A survey. arXiv 2020, arXiv:2006.11371
[33] A. Adadi and M. Berrada, "Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)," in *IEEE Access*, vol. 6, pp. 52138-52160, 2018
[34] S. Wang, M. A. Qureshi, L. Miralles-Pechuán, T. Huynh-The, T. R. Gadekallu and M. Liyanage, "Explainable AI for 6G Use Cases: Technical Aspects and Research Challenges," in *IEEE Open Journal of the Communications Society*, vol. 5, pp. 2490-2540, 2024
[35] S. Garg, K. Kaur, G. S. Aujla, G. Kaddoum, P. Garigipati and M. Guizani, "Trusted Explainable AI for 6G-Enabled Edge Cloud Ecosystem," in *IEEE Wireless Communications*, vol. 30, no. 3, pp. 163-170, June 2023
[36] F. Rezazadeh, H. Chergui and J. Mangues-Bafalluy, "Explanation-Guided Deep Reinforcement Learning for Trustworthy 6G RAN Slicing," *2023 IEEE International Conference on Communications Workshops (ICC Workshops)*, Rome, Italy, 2023, pp. 1026-1031
[37] M. S. Munir *et al.*, "Neuro-Symbolic Explainable Artificial Intelligence Twin for Zero-Touch IoE in Wireless Network," in *IEEE Internet of Things Journal*, vol. 10, no. 24, pp. 22451-22468, 15 Dec.15, 2023
[38] P. Barnard, I. Macaluso, N. Marchetti and L. A. DaSilva, "Resource Reservation in Sliced Networks: An Explainable Artificial Intelligence (XAI) Approach," *ICC 2022 - IEEE International Conference on Communications*, Seoul, Korea, Republic of, 2022, pp. 1530-1535
[39] N. Khan, S. Coleri, A. Abdallah, A. Celik and A. M. Eltawil, "Explainable and Robust Artificial Intelligence for Trustworthy Resource Management in 6G Networks," in *IEEE Communications Magazine*, vol. 62, no. 4, pp. 50-56, April 2024

[40] Małota, A., Koperek, P., Funika, W. (2023). Towards Understanding of Deep Reinforcement Learning Agents Used in Cloud Resource Management. In: Mikyška, J., de Mulatier, C., Paszynski, M., Krzhizhanovskaya, V.V., Dongarra, J.J., Sloot, P.M. (eds) Computational Science – ICCS 2023. ICCS 2023. Lecture Notes in Computer Science, vol 14074

[41] 5GPPP, "Edge Computing for 5G Networks V1.0", 0.5281/zenodo.3698117, Jan. 2021.

[42] Ericsson Technology Review, "Next-generation Edge-Cloud Ecosystem", Feb. 2020

[43] A. Cañete Garrucho, X. Palomo Teruel, O. Feliu Juárez, E. Kartsakli, and E. Quiñones Moreno, "EXTRACT: A distributed data-mining software platform for EXTReme dAta aCross The compute continuum," in Proc. 11th ACM Celebration of Women in Computing: womENcourage 2024, Madrid, Spain, Jun. 2024, pp. 1-2

[44] X. Zhang, Z. Li, C. Lai, and J. Zhang, "Joint Edge Server Placement and Service Placement in Mobile-Edge Computing," *IEEE Internet of Things Journal*, vol. 9, no. 13, pp. 11261-11274, Jul. 2022. doi: 10.1109/JIOT.2021.3125957

[45] Z. Zhao, H. Cheng, X. Xu, and Y. Pan, "Graph Partition and Multiple Choice-UCB Based Algorithms for Edge Server Placement in MEC Environment," *IEEE Transactions on Mobile Computing*, vol. 23, no. 5, pp. 4050-4061, May 2024. doi: 10.1109/TMC.2023.3284994

[46] D. D. Sánchez-Gallegos, A. Galaviz-Mosqueda, J. L. Gonzalez-Compean, S. Villarreal-Reyes, A. E. Perez-Ramos, D. Carrizales-Espinoza, and J. Carretero, "On the Continuous Processing of Health Data in Edge-Fog-Cloud Computing by Using Micro/Nanoservice Composition," *IEEE Access*, vol. 8, pp. 120255-120268, 2020. doi: 10.1109/ACCESS.2020.3006037

[47] "COMPSs Documentation," accessed Oct. 17, 2024. Online. Available: https://compss-doc.readthedocs.io/en/stable/

[48] Rojas Pérez, "5G-enabled edge deployments for real-time services in smart city environments," M.S. thesis, Fac. d'Informàtica de Barcelona, Universitat Politècnica de Catalunya, Barcelona, Spain, 2023. Online. Available: https://upcommons.upc.edu/handle/2117/413951

[49] D. Yin, L. Chen, J. Yang, and K. Deng, "An Efficient Multi-Edge Server Coalition Computation Offloading Scheme of Sensor-Edge-Cloud," IEEE Access, vol. 12, pp. 12909-12921, 2024. doi: 10.1109/ACCESS.2023.3344389

[50] H. Hu, W. Song, Q. Wang, R. Q. Hu, and H. Zhu, "Energy Efficiency and Delay Tradeoff in an MEC-Enabled Mobile IoT Network," *IEEE Internet of Things Journal*, vol. 9, no. 17, pp. 15942-15953, Sept. 2022. doi: 10.1109/JIOT.2022.3153847

[51] P. Yu, J. Zhang, H. Fang, W. Li, L. Feng, F. Zhou, P. Xiao, and S. Guo, "Digital Twin Driven Service Self-Healing with Graph Neural Networks in 6G Edge Networks," IEEE Transactions on Mobile Computing, vol. 41, no. 3, pp. 720–734, Mar. 2023. doi: 10.1109/JSAC.2022.3229422

[52] Y. Yao, B. Gu, Z. Su, and M. Guizani, "MVSTGN: A Multi-View Spatial-Temporal Graph Network for Cellular Traffic Prediction," IEEE Transactions on Mobile Computing, vol. 22, no. 5, pp. 2837–2849, May 2023. doi: 10.1109/TMC.2021.3129796

[53] Z. Ning, P. Dong, X. Wang, X. Hu, L. Guo, B. Hu, Y. Guo, T. Qiu, and R. Y. K. Kwok, "Mobile Edge Computing Enabled 5G Health Monitoring for Internet of Medical Things: A Decentralized Game Theoretic Approach," IEEE Journal on Selected Areas in Communications, vol. 39, no. 2, pp. 463–476, Feb. 2021. doi: 10.1109/JSAC.2020.3020645

[54] L. Zeng, C. Yang, P. Huang, Z. Zhou, S. Yu, and X. Chen, "GNN at the Edge: Cost-Efficient Graph Neural Network Processing Over Distributed Edge Servers," IEEE Journal on Selected Areas in Communications, vol. 41, no. 3, pp. 720–734, Mar. 2023. doi: 10.1109/JSAC.2022.3229422

[55] 2020 Trustwave Global Security Report URL: https://www.trustwave.com/hubfs/Web/Library/Documents_pdf/D_16791_2020-trustwave-global-security-report.pdf

[56] Alekseeva, Daria, et al. "The future of computing paradigms for medical and emergency applications." Computer Science Review 45 (2022): 100494.
[57] Sabia, Alessia, and Beniamino Di Martino. "Towards a Patterns Driven Cloud Edge Continuum Architecture for eHealth." International Conference on Advanced Information Networking and Applications. Cham: Springer Nature Switzerland, 2024.
[58] Vidakis, Konstantinos, et al. "A comparative study of short-range wireless communication technologies for health information exchange." 2020 International conference on electrical, communication, and computer engineering (ICECCE). IEEE, 2020.
[59] Zhang, Ting, et al. "Bluetooth low energy for wearable sensor-based healthcare systems." 2014 IEEE healthcare innovation conference (HIC). IEEE, 2014.
[60] Sethia, Divyashikha, et al. "NFC based secure mobile healthcare system." 2014 sixth international conference on communication systems and networks (COMSNETS). IEEE, 2014.
[61] Ge, Yao, et al. "Contactless WiFi sensing and monitoring for future healthcare-emerging trends, challenges, and opportunities." IEEE Reviews in Biomedical Engineering 16 (2022): 171-191.
[62] Zhang, Zhongwei, and Xiaohu Hu. "ZigBee based wireless sensor networks and their use in medical and health care domain." 2013 Seventh International Conference on Sensing Technology (ICST). IEEE, 2013.
[63] Wamba, Samuel Fosso, Abhijith Anand, and Lemuria Carter. "A literature review of RFID-enabled healthcare applications and issues." International Journal of Information Management 33.5 (2013): 875-891.
[64] Hong, Keum-Shik, and M. Atif Yaqub. "Application of functional near-infrared spectroscopy in the healthcare industry: A review." Journal of Innovative Optical Health Sciences 12.06 (2019): 1930012.
[65] Chakraborty, Shouvik, Kalyani Mali, and Sankhadeep Chatterjee. "Edge computing based conceptual framework for smart health care applications using z-wave and homebased wireless sensor network." Mobile Edge Computing (2021): 387-414.
[66] Ahad, Abdul, Mohammad Tahir, and Kok-Lim Alvin Yau. "5G-based smart healthcare network: architecture, taxonomy, challenges and future research directions." *IEEE access* 7 (2019): 100747-100762.
[67] Mdhaffar, Afef, et al. "IoT-based health monitoring via LoRaWAN." *IEEE EUROCON 2017-17th international conference on smart technologies*. IEEE, 2017.
[68] Li, Huan-Bang, et al. "Wireless body area network combined with satellite communication for remote medical and healthcare applications." Wireless personal communications 51 (2009): 697-709.
[69] Yen, Yun-Sheng, et al. "Using WiMAX network in a telemonitoring system." 2011 3rd International Conference on Computer Research and Development. Vol. 1. IEEE, 2011.
[70] Mauwa, Hope, et al. "Community healthcare mesh network engineering in white space frequencies." 2019 ITU Kaleidoscope: ICT for Health: Networks, Standards and Innovation (ITU K). IEEE, 2019.
[71] Bhuiyan, Md Zakirul Alam, et al. "Blockchain and big data to transform the healthcare." Proceedings of the international conference on data processing and applications. 2018.
[72] Park, Eunjeong, et al. "Requirement analysis and implementation of smart emergency medical services." IEEE Access 6 (2018): 42022-42029.
[73] Wang, Mengying, et al. "Method and application of information sharing throughout the emergency rescue process based on 5G and AR wearable devices." Scientific Reports 13.1 (2023): 6353.
[74] van Stigt, Maritta N., et al. "ELECTRA-STROKE: Electroencephalography controlled triage in the ambulance for acute ischemic stroke—Study protocol for a diagnostic trial." Frontiers in neurology 13 (2022): 1018493.
[75] Antevski, Kiril, et al. "A 5G-based eHealth monitoring and emergency response system: Experience and lessons learned." IEEE Access 9 (2021): 131420-131429.
[76] Peralta-Ochoa, Angélica M., et al. "Smart healthcare applications over 5G networks: A systematic review." Applied Sciences 13.3 (2023): 1469.

[77] Zhang, Jun, et al. "Intelligent speech technologies for transcription, disease diagnosis, and medical equipment interactive control in smart hospitals: A review." Computers in Biology and Medicine 153 (2023): 106517.
[78] Dennis, Michael V. "Digital Photographs in the ED: Images of an accident can offer ED staff a clearer picture of the cause and extent of victims' injuries." AJN The American Journal of Nursing 103.12 (2003): 44-46.
[79] Zhai, Yunkai, et al. "5G-network-enabled smart ambulance: architecture, application, and evaluation." IEEE Network 35.1 (2021): 190-196.
[80] Kaymak, Burcu Azapoglu, and Merve Eksioglu. "Rapid Ultrasonography for Shock and Hypotension Protocol Performed using Handheld Ultrasound Devices by Paramedics in a Moving Ambulance: Evaluation of Image Accuracy and Time in Motion." Prehospital and Disaster Medicine (2024): 1-6.
[81] Li, Shing-Han, et al. "Developing an active emergency medical service system based on WiMAX technology." Journal of medical systems 36 (2012): 3177-3193.
[82] Chiti, Francesco, et al. "A broadband wireless communications system for emergency management." IEEE Wireless Communications 15.3 (2008): 8-14.
[83] Sisinni, Emiliano, Dhiego Fernandes Carvalho, and Paolo Ferrari. "Emergency communication in IoT scenarios by means of a transparent LoRaWAN enhancement." IEEE Internet of Things Journal 7.10 (2020): 10684-10694.
[84] Drăgulinescu, Ana Maria Claudia, et al. "LoRa-based medical IoT system architecture and testbed." Wireless Personal Communications (2020): 1-23.
[85] Kose, Serdar, Enes Koytak, and Yusuf S. Hascicek. "An overview on the use of satellite communications for disaster management and emergency response." International Journal of Emergency Management 8.4 (2012): 350-382.
[86] Cid, Victor H., Andrew R. Mitz, and Stacey J. Arnesen. "Keeping communications flowing during large-scale disasters: leveraging amateur radio innovations for disaster medicine." Disaster medicine and public health preparedness 12.2 (2018): 257-264.
[87] Kirubarajan, Abirami, et al. "Artificial intelligence in emergency medicine: a scoping review." Journal of the American College of Emergency Physicians Open 1.6 (2020): 1691-1702.
[88] Meier, Lea, Jan Gabriel Bauer, and Kerstin Denecke. "Speech-based Documentation in Emergency Medical Services with the Electronic Language Interface for Ambulance Services." 2020 IEEE International Conference on Healthcare Informatics (ICHI). IEEE, 2020.