DECICE

DEVICE-EDGE-CLOUD INTELLIGENT COLLABORATION FRAMEWORK

Grant Agreement: 101092582

D5.1 Use Cases Requirements

This project has received funding from the European Union's Horizon Europe Research and Innovation Programme under Grant Agreement No 101092582.

Document Information

Deliverable number:	D5.1
Deliverable title:	Use Cases Requirements
Deliverable version:	1.3
Work Package number:	WP5
Work Package title:	Deployment, Validation and Performance Assessment
Responsible partner	TOP-IX
Due Date of delivery:	2023-08-31
Actual date of delivery:	2023-08-30
Dissemination level:	PU
Type:	R
Editor(s):	Christian Racca, Gabriele Rocca, Luca Rocco (TOP-IX)
Contributor(s):	Francesco Barchi (UNIBO), Andrea Bartolini (UNIBO), Mujdat Soyturk (MARUN), Felix Stein (Uni-Goettingen), Berkay Yaman (BigTri)
Reviewer(s):	Jonathan Decker (Uni-Goettingen), Riccardo Cavadini (E4), Dirk Pleiter (JSC)
Project name:	Device-Edge-Cloud Intelligent Collaboration framEwork
Project Acronym:	DECICE
Project starting date:	2022-12-01
Project duration:	36 months
Rights:	DECICE Consortium

Document History

Version	Date	Partner	Description
0.1	(2023-07-12)	(TOP-IX)	(Initial internal version)
0.2	(2023-07-14)	(TOP-IX)	(Second internal version)
0.3	(2023-07-18)	(TOP-IX)	(Third internal version)
1.0	(2023-07-21)	(TOP-IX)	(First official version)
1.1	(2023-08-02)	(TOP-IX)	(Comments by reviewers)
1.2	(2023-08-22)	(TOP-IX)	(Integrated additional contributions)
1.3	(2023-08-25)	(TOP-IX)	(Updated with high resolution image)

Acknowledgement: This project has received funding from the European Union's Horizon Europe Research and Innovation Programme under Grant Agreement No 10192582.

Executive Summary

The deliverable titled "Use Cases Requirements" presents a comprehensive compilation of the three use cases playing a crucial role in developing the DECICE project. The deliverable, led by TOP-IX with support from E4, collected technical requirements from various stakeholders to define essential prerequisites for the project's test bed.

The use cases cover a diverse range of applications and will be utilized to test the DECICE framework, evaluate performance and provide feedback for ongoing developments using a co-design approach.

The first use case involves smart city infrastructure provisioning for Connected and Autonomous Driving and Cooperative Intelligent Transportation Systems. It includes real-time processing for object detection and safety in intersections, with edge devices strategically placed for optimal latency and reliability, adhering to job placement and scheduling standards.

The second use case focuses on Magnetic Resonance Imaging scans, standard medical examinations conducted on edge nodes to create high-resolution 3D images. Due to the sensitive personal information in scans, special care is required in storage, transfer, and access of the data. Challenges include computing and storage complexities, moving analysis jobs to the Cloud, network transfer limitations, and data access management.

The third use case addresses infrastructure provisioning for disaster management and emergency response. It aims to develop an open digital platform supporting operators using data from drones and satellites, enabling the embedding of machine learning algorithms for autonomous flight and mission operations. The platform orchestrates processing and communication between Cloud/edge/drones and provides adaptability to various emergency conditions.

The deliverable contributes mainly to project **Objective 5** demonstrating the usability and benefits of the DECICE solution for real-life use cases. The information included in this deliverable also helps to achieve project objectives 1, 2 and 4.

Challenges encountered during requirements collection include stakeholder diversity, ambiguity, incompleteness, lack of domain knowledge, and communication barriers. Clear communication channels and cross-functional team meetings were established to mitigate these difficulties.

The target audience for this report includes mainly the project partners.

Requirements were collected through two online workshops and appropriate templates for specification collection, with participation from representatives across various organizations involved in the DECICE project.

Each use case will be presented through a technical overview, a description of the working scenario(s), a detailed analysis of the network and infrastructure components as well as computation, storage and time constraints. A conclusive section is dedicated to performance and functional requirements.

Contents

1 Purpose and Scope of the Deliverable

This deliverable aims to compile a comprehensive list of DECICE project use cases and test applications from various stakeholders, which play a crucial role in the project's development and help define the requirements for the test bed. These use cases encompass a diverse range of applications and will be utilized to test the DECICE framework, evaluate performance, and provide valuable feedback for ongoing developments, emphasizing a co-design approach.

TOP-IX has the lead of this task and, with the support of E4 (leader of the related Work Package), collected the technical requirements consistently across different applications, ensuring clarity and facilitating the implementation process. In particular, GWDG, MARUN (Marmara University) and UNIBO (University of Bologna) contributed as the use case owners.

The deliverable will be a foundation for subsequent project activities, contributing essential insights for developing the DECICE framework and enabling effective collaboration among project partners.

2 Use Cases Abstracts

The **first use case** is about smart city infrastructure provisioning. This use case focuses on real-time processing in Connected and Autonomous Driving and Cooperative Intelligent Transportation Systems. Connected and Autonomous Vehicles use onboard sensors for object detection and situation awareness. Intersections introduce additional difficulties, challenges, and safety issues due to varying road users (cars, trucks, buses, pedestrians, bicycles, motorcycles, wheelchairs users, etc.) in varying numbers moving in different directions. Detection of road users (including Vulnerable Road Users - VRU) ensures safety and traffic efficiency. The use case also includes complex scenarios with multiple intersections. Real-time processing can be done with single or multiple-edge devices according to the needs of momentary processing power and network bandwidth. These edge devices can be located at the intersections or in a location that is close to the crossroads to fulfil the latency and reliability needs defined in the standards. It is important to note that the standards provide constraints for the placement and scheduling of the jobs.

The **second use case** focuses on MRI (Magnetic Resonance Imaging) scans, a standard medical examination tool well-diffused due to the noninvasive way of viewing the inner body parts. Typically, MRI scans are conducted on edge nodes; they only cover individual body parts of a patient and create a high-resolution 3D image of the body part using magnetic fields. Advances in technology lead to higher resolutions and, consequently, more significant data sizes. Like other medical data, MRI scans often contain sensitive personal information about the patient. Therefore, special care concerning storage, transfer and access is necessary. The analysis of the MRI scans poses difficulties, both from a computing and storage perspective. Each MRI scan consists of multiple slices of data, thus creating a complex object for analysis. Complex models for research can be challenging for the computing capabilities of edge devices; therefore, moving analysis jobs to the Cloud is necessary. Additionally, the computation time can be a limiting factor. Finally, managing access rights to the data, the storage location and preventing data loss are vital components from the storage perspective.

The **third use case** concerns infrastructure provisioning for disaster management and emergency response. It aims at developing an open digital platform to support emergency response operators exploiting data from drones and satellites. The objective is to bring intelligence to the operational activities in the field by providing computation for embedding machine learning algorithms to support drone autonomous flight and mission operations. The platform will implement an efficient and adaptive analysis of AI algorithms in the field, orchestrating the processing and the communication between Cloud/edge/drone, allowing the operator to control the data flows between the computation entities and providing adaptability to emergencies, such as the size of the area to be explored, recognition tasks to be performed, communication link bandwidth, energy availability and weather conditions.

3 Contribution to Project Objectives

The DECICE project endeavours to compile an exhaustive inventory of use cases and test applications from diverse stakeholders. This compilation assumes a pivotal role in the project's progression, acting as a guide to define essential prerequisites for the test bed.

The use cases are utilized for testing the environment, particularly the DECICE framework, evaluating the performance and providing feedback on the developments while pursuing an approach of co-design.

A particular emphasis is given to the performance requirements in each use case; therefore, performance metrics are derived to evaluate the fulfilment of these requirements.

The deliverable contributes mainly to project **Objective 5** demonstrating the usability and benefits of the DECICE solution for real-life use cases. It also helps to achieve the following other project objectives.

- **Objective 1** - Develop a solution that allows to leverage a compute continuum ranging from Cloud and HPC to edge and IoT: D5.1 provides important inputs to the development efforts, e.g. technical requirements.
- **Objective 2** - Develop a scheduler supporting dynamic load balancing for energy-efficient compute orchestration: D5.1 allows to identify scheduling requirements, which would impact decisions on scheduling algorithms, as well as information required to make such decisions.
- **Objective 4** - Design and implement a Dynamic Digital Twin of the system with AI-based prediction capabilities: D5.1 could allow to identify information, which the digital twin may have to provide in order to make scheduling decisions.

4 Difficulties Encountered

Projects involve multiple stakeholders with diverse backgrounds, perspectives, and interests. Aligning their expectations and requirements can be challenging, as each stakeholder may prioritize different aspects of the project. Some of the encountered difficulties are listed below:

- **Ambiguity**: Requirements may be communicated in ambiguous or vague terms, leading to misunderstandings and difficulties translating them into actionable specifications.

- **Incompleteness**: Sometimes, stakeholders may not fully understand or articulate their needs, leading to incomplete requirements that lack crucial details.
- **Lack of domain knowledge**: Project teams may not understand the domain or industry in which the use case is being executed, making it challenging to collect accurate and relevant requirements.
- **Communication barriers**: Miscommunication, different ways of thinking, or lack of effective communication channels can hinder the collection of accurate and precise requirements.

5 Sustainability and Lessons Learned

The two main strategies adopted to mitigate above difficulties are:

- **Establish common approach and proper communication channels**: A common approach helps to find a way towards a common understanding of the use cases. Well-defined communication channels facilitate smooth collaboration between stakeholders. Regular meetings and workshops should be utilized to encourage open dialogue and information sharing.
- **Cross-functional team meetings**: Conduct periodic cross-functional team meetings involving representatives from other related work packages. These meetings can help identify potential areas of overlap or cross-contamination, enabling proactive resolution of conflicts.

6 Engagement activities linked to this deliverable

The requirements were collected through two online collective and interactive workshops combined with compiling appropriate templates for collecting specifications. The activity was carried out mainly within Work Package 5 but was supported by activities within the other Work Packages (in particular Work Package 2).

6.1 Target audience

The audience for this deliverable is:

	The general public (PU)
✓	The project partners, including the Commission services (PP)
	A group specified by the consortium, including the Commission services (RE)
	This report is confidential, only for members of the consortium, including the Commission services (CO)

6.2 The first workshop

On Thursday, April 20th 2023, a first workshop was held online to kick off the collection of requirements from stakeholders, with a particular emphasis on the three critical use cases identified within the project.

The workshop was attended by representatives from various organizations involved in the DECICE project, including researchers, engineers, and other professionals. The workshop's goal was to initially present and gather valuable insights from these stakeholders to better understand the requirements for infrastructure provisioning and orchestrator development in the context of the three identified use cases.

Participants engaged in a collaborative discussion and brainstorming throughout the first workshop. They finalized a template, initially prepared by TOP-IX, to collect the requirements for infrastructure provisioning in the three use cases. This template was then filled by each use case owner in the following two weeks and uploaded to the project repository.

6.3 The second workshop

Following the analysis of the compiled templates, a second shared work session with the use case managers became necessary. This necessity became even more evident following the first steps of the Scheduler, a core element in the design work carried out within Work Package 2.

One of the main project objectives is designing and implementing a resource scheduler capable of dynamically and more efficiently orchestrating computing power while reducing environmental impact.

The definition of the metrics needed to optimise the resource scheduling process (the focus of Deliverable 2.1) required a more precise formulation of the use cases, especially concerning the metrics and logical workflows.

The second workshop followed a reverse approach from the first event. A specification collection document was shared in advance, and during the workshop, the three use case managers outlined the design guidelines, dependencies and constraints.

Participation in the second workshop was also extended outside Work Package 5 to increase the Project members' awareness of the use cases and, simultaneously, present the use case managers with the progress made at the core infrastructure design level.

The second workshop was hosted online on July 13th 2023.

6.4 Publications

Each of the workshops presented above has been documented with a post on the project's web channels.

7 Use Cases Analysis

7.1 Intelligent intersection with VRU detection

7.1.1 Technical overview of the use case

Use case owner: MARUN (Marmara University)

The use case is about locating and identifying (through image processing) vulnerable road users (VRUs), including pedestrians, cyclists and motorcyclists, at the Road intersection to generate alerts for drivers and prevent potential dangers using V2X (Vehicle-to-everything) communications.

The object detection models will be employed to evaluate the spatiotemporal variations of users (such as vehicles and VRUs) and environmental conditions (such as light, illumination, weather conditions, etc.).

The main goals in the use case are:

- To detect VRUs robustly and accurately within the intersection area using 3D localization models.
- To reduce the risk of accidents that involve VRUs by providing drivers with real-time warnings via V2X communication (warnings are received by the vehicles that are equipped with On-Board Units, OBUs in short).
- To enable real-time processing within the computing continuum for ensuring immediate response and timely alerts to mitigate accident risks.
- To gracefully adjust the inference and model updates.

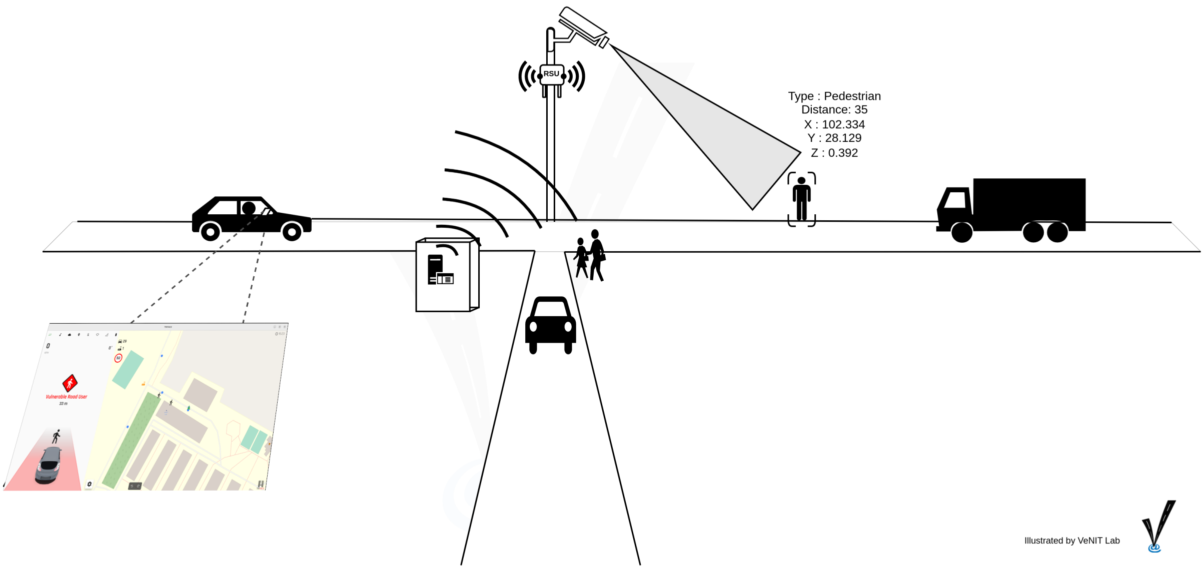

Figure 1: Use case 1 reference schema

The edge device provides the processing power and services for fast response to the stakeholders at the intersection for one or more intersections (according to the processing power needs). Edge is located near the intersection(s) to fulfil the latency and reliability requirements. An edge device is assigned to each intersection; however, if the need for processing power in a specific intersection is low, the edge device(s) located at other intersections can be used. The edge devices also communicate with the Cloud to transfer the model updates and inferences. The Cloud will host the main model for VRU detection, which edge devices will update aperiodically. It will also transfer inferences to the edge devices resulting from occasional updates. The Cloud can also host some programs that collect logs and other metrics from the edge devices that can be used for future improvements. The Kubernetes cluster and orchestrator can be run in a Cloud platform located in an enterprise or public network. Detection and localization models should be executed in the same container running in an edge device. As the model will be updated and adapted to its assigned intersection, the models should be stored in persistent storage. Cloud should host a container image for the main model and the program to manage the operations for federated learning (federated learning methods can be applied to update and adapt the models incrementally). During the project, additional containers might be needed to collect metrics, logs, etc. The use case is currently in the development stage. The V2X communication and connectivity part of the use case has been completed. In the current stage, the models for detection and localization are being developed.

7.1.2 Working scenario

The system will be tested at the intelligent intersection (at Marmara University), where cameras monitor the intersection for such cases given above, and the traffic lights are controlled by the Traffic Light Controller, which interacts with the Roadside Unit Base Station (RSU) at the intersection and the vehicles.

Three specific application scenarios have been defined.

Scenario 1

The DECICE framework deploys the inference and (edge) training applications to fulfil and match constraints. The current scenario consists of single or multiple intersections. In a multi-intersection case, a model can run on other intersections' edge devices (or clusters) if the network delay, device processing power, and model requirements are satisfied. Considering the network delay, the framework should predict network congestion and relocate the model before the system starts to fail meeting performance and functional requirements. Figure 2 shows an example of this scenario. It is seen in the picture that the inference that is processing Intersection 3's camera frames is deployed on the edge device which is physically mounted on Intersection 2. This is a possible case if the requirements (e.g. latency) hold. However, it might not be possible to run or a reschedule might be needed when the requirements don't hold (Intersection 1). Running multiple containers on the same edge device can be advantageous regarding energy efficiency since it is possible to put unused devices into sleep mode during that time.

Scenario 2

The DECICE framework must manage an inference model update. Therefore, it replaces the currently running container after receiving a software update request. The old model persists until the healthy status is achieved on the newly deployed model and inference. The framework assigns the proper device based on the new model's resource constraints and available devices.

Scenario 3

Parallelizing for time constraints. The inference process should provide outputs with specified frequency (see section "Performance and functional requirements"). Therefore, if the delay of a single instance (the inference assigned for a specific intersection's camera) is insufficient for providing the minimum frequency, the framework can spawn duplicates of the inference in parallel to satisfy the frequency constraint. The number of duplicates required is based on the delay caused by processing a single frame (inference for detection and localization) and the device's processing power to be used.

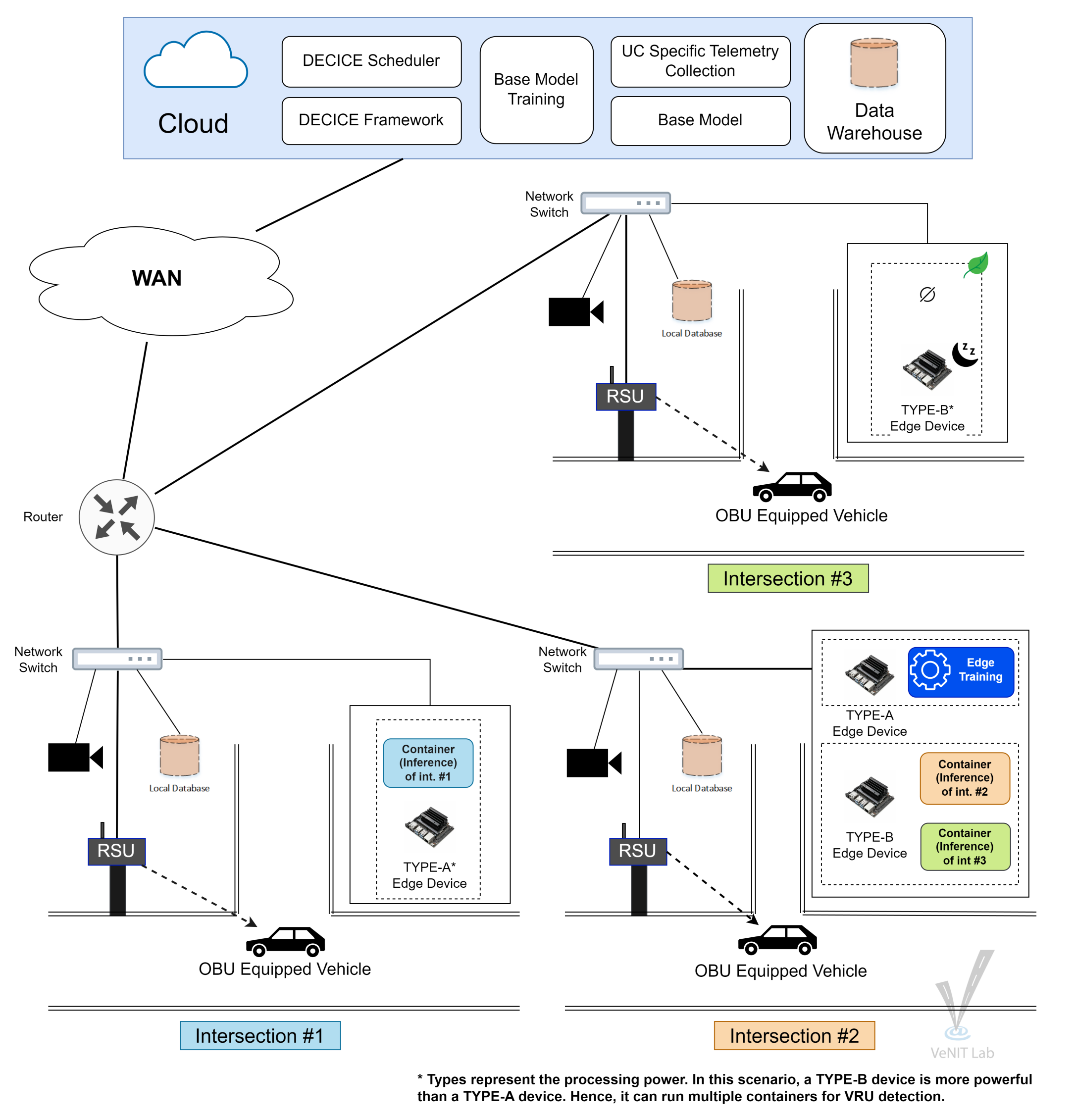

Figure 2: An illustration representing the scenario of VRU detection and localization container scheduling for multiple intersections

7.1.3 Network characteristics

In terms of network reliability:

- The connection between the intersection equipment (RSU, edge devices, camera) and the Cloud is assumed not stable and can be best-effort.
- The connection between different intersections' local networks is assumed not stable and can be best-effort.
- The connection between the camera, edge devices, RSU and the OBUs (that belong to the vehicles passing the intersection) should be 99% reliable according to 3GPP TS 22.186 version 17.0.0 Release 17, for sensor information sharing on low-autonomy.

In terms of bandwidth:

- The bandwidth between the Cloud and the intersection equipment can be varying and less than 1Gbps.
- The bandwidth between different intersections' local networks can be varying and less than 1Gbps.
- At least 1Gbps bandwidth is required for the network connecting camera to edge device, and edge device to RSU.

7.1.4 Use case infrastructure components

Typically, one edge device is located on a single intersection. However, the jobs can be consolidated, reducing the number of edge devices when the need for processing power is low. Marmara University will record video frames in their test site and transfer them to edge devices to simulate this environment in the development and testing platform. In an advanced version, on the other hand, cameras will directly feed edge devices with frames.

As the jobs for different intersections can be run on a single-edge device, latency and processing power requirements should be considered during the placement of the devices. Edge devices have a connection with the Cloud. However, this connection has flexible requirements.

To simplify the use case, a single intersection can be considered where one edge device will be enough. However, Cloud support is still needed. Furthermore DECICE is supposed to focus on scheduling challenges. Having only a single RSU would eliminate relevant scheduling decisions.

In previous research and in the preliminary stages of the project a simulation environment was used to define the specifications and test the different conditions.

7.1.5 Computation, storage and time constraints

Computation The needed computing power for inference on edge should be at least the reference device: Nvidia Jetson Nano (NVIDIA Maxwell GPU, Quad-Core ARM Cortex-A57 Processor, 4GB LPDDR4). There is no specific restriction for computation needed on the Cloud (for model training

Component name	Component role	Additional details
Edge device	Running object detection model and inference, edge training.	Off-the-Shelf components (i.e. Jetson Orin, Nano, Atlas500 that will be placed in varying proximity to the intersection
RSU (Road Side Unit)	Roadside V2X communication device	One for each intersection
Server (virtual machine in the Cloud	To maintain applications' services and database	-
HPC Cloud	Aimed at base model training and improvement (i.e. incremental learning, federated learning, etc.)	-
Cameras	To collect frames in real time	It is paired to one RSU and not directly communicating with any other components and thus outside of the DECICE infrastructure.

Table 1: Infrastructural components

or other operations needed for the use case). However, it should be sufficient enough for the required operations.

Storage: There is no specific restriction for the storage. However, a sufficient storage should be present in the intersections' local networks and the Cloud for recording captured video frames. Also, the edge devices should have enough storage for storing the models for VRU detection and localization.

Time: The ideal target latency between the edge device and RSU is 20 ms according to 5GAA recommendation "C-V2X Use Cases and Service Level Requirements. Volume I. Technical Report". 100 ms can be considered the worst case.

This time includes transmitting the inference output to the RSU, broadcasting the results via the Road Side Unit and receiving from the OBU. There is no time constraint for connecting edge devices and the Cloud. The latency between the different intersection's local networks is assumed to vary.

7.1.6 Performance and functional requirements

Based on the collection of information, the following set of performance and functional requirements relating to the specific use case is assumed.

Detection Accuracy

Detection of hidden/undetected objects at the edge with more than 95% accuracy. The system should detect risky/unsafe conditions with a rate 10% higher than the baseline. This requirement is not directly related to DECICE framework.

End-to-end Latency

End-to-end latency should be around 20 ms (milliseconds) in ideal conditions (according to 5GAA recommendation "C-V2X Use Cases and Service Level Requirements. Volume I. Technical Report") and 100 ms (3GPP TS 22.186 v17.0.0) in the worst case. End-to-end latency is measured from the moment the output is transmitted by the application (inference) to the moment it is received at the destination (OBU).

For the processing of a single frame captured from the intersection the delay caused by transferring a single frame to the edge device + the delay caused by processing a single frame (object detection and localisation, a.k.a. inference) + delay caused by transferring the output of the inference to RSU (and to the cars by V2X communication) must be <= 300 ms (ETSI TR 103 300-1 V2.3.1, ETSI TS 103 300-2 V2.2.1 and ETSI TS 101 539-1 V1.1.1).

Inference Frequency

Minimum frequency for providing object detection and localization output for a single intersection >= 10 Hz (ETSI TR 103 562 V2.1.1) based on the total number of parallel instances.

Productivity - Dynamic Load Balancing

Rescheduling when constraints aren't fulfilled should be fast enough to keep the system alive. A pre-planned recovery mechanism (even temporarily sacrificing energy efficiency) might be helpful if scheduling puts too much overhead.

Energy Efficiency

The system should implement efficient resource management and adapt to the varying load conditions saving energy compared to the baseline. A system based on the DECICE framework should allow reducing energy consumption compared to a setup where intersections are equipped with independent devices.

7.2 Magnetic Resonance Imaging (MRI) scans

7.2.1 Technical overview of the use case

Use case owner: GWDG

Magnetic Resonance Imaging (MRI) is a type of scan that uses strong magnetic fields and radio waves to produce detailed images of the inside of the body. MRI scans consists of multiple slices of data, thus creating a complex object for analysis. Complex models for analysis can be challenging for the computing capabilities of edge devices. Therefore, moving analysis jobs to the Cloud or HPC system is necessary. Edge devices need to safely aggregate MRI data and send it to the Cloud or HPC for further computation as well as guarantee data security and time sensitivity.

The edge nodes are located as close as possible to the actual object of interest (network, room, etc.) to be able to safely interact with and extract data from the MRI scanner as well as provide radiologists with information after a successful computation.

The volumetric analysis of the brain only relies on a single sequence, which can be started first and can be extracted from the MRI scanner. While the other measurement sequences are running, processing of these first images can happen simultaneously on the edge devices, Cloud or HPC cluster.

The use case was already implemented in a preliminary form before the start of the DECICE project.

7.2.2 Working scenario

This use case consists of a single scenario where edge devices acquire data from the MRI scanner. To guarantee time critical performances (results should become available within 10 minutes), the computation can be distributed across multiple edge devices or computed in the Cloud (using HPC infrastructure). Depending on the load of the edge and the available bandwidth between the edge and the Cloud, the fastest infrastructure is determined. When the job is finished, the results are displayed via edge devices for evaluation.

The main steps are:

- Scan upload.
- Processing.
- Operations of sending back results.

Particularly, "processing" is the component that can be scheduled and orchestrated by the DECICE framework. Below is a flowchart illustrating the processing of two types of images.

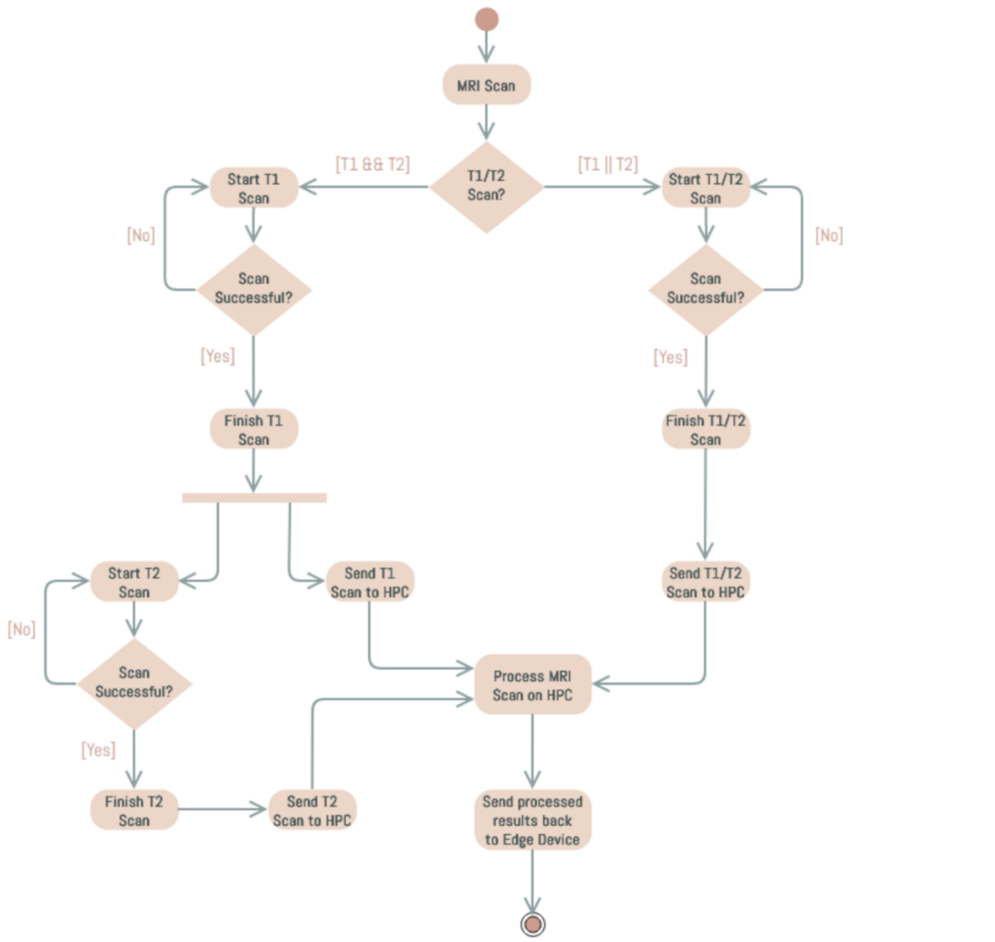

Figure 3: Use case 2 workflow

T1-weighted and T2-weighted images are two different types of MRI images that provide distinct types of contrast for visualizing tissues within the body. They are created by manipulating the relaxation times of hydrogen protons in tissues after exposure to radio frequency pulses. When using an MRI scan radiologists can decide whether they want to create T1 OR T2 images or T1 AND T2 images. T1-weighted images emphasize the anatomical structure of the tissue. In T1 images, fatty tissue appears bright, while fluids and free hydrogen atoms appear dark. Bone structures are typically well depicted. T1 images provide high-contrast images and are useful for depicting tissue anatomy and identifying tissue types. T2-weighted images on the other hand emphasize the water content in the tissue. Fluids and fluid accumulations appear bright, while fatty tissue and bone structures appear darker. T2 images are particularly useful for detecting fluid accumulations, infections, edema, and pathological changes in the tissue.

7.2.3 Network characteristics

In this case there are no specific requirements on the configuration or on the characteristics of the network and the connection between the various components as long as the topology and the specifications guarantee the achievement of the computation time constraints (as better detailed in the following paragraph). Similarly the bandwidth capacity can vary according to the computational capacity of the individual components which determines the quantity of information to be transferred on the network.

7.2.4 Use case infrastructure components

Edge devices acquire data from the MRI scanner and in order to guarantee time-critical results, the computation needs to be distributed across multiple edge devices or has to be computed in the Cloud.

Component name	Component type	Additional details
Edge device	Computation and storage	In charge of data input, computation and data output
Cloud or HPC system	Computation	CPU and GPU
Cloud storage	Storage	Responsible for storing the results of image processing

Table 2: Infrastructural components

MRI scans can be computationally intensive, especially when producing high-resolution, detailed images or when applying complex imaging techniques. Therefore computational demands of processing these MRI scans might not be met by edge devices and must consequently be processed on other devices in the compute continuum such as in a Cloud or HPC environment. Edge devices play, in any case, a crucial role for data transfer, security and reliability to send data to the Cloud or even HPC and receive results back for further analysis. The main task of the scheduler must therefore be to adapt the allocation of resources in order to provide sufficient resources to operate successfully within the established constraints. High availability in this context means reliability. Clinics and radiologists cannot afford to wait for a system that is only available 50% of the time or needs longer than 10 minutes of waiting time for a processed MRI scan.

To guarantee time-critical results, orchestration is necessary for both edge devices and the Cloud. The orchestration framework will ensure efficient distribution of computational tasks between edge devices and the Cloud to minimize latency and maximize processing speed. By leveraging edge computing capabilities, certain computations can be offloaded to edge devices, reducing the dependency on the Cloud. Most significantly, once an execution of this use case starts, resources for upload and computation should be reserved without significant delay.

Furthermore, due to the sensitivity of the data involved in the use case, particularly MRI scans of patients, privacy and compliance with GDPR regulations are paramount. Image processing and computation must be carefully orchestrated to ensure data privacy and compliance with applicable data protection laws. Therefore, to safeguard user privacy and comply with GDPR, special consideration must be given to selecting nodes where image computation occurs. Personal data, including images, should only be processed on nodes authorized and certified to handle sensitive information following

GDPR requirements.

7.2.5 Computation, storage and time constraints

Computation: The use case involves extremely heavy computation, which requires significant processing power. To handle the computational load efficiently, parts of the computation can be performed on GPUs (Graphics Processing Units), leveraging its massive parallel processing capabilities. Other parts of the computation can only be performed on CPUs (Central Processing Units).

Storage: No specific information is provided regarding storage constraints. However, it is essential to ensure sufficient storage capacity for storing intermediate and final results during the computation process. A detailed analysis of storage requirements should be performed based on the nature and scale of the data involved in the use case.

Time: The computation must be completed within a strict time limit and should not exceed 10 minutes. Meeting this deadline is critical for delivering timely results to users and ensuring the overall effectiveness of the use case.

7.2.6 Performance and functional requirements

The following set of performance and functional requirements is relevant for the use case.

Computation Time

As repeatedly stated, one of the most important metrics is the average time taken to process an individual image from the GPU segmentation step to the CPU analysis step. This metric directly measures the efficiency of the computation process and adherence to the strict 10-minute time constraint.

Orchestration Efficiency

Another important metric is the percentage of computational tasks efficiently distributed between edge devices and the Cloud. This metric measures how well the orchestration framework balances the workload to minimize latency and maximize processing speed, thereby ensuring time-critical results.

Privacy Compliance

The adherence to GDPR regulations regarding data privacy and protection during image processing is strongly required. Particularly, it is important to evaluate whether personal data, including images, is processed only on authorized nodes and follows the necessary privacy safeguards. Data protection must be guaranteed both on (possibly a subset of) the edge and Cloud nodes.

7.3 In-the-field intelligence supporting emergency response

7.3.1 Technical overview of the use case

Use case owner: UNIBO (University of Bologna)

The use case is aimed to develop an open digital platform to support emergency response operators exploiting data gathered from robots and satellites and bringing intelligence to the operational activities. The platform will implement efficient and adaptive AI algorithms for autonomous robots' navigation, exploration and planning and will orchestrate the processing and communication between Cloud/edge/robots. The operators will be able to access intelligence data and control mission operations.

7.3.2 Working scenario

The optimal solution consists of:

- An **edge system**, the Base Station (**BS**) connected to the Cloud via a reliable connection, to manage and orchestrate the operations of the autonomous robotic agents (**AR**).
- **Autonomous Robotic (AR)** agents, capable of operating in the environment through ground and/or air movement.
- The **Cloud** aimed at performing computationally intensive but more precise tasks than those performed on the edge system.
- Not time-critical **activities**, aimed to refine the information in the field (e.g. integrating information from satellite maps with filed data).

A minimum viable version of the above scenario consists of the following:

- The Base Station edge system.
- One robotic agent.
- The Cloud environment.

7.3.3 Network characteristics

At the current design state of the use case we expect the different components to be connected to the network in the following ways:

- **BS**: at least 4G connection. 4G/5G or Satellite (e.g.: Starlink) connection in the optimal solution.
- **AR**: at least WiFi with Mesh support + Bluetooth. 4G/5G + WiFi with Mesh support + Bluetooth + NB-IoT in the optimal solution.
- **Cloud**: reliable network connection (>1Gbps).

7.3.4 Use case infrastructure components

Edge devices are Autonomous Robots (AR) and the Base Station (BS). Their main activities are: AR mission planning, local and global mapping and Autonomous Robotic agents' navigation.

Cloud can be a remote computing infrastructure with at least a Deep Learning training accelerator module. Cloud main activities include global mapping optimization, data analysis, and training for the edge-device AI deployed.

The orchestration is required between BS and AR concerning the mission management activities. For instance, in the case of AR in line of sight with BS, some containerized tasks can be executed in the BS to decrease the time constraints of computation activities. For out-line-of-sight operations or in case of network disruption the same task must be executed in the AR (if capable) to maintain the same working memory data/state.

Component name	Component type	Additional details
BS compute unit	Computation	Workstation
AR compute unit	Computation	Like Nvidia jetson
AR real-time compute unit	Computation	Like Holybro Pixhawk
Cloud	Computation	High-end CPU and GPU for GPU
AR Battery	Energy providing	Not included in the DECICE perimeter

Table 3: Infrastructural components

7.3.5 Computation, storage and time constraints

Cloud Computation: Cloud computational power is used for deep learning algorithms and image processing activities.

BS Computation: Base Station Computation power is used for mapping management, mission management, computer vision, and deep learning inference. Technical requirements are:

- Multi-core, x8086, +2GHz CPU.
- CPU.
- +8GB RAM.

AR Computation: AR computational power is used for computer vision and deep learning inference. Technical requirements are:

- Multi-core, ARM, 1-2GHz CPU.
- CPU.
- 1-4 GB RAM in case of optimal solution. Up to +8GB if the solution is not optimized.

AR Real-time Computation: AR computational power is used for autonomous navigation. Technical requirements are:

- Single-core, ARM, <1GHz CPU.
- <1 GB RAM in case of optimal solution. Up to 4GB if the solution is not optimized.

Cloud Storage: 100 GB - 1 TB. This storage is needed for neural networks models, environment maps, intelligence-relevant data (e.g. images, videos) and sensor data. If the solution is not optimized from the software point of view, storage requirements can increase up to >1TB. There are no constraints on memory availability on edge devices.

Time Critical Components:

- Robots' autonomous navigation is time-critical for people and robot safety.
- Point of interest recognition (e.g. survivors) for maximizing the success of emergency operations.
- Communication between AR and BS to allow the mission management in BS and allow a higher movement speed of AR.

Map optimization and neural network model training are not time-critical.

7.3.6 Performance and functional requirements

The following set of performance and functional requirements is relevant for the use case.

Data Security

It is mandatory to avoid the storage of data on insecure devices.

Productivity

Real-time detection (e.g. people) must be lower than 10 seconds.
"Navigation speed" and "Explored area per time unit" are additional metrics to be monitored.

Mission Replanning and Task Migration

Mission re-planning based on real-time data must be lower than 1 minute.
An additional related metric is the "task migration latency".

Latency

The main metrics to monitor here are the "communication latency" and the "available bandwidth between components".

Energy Efficiency

Deployment choices can affect energy autonomy. Therefore this metric has to be carefully monitored. A system based on the DECICE framework should allow reducing energy consumption compared to a setup with independent devices.

Glossary

AR AR. 20–22

BS Base Station. 20–22

HPC High Performing Computing. 7, 14–16, 18

MRI Magnetic resonance imaging. 5, 6, 15–18

OBU On-Board Units. 10, 13–15

RSU Roadside Unit Base Station. 11, 13–15

V2X Vehicle-to-everything. 10, 11, 14, 15

VRU Vulnerable Road User. 5, 6, 9, 10, 12, 14