17th International Conference on Space Operations, Dubai, United Arab Emirates, 6 - 10 March 2023.

SpaceOps-2023, ID # **600**

Augmented and Virtual Reality for Ground Station and Telescope Maintenance at ESOC

Nebras Nassar[a]*, Sebastian Martin[b], Ruediger Gad[a], Pouya Haschemi[a], Manuel Olbrich[c], Fabian Rücker[c]

[a] *Terma GmbH, Bratustraße 7, 64293 Darmstadt, Germany*, nena@terma.com

[b] *European Space Agency (ESA), European Space Operations Centre (ESOC), Robert-Bosch-Straße 5, 64293 Darmstadt, Germany,* *sebastian.martin@esa.int*

[c] *Fraunhofer Institute for Computer Graphics Research IGD, Fraunhoferstraße 5, 64283 Darmstadt, Germany,* *manuel.olbrich@igd.fraunhofer.de*

* Corresponding Author

Abstract

Augmented Reality (AR) and Virtual Reality (VR) technology can potentially enhance the work of maintenance engineers by providing them with real-time visual and textual information overlaid on their physical environment (AR) and allowing them to simulate tasks in a virtual environment before executing them in real-world (VR). This study aimed to explore the use of AR and VR in supporting engineers in their daily tasks and providing support from remote experts for the Ground Station and Telescope maintenance at the European Space Operations Centre (ESOC). The study presents conceptual and practical results, including use case identification, technology analysis, design and implementation of a proof-of-concept prototype, and demo application scenarios.

Keywords: Maintenance, Remote Support, Augmented Reality, Virtual Reality, Ground Station, Telescope

Acronyms/Abbreviations

European Space Agency (ESA), European Space Operations Centre (ESOC), Maintenance Engineer (ME), Remote Expert (RE), Proof-of-Concept (PoC), Commercial-off-the-Shelf (COTS), Network Data Interface Unit (NDIU), Head Mounted Display (HMD), Content Management System (CMS), Light Detection and Ranging (LiDAR), Field of View (FoV), Augmented Reality (AR), Virtual Reality (VR).

1 Introduction

The European Space Operations Centre (ESOC) in Darmstadt controls most unmanned space probes of the European Space Agency (ESA). To communicate with, e.g., spacecrafts, by the transmission of commands to the spacecraft and reception of status and scientific data from the spacecraft, a network of Ground Stations distributed worldwide (ESA's ESTRACK network) is operated from ESOC. These Ground Stations are often in remote geographical locations, and local staff do often not possess the required specialization and/or knowledge to perform all tasks related to configuration, maintenance, and operation of the hardware equipment. Maintaining Ground Stations is essential to ensure reliable communication, prevent potential issues and failures, extend the lifespan of the equipment, and ensure safety and security. As a result, more experienced engineers must travel to the sites to perform their respective tasks. For ESOC, this problem is not unique to Ground Stations, which is why remote maintenance of Telescopes is also considered within the scope of this work.

Augmented and Virtual Reality. Augmented Reality (AR) and Virtual Reality (VR) are rapidly evolving technologies that have the potential to improve various industries. AR superimposes digital information onto the physical world, while VR creates a fully immersive, computer-generated environment. AR/VR can help engineers to work more efficiently by overlaying digital data onto equipment and allowing them to interact with it through interactive and visual representation. These technologies have various applications, including training, visualization, and communication. In the commercial building management sector, similar challenges exist as described in this paper but with very different equipment and data. For example, the INSITER project [4] has shown that AR can be a promising tool to support workers in unknown environments by improving the usability of digital building models with the use of AR applications. AR has proven to be a valuable tool in iterative engineering projects. In [5], the study shows that superimposing planned geometry in partially constructed machines helps to better visualize and communicate planned iterations and identify problems. Furthermore, the automotive industry has, for several years, invested in the development of maintenance procedure support tools that employ AR techniques. Workshop employees use a tablet to support car maintenance procedures, with the tool augmenting their view of the hardware equipment with additional information relevant to the current procedure step. This information can be drawings, technical data, visual augmentation, and animations (e.g., color-coding the right place where a screw shall be opened or animating the installation of a hose into the fitting slot). The same techniques are also used in training scenarios, telemedicine, as well as for teleoperations, where a local operator is performing the tasks, while a remote operator sees his/her exact view and can provide detailed information by directly augmenting the view of the local operator.

Study context. The activity presented here focused on the specific use case of introducing and utilizing AR and VR technologies in support of (remote) maintenance tasks in Ground Stations and Telescope at ESA/ESOC.
The study is executed in the context of previous and parallel activities on leveraging AR/VR for space operations activities at ESOC. The previous and parallel activities focused on technology assessment, demonstrators and use case discovery, and developments of proof-of-concept demonstrators in spacecraft operations. Unique to this activity compared to other AR/VR related activities in spacecraft operations is the in-situ handling of real hardware, e.g., Ground Station or Telescope equipment. In contrast to this, in spacecraft mission operations, no real hardware can be accessed directly as spacecraft or rovers typically operate in space or on other celestial bodies without physical access. This has a strong implication on the suitability for using AR. Due to the nature that in Ground Stations and Telescope operations, real hardware can be directly accessed, advantages of AR can be better leveraged. It is possible to combine real objects with augmented functionality like virtual computer-generated content for, e.g., displaying instructions, interactively taking notes, recording status information, or navigating in real environments with the help of AR overlays. The activities of this study were split into three phases:

- **Explorative phase:** This phase explored potential use cases of AR/VR with the domain and the required technology assessments. The use case analysis aimed to identify useful application scenarios and use cases which could also streamline the design and development of the actual prototype to be developed later. The technology assessment aimed to allow a methodologically driven approach for the selection of required Commercial-off-the-Shelf (COTS) hard- and software solutions, which were be an integral part of the prototype. The use case scenarios influenced the technology assessment on one hand, while the feasibility of use cases depended on available technologies on the other hand. Because of this interdependence, both tasks were executed simultaneously in an iterative process.
- **Development phase:** In this phase, a prototype was implemented covering the selected use cases and their assessment. It was initiated by an architectural design followed by the agile and iterative implementation of the envisaged prototype. The architecture principle and the design of the system followed the client-server model. The system was divided into three layers: the client layer, the service layer, and the resource layer.
- **Application phase:** In this phase, a demo application using our developed proof-of-concept (PoC) implementation was applied at ESOC.

Envisaged system. To understand the landscape of the needed system, we sketch a simple block diagram showing our system, namely the AVRGST system and all the other systems that it interfaces with.

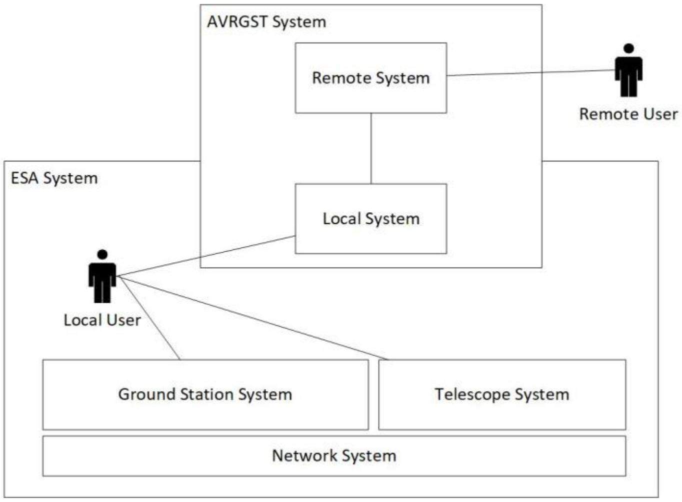


Figure 1-1 Block diagram showing our AVRGST system interacting with other systems and users

The AVRGST system's primary goal is to support the maintenance engineer (ME), located in the Ground Station or Telescope, in the daily work by, e.g., providing information about the equipment and allowing remote support from the remote expert (RE), who is located in the study case in ESOC premises, by using AR and VR technologies. As shown in Figure 1-1, the AVRGST system was divided into two subsystems:

- *The Remote System* with users called remote experts (REs), who traditionally would travel to Ground Station sites - but with the help of the tool should stay located in ESOC offering remote guidance, and
- *The Local System* with users called maintenance engineers (MEs) who have direct physical interaction with hardware (Ground Station system or Telescope system).

Contribution and structure. The main contributions of the paper are listed as follows:

- Exploration of use cases for AR/VR in Ground Station and Telescope maintenance
- Technology assessments, including the selection of the AR/VR devices for implementing the use cases
- System functionality identification
- System architecture design
- Proof-of-concept (PoC) implementation
- Demo application scenarios

The paper is structured as follows: Section 2 presents the methodology and results of applying the explorative phase. The development approach, system architecture, and implementation are presented in Section 3. Section 4 presents and discusses the application scenarios performed using the development software at ESOC, while Section 5 concludes the paper.

2 Explorative phase: methodology and results

This section gives an overview of the followed approach in the explorative phase and presents its outputs.

2.1 Approach

Our approach to this exploratory phase was divided into two parts. We started by analyzing the business use cases mentioned in the Statement of Work [7] to perform a functionality and requirements analysis and generate the system requirements. The use case ideas were collected during several workshops with ESOC staff. Due to measures to prevent the spread of Covid-19, the workshops took place in an online format. The ideas shared and discussed during the workshops were written and exchanged on the interactive, collaborative platform called Miro Board[1]. Thereafter, high-level user stories were derived from these ideas. Depending on the employed software development process, e.g., Kanban, Scrum, etc., these high-level user stories can also be seen as "epics". These high-level user stories were designed to have representative requirements, which were called within the scope of agile development "acceptance criteria". The aim was to establish a good trade-off between, e.g., the level of detail, the communication of software development, and design considerations versus the overhead spent on generating, maintaining, and tracing requirements. Thus, we were able to achieve high efficiency within the development process. Furthermore, the AR/VR devices and technologies were evaluated and selected to implement the use case ideas, and toolchain options that may be capable of realizing the use case ideas were suggested as well.

2.2 Use cases

In this section, we outline the process and the output of collecting, analysing the uses case ideas, and deriving the related user stories. As an initial starting point, we used the business use cases as described in the Statement of Work (SoW) [7] provided initially by ESA as a basis. These reference use cases provide a good scope for demonstrating AR/VR capabilities in the envisaged application domain of Ground Stations and Telescopes. This assessment was based on our experience from previous activities [1, 6] in which we also performed use case identification and selection and on the currently on ongoing parallel activity [8] that focuses on ESOC mission operations. Three different use cases were presented in the SoW:

- Local AR maintenance use case for Ground Station: This use case is about a maintenance engineer familiar with the Ground Station system, who shall carry out maintenance works or troubleshooting on the equipment level. The equipment to be serviced is known or well enough documented in the system.

---

[1] https://miro.com/

- Remote AR maintenance use case for Ground Station: This use case features two locations staffed by a remote expert and a local maintenance engineer. The purpose is for the remote expert to remotely help the maintenance engineer perform the correct actions on the equipment, as well as for the remote expert to show the ME the proper information.
- Telescope use case: This use case requires specific hands-free capabilities for the engineer to access and move within the target telescope, limiting the suitable hardware. Additionally real-time calibration information from third-party systems shall be available to the engineers during maintenance tasks within the telescope.

We lifted these SoW use cases to groups called “Areas of Interest” (AoI). These Areas of Interest then contained detailed user stories for specific use cases, which will be described in later stages. As part of the execution of this activity, a specific workshop with ESOC staff was performed. Within this, potential users of the envisaged solutions were invited to explain from their point of view the potential use cases in detail, to identify potential applications and limitations and to understand current workflows. Multiple use case ideas were generated during this workshop. These ideas were then mapped to the three pre-mentioned AoI. After this mapping, clusters of ideas were identified. These clusters were cross-cutting the Areas of Interest. This means that a cluster could spread over several AoI. During the workshop, five different and independent clusters were identified. The participating users were then asked to vote for these clusters. Two of the five clusters were identified as the most relevant. Table 2-1 contains both clusters including the corresponding use case ideas as defined during the workshop. Each idea (with its idea number) is correlated to a specific area of interest with a corresponding identifier (Remote (R), Local (L), or Telescope (T)).

Table 2-1 List of collected ideas

ID	Description	Cluster	Areas of Interest
I1	Configure the Network Data Interface Unit (NDIU) remotely with the support of an ESOC Operator	Cluster A	R
I2	Virtually highlighting objects with e.g., a virtual circle, arrows, and descriptions shall be possible		R
I3	For a maintenance procedure, the AR system can display steps and provide system description, interface definitions, etc		L
I4	Allow AR overlays from different devices		L
I5	From procedure document to tablet/AR guided instruction process		L
I6	ME at equipment site, the AR system provides with the set of all available of drawings, procedures, wiring diagrams, etc. available for this equipment		L
I7	ME selects relevant content. The AR system overlays information about the equipment into the field of view (e.g., port number, signal name, cable types, connectivity, etc.)		L
I8	The operator can get displayed instructions on how to adjust the different lenses to achieve the best possible focus		T
I9	The operator can identify the Telescope elements while doing maintenance (i.e., Camera ID, etc.)		T
I10	The operator can see in his/her AR FoV sensitive "no-touch" areas		T
I11	RE and ME can communicate during AR operation via Audio	Cluster B	R. T
I12	The RE can use virtual hands to show the ME instructions		R. T
I13	The RE can look at the environment of the local ME without the ME having to look himself		R
I14	"Real time" 3D models or 360° videos of a room/equipment to enable a RE to look independently of the ME field of view and make markings in the field (instructions)		R
I15	The field of view of the AR system can be streamed to a remote expert (RE) who shall be able to interact with the ME		R, T
I16	Not only having video and audio exchange between sites but also showing virtual hands		R, T

Table 2-1 covers only high-level ideas, and there are no defined user stories yet. Since a user story may involve several ideas, the following process was approached to generate the user stories: At first, the ideas were analyzed and classified to their high-level functionalities. Thereafter, possible user stories were derived by combining several possible ideas. As a result, 5 generic ideas were identified (the ideas with high-level functionalities, namely ideas I2, I3, I4, I14, and I15). Based on the resulting generic ideas, backlog items (later user stories or subtasks) were produced by calculating all possible combinations of the generic ideas. More than 25 backlog items were produced. Table 2-2 shows some of the backlog items out of 25 items.

Table 2-2 A subset of the backlog items (user stories, tasks)

ID	Derived from Ideas	Description	Subtask of
BL 7	I2 + I3 + I15	As a user (ME), I can communicate with RE, highlight objects, and get detailed information	-
BL 8	I2 + I3 + I14	As a user (RE), I can make a virtual tour, highlight objects, and get detailed information	-
BL 10	I5	As a user (ME), I can get tablet/AR guided instructions from a procedure document	BL 7
BL 11	I7	As a user (RE), I can overlay information about the equipment into the field of view of ME (e.g., port number, signal name, cable types, connectivity, etc.)	BL7 & BL 10
BL 13	I9	As a user (ME), I can identify the Telescope elements while executing maintenance (i.e., Camera ID, etc.)	BL 7
BL 14	I10	As a user (ME), I can see in my AR FoV sensitive "no-touch" areas	BL 7
BL 24		As a user (ME), I can use a 3D authoring tool to create guided instructions based on e.g., a user manual	BL 10

2.3 Technology assessment

This section is an assessment of state-of-the-art hard- and software that was considered for the project.

2.3.1 Hardware selection

Most use cases require equipping the ME with at least a tablet computer, while some require more specific hardware for hands-free operation.

Tablet. The main interaction device for the maintenance engineer (ME) is the tablet. For our proposed solutions, we depend on a target platform with strong AR support, which means the device has to offer functionality to track itself in a room. Additionally, environment meshing to support occlusion is a valuable feature to make the augmentations easier to understand by only superimposing data that is not hidden by real-world objects. These features led to using the apple “iPad Pro” as a tablet device. The “Pro” variant offers an integrated LiDAR (Light Detection and Ranging) sensor to sense and measure the distance to objects in the environment. Since Apple[2] offers a standardized interface to this depth sensor data, there are several COTS 3D scanning applications available for iPad Pro/iPhone Pro devices.

Virtual Reality headsets (VR HMD). There are several VR headsets with different specifications and features on the market. Fundamentally they can be split into two groups: standalone devices like “Meta Quest 2”[3] or “Pico 4” and tethered devices that need to be connected to a powerful computer like “Varjo XR3” or “HTC Vive Pro 2”. For our project, we went with the “Meta Quest 2” because it can be used both as a PC-VR headset but also as a standalone device. Additionally, there are no external tracking sensors needed, which reduces the setup time immensely. This

---

[2] https://www.apple.com/

[3] https://www.meta.com/de/en/quest/products/quest-2/

approach offered the most flexibility. The device is marketed towards end users, which led to a very user-friendly setup procedure, which we hope increases the acceptance of the device in our use cases.

Augmented Reality headsets (AR HMD). Current AR headsets can also be split into two groups: optical see-through (OST) and video see-through devices. The former category of headsets allows the user to see his surrounding environment as if the user was looking through a normal pair of glasses. Additional content like 3D models is superimposed onto and anchored into the real world. Video see-through headsets achieve the AR effect by capturing the environment with built-in cameras and streaming it to the displays in front of the user's eyes. Since the targeted use cases aim on an engineer working on small and sensitive devices while wearing the HMD, we opted for optical see-through since it allows the best real-world perception without any delays. We decided to use the "HoloLens 2"[4], which also offers a standalone experience and detaches the user less from his real surrounding. Other AR headsets like "Google Glass", "Nreal Light" or "Magic Leap" did not meet the project's needs due to lacking spatial awareness or other features.

2.3.2 Software selection

3D-Scanning. Many features of the project depended highly on the availability of 3D models. We know from previous activities that 3D models are often hard to obtain since they are either designed by third parties or simply non-existent. Therefore, a simple solution to create 3D representations of real objects was needed. There is a large variety of 3D scanning software on the market. Some of them use a collection of normal 2D photos to reconstruct the 3D geometry (Photogrammetry), while others make use of a depth sensor. The former category of scanning software is more accessible since a normal 2D camera suffices to create 3D models. Nevertheless, the editing process is rather time-consuming and, therefore, more suited for experienced users. Common computer editing software is e.g., "Meshroom"[5] and "RealityCapture". Recently, a photogrammetry mobile app called "RealityScan" was released that allows photogrammetry on a mobile device. Creating 3D models with depth sensors on the other hand is comparatively fast and beginner friendly since there are apps available with an intuitive user experience like "3D Scanner App"[6] and "Scaniverse"[7]. For this project we used an "iPad Pro", which is equipped with a LiDAR depth-sensor. While the level of detail of the 3D model was not perfect, it satisfied the requirements and the high-quality texture taken with the high-resolution camera upvalued the resulting models.

3D-Engines. There are two popular game engines that most games are developed with and used to create augmented reality experiences: Unreal Engine and Unity Engine. The development in Unreal is based on C++ while Unity uses C#. While both engines target a similar developer audience, support, and documentation, special use cases like mixed reality seem to be better on Unity's side. This also is reflected in the hardware manufacturers' documentation, which seems more comprehensive for Unity. Additionally, with the project team's experience, the decision was made for Unity.

Authoring tools. There are multiple authoring tools that allow the user to design and create, e.g., VR training or animated step-by-step tutorials. "Scope AR" and "Reflekt One" are two exemplary solutions that allow the user to create instruction routines. "MobiPV" is a solution that ESA used in previous activities. For our cases of application, these solutions were insufficient because they could not be embedded in a live-remote assistance system.

Remote assistance solutions. Microsoft offers its own remote assistance solution with "Microsoft Dynamics"[8]. The user can call a remote expert using a tablet or "HoloLens" and can see annotations set by the remote expert during the live session. "TeamViewer Assist AR"[9] offers similar capabilities. Our project required annotations to be saved such they could be reviewed at any time, which is why the aforementioned solutions were insufficient.

In general, many of the mentioned software are rather island solutions for a specific use case and, therefore, they might not be suitable for this project, which targets multiple use cases with a single solution.

---

[4] https://www.microsoft.com/en-us/hololens/hardware
[5] https://alicevision.org/
[6] https://apps.apple.com/us/app/3d-scanner-app/id1419913995
[7] https://scaniverse.com/
[8] https://dynamics.microsoft.com
[9] https://www.teamviewer.com

2.4 Toolchain

This section describes the toolchains that were used to implement the use-cases and ideas, which were considered useful during the workshops. Elements of these toolchains are either already available software packages or elements of our developed applications, such as the ME application on portable devices (the AVRGST ME Application) and the web applications, such as the AVRGST Content Management System (CMS) and Communication Application (see Section 3). Some of the use case ideas with suggested toolchain solutions are listed as follows:

- *Configure the Network Data Interface Unit (NDIU) remotely with the support of an ESOC operator.* Any configurations that needed to be made directly on a computer could be addressed by using an established bi-directional video streaming service like "MS Teams" and the screen sharing functionality. Since the environment of the on-site engineer should be streamed by using a portable device like "iPad Pro" such that the remote expert can interact with the 3D environment, we developed our own solution. This was achieved with the developed ME application and the communication application based on the WebRTC[10] technology.
- *Virtually highlighting objects with e.g., a virtual circle, arrows, and descriptions shall be possible.* For some real-time remote assistance scenarios Commercial off-the-Shelf (COTS) products like "TeamViewer Assist AR" are sufficient. The remote assistant can draw in the on-site user's field of view and highlight and annotate real objects in his surroundings. Nevertheless, this solution is not sufficient for an offline / not-live scenario, because remote assistance sessions and annotations cannot be saved and revisited later. Therefore, a custom functionality was built that allows the user to save annotations and highlights with respect to the real environment, while also allowing live annotation in shared sessions.
- *From procedure document to tablet/AR guided instruction process.* We developed the ME application to allow playing back step-by-step guides in AR, which were created in the developed Content Management System (CMS). The system allows different levels of fidelity, from simply having access to a procedure document in the 3D environment, to AR supported procedures with 3D models and spatial information. This spectrum allows easy transfer, without restricting the potential of AR user guidance.
- *ME selects relevant content. The AR system overlays information about the equipment into the field of view (e.g., port number, signal name, cable types, connectivity, etc.).* The data entered via the CMS is displayed relative to the real object by placing it into the field of view with respect to the scanned QR-code.
- *Telescope use-case: The operator gets displayed instructions on how to adjust the different lenses to achieve the best possible focus.* The engineer uses an AR HMD to live stream the Telescope's desktop view. Therefore, the user gets direct feedback while adjusting and sees how the Telescope view changed. This livestream is produced on the host machine by capturing software like VLC[11] and using our developed applications, the ME and communication applications.
- *The RE can use virtual hands to show the ME instructions.* The RE is either able to interact with the ME by placing annotations directly in the environment using his computer screen or the user can use a VR device to view the 3D model spatially and move within the virtual environment as if he was physically present. These features are supported using our developed applications.
- *Not only having video and audio exchange between sites but also showing virtual hands.* The developed communication solution offers video and audio communication and data exchange channel for sharing the positions of the RE's input devices (cursor, controller or tracked hands).

3 Development phase: methodology and results

This section gives an overview of the development approach and the key features and architecture of the developed system, namely the AVRGST system.

3.1 Approach

Our approach to designing and developing the needed system involved a systematic process that began with the analysis of the system to identify its main functionalities. This step was crucial in ensuring that the final system met the requirements of the users, suggesting the solutions, and estimating them. Based on analyzing the user ideas and requirements, we identified the main functionalities needed to realize a comprehensive solution from the development point of view as follows:

---

[10] https://webrtc.org/

[11] https://www.videolan.org/vlc/

- Building AR objects (3D models): It includes obtaining or creating 3D models for assets and rooms and persisting them to be shared later.
- Displaying and overlaying dynamic data: It involves dynamic and user-friendly overlaying of different AR objects, such as plain data, diagrams, and media, superimposed on the camera view.
- Highlighting and annotating: It aims at highlighting objects in the view and annotating real and virtual objects. The annotated AR objects can be persisted and shared for later use.
- Communicating and remote support: It provides audio and video communication, enhanced with real-time annotation and direct support functionality.
- Providing services: All data for the previously mentioned functionalities must to be provided by several data services, such as data processing and delivery.

This list helped us to organize the backlog items (which were created based on the collected user stories). Then, we divided the system into three layers: the client layer (including ME and RE devices), the service layer (data services and web applications), and the resource layer (the data set).

To realize most of the use cases, we proposed and evaluated various solutions. We first tried to support each use case by investing in the solutions based on existing tools as much as possible. Therefore, we went over each use case and foreseen the primary development obstacles and constraints, such as the absence of the COTS tools and the complexity of the configuration and integration. Nevertheless, some cases could not be covered by existing tools, and some solutions using existing tools might not have fulfilled the user's desire. Therefore, we proposed three complexity-based solutions to cover many use cases while keeping time and cost in mind. The complexity factor is based on the components of the AVRGST system that must be developed. All these solutions depend on developing AR applications and services and integrating them into existing systems. In the end, we developed a prototype implementation of an advanced solution that covers all the functionalities. It includes our custom-built remote support tools and contains advanced AR applications, services, and web-based applications. This will allow us to continue developing it so that we can get a ready and advanced software solution.

Thereafter, we selected the technologies to implement the system. They were selected to solve several challenges, such as providing platform-independent applications, dealing with different distributed systems, increasing modularity, supporting asynchronous and synchronous communications, and considering the interoperability and compatibility among software components. The development activity during this project followed the agile principle.

3.2 System architecture and implementation

This section briefly presents some details about the system architecture and its implementation. Our system's architecture principle and design goal followed the client-server model since it efficiently delivers resources to the client and allows separating concerns into each layer, increasing the modularity. Figure 3-1 shows an overview of the concrete components of the system. We distinguish two kinds of components, client components, and service and web application components.

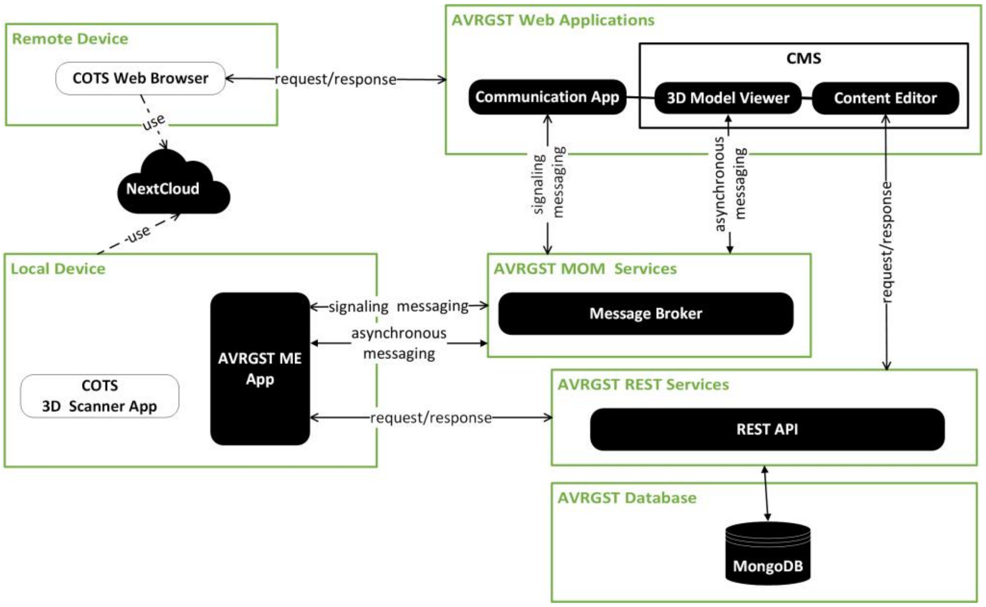


Figure 3-1 Detailed diagram of the AVRGST components

Client components. The components in the client layer were designed to be a platform-independent application that can be easily adapted for different operating environments. Different components were developed based on the user's device (iPad, HoloLens2, PC/Laptop), role (remote, user, or admin), and required functionalities. The local device of the ME may contain the followings:

- AVRGST ME Application: It was developed to provide different functionalities, such as displaying 3D models, overlying information, interacting with AR objects, and establishing a communication channel.
- COTS Applications: Some COTS apps could be used such as the 3D Scanner app (Scaniverse[12]) to create a 3D model for an asset and a browser app to interact with the AVRGST Web Applications.

The remote device of the RE should contain COTS apps, such as a web browser to interact with the ME via the AVRGST Web Applications.

Service and web application components. The service components were developed to mainly provide a collection of highly maintainable and loosely coupled services for different purposes. The REST services[13] and the Message-Oriented Middleware (MOM)[14] services were developed to make the data available as resources for the client applications and other system components. We developed several components for various purposes as follows:

- AVRGST Web Applications: It has two main applications:
    - Content Management System (CMS): It consists of a web-based Content Editor, designed to allow users to manage the database data and retrieve them through a user-friendly web-based interface, which is extended by a 3D viewer. This extension allows viewing 3D models and creating 3D data.
    - Communication Application: It was provided to establish the audio and video communication between the RE and the ME and to allow sharing of data and camera feed. The RE could use several added extensions, such as remote drawing to support the ME in the daily work.
- AVRGST REST Services: A REST API was developed to receive requests from the AVRGST ME Application and return the needed data in a JSON format. Additionally, the API was used by the CMS to manage the AVRGST Database.
- AVRGST MOM Services: It was developed to allow the 3D Model Viewer and AVRGST ME Application to exchange messages asynchronously. The exchanged messages appear on both sides without manually refreshing or rendering the apps. It was also played as a signaling server to prepare the bidirectional communication channel between RE and ME devices.

---

[12] https://scaniverse.com
[13] https://www.redhat.com/en/topics/api/what-is-a-rest-api
[14] https://www.ibm.com/cloud/learn/message-brokers

In the following, we provide more details about the developed components:

AVRGST ME Application: The application was developed to provide features such as: identifying QR codes, displaying and repositioning AR objects, overlaying dynamic information and annotations, and establishing a communication channel. The application aligns the room model with the real room by scanning two or more known QR codes. Based on the scanned QR codes, the application provides graphical user interfaces (UI) on each known device to request and receive additional data, such as getting instructions, PDF files, diagrams, and 3D models and hide information related to the selected object. Furthermore, additional functionalities like highlighting "no touch areas" or displaying annotations are supported to help the user to get a first understanding of assets without the help of experts. Figure 3-2 shows an example of the interactive graphical user interfaces displayed on specific positions of devices in a room. Using the UIs, the user can request additional information about the device parts. Subcomponents are identified and marked with a box around them. The user can reposition the AR objects, such as 3D models, by moving them via a specific relocation interface on the screen of the tablet and save their current state back to the data service in the backend. Moreover, the application allows the users to configure their session information to start/end the video and audio calls. Mobile users can share their camera feed. The RE's video feed always stays in the ME's field of view (FoV), while the ME's video feed is streamed to the RE. The application was first developed for iPad and then customized for HoloLens 2. Note, the core functionality of the client components can be transferred to android devices. The iPad Pro was specifically selected due to its superior indoor tracking capabilities. Some contemplated features rely on the iPad Pro's depth camera, but those could be omitted for other platforms.

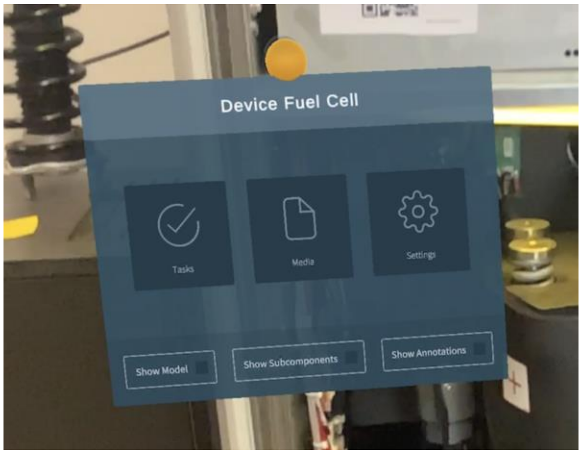


Figure 3-2 An example of the user interface (UI) based on a scanned QR code

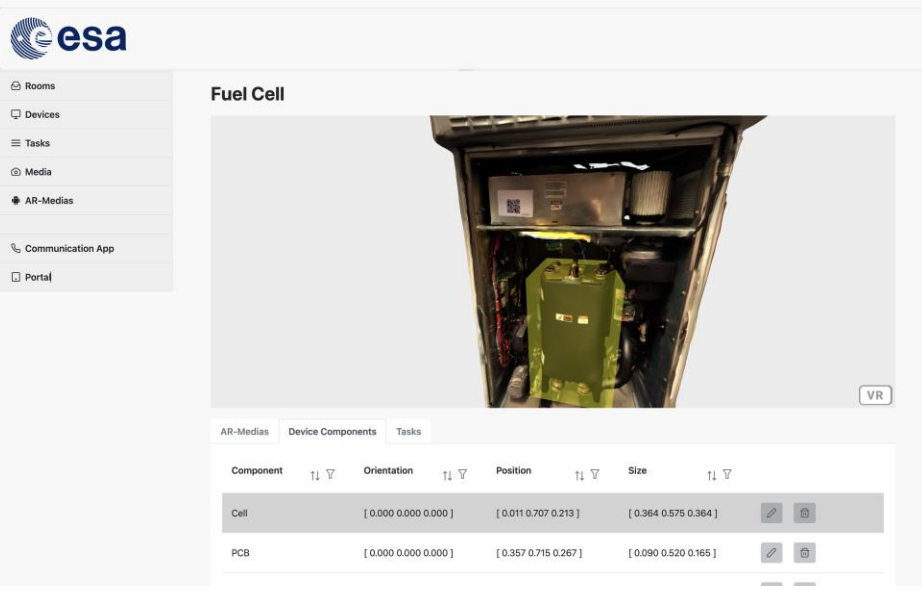
AVRGST Content Management System (CMS). It was developed to maintain the content data of the project, and thus we provide a common interface for collecting data in accordance with the project data structure. The AVRGST CMS is a web-based administration tool to manage all the needed data of rooms, assets, and media attached to them. The data was persisted in the database and managed and retrieved via the AVRGST REST API Services. Additional data, such as 3D scans/models, can be uploaded to a file service and afterward be linked via the corresponding URI to the relevant item via the CMS. This also works for URIs of existing data from the internal or external network, e.g., a PDF. Our web-based application was implemented using the Angular framework[15], and thus it can run on every device. It is also designed to be responsive to adapt to common screen sizes. The CMS consists of a main page and contains several sub-pages to manage the content of the project entities such as:

- Room Page: As a basic item to start with, the content manager can create a room item for every individual room that shall be set up. This is like the container for every asset that later can be attached to it. A 3D representation of the room is recommended (e.g., through a 3D scan) to help the RE to navigate the room.
- Task Page: Here, we can prepare tasks, or, more precisely, step-by-step instructions to guide an engineer through different stages of a procedure.

---

[15] https://angular.io/

To display 3D Models, a customized 3D Model Viewer based on A-Frame.js[16] was used. It can be accessed and controlled directly through the web interface. Additionally, it is possible to use the same 3D Model Viewer via a VR-HMD, which allows the user to get a better spatial understanding. The 3D model can be viewed from different angles, and custom annotations and media information can be placed regarding the object. An example of the CMS page is shown in Figure 3-3.



Component	Orientation	Position	Size
Cell	[0.000 0.000 0.000]	[0.011 0.707 0.213]	[0.364 0.575 0.364]
PCB	[0.000 0.000 0.000]	[0.357 0.715 0.267]	[0.090 0.520 0.165]

Figure 3-3 An example of AVRGST CMS page including the Model Viewer

AVRGST Meta-model and Database. We designed the meta-model based on our analysis of the use cases. Several entities are identified to hold the data of a room, a device and its components, and related media, and to hold the procedure tasks. To implement the database, we used MongoDB[17], an open-source document database that employs a flexible schema for data storage. Applications can then retrieve the data in a JSON[18] format which has many benefits: human-readable, structured, and unstructured data can be stored in the same document.

AVRGST REST API. We developed the REST API to include a set of subroutines and protocols that allow client and application components to communicate with the database and to manage its data through a set of HTTP verbs such as GET, PUT and DELETE. Since the REST architecture treats every content as a resource, the API was developed to respect the meta-model defined previously by specifying at least one resource for each entity. Additionally, further resources were defined to provide further services, such as performing specific queries. Totally our API included more than 20 REST resource types being developed to meet the requirements for the ME application and CMS. For each resource/entity the main functionalities were developed to get, add, update, and delete a resource. Each resource was identified by URIs. The API was implemented using Python[19].

AVRGST Communication Application. Our system provides a communication tool that allows two distant users to collaborate efficiently. It provides the ability to establish audio and video communication and exchange data interactively. This allows remote experts to support on-site engineers with specific problem cases. Additionally, the remote user can interact with the local user's real environment by highlighting specific objects or points of interest and annotating them if needed. Figure 4-4 shows an example of the web page of the application. This web interface consists of three windows: one displays a 3D model of the room, another shows the MR's camera view, and the third shows the RE's camera view.

---

[16] https://aframe.io/docs/1.3.0/introduction
[17] https://www.mongodb.com/home
[18] https://www.json.org
[19] https://www.python.org/

To realize these two features of communication, we developed a communication application based on WebRTC and message broker technologies. While devices such as the iPad allow using software like Microsoft Teams in parallel with custom applications, the same is not true for the HoloLens 2. Also, such a connection would degrade the remote expert to a spectator that can communicate verbally or force him to open a second window to have some interaction with the ME's system. With the integration of WebRTC into the ME applications on iPad and HoloLens, the call is handled from inside the application, and the RE can have the communication in a rich web application.

WebRTC needs a signaling server to prepare the communication between two peers, e.g., to send and receive communication offers from the caller and callee and to exchange the needed information. In our system, the signaling was handled via the AVRGST MOM Services and thereby simplified the clients' connections to the backend services. To implement the MOM services, we used Eclipse Mosquitto[20], an open-source message broker.

To exchange the data such as annotations and drawing lines between the RE and ME, we developed annotation and drawing functionalities so that the RE can annotate 3D models and draw on the FoV of ME, for example. Since the RE may use a flat 2D screen while the ME's surrounding is 3D, a ray cast is used to place them within the 3D environment, more specifically, by placing them on the closest 3D geometry relative to the spot marked on the 2D screen. If the RE uses a device like the Meta Quest, annotations can be placed freely in 3D space.

3.3 System deployment

To ease the process of deploying the services and web applications and running them on any target system, we used the containerization concept for the backend services as far as possible. We packaged the applications with their environments and dependencies so that they remain completely independent of the host environment. Containers enable us to securely run the apps in complete isolation with all runtime requirements pre-packaged with the app itself. Thus, this approach frees us from worrying about what is running on the target host machine. Furthermore, the big advantage is that the applications are built once and then can be deployed several times. Docker[21] is an open platform for developing, shipping, and running applications. We defined a Docker[22]file to package the REST API and the related artifacts automatically, as an example. The docker file is used to build a docker image which can be used to create several instances of the REST API as well. To ease the process of managing the whole lifecycle of the services, we used the Docker Compose tool.

Regarding the client components of the iPad, we used the TestFlight platform[23], which allows us to upload the deployed AVRGST ME Application to the store and then share its link with specific users to download and install the application. The AVRGST ME Application of the HoloLens2 was hosted on a cloud portal, such as Nextcloud and shared with the users.

4 **Application phase: scenarios and discussion**

This section presents the application of our developed AVRGST software at GSRF (Ground Segment Reference Facility) at ESOC. The GSRF comprises a full spectrum of systems constituting a ground segment and the simulated space segment. Engineers use the GSRF, for example, to validate and experiment with systems under development. The GSRF contains a large variety of hardware and racks deployed in the Ground Stations and Telescope facilities. For asset management and tracking, each hardware item is already equipped with QR codes. Hence the GSRF was a representative environment to test and validate the products of this study.

In the following, we show the main features and identify the steps and control flow of applying the AVRGST software. Thus, we can evaluate several characteristics such as complexity, time expenditure, and usability. While the application can be used for simple remote support without any preparation, the full features depend on preexisting data in the application's backend. This allows the ME to access information and the RE to navigate in the MEs surroundings. Therefore, the following section is separated into Preparation Phase and Operational Phase.

4.1 Preparation phase

This phase covers all steps to set up a digital twin of a room in the backend. This includes:

- Creating (or otherwise obtaining) 3D models for the room and assets

---

[20] https://mosquitto.org/
[21] https://www.docker.com/
[22] https://www.docker.com/
[23] https://developer.apple.com/testflight/

- Capturing QR codes in the room (at least relevant assets, ideally room markers)
- Aligning the 3D model of the room (alignment between the real world and the virtual representation)
- Inserting the room and device data (models, files, URLs, annotations, tasks, etc.)

To fully use the applications' features, a digital representation of the targeted room needs to be created and aligned with the actual direction of the room. The 3D models of the room and assets were created using the 3D Scanner app (Scaniverse[24]). This step took just about 15-20 minutes. Then, the data of the room and its devices had to be prepared. Adding a room to the backend database can be done via the CMS or the ME application (the AVRGST ME Application). For room alignment (alignment between the real world and the virtual representation), we just scanned two QR codes there using the ME application. This again can be done via the CMS (if QR code positions are known), or they can be captured with the ME application. For easy realignment, some QR codes for this purpose were fixed in the room.

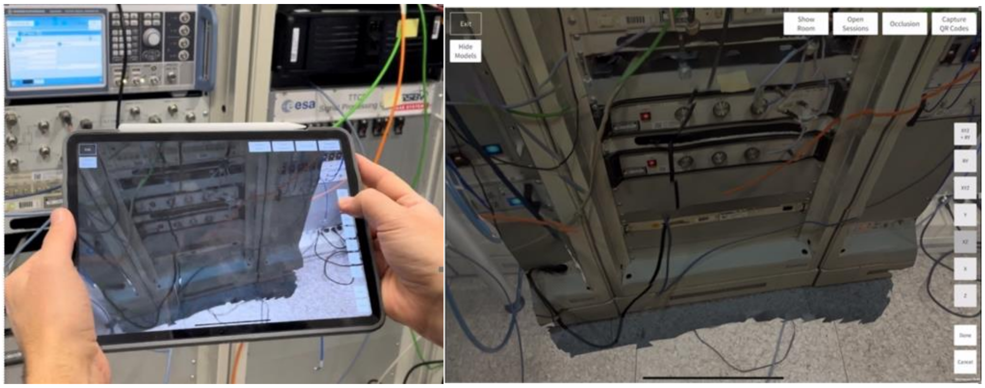


Figure 4-1 Aligning the 3D model of a device in the FoV via the ME app

After room alignment was dealt with, additional devices were added via the CMS or by scanning the QR code using the ME application. Additionally, the CMS was used to add additional media to devices, such as 3D models, documents, or URLs. Similarly, task data were added to the device. After adding a 3D model of a device, we used the CMS to mark subcomponents of the device and add annotations, which can later be viewed by the ME on site.
The RE can use the Model Viewer in the CMS to navigate and annotate the 3D model of the room, which contains the representation of the devices. But the 3D models of the devices have no inherent position and orientation in the virtual room, so these devices' models need to be aligned. To easily position the 3D model of a device into the correct position of the physical device, the ME can use the application transformation control provided by the ME application, as shown in Figure 4-1 (right), to align the 3D model and thus these positions will be automatically saved back into the backend. At this point, all possible preparations were done, and the ME application can be used. The preparation phase using our developed prototype took about one hour. However, this should be done once.

4.2 Operational phase

The application is designed as an information system used by the ME on his/her own. But it also offers extensive remote support capabilities, which allow an RE to join a session and support an ME. The experience for the ME will be described with the ME application on the tablet (iPad Pro) and on the AR-HMD (HoloLens 2). The required step in this phase is scanning QR codes so the ME application can be used to its full potential.

4.2.1 Maintenance engineer with tablet (iPad Pro)

To use the ME application as an information system, the ME just ran the ME application and was greeted with the request to scan QR codes to align the virtual representation with the real environment. This was achieved by scanning at least two known QR codes, ideally room markers or known device QR codes. After this step, all known devices were augmented with their specific information interface in the camera view. At this point, the ME was able

---

[24] https://scaniverse.com

to approach the device of interest and start to interact with its interface. Figure 4-2 (left) shows the room overview with UIs on each known device, and (right) shows a UI with subcomponents being highlighted in the background.

Figure 4-2 (left) Room overview with UIs in on each known device and (right) Closeup of a device UI showing the device components being highlighted

The ME was able to do the followings:

- Overlay the device model on the real object, as shown in Figure 4-1. This was to verify the device orientation was correct and to see parts of the device which are currently obscured by other materials.
- Display device components, as shown in Figure 4-2 (right). This would help to identify specific parts of the device and highlight sensitive parts that should be avoided.
- Switch to the Media view to view and relocate spatial media like 3D models or pictures placed in 3D, view annotations which are linked to the media objects, and to open media like documents, images (without 3D position) or other URLs.
- Switch to the Task view to select a main task to work through and select the current active subtask as shown in Figure 4-3, which contains textual information about the subtask, as well as media references linked to the subtask. Depending on the subtask definition, 3D media and/or annotations will automatically be displayed in the camera view.

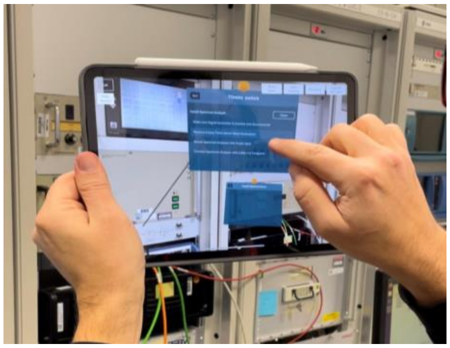


Figure 4-3 The interaction with the UI device

4.2.2 *Maintenance engineer with HMD (HoloLens 2)*

The ME application of the HMD is similar to the one on the iPad. While wearing the HMD, the ME has his/her hands-free to interact with the environment and all augmentations. This benefit comes at the cost of having to interact with the application via hand gestures instead of simply tapping a screen. The opened interfaces can follow the user through space and can be spatially locked, so its position remains static requiring the ME to come back to it for interaction.

Regarding our use cases, the use of HMD with our ME application is mainly useful to implement the Telescope use cases where the ME needs to adjust and maintain the Telescope while reading some data from a computer connected to the Telescope (controlling computer). We supported these use cases using the ME application of HMD with a remote desktop viewing integration. This allowed the ME to get information from an external desktop computer while being hands-on with a device. This simplified interactions, where a local change (like turning a calibration pot) has direct results, which are not visible on the device itself but on a controlling computer. To have this feature as versatile as possible, the ME just received an MJPEG stream, which can be positioned in the FoV. This versatility means that arbitrary sources can be viewed, like a desktop stream or even network cameras. This option was specifically chosen so that frame-grabber devices can be used on machines where no software can be changed, or which are not connected to a shared network.

4.2.3 Remote support

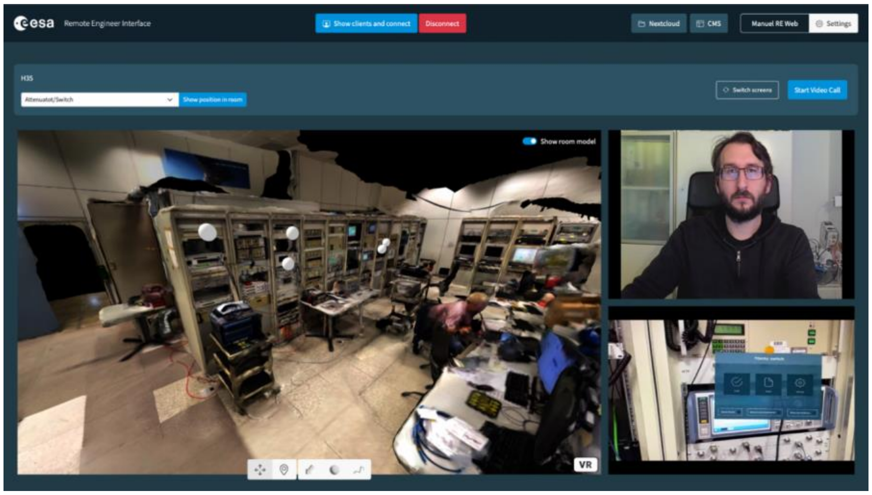


Figure 4-4 Active session in the RE communication web application. Large window on the left shows a 3D model of the room, lower right shows the ME camera view, and upper right shows the RE camera view

The RE used the Communication Application to initiate a support session by starting a voice/video call to support the ME. With this call active, the RE did the followings:

- Make a virtual tour using the 3D view of the room, in which all known devices were marked as shown in Figure 4-4.
- Send go-to messages and annotations to the ME to navigate to a device or a point in the room. The ME just needed to follow an on-screen arrow as shown in Figure 4-5 (left).
- Start the remote drawing feature to leave more detailed information in the ME's 3D environment. On the ME device, these drawings were projected back on the 3D environment, where they were displayed as lines, as shown in Figure 4-5 (right).

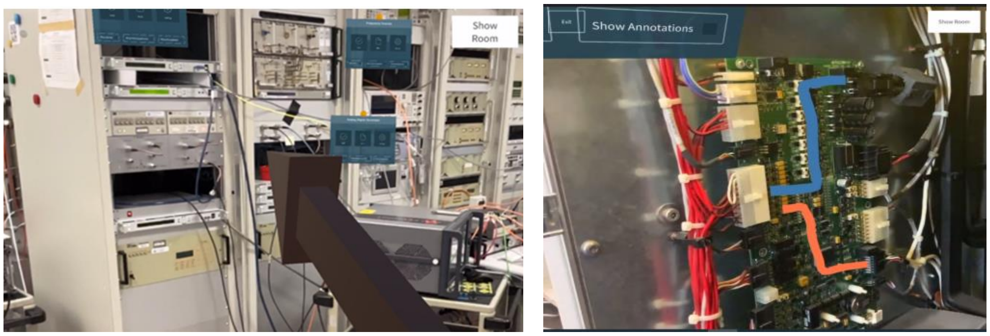


Figure 4-5 Support information from the RE. (left) ME is guided to a navigation point set by the RE. (right) ME screen looking at the drawing

4.3 Discussion

We developed the AVRGST applications with several features promising to increase the efficiency of the work. These developed features provided the ME and RE with real-time and dynamic information, which helped them to make more informed decisions and work more efficiently. Moreover, the applications aimed at enhancing the collaboration between the RE and ME located in different parts of the world, and thus travel costs and time were reduced. The AVRGST software was able to simulate situations that would not be applicable or easy to experience in real life. This is very important if we want to train employees by providing 3D visualizations with dynamic data, which can enhance their understanding, perception of the situation, and their reactions.

Although that HoloLens 2 (AR HMD) and the Meat Quest 2 (VR HMD) are both examples of advanced head-mounted displays (HMDs) that have been designed with comfort and usability in mind, in practice, the usability of AR and VR HMDs can vary widely depending on the intended use case. Several hardware properties, such as battery life, field-of-view, and resolution may affect usability and applicability. Most current HMD devices have a battery life of about a few hours when used actively. The Meta Quest 2 battery life can be upgraded to up to 8 hours with a battery pack that is attached to a special head strap. This, of course, makes the device heavier but also more comfortable due to optimized weight distribution. The battery life of an iPad is about 9 hours. Therefore, the choice depends mostly on the exact use-case scenario and the expected usage time per session. The field of view determines how much of a virtual scene the user can perceive. Most current HMDs provide a field of view between 90° and 120°. See-through devices like HoloLens 2, on the other hand, only extend the user's perceived reality. This means that the user perceives his real environment as usual, but additional digital content is displayed within reality. The field of view of the digital overlay is limited to about 50° for most see-through devices. In our demo application, the field of view was not an issue but increasing the range could be a useful feature to return to the normal range of the FoV of the human eyes. The resolution indicates how many details are visible to the user. The used VR devices are promoted to have "human-eye" resolution, which means that no difference between the real world and the virtual world can be detected with regards to the resolution (i.e., "sharpness" of the view). While using the HoloLens 2, we experienced some cases where reading small fonts might necessitate the use of a zooming feature. A study about how comfortable and accurate working with HMD devices and AR/VR technology should be investigated in the future.

5 Conclusion

To assist local engineers in maintaining Ground Station equipment and Telescopes using Augmented Reality (AR) and Virtual Reality (VR), we conducted the following activities during the study's exploration, development, and application phases: identifying key use cases, assessing and selecting appropriate technologies and hardware, designing the system architecture, developing a prototype of the system, and demonstrating various user scenarios using the developed applications. The use cases were derived based on analyzing the business ideas collected through workshops with the ESOC staff. The identified use cases provided a range of potential applications and were broadly classified into two main categories: *(a) Overlaying dynamic digital data on physical assets*: This category included overlying various types of data based on user requests and presenting AR tasks to guide the maintenance engineer step by step. *(b) Interactive remote sessions*: This category included activities such as video and audio calls, interactive data exchange between a remote expert and a maintenance engineer and the ability for the remote expert to tour the maintenance engineer's environment virtually. After evaluating the user and system requirements, we proposed our solutions to cover a wide range of use cases while keeping time and cost in mind. The developed prototype software can provide several functionalities, such as building AR objects and environments, displaying and overlaying dynamic information, remote support, offering highlighting and annotation aims, and providing different kinds of data services. Overall, the study's outcome demonstrated the potential of AR and VR technology to reduce time and costs, improve efficiency, and enhance communication and collaboration among engineers.

Acknowledgements

This activity was carried out under a programme of, and funded by, the European Space Agency. We thank the members of the Ground Station (GSRF) and Telescope at ESOC for their support throughout the course of this study and demo. We thank the project members who have played a key role in the success of this study, namely Holger Graf, Meriem Drira, Julius Kühn, Robin Horst, Sheetal Rahul Sinha, Andreas Zapf, Rahma Sfaxi, and Lisa Baule.

6 References

[1] R. Gad, M. Valadas, H. Graf, M. Olbrich, J. Keil, M. Sarkarati, S. Bamfaste, F. Nicolini and A. Cowley, "Assessing the Potential of Virtual Reality for Improving ESA’s Space Operations," in *2018 SpaceOps Conference*.

[2] Olbrich, M., Graf, H., Keil, J., Gad, R., Bamfaste, S., Nicolini, F. (2018). Virtual Reality Based Space Operations – A Study of ESA’s Potential for VR Based Training and Simulation. In: Chen, J., Fragomeni, G. (eds) Virtual, Augmented and Mixed Reality: Interaction, Navigation, Visualization, Embodiment, and Simulation. VAMR 2018. Lecture Notes in Computer Science(), vol 10909. Springer, Cham. https://doi.org/10.1007/978-3-319-91581-4_33.

[3] P. Hořejší, Augmented reality system for virtual training of parts assembly, Procedia Engineering, 2015; 100(1):699–706.

[4] R. Günther, A. Kluth, M. Olbrich, J. Braun, and T. Bauernhansl. “Mixed Reality for on-Site Self-Instruction and Self-Inspection with Building Information Models,” -1, 2018. https://publica.fraunhofer.de/handle/publica/253936.

[5] M. Shahid, B. Jesche, M. Olbrich, H. Graf, A. Franek, A. Kuijper, U. Bockholt, and M. Schmitt. “Measurement Based AR for Geometric Validation within Automotive Engineering and Construction Processes,” -1, 2020. https://publica.fraunhofer.de/handle/publica/408435.

[6] R. Gad, H. Graf, M. Sarkarati, C. Scott, “Study Results on Employing Virtual and Augmented Reality for Spacecraft Operations and Astronaut Training at ESOC and EAC,” in 2019 AR/VR for European Space Programmes Workshop.

[7] ESA/ESOC, Statement of Work ESA Express Procurement Plus EXPRO+, “Augmented and Virtual Reality Support Tool for Ground Station Equipment and Telescopes”, 2021.

[8] R. Gad, S. Martin, M. Olbrich, M. Fischer, S. Baci, F. Ruecker, R. Sfaxi, , “Towards Leveraging Augmented and Virtual Reality for Spacecraft Mission Operations at ESOC”, in 2023 SpaceOps Conference (to appear).